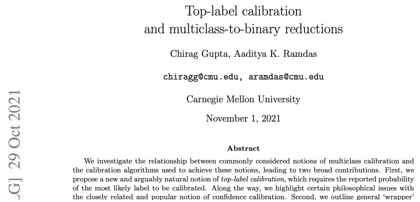
Top-label calibration and multiclass-to-binary reductions

Presented by: Shiyun Xu

University of Pennsylvania

April 1, 2022



Top-label calibration
and multiclass-to-binary reductions

Chirag Gupta, Aaditya K. Ramdas

chiragg@cmu.edu, aramdas@cmu.edu

Carnegie Mellon University

November 1, 2021

Abstract

We investigate the relationship between commonly considered notions of multiclass calibration and the calibration algorithms used to achieve these notions, leading to two broad contributions. First, we propose a new and arguably natural notion of *top-label calibration*, which requires the reported probability of the most likely label to be calibrated. Along the way, we highlight certain philosophical issues with the closely related and popular notion of confidence calibration. Second, we outline general 'wrapper'

Content

1. Calibrated predictors
2. Calibration algorithms for 'M2B' calibrators
   1. Experiments: M2B calibration with histogram binning
   2. Distribution-free top-label calibration using histogram binning
   3. Top-label calibration using histogram binning
3. Binning based calibrators for canonical multi-class calibration

Calibrated predictors

1. P: a data-generating distribution over $\mathcal{X} \times [L]$
2. (X, Y): a random datapoint from the distribution P
3. $c : \mathcal{X} \to [L]$: a classifier
4. $h : \mathcal{X} \to [0, 1]$: a confidence predictor for the predicted label $c(X)$

Definition (Guo et al., 2017)

A predictor is **confidence calibrated** for P if

$$P(Y = c(X)|h(X)) = h(X)$$

for $(c, h) = (\arg\max_{l \in [L]} h_l(\cdot), \max_{l \in [L]} h_l(\cdot))$.

Calibrated predictors

Example

Suppose the feature space is $\mathcal{X} = \{a, b\}$, with $L = 2$. (e.g.: X is a patient and Y is the disease they are suffering from.) Consider a predictor pair (c, h) and let the values taken by (X, Y, c, h) be as follows:

x	$P(X = x)$	Prediction $c(x)$	Confidence $h(x)$	$P(Y = c(X) \mid X = x)$
a	0.5	1	0.6	0.2
b	0.5	2	0.6	1.0

Confidence calibration: $P(Y = c(X) \mid h(X) = 0.6) = 0.5[P(Y = 1 \mid X = a) + P(Y = 2 \mid X = b)] = 0.5(0.2 + 1) = 0.6.$

'Among all patients who have probability 0.6 of having some unspecified disease, the fraction who have that unspecified disease is also 0.6.'

However, for either $X = a$ or $X = b$, the probabilistic claim of 0.6 bears no correspondence with reality.

Calibrated predictors

'What if a patient wants to know the probability of having disease D among patients who were predicted to have disease D with confidence 0.6?'

Definition

A predictor is **top-label calibrated** for P if

$$P(Y = c(X)|h(X), c(X)) = h(X)$$

1. The expected calibration error (ECE) associated with confidence calibration is defined as $\text{conf-ECE}(c, h) := \mathbb{E}_X|P(Y = c(X) \mid h(X)) - h(X)|$.
2. We define top-label-ECE (TL-ECE) in an analogous fashion, but also condition on $c(X)$:
$\mathrm{TL} - \mathrm{ECE}(c, h) := \mathbb{E}_X|P(Y = c(X) \mid c(X), h(X)) - h(X)|$.

The predictor in Example has $\text{conf-ECE}(c, h) = 0$. However, it has $\mathrm{TL} - \mathrm{ECE}(c, h) = 0.4$, revealing its miscalibration.

Calibrated predictors

For a given class l and bin B_b, define

$$\Delta_{b,l} := |\widehat{P}(Y = c(X) \mid c(X) = l, h(X) \in B_b) - \widehat{\mathbb{E}}[h(X) \mid c(X) = l, h(X) \in B_b]|$$

where $\hat{P}, \hat{\mathbb{E}}$ refer to the empirical distribution of the test data.

The overall miscalibration is then
$\Delta_b :=$ Weighted-average $(\Delta_{b,l}) = \sum_{l \in [L]} \widehat{P}(c(X) = l \mid h(X) \in B_b)\Delta_{b,l}$.

Calibrated predictors

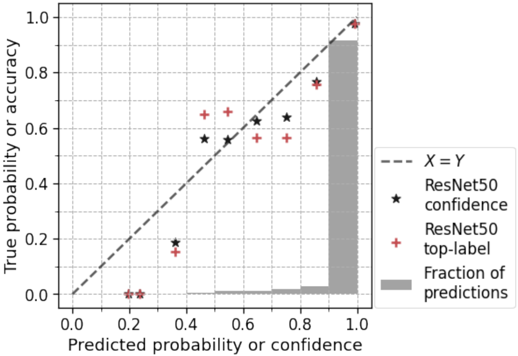


Figure: Confidence reliability diagram (points marked ★) and top-label reliability diagram (points marked +) for a ResNet-50 model on the CIFAR-10 dataset. The **gray bars** denote the fraction of predictions in each bin. The confidence reliability diagram (mistakenly) suggests better calibration than the top-label reliability diagram.

