Supplementary Material for Trainable Calibration Measures for Neural Networks from Kernel Mean Embeddings

Aviral Kumar [1] **Sunita Sarawagi** [1] **Ujjwal Jain** [1]

1. Proof of Theorem 2

Proof. Our proof technique is similar to (Gretton et al., 2012)'s for MMD. Define $g(\mathrm{D}) = |\mathrm{MMCE}_m(\mathrm{D}) - \mathrm{MMCE}(P)|$. The maximum change in $g(\mathrm{D})$ when one sample (r_i, c_i) is replaced by a random other sample is $\frac{2\sqrt{K}}{m}$ in the expression for MMCE_m. Applying McDiarmid inequality, we get that

$$\Pr(g(\mathrm{D}) - E_{\mathrm{D}}[g(\mathrm{D})] > \epsilon) < \exp(-\frac{m\epsilon^2}{2K}) \qquad (1)$$

Next we upper bound $E_{\mathrm{D}}[g(\mathrm{D})]$ starting from the definition of MMCE.

$$E_{\mathrm{D}}[|\mathrm{MMCE}_m(\mathrm{D}) - \mathrm{MMCE}(P)|]$$
$$= E_{\mathrm{D}}[|\sup_{f\in\mathcal{F}}[\sum_{i=1}^{m}\frac{(c_i - r_i)f(r_i)}{m}] - \sup_{f\in\mathcal{F}} E_P[(c-r)f(r)]|]$$
$$\leq E_{\mathrm{D}} \sup_f \left|\sum_{i=1}^{m}\frac{(c_i - r_i)f(r_i)1}{m} - E_P[(c-r)f(r)]\right|$$
$$= E_{\mathrm{D}} \sup_f \left|\sum_{i=1}^{m}\frac{(c_i - r_i)f(r_i)}{m} - E_{\mathrm{D}'}[\sum_{i=1}^{m}\frac{(c'_i - r'_i)f(r'_i)}{m}]\right|$$
$$\leq E_{\mathrm{D},\mathrm{D}'} \sup_f \left|\sum_{i=1}^{m}\frac{(c_i - r_i)f(r_i)}{m} - \sum_{i=1}^{m}\frac{(c'_i - r'_i)f(r'_i)}{m}\right|$$
$$= E_{\mathrm{D},\mathrm{D}',\sigma} \sup_f \left|\sum_{i=1}^{m}\sigma_i\left(\frac{(c_i - r_i)f(r_i)}{m} - \frac{(c'_i - r'_i)f(r'_i)}{m}\right)\right|$$
$$\leq 2\sqrt{\frac{4K}{m}}$$

In the above σ_i denotes random variables that can take values +1 or -1 with equal probability and the last inequality is due to (Bartlett & Mendelson, 2002),Lemma 22. Combining Equation 1 and the above we get that $\Pr(g(\mathrm{D}) > 4\sqrt{\frac{K}{m}} + \epsilon) < \exp(-\frac{m\epsilon^2}{2K})$ Rearranging terms and substituting δ on the RHS proves the inequality. □

[1]Department of Computer Science and Engineering, IIT Bombay, Mumbai, India. Correspondence to: Aviral Kumar <aviralkumar2907@gmail.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

Dataset	Model	Improvement	
		Accuracy	ECE
MNIST	LeNet 5	+0.02%	-0.30%
CIFAR 10	Resnet 50	-0.02%	-2.05%
CIFAR 10	Resnet 110	+0.01%	-2.50%
CIFAR 100	Resnet 32	+1.20%	-2.58%

Table 1. The change in ECE and accuracy compared to Baseline when fine-tuning a pre-trained model using MMCE.

2. Proof of Theorem 3

Proof. Starting from the RHS,

$$\mathrm{ECE}(P(r,c)) = \mathbb{E}_r\left[|r - \tfrac{p(c=1,r)}{p(r)}|\right]$$
$$= \mathbb{E}_r\left[|\tfrac{r\cdot p(c=0,r) - (1-r)\cdot p(c=1,r)}{p(r)}|\right]$$
$$= \int_r |r\cdot p(c=0,r) - (1-r)\cdot p(c=1,r)|dr$$

Now, we can rewrite

$$\mathrm{MMCE}(P(r,c)) = \sup_{f\in\mathcal{F}} \sum_c \int_r (c-r)\cdot f(r)dP(r,c)$$
$$= \int_r \big((1-r)\cdot p(c=1,r) - r\cdot p(c=0,r)\big)\cdot f(r)dr$$

It is easy to see that $M(\mathcal{D}_L, P(r,c)) = \mathrm{ECE}(P(r,c))$ where $\mathcal{D}_L = \{f \mid ||f||_\infty \leq L\}$. We pick $f(r) = L\cdot sign((1-r)\cdot p(c=1,r) - r\cdot p(c=0,r))$. Note that the set $\mathcal{D}_L$ also includes discontinuous functions. In contrast $\mathrm{MMCE}(P) = M(\mathcal{F}_K, P)$ where $\mathcal{F}_K$ is the space of continuous functions in RKHS with maximum kernel value limited to K. $\mathcal{F}_K$ is included in $\mathcal{D}_L$ when $L \geq \sqrt{K}$. This proves our required result.

□

3. Finetuning using MMCE

Table 1 shows the ECE and Accuracy numbers for some models when MMCE is used to finetune them, post-training.

4. Comparison of Running times

Table 2 summarizes the running time per epoch for training using MMCE+NLL and NLL objectives. MMCE, on an average, doesn't create an overhead of more than 10% over the baseline.

Dataset	Model	Baseline	MMCE
CIFAR 10	Resnet 50	4.6s	4.7s
CIFAR 10	Resnet 110	10.5s	11.1s
CIFAR 100	W. Resnet 28-10	48.0s	55.0s
CIFAR 100	Resnet 32	11.5s	11.5s
20 Newsgroups	Global Pool	5.7s	6.0s
IMDB Reviews	HAN	226.0s	227.0s
UCI HAR	LSTM	0.6s	0.7s

Table 2. Running time per epoch in seconds for Baseline and MMCE methods for different models and datasets

References

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. J. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.