On Calibration of Modern Neural Networks

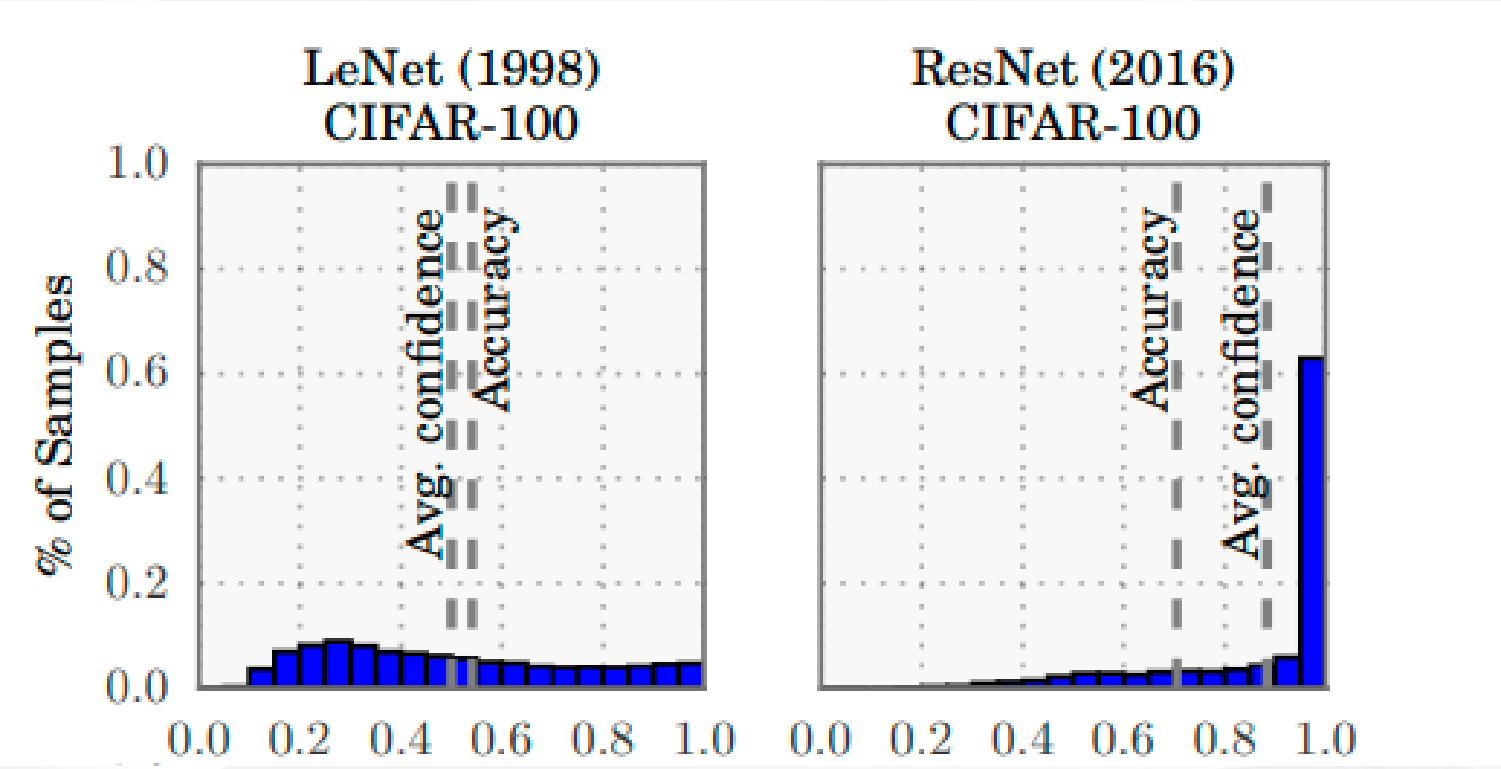
Chuan Guo, Geoff Pleiss, Yu Sun, Kilian Q. Weinberger

Coby Penso

What we'll see

1. Introduction
2. Calibration
3. Calibration Measurements
4. Miscalibration reasons
5. Calibrating Binary Models
6. Calibrating Multiclass Models
7. Experiments

Introduction



Calibration

Problem settings:
The input $X \in \mathcal{X}$ and label $Y \in \mathcal{Y} = \{1, ..., K\}$
are r.v that follow a ground truth joint distribution $\pi(X, Y) = \pi(Y|X)\pi(X)$