Calibrated predictors

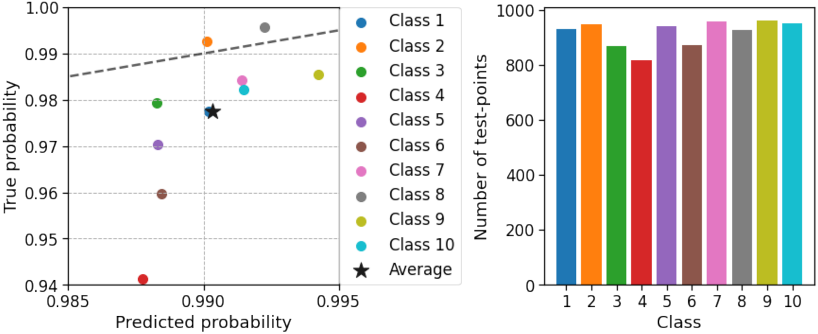


Figure: Class-wise and zoomed-in version of Figure 1 for bin 10. The ★ markers are in the same position as Figure 1, and denote the average predicted and true probabilities. The colored points denote the predicted and true probabilities when seen class-wise. The histograms on the right show the number of test points per class within bin 10.

Calibrated predictors

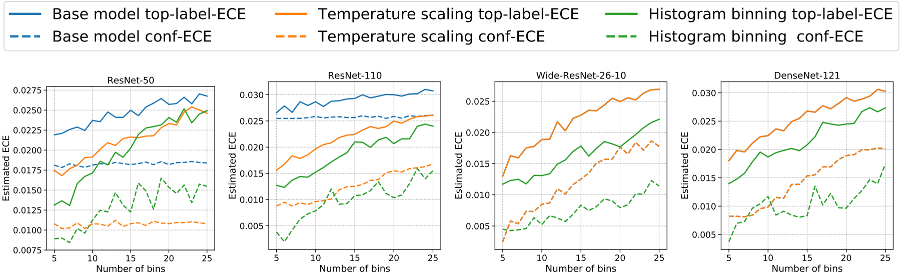


Figure: Conf-ECE (dashed lines) and TL-ECE (solid lines) of four deep nets on CIFAR-10, as well as with recalibration using histogram binning and temperature scaling.

1. The TL-ECE is often 2–3 times larger than the conf-ECE.
2. Top-label histogram binning typically performs better than temperature scaling.

Calibrated predictors

Other 'Multiclass-to-binary' (or M2B) calibration notions

Definition (Kull et al., 2017)

A predictor $\boldsymbol{h} = (h_1, h_2, \cdots, h_L)$ is **class-wise calibrated** if

$$\forall l \in [L], \qquad P\left(Y = l \mid h_l(X)\right) = h_l(X).$$

Definition ((Kartik) Gupta et al. 2021)

For some $l \in [L]$, let $c^{(l)} : \mathcal{X} \to [L]$ denote the l-th highest class prediction, and let $h^{(l)} : \mathcal{X} \to [L]$ denote the confidence associated with it. For a given $K \leq L$, **top-K-confidence calibration** holds if

$$\forall k \in [K], \qquad P\left(Y = c^{(k)}(X) \mid h^{(k)}(X)\right) = h^{(k)}(X).$$

* Special case: $c = c^{(1)}, h = h^{(1)}$.

Calibrated predictors

Other 'Multiclass-to-binary' (or M2B) calibration notions

Definition ((Chirag) Gupta et al. 2021)

Similarly, **top-K-label calibration** is defined by

$$\forall k \in [K], P\left(Y = c^{(k)}(X) \mid h^{(k)}(X), c^{(k)}(X)\right) = h^{(k)}(X)$$

Calibration notion	Quantifier	Prediction (pred(X))	Binary calibration statement
Confidence	-	$h(X)$	$P(Y = c(X) \mid \text{pred}(X)) = h(X)$
Top-label	-	$c(X), h(X)$	$P(Y = c(X) \mid \text{pred}(X)) = h(X)$
Class-wise	$\forall l \in [L]$	$h_l(X)$	$P(Y = l \mid \text{pred}(X)) = h_l(X)$
Top-K-confidence	$\forall k \in [K]$	$h^{(k)}(X)$	$P(Y = c^{(k)}(X) \mid \text{pred}(X)) = h^{(k)}(X)$
Top-K-label	$\forall k \in [K]$	$c^{(k)}(X), h^{(k)}(X)$	$P(Y = c^{(k)}(X) \mid \text{pred}(X)) = h^{(k)}(X)$

Table: Multiclass-to-binary (M2B) notions verify one or more binary calibration statements. The statements in the rightmost column are required to hold almost surely.

Calibrated predictors

'Multiclass-to-binary' (or M2B) notions of calibration

Each binary calibration requirement corresponds to verifying if the distribution of Y, conditioned on some prediction pred(X), satisfies **a single binary calibration claim** associated with pred(X).

In **canonical calibration** (Widmann et al., 2019), the conditioning occurs on the L-dimensional prediction vector $\mathrm{pred}(X) = \mathbf{h}(X)$. After conditioning, the L statements $P(Y = l \mid \mathrm{pred}(X)) = h_l(X)$ should **simultaneously** be true.

Non-M2B notions of calibration are **harder** to achieve due to the conditioning on a multi-dimensional prediction. For the same reason, it is perhaps easier for humans to interpret binary calibration when making decisions.

Calibrated predictors

An example of the philosophy of M2B calibration

	$g(a)$	$g(b)$	$g(c)$	$g(d)$	$g(e)$	$g(f)$
$g_1(\cdot)$	0.1	0.6	0.2	0.0	0.0	0.9
$g_2(\cdot)$	0.0	0.0	0.7	0.1	0.1	0.1
$g_3(\cdot)$	0.6	0.1	0.0	0.1	0.8	0.0
$g_4(\cdot)$	0.3	0.3	0.1	0.8	0.1	0.0
$g(\cdot)$	0.6	0.6	0.7	0.8	0.8	0.9
$c(\cdot)$	3	1	2	4	3	1
Y	3	4	2	1	4	1

(a) Predictions of a fixed base model $\mathbf{g} : \mathcal{X} \to \Delta^3$ on calibration/test data $\mathcal{D} = \{(a, 3), (b, 4), \ldots, (f, 1)\}$.

