On Calibration of Modern Neural Networks: Temperature Scaling

Paper Authors: Chuan Guo, Geoff Pleiss, Yu Sun
Kilian, Q. Weinberger

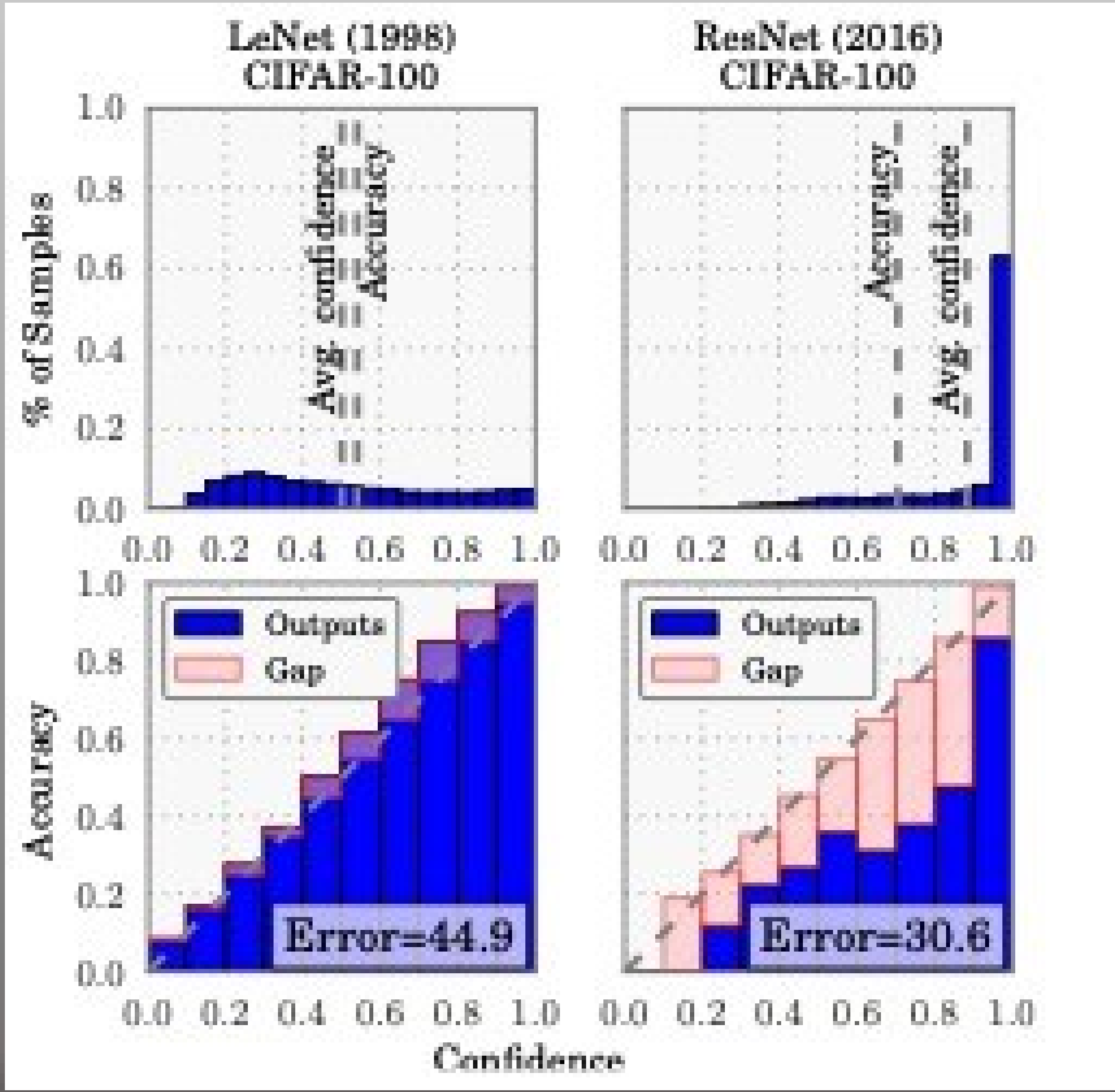


Image reproduced under fair use from
https://arxiv.org/abs/1706.04599

Overview

- Calibration: Predict probability representative of correctness likelihood
- Modern neural networks are poorly calibrated
  - unlike those from a decade ago
- Calibration influenced by
  - depth, width
  - weight decay, and
  - Batch Normalization
- Evaluate post-processing calibration on state-of- the-art architectures
- Temperature scaling is surprisingly effective at calibration
  - single- parameter variant of Platt Scaling

Motivation - I

- neural networks produced well-calibrated probabilities on binary classification tasks
  - Niculescu-Mizil & Caruana (2005)
- Comparison
  - 5-layer LeNet (LeCun et al., 1998)
  - 110-layer ResNet (He et al., 2016)
  - CIFAR-100