Prefect Calibration

Let h neural networks with $h(X) = (\hat{Y}, \hat{P})$.
$\hat{Y}$ - Class prediction. $\hat{P}$ - Confidence (probability of correctness).
Perfect Calibration is when $\hat{P}$ represents a true probability

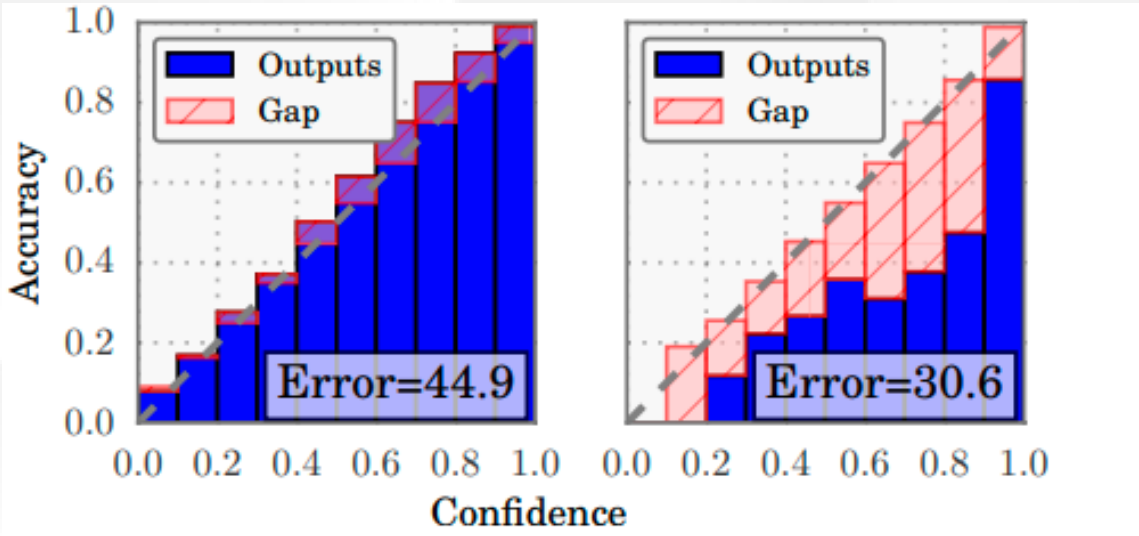
$$P(\hat{Y} = Y|\hat{P} = p) = p, \quad \forall p \in [0, 1]$$

Reliability Diagrams

Visual representation of model calibration.

Algorithm 1 Reliability diagram construction

1: Group prediction into M interval bins
2: Calculate accuracy of each bin.
3: $\mathcal{B}_m$ indices of predictions with confidence falls into the interval $\mathcal{I}_m = (\frac{m-1}{M}, \frac{m}{M}]$
4: $acc(\mathcal{B}_m) = \frac{1}{|\mathcal{B}_m|} \sum_{i \in \mathcal{B}_m} 1_{(\hat{y}_i = y_i)}$
5: $conf(\mathcal{B}_m) = \frac{1}{|\mathcal{B}_m|} \sum_{i \in \mathcal{B}_m} \hat{p}_i$



Expected Calibration Error (ECE)

One notion of miscalibration is the difference in expectation between confidence and accuracy

$$E_{\hat{P}}\left[\left|P\left(\hat{Y}=Y|\hat{p}=p\right)-p\right|\right]$$

ECE is taking a weighted average of the bins' accuracy/confidence difference

$$ECE=\sum_{m=1}^{M}\frac{|\mathcal{B}_m|}{n}\left|acc(\mathcal{B}_m)-conf(\mathcal{B}_m)\right|$$

where n in number of samples and M number of bins.

Maximum Calibration Error (MCE)

In high-risk applications where reliable confidence measures are absolutely necessary, we may wish to minimize the worst-case deviation between confidence and accuracy:

$$\max_{p \in [0,1]} \left| P\left(\hat{Y} = Y | \hat{p} = p\right) - p \right|$$

MCE estimates the deviation

$$MCE = \max_{m \in \{1,\ldots,M\}} |acc(\mathcal{B}_m) - conf(\mathcal{B}_m)|$$

Negative log likelihood

Also called Cross entropy loss.
Given a probabilistic model $\hat{\pi}(Y|X)$ and n samples

$$NLL = -\sum_{i=1}^{n} \log[(\hat{\pi}(y_i|x_i)]$$

NLL minimized iff $\hat{\pi}(Y|X) = \pi(Y|X)$

Miscalibration reasons

Model capacity

Model capacity increased dramatically.
Higher capacity $\rightarrow$ better classification error, worst model calibration.