$g(\cdot)$	0.6	0.6	0.7	0.8	0.8	0.9
$Y = c(\cdot)$	1	0	1	0	0	1

(b) Confidence calibration

$c(\cdot) = 1$

$g(\cdot)$	0.6	0.9
$Y = c(\cdot)$	0	1

$c(\cdot) = 2$

$g(\cdot)$	0.7
$Y = c(\cdot)$	1

$c(\cdot) = 3$

$g(\cdot)$	0.6	0.8
$Y = c(\cdot)$	1	0

$c(\cdot) = 4$

$g(\cdot)$	0.8
$Y = c(\cdot)$	0

(c) Top-label calibration

$g_1(\cdot)$	0.1	0.6	0.2	0.0	0.0	0.9
$Y = 1$	0	0	0	1	0	1

$g_2(\cdot)$	0.0	0.0	0.7	0.1	0.1	0.1
$Y = 2$	0	0	1	0	0	0

$g_3(\cdot)$	0.6	0.1	0.0	0.1	0.8	0.0
$Y = 3$	1	0	0	0	0	0

$g_4(\cdot)$	0.3	0.3	0.1	0.8	0.1	0.0
$Y = 4$	0	1	0	0	1	0

(d) Class-wise calibration

$g(a)$	$Y(a)$
0.1	0.0
0.0	0.0
0.6	1.0
0.3	0.0

$g(b)$	$Y(b)$
0.2	0.0
0.7	0.0
0.0	0.0
0.1	1.0

...
..
..
..
..

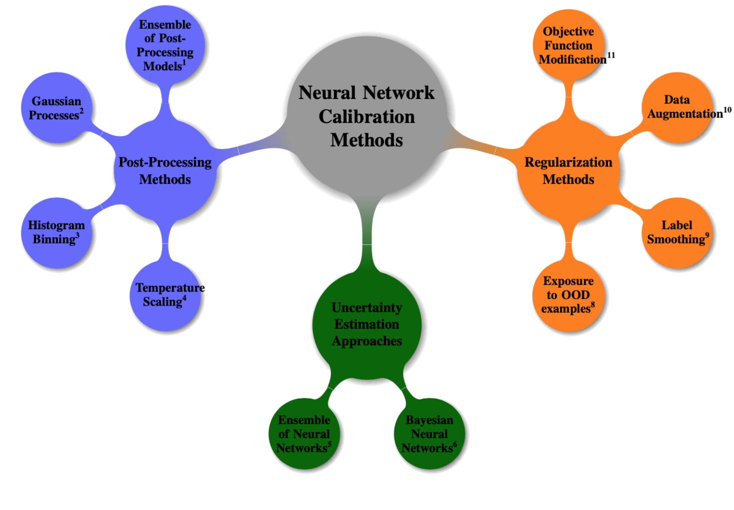
$g(f)$	$Y(f)$
0.9	1.0
0.1	0.0
0.0	0.0
0.0	0.0

(e) Canonical calibration

Figure: Illustrative example for the M2B notions. The numbers in plot (a) correspond to the predictions made by $\mathbf{g}$ on a dataset $\mathcal{D}$. If $\mathcal{D}$ were a test set, plots (b–e) show how it should be used to verify if $\mathbf{g}$ satisfies the corresponding notion of calibration. Consequently, we argue that if $\mathcal{D}$ were a calibration set, and we want to achieve one of the notions (b–e), then the data shown in the corresponding plots should be the data used to calibrate $\mathbf{g}$ as well.

Calibration algorithms for ‘M2B’ calibrators

(A Survey of Uncertainty in Deep Neural Networks, Gawlikowski, 2021)
Calibration methods in neural networks



[1] [48] [2] [47] [3] [276] [4] [15], [68], [274] [5] [31], [275] [6] [272], [209], [273], [274], [16] [8] [277] [9] [268], [270]
[10] [278], [279], [234], [280] [11] [269], [266], [11], [271], [279]

Calibration algorithms for ‘M2B’ calibrators

The goal of **post-hoc calibration** is to use some given calibration data $\mathcal{D} = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)\} \in (\mathcal{X} \times [L])^n$, typically data on which $\mathbf{g}$ was not trained, to recalibrate $\boldsymbol{g}$. In practice, the calibration data is usually the same as the validation data.

Motivation:

1. We verify if $\boldsymbol{g}$ is calibrated on a certain dataset based on some M2B notion of calibration
2. We split the test data into a number of sub-datasets, each of which are used to verify one of the binary calibration claims.

Calibration algorithms for 'M2B' calibrators

Algorithm 5: Post-hoc calibrator for a given M2B calibration notion $\mathcal{C}$

Input: Base (miscalibrated) multiclass predictor $\mathbf{g}$, calibration data $\mathcal{D} = (X_1, Y_1), \ldots, (X_n, Y_n)$, binary calibrator $\mathcal{A}_{\{0,1\}} : [0,1]^{\mathcal{X}} \times (\mathcal{X} \times \{0,1\})^\star \to [0,1]^{\mathcal{X}}$

$K \leftarrow$ number of distinct calibration claims that $\mathcal{C}$ verifies;
for *each claim* $k \in [K]$ **do**
    From $\mathbf{g}$, infer $(\tilde{c}, \tilde{g}) \leftarrow$ (label-predictor, probability-predictor) corresponding to claim k;
    $\mathcal{D}_k \leftarrow \{(X_i, Z_i)\}$, where $Z_i \leftarrow \mathbb{1}\{Y_i = \tilde{c}(X_i)\}$;
    if *conditioning does not include class prediction* $\tilde{c}$ **then**
        — (confidence, top-K-confidence, and class-wise calibration) —
        $h_k \leftarrow \mathcal{A}_{\{0,1\}}(\tilde{g}, \mathcal{D}_k)$;
    end
    else
        — (top-label and top-K-label calibration) —
        for $l \in [L]$ **do**
            $\mathcal{D}_{k,l} \leftarrow \{(X_i, Z_i) \in \mathcal{D}_k : \tilde{c}(X_i) = l)\}$;
            $h_{k,l} \leftarrow \mathcal{A}_{\{0,1\}}(\tilde{g}, \mathcal{D}_{k,l})$;
        end
        $h_k(\cdot) \leftarrow h_{k,\tilde{c}(\cdot)}(\cdot)$ (h_k predicts $h_{k,l}(x)$ if $\tilde{c}(x) = l$);
    end
end
— (the new predictor replaces each $\tilde{g}$ with the corresponding h_k) —
return (label-predictor, h_k) corresponding to each claim $k \in [K]$;