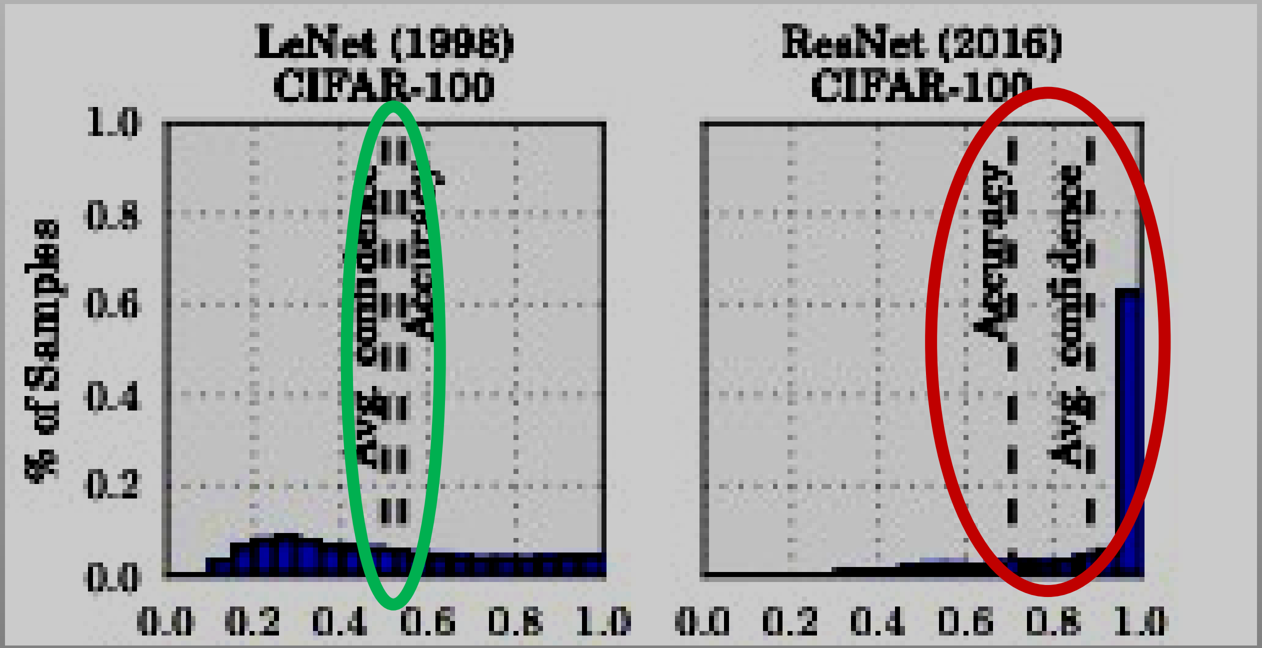


Image reproduced under fair use from https://arxiv.org/abs/1706.04599

Motivation - II

- neural networks produced well-calibrated probabilities on binary classification tasks
  - Niculescu-Mizil & Caruana (2005)
- Comparison
  - 5-layer LeNet (LeCun et al., 1998)
  - 110-layer ResNet (He et al., 2016)
  - CIFAR-100
- Reliability Diagram

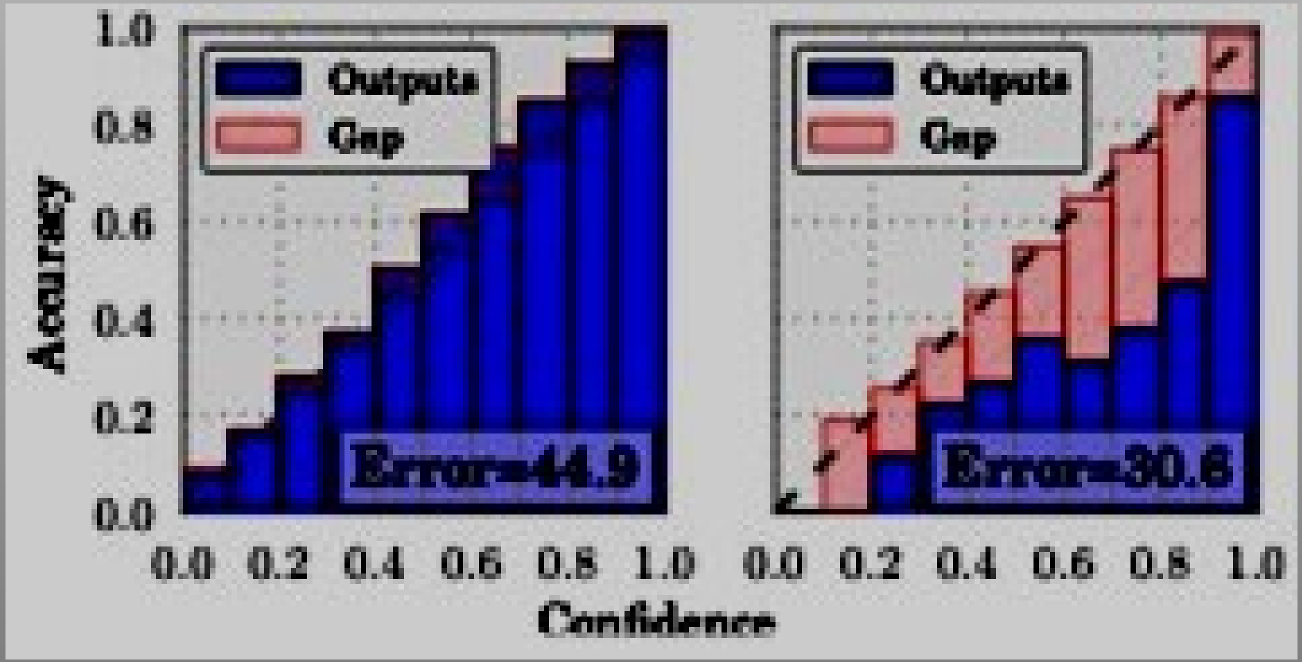


Image reproduced under fair use from
https://arxiv.org/abs/1706.04599

Calibration Definition

- Let h be a neural network with $h(X) = (\hat{Y}, \hat{P})$
  - $\hat{Y}$ is a class prediction
  - $\hat{P}$ is its associated confidence, i.e. probability of correctness.
- Expect confidence estimate $\hat{P}$ to be calibrated

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

- For example,
  - given 100 predictions,
  - each with confidence of 0.8,
  - expect that 80 should be correctly classified.

