Measuring Calibration in Neural Networks

Evaluation and Methods

Yi Zhou

zhouyi1023@tju.edu.cn

IDPT & XJTLU

Jiangsu Industrial Technology Research Institute

August 31, 2021

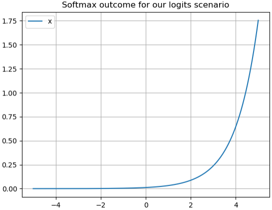
Table of Contents

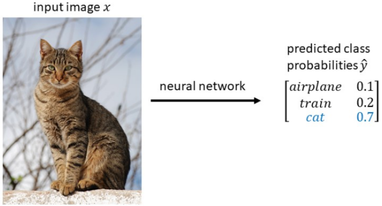
Predicted Probability

Suppose $z \in \mathbb{R}^K$ is the logits in the last layer, then the predicted probability $\hat{p}_i$ and its corresponding predicted class $\hat{y}_i$ are derived using the softmax function of z

$$\sigma_{SM}(z_i^k) = \frac{\exp(z_i^k)}{\sum_{j=1}^{K} \exp(z_i^j)}, \quad \hat{p}_i = \max_k \sigma_{SM}(z_i^k), \quad \hat{y}_i = \arg\max_k z_i^k$$

Interpretation of the predicted probability: For example, given 100 predictions of cats, each with confidence of 0.7, we expect that 70 should be correctly classified





Model Calibration

A model is perfect calibrated if

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

Empirical approximations

- Expected Calibration Error (ECE)
- Reliability diagrams
- Brier score

Expected Calibration Error (ECE)

Group predictions into M equal-width bins, the accuracy and confidence within the bin set B_m are

$$Acc(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \mathbb{1}(\hat{y}_i = y_i), \quad Conf(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i$$

Perfect calibration indicates $Acc(B_m) = Conf(B_m)$ for all $m \in M$

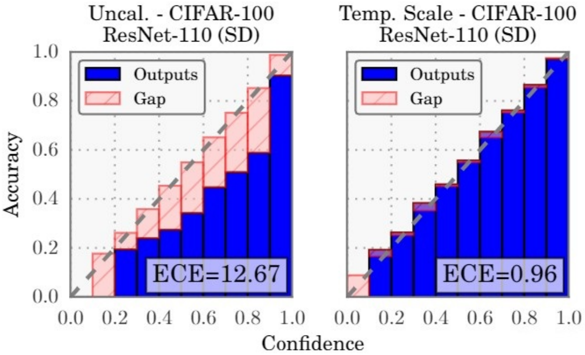
ECE: taking a weighted average of the bins' accuracy and confidence difference

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} |\, Acc(B_m) - Conf(B_m)|$$

Reliability Diagram

Visualization of every bin's confidence and accuracy

- Perfect calibration follows the diagonal
- Under the diagonal indicates over-confidence
- Over the diagonal indicates under-confidence



Variants of ECE

Problems of ECE

- Failing to condition on the class
- Consider only the predicted class, the other K-1 classes are omitted
- Use evenly spaced binning but the predicted probability is skewed

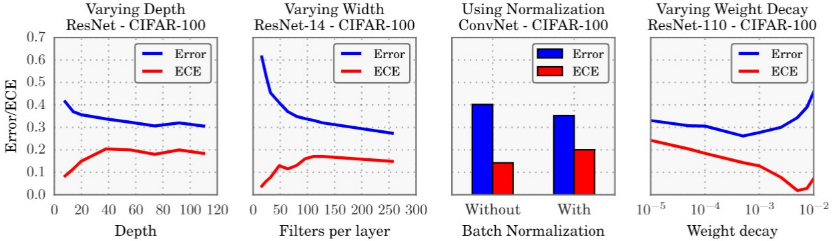
$$AdaECE = \sum_{i=1}^{M} \frac{|B_i|}{N} |Acc(B_m) - Conf(B_m)| \quad s.t. \forall i, j \; |B_i| = |B_j|$$

$$ClasswiseECE = \frac{1}{K} \sum_{i=1}^{M} \sum_{j=1}^{K} \frac{|B_i, j|}{N} |Acc(B_{i,j}) - Conf(B_{i,j})|$$

Modern Networks are Poorly Calibrated

Modern neural networks, e.g. ResNet, are poorly calibrated[1]

- Increasing depth and width may reduce classification error, but negatively affect model calibration (over confidence)
- Models trained with Batch Normalization tend to be more miscalibrated
- Training with less weight decay has a negative impact on calibration



[1] On Calibration of Modern Neural Networks[1]

Model Calibration Under Dataset Shift

Out-of-distrbution(O.O.D.) data

- Covariate shift: corruptions and perturbation
- Unseen data whose label is not with in the original k classes

- Most methods demonstrate very low entropy and give high confidence predictions on data that is entirely OOD
- Along with accuracy, the quality of uncertainty consistently degrades with increasing dataset shift
- Calibrating on the validation set leads to well-calibrated predictions on the test set, but it does not guarantee calibration on shifted data[2]

[2] Improving model calibration with accuracy versus uncertainty optimization[2]

Recall: Negative Log Likelihood

Cross Entropy Loss

Cross entropy measures the dissimilarity between two distrbutions

$$H(g,f) = -\sum_x g(x)\log\frac{1}{f(x)} = -\sum_x g(x)\log f(x)$$

where $g(x)$ is the true distribution, $f(x)$ is the predicted distribution.