Calibration algorithms for ‘M2B’ calibrators

‘For every class $l \in [L]$, $P(Y = l \mid c(X) = l, h(X)) = h(X)$.’
Binary calibrator $\mathcal{A}_{\{0,1\}} : [0,1]^{\mathcal{X}} \times (\mathcal{X} \times \{0,1\})^{\star} \to [0,1]^{\mathcal{X}}$, base multiclass predictor $\mathbf{g} : \mathcal{X} \to \Delta_{L-1}$, calibration data $\mathcal{D} = (X_1, Y_1), \ldots, (X_n, Y_n)$.

Algorithm 1: Top-label calibrator

$c \leftarrow$ classifier or top-class based on $\mathbf{g}$;
$g \leftarrow$ top-class-probability based on $\mathbf{g}$;
for $l \leftarrow 1$ ***to*** L **do**
  $\mathcal{D}_l \leftarrow \{(X_i, \mathbb{1}\{Y_i = l\}) : c(X_i) = l)\}$;
  $h_l \leftarrow \mathcal{A}_{\{0,1\}}(g, \mathcal{D}_l)$;
end
$h(\cdot) \leftarrow h_{c(\cdot)}(\cdot)$ (predict $h_l(x)$ if $c(x) = l$);
return (c, h);

---

* Examples of $\mathcal{A}_{\{0,1\}}$ are histogram binning (Zadrozny and Elkan, 2001), isotonic regression (Zadrozny and Elkan, 2002), and Platt scaling (Platt, 1999).
* The features in $\mathcal{D}_l$ are the X_i's for which $c(X_i) = l$, and the labels are $\mathbb{I}\{Y_i = l\}$.
*The top-label predictor $\mathbf{c}$ does not change in this process. Thus the accuracy of (c, h) is the same as the accuracy of $\mathbf{g}$ irrespective of $\mathcal{A}_{\{0,1\}}$.

Calibration algorithms for 'M2B' calibrators

Binary calibrator $\mathcal{A}_{\{0,1\}} : [0,1]^{\mathcal{X}} \times (\mathcal{X} \times \{0,1\})^{\star} \to [0,1]^{\mathcal{X}}$, base multiclass predictor $\mathbf{g} : \mathcal{X} \to \Delta_{L-1}$, calibration data $\mathcal{D} = (X_1, Y_1), \ldots, (X_n, Y_n)$.

Algorithm 1: Top-label calibrator

$c \leftarrow$ classifier or top-class based on $\mathbf{g}$;
$g \leftarrow$ top-probability based on $\mathbf{g}$;
for $l \leftarrow 1$ ***to*** L **do**
    $\mathcal{D}_l \leftarrow \{(X_i, \mathbb{1}\{Y_i = l\}) : c(X_i) = l)\}$;
    $h_l \leftarrow \mathcal{A}_{\{0,1\}}(g, \mathcal{D}_l)$;
end
$h(\cdot) \leftarrow h_{c(\cdot)}(\cdot)$ (predict $h_l(x)$ if $c(x) = l$);
return (c, h);

Algorithm 2: Class-wise calibrator

Write $\mathbf{g} = (g_1, g_2, \ldots, g_L)$;
for $l \leftarrow 1$ ***to*** L **do**
    $\mathcal{D}_l \leftarrow \{(X_i, \mathbb{1}\{Y_i = l\}) : i \in [n]\}$;
    $h_l \leftarrow \mathcal{A}_{\{0,1\}}(g_l, \mathcal{D}_l)$;
end
return $(h_1, h_2, \ldots, h_L)$;

Algorithm 3: Confidence calibrator

$c \leftarrow$ classifier or top-class based on $\mathbf{g}$;
$g \leftarrow$ top-class-probability based on $\mathbf{g}$;
$\mathcal{D}' \leftarrow \{(X_i, \mathbb{1}\{Y_i = c(X_i)\}) : i \in [n]\}$;
$h \leftarrow \mathcal{A}_{\{0,1\}}(g, \mathcal{D}')$;
return (c, h);

Algorithm 4: Normalized calibrator

Write $\mathbf{g} = (g_1, g_2, \ldots, g_L)$;
for $l \leftarrow 1$ ***to*** L **do**
    $\mathcal{D}_l \leftarrow \{(X_i, \mathbb{1}\{Y_i = l\}) : i \in [n]\}$;
    $\widetilde{h}_l \leftarrow \mathcal{A}_{\{0,1\}}(g_l, \mathcal{D}_l)$;
end
Normalize: for every $l \in [L]$,
  $h_l(\cdot) := \widetilde{h}_l(\cdot) / \sum_{k=1}^{L} \widetilde{h}_k(\cdot)$;
return $(h_1, h_2, \ldots, h_L)$;