Batch Normalization

Improves optimization by minimizing distribution shifts in activations.
Enables deep architectures, improves training time, reduce the need for regularization, but, higher miscalibration.

Weight decay

Improves regularization and prevent overfitting.
Less weight decay $\rightarrow$ worst calibration.
More regularization $\rightarrow$ Better calibration

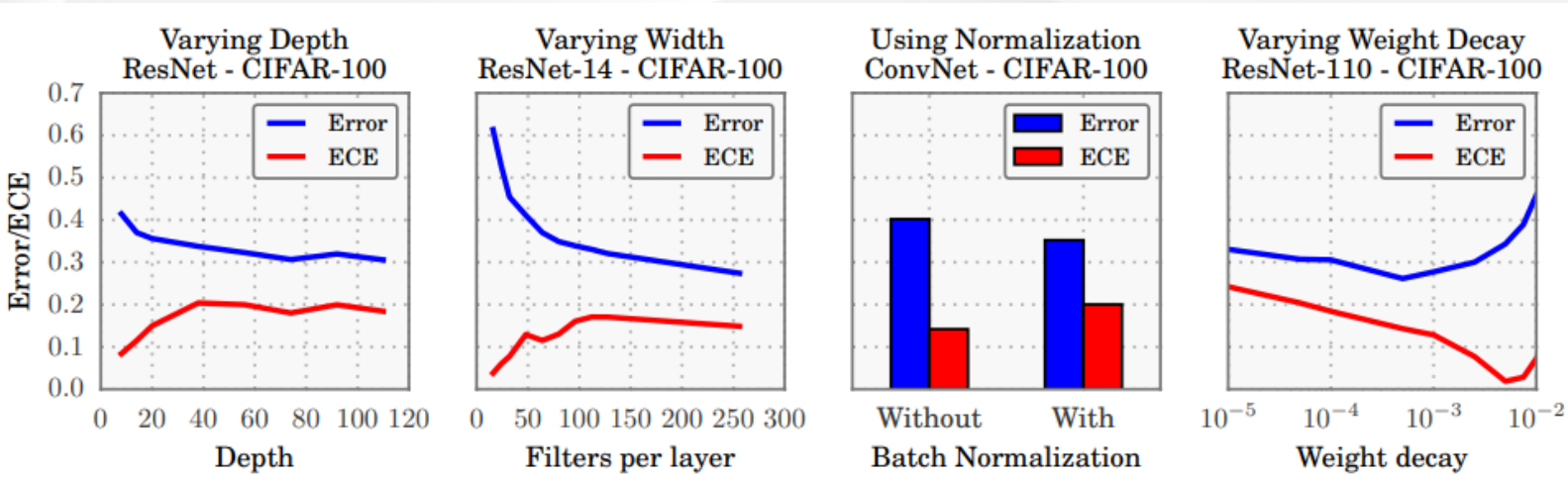
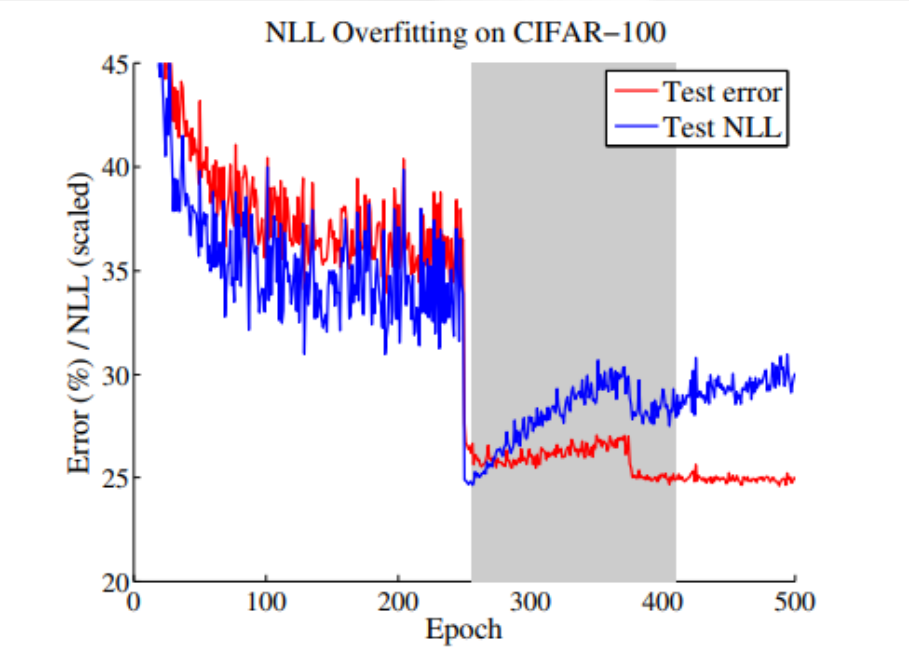