Softmax cross entropy can be interpreted as a negative log likelihood

Negative Log Likelihood

The ground truth $y \in \mathbb{R}^k$ is a one hot vector (Dirac delta function), likelihood of the observation is $\prod_{i=1}^{K} \hat{y}_i^{y_i}$, then

$$CE = NLL = -\sum_{i=1}^{K} y_i \log \hat{y}_i$$

What Causes Miscalibration?

High capacity of neural networks leaves them vulnerable to overfitting on the NLL loss [3]

The optimiser may try to further reduce the training NLL by increasing the confidences for the correctly classified samples

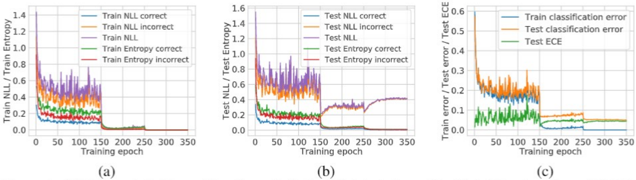


Figure 1: Metrics related to calibration plotted whilst training a ResNet-50 network on CIFAR-10.

After the 150th epoch

- Rise in test NLL rise indicates overfitting
- Rise in test ECE indicates miscalibration
- Entropies keep dropping indicating the distributions get peakier, but could at wrong places

[3] Calibrating Deep Neural Networks using Focal Loss[3]

Post Processing: Temperature Scaling

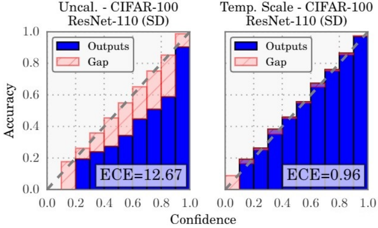
Use a single scalar parameter $T > 0$ for all classes to scale the logits before softmax

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

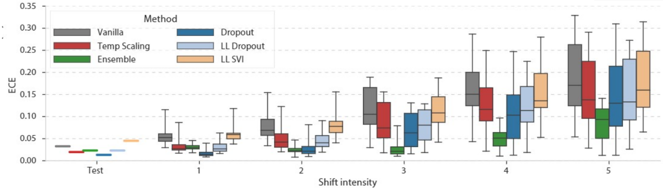
- T is called the temperature, and it "softens" the softmax (i.e. raises the output entropy) with $T > 1$
- T is optimized with respect to NLL on the validation set
- Temperature scaling does not affect the model's accuracy

Temperature Scaling Performance

- Temperature scaling leads to well-calibrated uncertainty on the i.i.d. test set and small values of shift



- But is significantly outperformed by methods that take epistemic uncertainty into account as the shift increases.



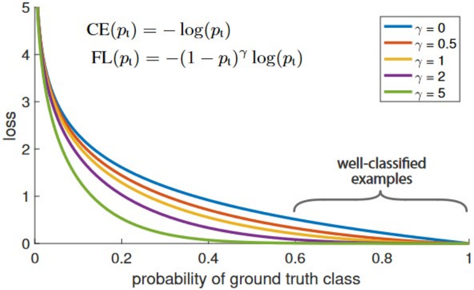
Modify the Loss: Focal Loss

Motivation: Encouraging the predicted distribution to have higher entropy can help avoid the overconfident predictions

Focal Loss: Weight loss components generated from individual samples in a mini-batch by how well the model classifies them

$$L_f = -(1 - \hat{p}_{i,y_i})^\gamma \log \hat{p}_{i,y_i}$$

where γ is the user-defined focusing parameter



Focal Loss Performance

- FLSD-53 produces the lowest calibration errors in general
- also perform better than other competitive loss functions under distribution shift

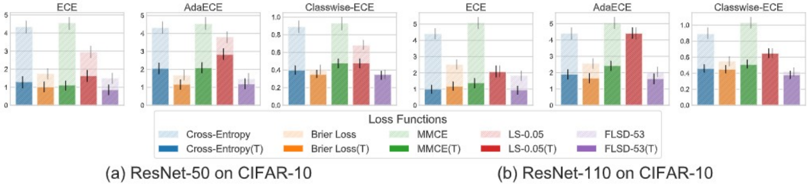


Figure 4: Bar plots with confidence intervals for ECE, AdaECE and Classwise-ECE, computed for ResNet-50 (first 3 figures) and ResNet-110 (last 3 figures) on CIFAR-10.