Experiments: M2B calibration with histogram binning

'Multiclass-to-binary' (or M2B) notions of calibration

Metric	Dataset	Architecture	Base	TS	VS	DS	N-HB	TL-HB
Top-label-ECE	CIFAR-10	ResNet-50	0.025	0.022	0.020	0.019	**0.018**	0.020
		ResNet-110	0.029	0.022	0.021	0.021	**0.020**	0.021
		WRN-26-10	0.023	0.023	0.019	0.021	**0.012**	0.018
		DenseNet-121	0.027	0.027	0.020	0.020	**0.019**	0.021
	CIFAR-100	ResNet-50	0.118	0.114	0.113	0.322	**0.081**	0.143
		ResNet-110	0.127	0.121	0.115	0.353	**0.093**	0.145
		WRN-26-10	0.103	0.103	0.100	0.304	**0.070**	0.129
		DenseNet-121	0.110	0.110	0.109	0.322	**0.086**	0.139
Top-label-MCE	CIFAR-10	ResNet-50	0.315	0.305	0.773	0.282	0.411	**0.107**
		ResNet-110	0.275	0.227	0.264	0.392	0.195	**0.077**
		WRN-26-10	0.771	0.771	0.498	0.325	0.140	**0.071**
		DenseNet-121	0.289	0.289	0.734	0.294	0.345	**0.087**
	CIFAR-100	ResNet-50	0.436	0.300	**0.251**	0.619	0.397	0.291
		ResNet-110	0.313	**0.255**	0.277	0.557	0.266	0.257
		WRN-26-10	0.273	**0.255**	0.256	0.625	0.287	0.280
		DenseNet-121	0.279	**0.231**	0.235	0.600	0.320	0.289

Figure: Top-label-ECE and top-label-MCE for deep nets (called 'Base' above) and various post-hoc calibrators: temperature-scaling (TS), vector-scaling (VS), Dirichlet-scaling (DS), top-label-HB or Algorithm 1 (TL-HB), and normalized-HB or Algorithm 4 (N-HB). Best performing method in each row is in bold.

Experiments: M2B calibration with histogram binning

$$\mathrm{TL-MCE}(c,h) := \max_{l\in[L]} \sup_{r\in\mathrm{Range}(h)} \left|P(Y=l \mid c(X)=l, h(X)=r) - r\right|$$

$$\mathrm{CW-ECE}(c,\mathbf{h}) := L^{-1}\sum_{l=1}^{L} \mathbb{E}_X \left|P\left(Y=l \mid h_l(X)\right) - h_l(X)\right|$$

Metric	**Dataset**	**Architecture**	**Base**	**TS**	**VS**	**DS**	**N-HB**	**CW-HB**
Class-wise-ECE $\times 10^2$	CIFAR-10	ResNet-50	0.46	0.42	0.35	0.35	0.50	**0.28**
		ResNet-110	0.59	0.50	0.42	0.38	0.53	**0.27**
		WRN-26-10	0.44	0.44	0.35	0.39	0.39	**0.28**
		DenseNet-121	0.46	0.46	**0.36**	**0.36**	0.48	**0.36**
	CIFAR-100	ResNet-50	0.22	0.20	0.20	0.66	0.23	**0.16**
		ResNet-110	0.24	0.23	0.21	0.72	0.24	**0.16**
		WRN-26-10	0.19	0.19	0.18	0.61	0.20	**0.14**
		DenseNet-121	0.20	0.21	0.19	0.66	0.24	**0.16**

Figure: Class-wise-ECE for deep nets and various post-hoc calibrators. All methods are the same as in Table 2, except top-label-HB is replaced with class-wise-HB or Algorithm 2 (CW-HB). Best performing method in each row is in bold.

Experiments: M2B calibration with histogram binning

1. For TL-ECE, N-HB is the best performing method for both CIFAR-10 and CIFAR-100. It could be because the data splitting scheme of the TL-calibrator (line 4 of Algorithm 1) splits datasets across the predicted classes, and some classes in CIFAR-100 occur very rarely.
2. For CW-ECE, TL-HB is the best performing method across the two datasets and all four architectures. The N-HB method which has been used in many CW-ECE baseline experiments performs terribly.

Distribution-free top-label calibration using histogram binning

Algorithm and theoretical guarantees:

Definition 1 (Marginal and conditional top-label calibration). Let $\varepsilon, \alpha \in (0,1)$ be some given levels of approximation and failure respectively. An algorithm $\mathcal{A} : (\mathbf{g}, \mathcal{D}) \mapsto (c, h)$ is

(a) (ε, α)-marginally top-label calibrated if for every distribution P over $\mathcal{X} \times [L]$,

$$P\Big(|P(Y = c(X) \mid c(X), h(X)) - h(X)| \leqslant \varepsilon \Big) \geqslant 1 - \alpha. \tag{10}$$

(b) (ε, α)-conditionally top-label calibrated if for every distribution P over $\mathcal{X} \times [L]$,

$$P\Big(\forall\, l \in [L], r \in \mathrm{Range}(h), |P(Y = c(X) \mid c(X) = l, h(X) = r) - r| \leqslant \varepsilon \Big) \geqslant 1 - \alpha. \tag{11}$$

1. Probabilities are taken over the test point $(X, Y) \sim P$, the calibration data $\mathcal{D} \sim P^n$ and any other inherent algorithmic randomness in $\mathcal{A}$.
2. Marginal calibration: with high probability, on average over the distribution of $\mathcal{D}, X, P(Y = c(X) \mid c(X), h(X))$.
3. Conditional calibration (strictly stronger): It requires the deviation to be at most ε for every possible prediction (l, r), including rare ones, not just on average over predictions (see e.g., medical settings...)