Reliability Diagram

- Visual representation of model calibration
- Plot accuracy vs. confidence
- Deviation from diagonal represents miscalibration

- Let B_m be the set of indices of samples
  - whose confidence falls into interval $I_m = (\frac{m-1}{M}, \frac{m}{M})$.
  - The accuracy of B_m is $\mathrm{acc}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \mathbf{1}(\hat{y}_i = y_i)$.

- Define the average confidence within bin B_m as

$$\mathrm{conf}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i,$$

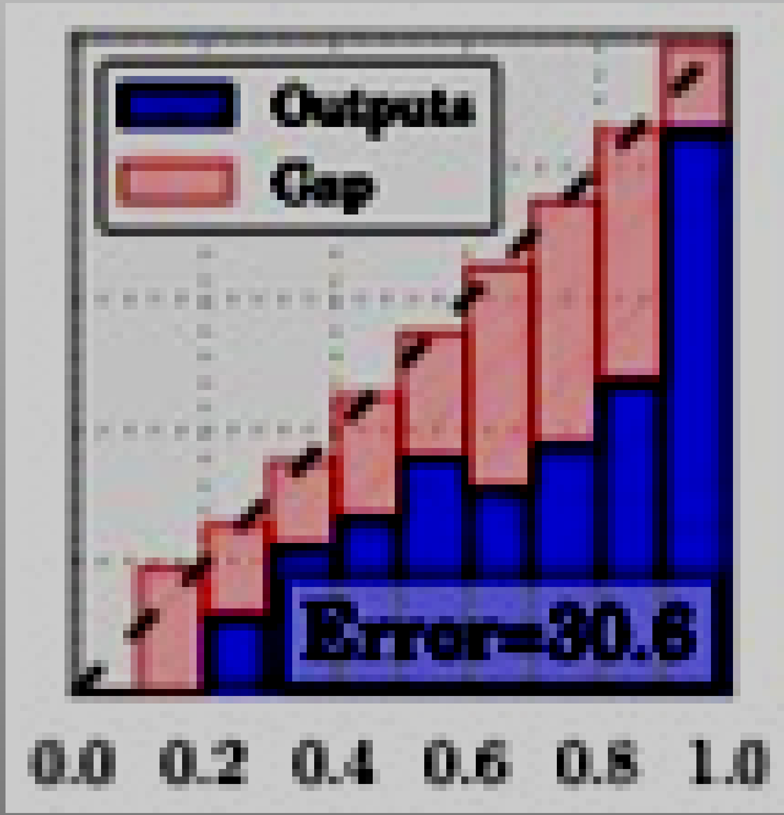


Image reproduced under fair use from https://arxiv.org/abs/1706.04599

Expected Calibration Error (ECE)

- Visual vs. Numeric
  - while reliability diagrams are useful visual tools,
  - it is more convenient to have a scalar summary statistic of calibration.
- Statistics comparing two distributions cannot be comprehensive(?)
- ECE: difference in expectation between confidence and accuracy

$$\mathop{\mathbb{E}}_{\hat{P}}\left[\left|\mathbb{P}\left(\hat{Y}=Y \mid \hat{P}=p\right)-p\right|\right]$$

- ECE approximation: $\mathrm{ECE}=\sum_{m=1}^{M}\frac{|B_m|}{n}\left|\mathrm{acc}(B_m)-\mathrm{conf}(B_m)\right|$

Maximum Calibration Error (MCE)

- high-risk applications
  - reliable confidence measures are absolutely necessary
- Minimize the worst-case deviation between confidence and accuracy

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

- Approximation involves binning (similar to ECE)

$$\mathrm{MCE}=\max_{m\in\{1,\ldots,M\}}|\mathrm{acc}(B_m)-\mathrm{conf}(B_m)|$$

Negative Log Likelihood (NLL)

- Negative log likelihood
  - a standard measure of a probabilistic model's quality
  - Friedman et al., 2001
- Also known as cross entropy loss
  - Bengio et al., 2015
- Given a probabilistic model $\hat{\pi}(Y|X)$, and n samples, NLL is defined as

$$\mathcal{L} = -\sum_{i=1}^{n} \log(\hat{\pi}(y_i|\mathbf{x}_i))$$

- In expectation, NLL is minimized if and only if $\hat{\pi}(Y|X)$ recovers the ground truth conditional distribution $\pi(Y|X)$.