Figure 2. The effect of network depth (far left), width (middle left), Batch Normalization (middle right), and weight decay (far right) on miscalibration, as measured by ECE (lower is better).

Miscalibration reasons

NLL

Overfitting to NLL beneficial to accuracy.
The network learns better classification accuracy at the expense of well-modeled probabilities.



Calibration Methods

Calibrating Binary Models

First, lets face calibration of binary models.

- $\mathcal{Y} = \{0, 1\}$
- Given a sample x_i, $\hat{p}_i$ - the network predicted probability of $y_i = 1$.
- $z_i \in \mathbb{R}$ - logit output
- $\hat{p}_i = \sigma(z_i)$
- **Goal:** produce calibrated probability $\hat{q}_i$

Histogram binning

All uncalibrated $\hat{p}_i$ divided into bins $\mathcal{B}_1, \ldots, \mathcal{B}_M$.
Assign calibrated score θ_m.
if $\hat{p}_i$ assign to $\mathcal{B}_m$ then $\hat{q}_i = \theta_m$
The predictions θ_i are chosen to minimize the bin-wise squared loss:

$$\min_{\theta_1,\ldots,\theta_M} \sum_{m=1}^{M} \sum_{i=1}^{n} 1_{\hat{p}_i \in \mathcal{B}_m} (\theta_m - y_i)^2$$

Given fixed bins bound, θ_m is the average number of positive-class samples in bin $\mathcal{B}_m$

Isotonic Regression

Learn piecewise constant function f:

$$\hat{q}_i = f(\hat{p}_i)$$

Isotonic regression produces $\{$ to minimize the square loss

$$\sum_{i=1}^{n} (f(\hat{p}_i) - y_i)^2$$

The optimization problem:

$$\min_{\substack{M \\ \theta_1,\ldots,\theta_M \\ a_1,\ldots,a_{M+1}}} \sum_{m=1}^{M} \sum_{i=1}^{n} 1_{a_m \leq \hat{p}_i \leq a_{m+1}} (\theta_m - y_i)^2$$

Subject to $0 = a_1 \leq a_2 \leq \ldots \leq a_{M+1} = 1, \quad \theta_1 \leq \theta_2 \leq \ldots \leq \theta_M$

Note: generalization of histogram binning where bin boundaries and bin predictions are jointly optimized.

Bayesian Binning into Quantiles (BBQ)

Extension of histogram binning using Bayesian model averaging.
Marginalize out all possible binning schemes to produce $\hat{q}_i$
Binning scheme s is a pair $(\mathcal{M}, \mathcal{I})$

$$P(\hat{q}_{te}|\hat{p}_{te}, \mathcal{D}) = \sum_{s \in \mathcal{S}} P(\hat{q}_{te}, \mathcal{S} = s|\hat{p}_{te}, \mathcal{D}) = \sum_{s \in \mathcal{S}} P(\hat{q}_{te}|\hat{p}_{te}, \mathcal{S} = s, \mathcal{D}) P(\mathcal{S} = s|\mathcal{D})$$

where $P(\hat{q}_{te}|\hat{p}_{te}, \mathcal{S} = s, \mathcal{D})$ is the calibrated probability using binning scheme s.
Using uniform prior

$$P(\mathcal{S} = s|\mathcal{D}) = \frac{P(\mathcal{D}|\mathcal{S} = s)}{\sum_{s' \in \mathcal{S}} P(\mathcal{D}|\mathcal{S} = s')}$$

Platt scaling

Parametric approach.
Learn scalar parameters $a, b \in \mathbb{R}$.
Outputs:

$$\hat{q}_i = \sigma(az_i + b)$$

a,b optimized using NLL over validation set.
Note: NN parameters are fixed during this stage.