Distribution-free top-label calibration using histogram binning

Algorithm 8: Top-label histogram binning

Input: Base multiclass predictor $\mathbf{g}$, calibration data $\mathcal{D} = (X_1, Y_1), \dots, (X_n, Y_n)$
Hyperparameter: # points per bin $k \in \mathbb{N}$ (say 50), tie-breaking parameter $\delta > 0$ (say 10^{-10})
Output: Top-label calibrated predictor (c, h)

```
c ← classifier or top-class based on g;
g ← top-class-probability based on g;
for l ← 1 to L do
    D_l ← {(X_i, 𝟙{Y_i = l}) : c(X_i) = l}  and  n_l ← |D_l|;
    h_l ← Binary-histogram-binning(g, D_l, ⌊n_l/k⌋, δ);
end
h(·) ← h_{c(·)}(·);
return (c, h);
```

Distribution-free top-label calibration using histogram binning

Formal algorithm and theoretical guarantees

Theorem 1. *Fix hyperparameters $\delta > 0$ (arbitrarily small) and points per bin $k \geqslant 2$, and assume $n_l \geqslant k$ for every $l \in [L]$. Then, for any $\alpha \in (0,1)$, Algorithm 8 is (ε_1, α)-marginally and (ε_2, α)-conditionally top-label calibrated for*

$$\varepsilon_1 = \sqrt{\frac{\log(2/\alpha)}{2(k-1)}} + \delta, \qquad \textit{and} \qquad \varepsilon_2 = \sqrt{\frac{\log(2n/k\alpha)}{2(k-1)}} + \delta. \tag{12}$$

Further, for any distribution P over $\mathcal{X} \times [L]$, we have $P(\text{TL-ECE}(c,h) \leqslant \varepsilon_2) \geqslant 1-\alpha$, and $\mathbb{E}\left[\text{TL-ECE}(c,h)\right] \leqslant \sqrt{1/2k} + \delta$.

1. $\widetilde{O}(1/\sqrt{k})$ dependence.
2. The proof is a multiclass top-label adaption of (Gupta and Ramdas, 2021).
3. Since δ can be chosen to be arbitrarily small, setting $k = 50$ gives roughly $\mathbb{E}_{\mathcal{D}}[\mathrm{TL-ECE}(h)] \leqslant 0.1$

Distribution-free top-label calibration using histogram binning

Remark

Gupta and Ramdas [2021] proved a more general result for general ℓ_p-ECE bounds. Similar results can also be derived for the suitably defined ℓ_p-TL-ECE. Additionally, it can be shown that with probability $1-\alpha$, the TL-MCE of (c,h) is bounded by ε_2.

Top-label calibration using histogram binning

n_l is the number of points predicted as class l.

Algorithm 8: Top-label histogram binning

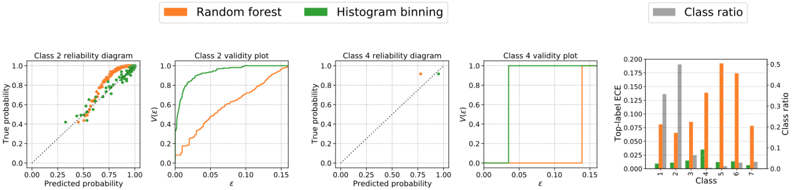
Input: Base multiclass predictor $\mathbf{g}$, calibration data $\mathcal{D} = (X_1, Y_1), \ldots, (X_n, Y_n)$
Hyperparameter: # points per bin $k \in \mathbb{N}$ (say 50), tie-breaking parameter $\delta > 0$ (say 10^{-10})
Output: Top-label calibrated predictor (c, h)

1 $c \leftarrow$ classifier or top-class based on $\mathbf{g}$;
2 $g \leftarrow$ top-class-probability based on $\mathbf{g}$;
3 **for** $l \leftarrow 1$ ***to*** L **do**
4     $\mathcal{D}_l \leftarrow \{(X_i, \mathbb{1}\{Y_i = l\}) : c(X_i) = l)\}$ and $n_l \leftarrow |\mathcal{D}_l|$;
5     $h_l \leftarrow$ Binary-histogram-binning$(g, \mathcal{D}_l, \lfloor n_l/k \rfloor, \delta)$;
6 **end**
7 $h(\cdot) \leftarrow h_{c(\cdot)}(\cdot)$;
8 **return** (c, h);

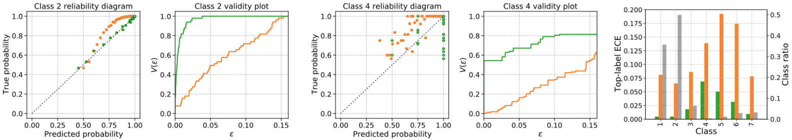
1. The function called in line 5 is Algorithm 2 of (Gupta and Ramdas, 2021).
2. (Gupta and Ramdas, 2021) suggests to choose the number of bins by fixing the number of points per bin, i.e., k in the algorithm.

Top-label calibration using histogram binning

Recalibration of a random forest using histogram binning on the class imbalanced COVTYPE-7 dataset (class 2 is roughly 100 times likelier than class 4).



(a) Top-label histogram binning (Algorithm 8) with $k = 100$ points per bin. Class 4 has only 183 calibration points. Algorithm 8 adapts and uses only a single bin to ensure that the TL-ECE on class 4 is comparable to the TL-ECE on class 2. Overall, the random forest classifier has significantly higher TL-ECE for the least likely classes (4, 5, and 6), but the post-calibration TL-ECE using binning is quite uniform.



(b) Histogram binning with $B = 50$ bins for every class. Compared to Figure 4a, the post-calibration TL-ECE for the most likely classes decreases while the TL-ECE for the least likely classes increases.