Observing Miscalibration - I

- **Model capacity**
- model capacity increased at a fast pace over the past decade.
- 100-1000 layers
  - (He et al., 2016; Huang et al., 2016)
- 100s of convolutional filters per layer
  - (Zagoruyko & Komodakis, 2016)
- increasing depth and width may reduce classification error
- Such increases negatively affect model calibration
  - ResNet on CIFAR-100

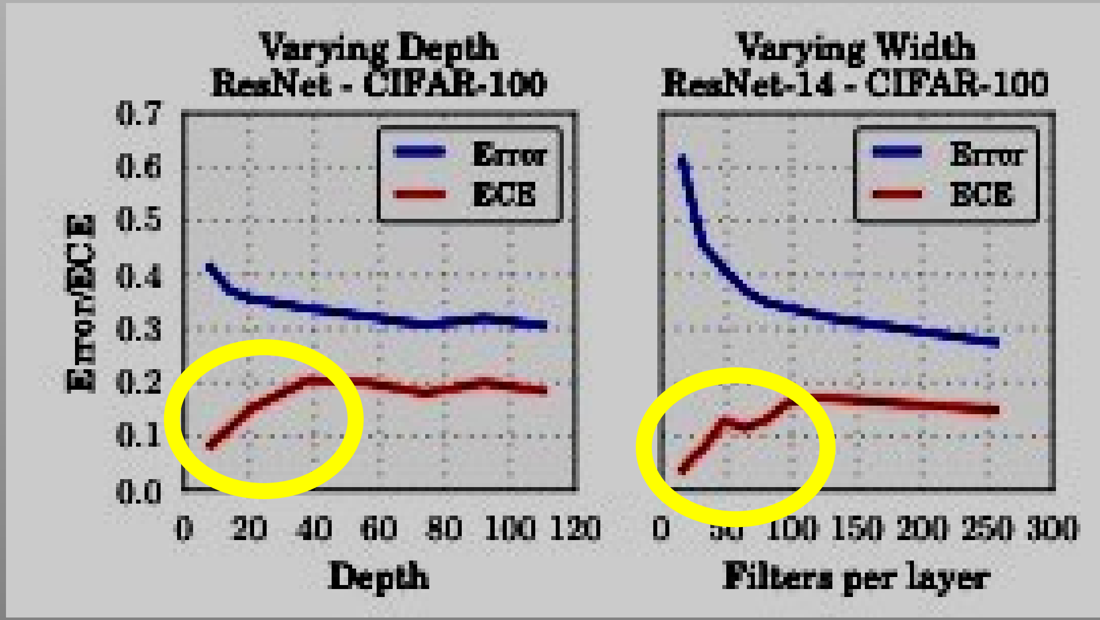
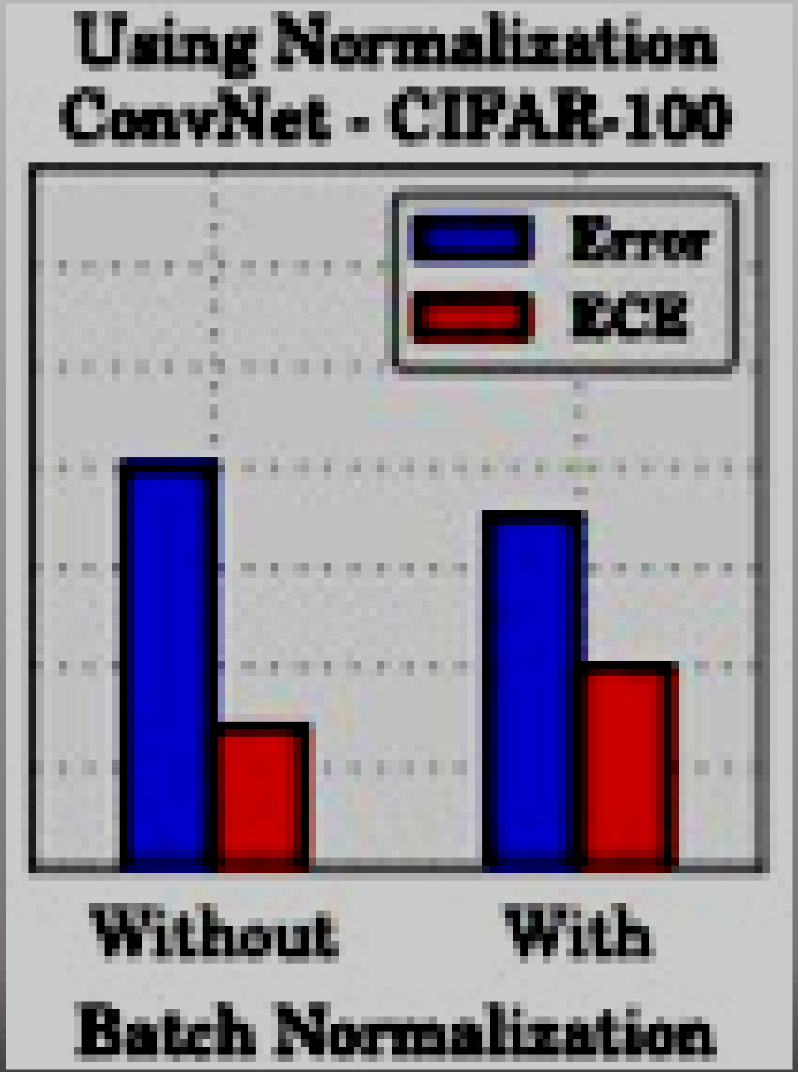


Image reproduced under fair use from https://arxiv.org/abs/1706.04599

Image reproduced under fair use from
https://arxiv.org/abs/1706.04599

Observing Miscalibration - II

- **Batch Normalization**
  - (Ioffe & Szegedy, 2015)
  - minimizes distribution shifts in activations
  - improves training time
  - reduces the need for more regularization
  - May improve accuracy
- Enable the development of very deep architectures
- Creates more miscalibrated models
  - regardless of hyperparameters



Observing Miscalibration - III

- **Weight decay**
  - used to be a predominant regularization mechanism for neural networks
  - Learning Theory Vapnik, 1998
    - regularization prevents overfitting
  - Ioffe & Szegedy, 2015
    - models with less L2 regularization generalizes better
  - Now common to train models with little weight decay, if any at all.
- more regularization improves calibration
  - well after optimal accuracy.