Calibrating Multiclass Models

- $\mathcal{Y} = \{1, ..., K\}, \quad K > 2$
- Given a sample x_i, $\hat{p}_i$ - confidence, $\hat{y}_i$ - prediction
- $z_i \in \mathbb{R}$ - logit output
- $\hat{y}_i = argmax_k(z_i^{(k)})$
- $\hat{(p)}_i$ derived:

$$\sigma(z_i)^{(k)} = \frac{exp(z_i^{(k)}}{\sum_{j=1}^{K} exp(z_i^{(j)})}$$

- **Goal:** produce calibrated probability $\hat{q}_i$

Extension of binning methods

K one-versus-all problem.

- For $k = 1, ..., K$ form binary calibration problem where the label is $1_{(y_i = k)}$
- predicted probability is $\sigma(z_i)^{(k)}$
- Hold K calibration models, each for given class.
- Calibrated probabilities $\left[\hat{q}_i^{(1)}, ..., \hat{q}_i^{(K)}\right]$

$$y_i = argmax \left[\hat{q}_i^{(1)}, ..., \hat{q}_i^{(K)}\right]$$

$$confidence = \frac{max\left[\hat{q}_i^{(1)}, ..., \hat{q}_i^{(K)}\right]}{\sum_{j=1}^{K} \hat{q}_i^{(j)}}$$

Matrix and vector scaling

Two multi-class extentions of Platt scaling.
Matrix scaling applies linear transformation $\mathcal{W}z_i + b$ to the logits:

$$\hat{q}_i = \max_k \sigma_{SM}(\mathcal{W}z_i + b)^{(k)}$$

$$\hat{y}_i' = \operatorname*{argmax}_k (\mathcal{W}z_i + b)^{(k)}$$

$\mathcal{W}, b$ optimized with respect to NLL on validation set.
Disadvantage: Number of parameters grows quadratically with number of classes K.

Vector scaling restricts $\mathcal{W}$ to be diagonal matrix.

Temperature scaling

Simplest extenstion of Platt scaling.
Use single scalar parameter $T > 0$ for all classes.

$$\hat{q}_i = \max_k \sigma_{SM}(z_i/T)^{(k)}$$

- softens the softmax (i.e raises output entropy) with $T > 1$
- $T \to \inf$ the probability $\hat{q}_i$ approaches $1/K$ which represents maximum entropy.
- $T = 1$ stay with same softmax
- $T \to 0$ probability collapses to point mass (i.e $\hat{q}_i = 1$).
- T optimized with respect to NLL on validation set.

Note: parameter T doesnt change the argmax
Temperature scaling does not affect the model's accuracy

Experiments and Results

Experiments

Dataset	Model	Uncalibrated	Hist. Binning	Isotonic	BBQ	Temp. Scaling	Vector Scaling	Matrix Scaling
Birds	ResNet 50	9.19%	4.34%	5.22%	4.12%	**1.85%**	3.0%	21.13%
Cars	ResNet 50	4.3%	**1.74%**	4.29%	1.84%	2.35%	2.37%	10.5%
CIFAR-10	ResNet 110	4.6%	0.58%	0.81%	**0.54%**	0.83%	0.88%	1.0%
CIFAR-10	ResNet 110 (SD)	4.12%	0.67%	1.11%	0.9%	**0.6%**	0.64%	0.72%
CIFAR-10	Wide ResNet 32	4.52%	0.72%	1.08%	0.74%	**0.54%**	0.6%	0.72%
CIFAR-10	DenseNet 40	3.28%	0.44%	0.61%	0.81%	**0.33%**	0.41%	0.41%
CIFAR-10	LeNet 5	3.02%	1.56%	1.85%	1.59%	**0.93%**	1.15%	1.16%
CIFAR-100	ResNet 110	16.53%	2.66%	4.99%	5.46%	**1.26%**	1.32%	25.49%
CIFAR-100	ResNet 110 (SD)	12.67%	2.46%	4.16%	3.58%	0.96%	**0.9%**	20.09%
CIFAR-100	Wide ResNet 32	15.0%	3.01%	5.85%	5.77%	**2.32%**	2.57%	24.44%
CIFAR-100	DenseNet 40	10.37%	2.68%	4.51%	3.59%	1.18%	**1.09%**	21.87%
CIFAR-100	LeNet 5	4.85%	6.48%	2.35%	3.77%	**2.02%**	2.09%	13.24%
ImageNet	DenseNet 161	6.28%	4.52%	5.18%	3.51%	**1.99%**	2.24%	-
ImageNet	ResNet 152	5.48%	4.36%	4.77%	3.56%	**1.86%**	2.23%	-
SVHN	ResNet 152 (SD)	0.44%	**0.14%**	0.28%	0.22%	0.17%	0.27%	0.17%
20 News	DAN 3	8.02%	**3.6%**	5.52%	4.98%	4.11%	4.61%	9.1%
Reuters	DAN 3	0.85%	1.75%	1.15%	0.97%	0.91%	**0.66%**	1.58%
SST Binary	TreeLSTM	6.63%	1.93%	**1.65%**	2.27%	1.84%	1.84%	1.84%
SST Fine Grained	TreeLSTM	6.71%	2.09%	**1.65%**	2.61%	2.56%	2.98%	2.39%