Binning for canonical multiclass calibration

Binning algorithms do not obviously extend for multiclass classification. Although their description is general for $L \geq 3$, some algorithms might only work well for reasonably small L, say if $L \leq 5$.
Denote $\mathbf{Y}$ as a 1-hot output vector, i.e., $\mathbf{Y}_i = \mathbf{e}_{Y_i} \in \Delta_{L-1}$. Here $\mathbf{e}_l$ corresponds to the l-th canonical basis vector in $\mathbb{R}^d$. Recall that for a canonically calibrated predictor $\mathbf{h} = (h_1, \cdots, h_L)$,

$$P(Y = l \mid \mathbf{h}(X)) = h_l(X) \text{ for every } l \in [L] \iff \mathbb{E}[\mathbf{Y} \mid \mathbf{h}(X)] = \mathbf{h}(X)$$

Canonical calibration implies class-wise calibration:

Proposition

If $\mathbb{E}[\boldsymbol{Y} \mid \boldsymbol{h}(X)] = \boldsymbol{h}(X)$, then for every $l \in [L]$, $P(Y = l \mid h_l(X)) = h_l(X)$. There exist predictors that are class-wise calibrated but not canonically calibrated (Vaicenavicius et al., 2019, Example 1).

Binning for canonical multiclass calibration

The binning scheme

Denote $\mathbf{g} : \mathcal{X} \to \Delta_{L-1}$ as the base model and $\mathbf{h} : \mathcal{X} \to \Delta_{L-1}$ as the model learned using some post-hoc canonical calibrator.
First, we partition Δ_{L-1} into $B \geq 1$ bins. We denote the binning scheme as $\mathcal{B} : \Delta_{L-1} \to [B]$ where $\mathcal{B}(\mathbf{s})$ corresponds to the bin to which $\mathbf{s} \in \Delta_{L-1}$ belongs.
To learn $\mathbf{h}$, we get the data indices for each bin index $b \in [B]$,

$$T_b := \{i : \mathcal{B}\left(\mathbf{g}\left(X_i\right)\right) = b\}, n_b = |T_b|$$

Then we compute the following estimates for the label probabilities for each bin index $(l, b) \in [L] \times [B]$:

$$\widehat{\Pi}_{l,b} := \frac{\sum_{i \in T_b} \mathbb{I}\left\{Y_i = l\right\}}{n_b} \text{ if } n_b > 0, \qquad \text{else } \widehat{\Pi}_{l,b} = 1/B$$

Then for every $l \in [L]$, set $h_l(x) = \widehat{\Pi}_{l,\mathcal{B}(x)}$.

Binning for canonical multiclass calibration

The binning scheme

Denote $\mathbf{g} : \mathcal{X} \to \Delta_{L-1}$ as the base model and $\mathbf{h} : \mathcal{X} \to \Delta_{L-1}$ as the model learned using some post-hoc canonical calibrator.
Then for every $l \in [L], h_l(x) = \widehat{\Pi}_{l,\mathcal{B}(x)}$.

* Using a **multinomial concentration inequality** (Devroye, 1983, Qian et al., 2020, Weissman et al., 2003), calibration guarantees can be shown for the learned $\mathbf{h}$. (Podkopaev and Ramdas 2021, Theorem 3) show such a result using the **Bretagnolle-Huber-Carol inequality**. These bounds decay as $1/n_b$ or $1/\sqrt{n_b}$.

Binning for canonical multiclass calibration

Sierpinski binning

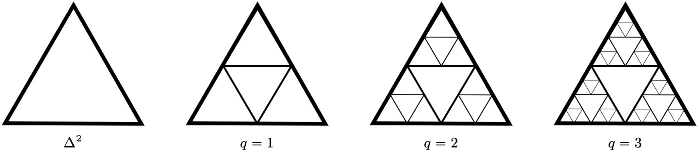


Figure 10: Sierpinski binning for $L = 3$. The leftmost triangle represents the probability simplex Δ^2. Sierpinski binning divides Δ^2 recursively based on a depth parameter $q \in \mathbb{N}$.

Binning for canonical multiclass calibration

Sierpinski binning for $L = 3$:

Given an $x \in \mathcal{X}$, let $\mathbf{s} = \mathbf{g}(x)$. For $q = 1$, the number of bins is $B = 4$. The binning scheme $\mathcal{B}$ is defined as

$$\mathcal{B}(\mathbf{s}) = \begin{cases} 1 & \text{if } s_1 > 0.5 \\ 2 & \text{if } s_2 > 0.5 \\ 3 & \text{if } s_3 > 0.5 \\ 4 & \text{otherwise.} \end{cases} \tag{1}$$

Since $s_1 + s_2 + s_3 + s_4 = 1$, only one of the conditions above can be true.
* If a finer resolution of Δ_2 is desired, $\mathcal{B}$ can be increased by further dividing the partitions above.

Binning for canonical multiclass calibration

Sierpinski binning for $L = 3$:

Each partition is itself a triangle; thus each triangle can be mapped to Δ_2 to recursively define the sub-partitioning. For $i \in [4]$, define the bins $b_i = \{\mathbf{s} : \mathcal{B}(\mathbf{s}) = i\}$. Consider the bin b_1. Let us 'reparameterize' it as $(t_1, t_2, t_3) = (2s_1 - 1, 2s_2, 2s_3)$. It can be verified that

$$
\begin{aligned}
b_1 &= \{(t_1, t_2, t_3) : s_1 > 0.5\} \\
&= \{(t_1, t_2, t_3) : t_1 + t_2 + t_3 = 1, t_1 \in (0, 1], t_2 \in [0, 1), t_3 \in [0, 1)\}.
\end{aligned}
$$

Binning for canonical multiclass calibration

Sierpinski binning for $L = 3$:

Based on this reparameterization, we can recursively sub-partition b_1 as per the scheme (1), replacing s with t. Such reparameterizations can be defined for each of the bins defined in (1):

$$b_2 = \{(s_1, s_2, s_3) : s_2 > 0.5\} : (t_1, t_2, t_3) = (2s_1, 2s_2 - 1, 2s_3),$$
$$b_3 = \{(s_1, s_2, s_3) : s_3 > 0.5\} : (t_1, t_2, t_3) = (2s_1, 2s_2, 2s_3 - 1),$$
$$b_4 = \{(s_1, s_2, s_3) : s_i \leq 0.5 \text{ for all } i\} : (t_1, t_2, t_3) = (1 - 2s_1, 1 - 2s_2, 1 - 2s_3),$$

If at every depth, we sub-partition all bins except the corresponding b_4 bins, then it can be shown using simple algebra that the total number of bins is $\left(3^{q+1} - 1\right)/2$. For example, in the figure above, when $q = 2$, the number of bins is $B = 14$, and when $q = 3$, the number of bins is $B = 40$.