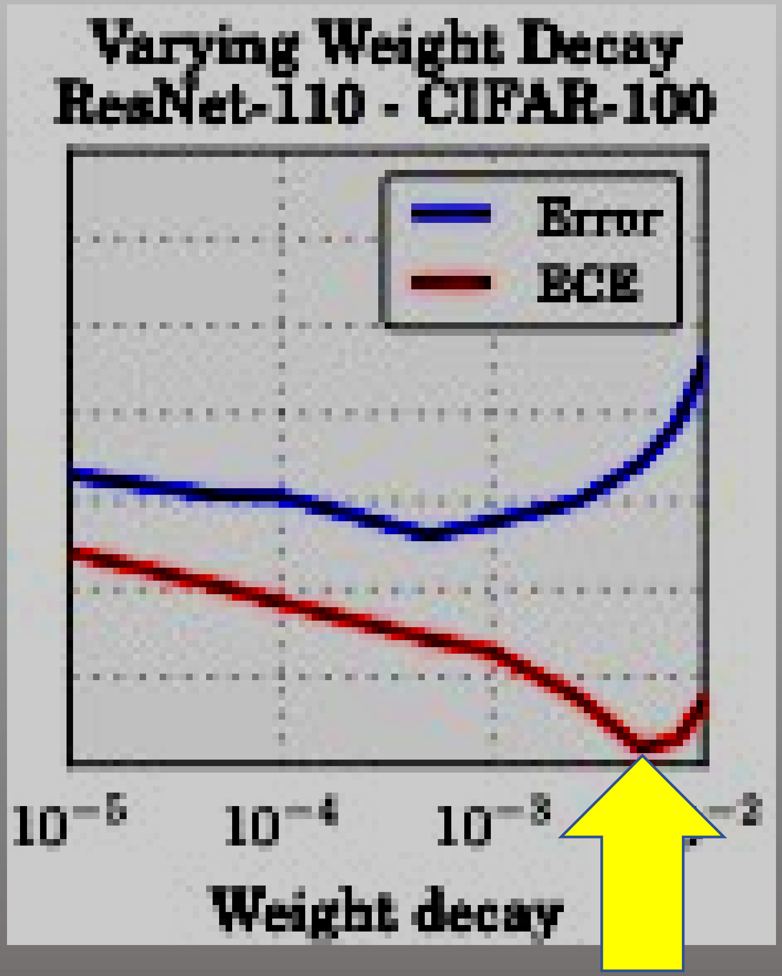


Image reproduced under fair use from
https://arxiv.org/abs/1706.04599

Observing Miscalibration - IV

- **NLL** indirectly measures model calibration.
- *In practice, we observe a disconnect between NLL and accuracy*
- Neural networks can overfit to NLL without overfitting to the 0/1 loss.
- Both error and NLL drop at epoch 250
  - when the learning rate is dropped
  - however, NLL overfits during the remainder of training.