Table 1. ECE (%) (with $M = 15$ bins) on standard vision and NLP datasets before calibration and with various calibration methods. The number following a model's name denotes the network depth.

Experiments

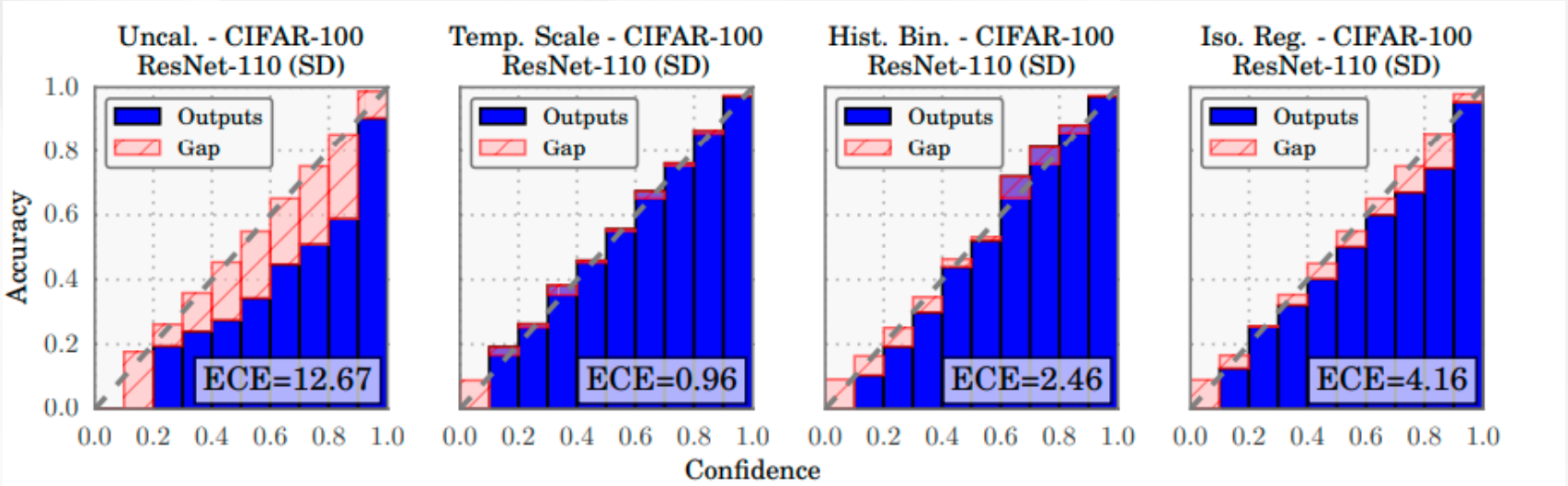


Figure 4. Reliability diagrams for CIFAR-100 before (far left) and after calibration (middle left, middle right, far right).

Experiments and Conclusions

Computation time:

- All methods scale linear with validation set size.
- Matrix and Vector scaling scales quadratically with Classes
- Matrix/Vector scaling, histogram binning is 2 order of magnitude more time than temperature scaling
- BBQ 3 order of magnitude than temperature scaling

Ease of Implementation

- BBQ by far the most complicated to implement.
- Other techniques are pretty straight-forward.

Temperature scaling

- Occam Razor at its best

Network miscalibration is intrinsically low dimensional

Questions?