Differentiable ECE Loss

Differentiable losses to improve calibration based on a soft (continuous) version of the binning operation [4]

- ECEs are not non-trainable since they are zero within bin boundaries and undefined at bin boundaries
- Suppose ξ_i is the center of bin i, define the soft bin-membership function as

$$u_{M,T}(c) = \sigma_{SM}(g_{M,T}(c))$$
$$g_{M,T,i}(c) = -(c - \xi_i)^2 / T$$

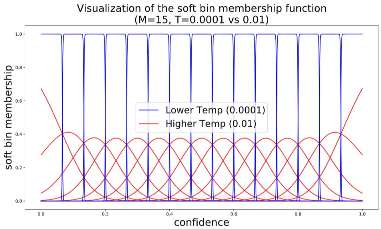


Figure 3: Visualization of the soft bin membership function which shows that the temperature parameter determines the sharpness of the binning. Soft binning limits to hard binning as temperature tends to zero.

[4] Soft Calibration Objectives for Neural Networks[4]

Differentiable ECE Loss Performance

We define the Expected Soft-Binned Calibration Error SB-ECE$_{\text{bin},p}(M, T, \hat{D}, \boldsymbol{\theta})$ and Expected Soft-Label-Binned Calibration Error SB-ECE$_{\text{lb},p}(M, T, \hat{D}, \boldsymbol{\theta})$:

$$\text{SB-ECE}_{\text{bin},p}(M, T, \hat{D}, \boldsymbol{\theta}) = \left(\Sigma_{i=1}^{M}\left(\frac{S_j}{\hat{N}}|A_j - C_j|^p\right)\right)^{1/p}, \tag{11}$$

$$\text{SB-ECE}_{\text{lb},p}(M, T, \hat{D}, \boldsymbol{\theta}) = \left(\frac{1}{\hat{N}}\Sigma_{i=1}^{\hat{N}}\Sigma_{j=1}^{M}(u^*_{\mathcal{M},T,j}(c_i)\cdot|A_j - c_i|^p)\right)^{1/p}. \tag{12}$$

The quantities S_j, C_j and A_j in these expressions are obtained by using the soft bin membership function $\boldsymbol{u}^*_{M,T}$ in place of the hard bin membership function $\boldsymbol{u}_{\mathcal{B}}$ in equations 8, 9 and 10 respectively.

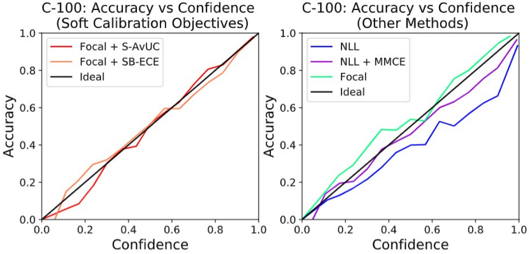
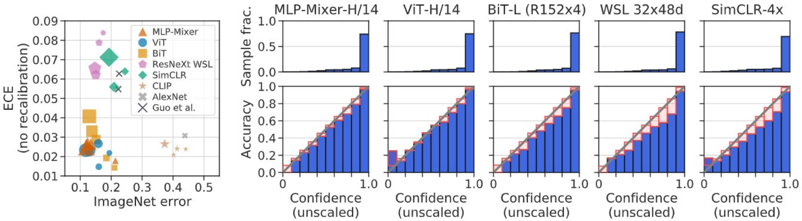


Figure 6: Accuracy vs Confidence plots for various methods on CIFAR-100. NLL is significantly overconfident and NLL + MMCE is somewhat overconfident. While Focal loss is underconfident, augmenting it with Soft Calibration Objectives fixes this issue, resulting in curves closest to the ideal.

Non-Convolutional Architectures

Architecture is an important determinant of model calibration[5]

- The non-convolutional MLP-Mixer and Vision Transformers are well calibrated and robust to distribution shift.
- In-distribution calibration slightly deteriorates with increasing model size
- Under distribution shift, calibration improves with model size
- Accuracy and calibration are correlated under distribution shift, such that optimizing for accuracy may also benefit calibration.



[5] Revisiting the Calibration of Modern Neural Networks[5]

Reference I

- Chuan Guo et al. "On calibration of modern neural networks". In: **International Conference on Machine Learning**. PMLR. 2017, pp. 1321–1330.
- Yaniv Ovadia et al. "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift". In: **arXiv preprint arXiv:1906.02530** (2019).
- Jishnu Mukhoti et al. "Calibrating deep neural networks using focal loss". In: **arXiv preprint arXiv:2002.09437** (2020).
- Archit Karandikar et al. "Soft Calibration Objectives for Neural Networks". In: **arXiv preprint arXiv:2108.00106** (2021).
- Matthias Minderer et al. "Revisiting the Calibration of Modern Neural Networks". In: **arXiv preprint arXiv:2106.07998** (2021).

Thank You !