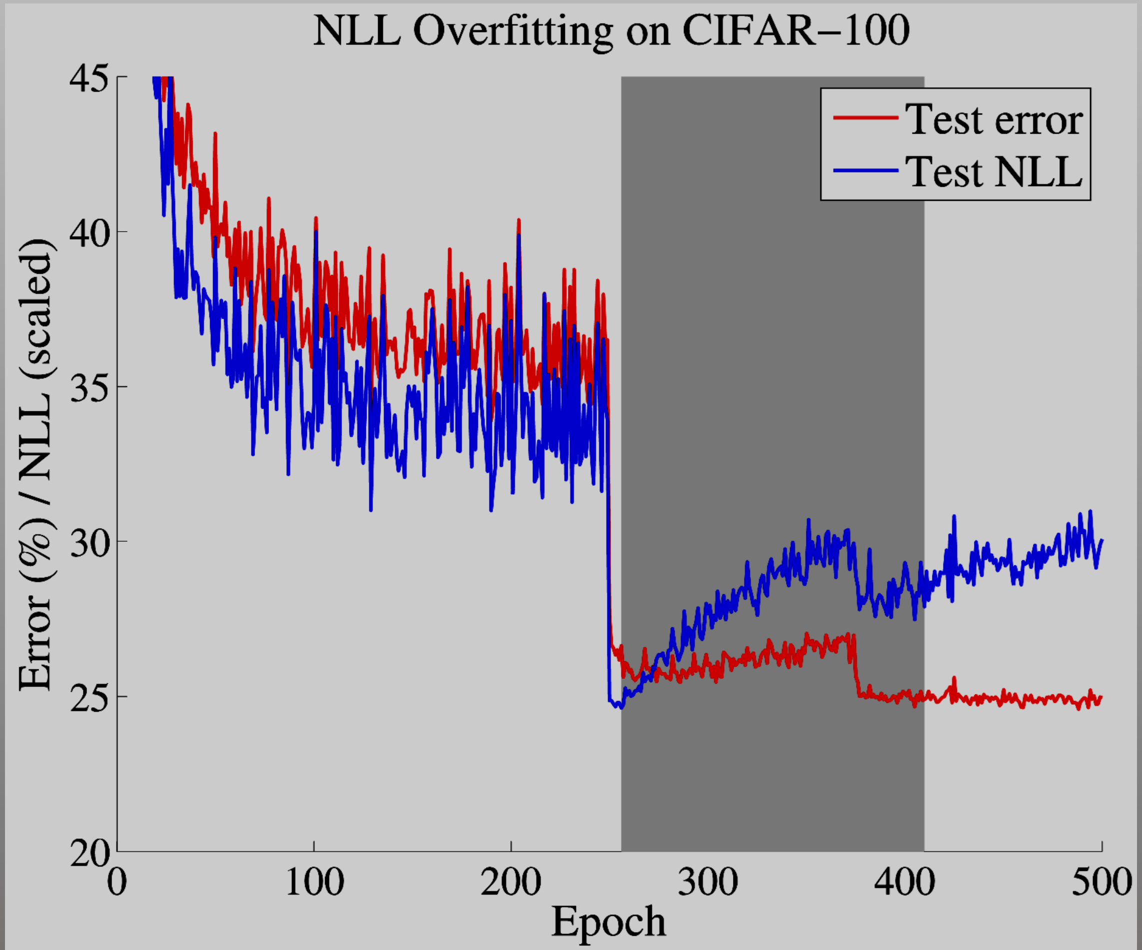


Image reproduced under fair use from https://arxiv.org/abs/1706.04599

Calibration Methods – I

- **Histogram binning** is a simple non-parametric calibration method
- all uncalibrated predictions $\hat{p}_i$ are divided into mutually exclusive bins $B_1, \ldots, BM$.
- Each bin is assigned a calibrated score θ_m; if $\hat{p}_i$ is assigned to bin B_m, then $\hat{q}_i = \theta_m$.
- For a fixed M, we define bin boundaries

$$0 = a_1 \le a_2 \le \ldots \le a_{M+1} = 1,$$

- The predictions θ_i are chosen to minimize the bin-wise squared loss:

$$\min_{\theta_1,\ldots,\theta_M} \sum_{m=1}^{M} \sum_{i=1}^{n} \mathbf{1}(a_m \le \hat{p}_i < a_{m+1}) \left(\theta_m - y_i\right)^2$$

- The solution results in θ_m that correspond to the average number of positive-class samples in bin B_m.

Calibration Methods – II

- **Isotonic regression**
  - learns a piecewise constant function *f* to transform uncalibrated outputs $\hat{q}_i = f(\hat{p}_i)$.
  - Generalizes histogram binning
    - bin boundaries and bin predictions are jointly optimized.
- Produces f to minimize the square loss $\sum_{i=1}^{n}(f(\hat{p}_i) - y_i)^2$.
- Optimization problem

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

$$\text{subject to} \quad 0 = a_1 \leq a_2 \leq \ldots \leq a_{M+1} = 1,$$

$$\theta_1 \leq \theta_2 \leq \ldots \leq \theta_M.$$

  - M is the number of intervals
  - $a_1, \ldots, a_{M+1}$ are the interval boundaries
  - and $\theta_1, \ldots, \theta_M$ are the function values

Calibration Methods – III

- **Bayesian Binning into Quantiles (BBQ)**
- Naeini et al., 2015
- an extension of histogram binning using Bayes model averaging
- BBQ marginalizes out all possible *binning schemes*
- The parameters of a binning scheme are $\theta_1, \ldots, \theta_M$
- Under this framework,
  - histogram binning and isotonic regression both produce a single binning scheme,
  - where BBQ considers a space *S* of all possible binning schemes for the validation data set D
- BBQ performs Bayesian averaging of the probabilities produced by each scheme

Calibration Methods – IV

- **Platt scaling** (Platt et al., 1999) is a parametric approach to calibration
- The non-probabilistic classifier predictions are used for logistic regression
    - trained on the validation set to return probabilities
- Platt scaling learns scalar parameters $a, b \in \mathbb{R}$ and
- outputs $\hat{q}_i = \sigma(az_i + b)$ as the calibrated probability.
- Parameters a and b optimized using NLL loss over validation set
- Neural network's parameters are fixed during this stage

Calibration – V

- **Extension to Multiclass Models**
- network outputs a class prediction $\hat{y}_i$ and confidence score $\hat{p}_i$ for each input $\mathbf{x}_i$.
- In this case, the network logits $\mathbf{z}_i$ are vectors, where $\hat{y}_i = \text{argmax}_k \, z_i^{(k)}$,
- $\hat{p}_i$ is typically derived using the softmax function

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

- Goal: produce a calibrated confidence and class prediction based on the above.