Binning for canonical multiclass calibration

$K = 4, \tau = 0.25$ $K = 5, \tau = 0.2$

Figure 11: Grid-style binning for $L = 3$.

Binning for canonical multiclass calibration

The projection histogram binning

To ensure that each bin contains $\Omega(n/B)$ points, we create the bins using estimated quantiles of $g(X)$.
The learned array T represents the thresholds for the directions given by q.
Each (q_b, T_b) pair corresponds to a hyperplane that 'cuts' Δ_{L-1} into two subsets given by $\left\{x \in \Delta_{L-1} : x^T q_b < T_b\right\}$ and $\left\{x \in \Delta_{L-1} : x^T q_b \geq T_b\right\}$.
The overall partitioning of Δ_{L-1} is created by merging these cuts sequentially.
This defines the binning function $\mathcal{B}$.
By construction, each bin contains at least $\left[\frac{n+1}{B}\right] - 1$ points in its interior. The interior points are then used to estimate the bin biases $\widehat{\Pi}$.
* As suggested by Gupta and Ramdas [2021], we do not include the points X_i that lie on the boundary, i.e., s.t. $\mathbf{g}(X_i)^\top q_s = T_s$ for some $s \in [B]$.

Binning for canonical multiclass calibration

The projection histogram binning

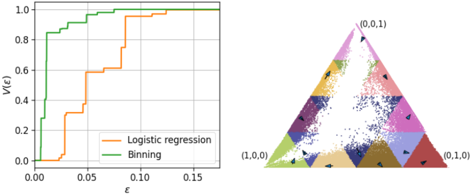
Algorithm 10: Projection histogram binning for canonical calibration

Input: Base multiclass predictor $\mathbf{g} : \mathcal{X} \to \Delta^{L-1}$, calibration data $\mathcal{D} = \{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$
Hyperparameter: number of bins B, unit vectors $q_1, q_2, \dots, q_B \in \mathbb{R}^L$,
Output: Approximately calibrated scoring function $\mathbf{h}$

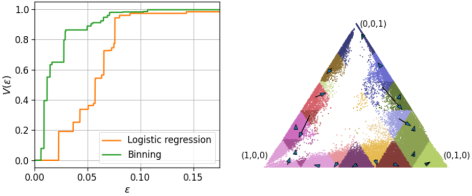
$S \leftarrow \{\mathbf{g}(X_1), \mathbf{g}(X_2), \dots, \mathbf{g}(X_n)\}$;
$T \leftarrow$ empty array of size B;
$c \leftarrow \lfloor \frac{n+1}{B} \rfloor$;
for $b \leftarrow 1$ ***to*** $B - 1$ **do**
  $T_b \leftarrow$ order-statistics(S, q_b, c);
  $S \leftarrow S \backslash \{v \in S : v^T q_b \leqslant T_b\}$;
end
$T_B \leftarrow 1.01$;
$\mathcal{B}(\mathbf{g}(\cdot)) \leftarrow \min\{b \in [B] : \mathbf{g}(\cdot)^T q_b < T_b\}$;
$\widehat{\Pi} \leftarrow$ empty matrix of size $B \times L$;
for $b \leftarrow 1$ ***to*** B **do**
  for $l \leftarrow 1$ ***to*** L **do**
    $\widehat{\Pi}_{b,l} \leftarrow \text{Mean}\{\mathbb{1}\{Y_i = l\} : \mathcal{B}(\mathbf{g}(X_i)) = b \text{ and } \forall s \in [B],\ \mathbf{g}(X_i)^T q_s \neq T_s\}$;
  end
end
for $l \leftarrow 1$ ***to*** L **do**
  $h_l(\cdot) \leftarrow \widehat{\Pi}_{\mathcal{B}(\mathbf{g}(\cdot)), l}$;
end
return h;

Binning for canonical multiclass calibration

Experiments with the COVTYPE dataset



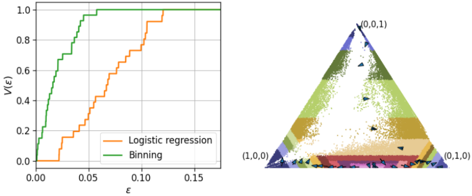
(a) Calibration using Sierpinski binning at depth $q = 2$.



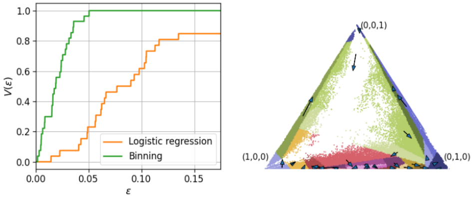
(b) Calibration using grid-style binning with $K = 5$, $\tau = 0.2$.

Binning for canonical multiclass calibration

Experiments with the COVTYPE dataset



(c) Projection-based HB with $B = 27$ projections: $q_1 = -\mathbf{e}_1, q_2 = -\mathbf{e}_2, \ldots, q_4, -\mathbf{e}_1, \ldots$, and so on.



(d) Projection-based HB with $B = 27$ random projections (q_i drawn uniformly from the ℓ_2-unit-ball in $\mathbb{R}^3$).

Summary

1. Confidence calibration is not enough for describing class-wise calibration – Top-label calibration
2. It is better to choose the number of bins by fixing the number of points per bin, i.e., k in the algorithm 8.
3. Some bins defined by these schemes may have very few calibration points n_b, leading to poor estimates $\widehat{\Pi}$ – Projection histogram binning