Calibration - VI

- **Extension of binning methods**.
  - Extend binary calibration methods to the multiclass setting
    - by treating the problem as K one-versus-all problems
- **Matrix and vector scaling**: multi-class extensions of Platt scaling.
- Let $\mathcal{Z}_i$ be the *logits vector* for input X_i.
- *Matrix scaling applies* a linear transformation $W\mathcal{Z}_i$ + b to the logits

$$\hat{q}_i = \max_k \; \sigma_{\mathrm{SM}}(\mathbf{W}\mathbf{z}_i + \mathbf{b})^{(k)},$$

$$\hat{y}_i' = \operatorname*{argmax}_k \, (\mathbf{W}\mathbf{z}_i + \mathbf{b})^{(k)}.$$

- The parameters W and b are optimized with respect to NLL on the validation set.
- # parameters for matrix grows quadratically with number of classes K
- Define *vector scaling:* W is restricted to be a diagonal matrix

Temperature Scaling

- Commonly used in other settings
  - knowledge distillation (Hinton et al., 2015)
  - statistical mechanics (Jaynes, 1957)
- Temperature scaling uses a single scalar parameter T > 0 for all classes
  - the simplest extension of Platt scaling
- Given the logit vector $\mathcal{Z}_i$, the new confidence prediction is $\hat{q}_i = \max_k \sigma_{\mathrm{SM}}(\mathbf{z}_i / T)^{(k)}$
- T is called the temperature
- It "softens" the softmax with $T > 1$.
- As $T \rightarrow \infty$, the probability $\widehat{q_i}$ approaches 1/K
  - which represents maximum uncertainty.
- T is optimized with respect to NLL on the validation set.
- Because the parameter T does not change the maximum of the softmax function,
- the class prediction remains unchanged.
- In other words, *temperature scaling does not affect the model's accuracy.*

Results – I

6 data sets for image classification

1. Caltech-UCSD Birds (Welinder et al., 2010): 200 bird species.
2. Stanford Cars (Krause et al., 2013): 196 classes of cars by make, model, and year.
3. ImageNet 2012 (Deng et al., 2009): Natural scene images from 1000 classes.
4. CIFAR-10/CIFAR-100 (Krizhevsky & Hinton, 2009): Color images (32 × 32) from 10/100 classes.
5. Street View House Numbers (SVHN) (Netzer et al., 2011): 32 × 32 colored images of cropped out house numbers from Google Street View.

Results – II

4 data sets for document classification

1. 20 News: News articles, partitioned into 20 categories by content.
2. Reuters: News articles, partitioned into 8 categories by topic.
3. Stanford Sentiment Treebank (SST) (Socher et al., 2013): Movie reviews, represented as sentence parse trees that are annotated by sentiment.
    - Each sample includes a coarse binary label and a fine grained 5-class label.

Results – III

CIFAR-100

Dataset	Model	Uncalibrated	Hist. Binning	Isotonic	BBQ	Temp. Scaling	Vector Scaling	Matrix Scaling
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%

Results – IV

CIFAR-10

Dataset	Model	Uncalibrated	Hist. Binning	Isotonic	BBQ	Temp. Scaling	Vector Scaling	Matrix Scaling
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%

Results – V

ImageNet/SVHN

Dataset	Model	Uncalibrated	Hist. Binning	Isotonic	BBQ	Temp. Scaling	Vector Scaling	Matrix Scaling
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%

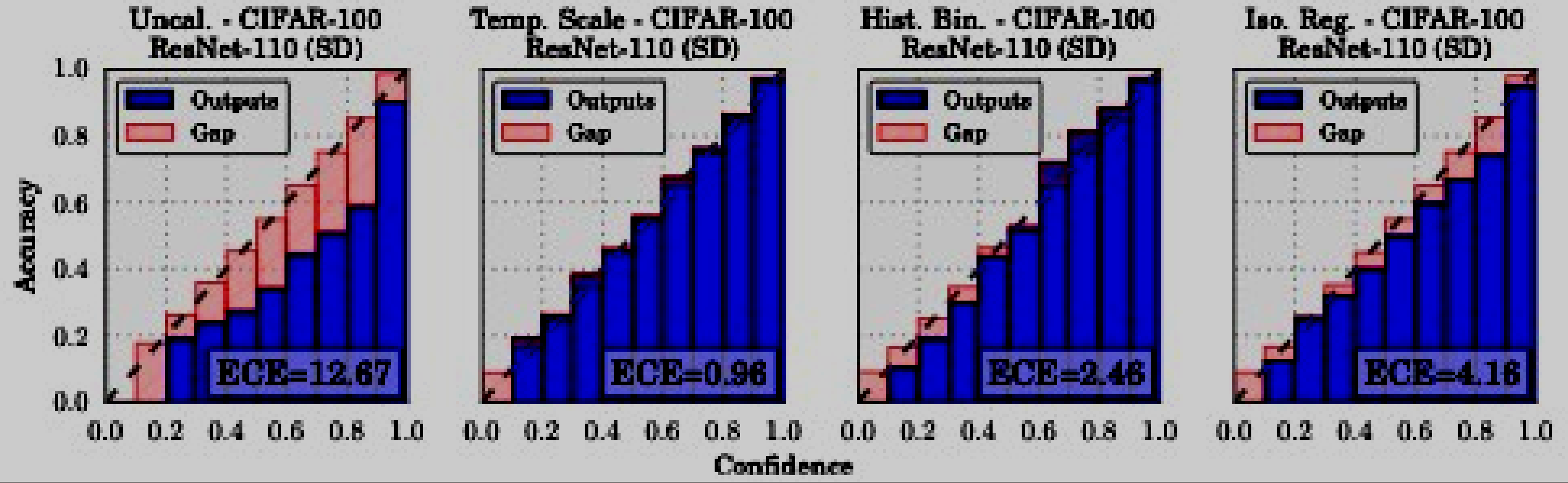
Results – VI

NLP

Dataset	Model	Uncalibrated	Hist. Binning	Isotonic	BBQ	Temp. Scaling	Vector Scaling	Matrix Scaling
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%

Results – VII

ResNet on CIFAR-100



Theoretical Result

Claim 1. *Given n samples' logit vectors $\mathbf{z}_1, \dots, \mathbf{z}_n$ and class labels $y_1, \dots, y_n$, temperature scaling is the unique solution q to the following entropy maximization problem:*

$$\begin{aligned}
\max_{q} \quad & -\sum_{i=1}^{n} \sum_{k=1}^{K} q(\mathbf{z}_i)^{(k)} \log q(\mathbf{z}_i)^{(k)} \\
\textit{subject to} \quad & q(\mathbf{z}_i)^{(k)} \geq 0 \qquad \forall i, k \\
& \sum_{k=1}^{K} q(\mathbf{z}_i)^{(k)} = 1 \quad \forall i \\
& \sum_{i=1}^{n} z_i^{(y_i)} = \sum_{i=1}^{n} \sum_{k=1}^{K} z_i^{(k)} q(\mathbf{z}_i)^{(k)}.
\end{aligned}$$

Proof

$$L = -\sum_{i=1}^{n}\sum_{k=1}^{K} q(\mathbf{z}_i)^{(k)} \log q(\mathbf{z}_i)^{(k)}$$
$$+ \lambda \sum_{i=1}^{n}\left[\sum_{k=1}^{K} z_i^{(k)} q(\mathbf{z}_i)^{(k)} - z_i^{(y_i)}\right]$$
$$+ \sum_{i=1}^{n} \beta_i \sum_{k=1}^{K} (q(\mathbf{z}_i)^{(k)} - 1).$$

Lagrangian

Proof

$$L = -\sum_{i=1}^{n}\sum_{k=1}^{K} q(\mathbf{z}_i)^{(k)} \log q(\mathbf{z}_i)^{(k)}$$
$$+ \lambda \sum_{i=1}^{n}\left[\sum_{k=1}^{K} z_i^{(k)} q(\mathbf{z}_i)^{(k)} - z_i^{(y_i)}\right]$$
$$+ \sum_{i=1}^{n} \beta_i \sum_{k=1}^{K} (q(\mathbf{z}_i)^{(k)} - 1).$$

$\Rightarrow$

$$\frac{\partial}{\partial q(\mathbf{z}_i)^{(k)}} L = -nK - \log q(\mathbf{z}_i)^{(k)} + \lambda z_i^{(k)} + \beta_i.$$

Lagrangian

Proof

$$L = -\sum_{i=1}^{n}\sum_{k=1}^{K} q(\mathbf{z}_i)^{(k)} \log q(\mathbf{z}_i)^{(k)}$$

$$+ \lambda \sum_{i=1}^{n}\left[\sum_{k=1}^{K} z_i^{(k)} q(\mathbf{z}_i)^{(k)} - z_i^{(y_i)}\right]$$

$$+ \sum_{i=1}^{n} \beta_i \sum_{k=1}^{K} (q(\mathbf{z}_i)^{(k)} - 1).$$

Lagrangian

$$\frac{\partial}{\partial q(\mathbf{z}_i)^{(k)}} L = -nK - \log q(\mathbf{z}_i)^{(k)} + \lambda z_i^{(k)} + \beta_i.$$

Setting derivative to 0,

$$q(\mathbf{z}_i)^{(k)} = e^{\lambda z_i^{(k)} + \beta_i - nK}$$

Proof

$$L = -\sum_{i=1}^{n}\sum_{k=1}^{K} q(\mathbf{z}_i)^{(k)} \log q(\mathbf{z}_i)^{(k)} + \lambda \sum_{i=1}^{n}\left[\sum_{k=1}^{K} z_i^{(k)} q(\mathbf{z}_i)^{(k)} - z_i^{(y_i)}\right] + \sum_{i=1}^{n}\beta_i \sum_{k=1}^{K}(q(\mathbf{z}_i)^{(k)} - 1).$$

Lagrangian

$$\frac{\partial}{\partial q(\mathbf{z}_i)^{(k)}} L = -nK - \log q(\mathbf{z}_i)^{(k)} + \lambda z_i^{(k)} + \beta_i.$$

Setting derivative to 0,

$$q(\mathbf{z}_i)^{(k)} = e^{\lambda z_i^{(k)} + \beta_i - nK}$$

Since probabilities sum to 1,

$$q(\mathbf{z}_i)^{(k)} = \frac{e^{\lambda z_i^{(k)}}}{\sum_{j=1}^{K} e^{\lambda z_i^{(j)}}}$$

Conclusions

- Probabilistic error and miscalibration worsen for modern neural nets
  - Even when classification error is reduced.
- Recent advances worsen network calibration
  - model capacity,
  - normalization,
  - regularization
- Future work:

Understand why these trends affect calibration while improving accuracy

- Temperature scaling is effective in calibrating models
  - simplest,
  - fastest, and
  - most straightforward