Function Norms for Neural Networks: Supplementary Material*

Amal Rannen-Triki †[1], Maxim Berman[1], Vladimir Kolmogorov[2], and Matthew B. Blaschko[1]

[1]KU Leuven, Belgium; email: name.surname@esat.kuleuven.be
[2]Institute of Science and Technology, Austria; email: vnk@ist.ac.at

1. On the complexity of neural network norm computation

1.1. Two-layer networks

We will define a norm on two layer ReLU networks by defining an inner product through a RKHS construction.

A two layer network with a single output can be written as

$$f(x) = w_1^T \sigma(W_2 x) \tag{1}$$

where $w_1 \in \mathbb{R}^m$ and $W_2 \in \mathbb{R}^{m \times d}$, and $\sigma(x) = \max(0, x)$ taken element-wise. In the following, such a network is represented by: (w_1, W_2), and we note:

Lemma 1 (Addition) *Let $u = (u_1, U_2)$ and $v = (v_1, V_2)$ be two functions represented by a 2-layer neural network. Then, the function $u + v$ is represented by $\left(\begin{pmatrix} u_1 \\ v_1 \end{pmatrix}, \begin{pmatrix} U_2 \\ V_2 \end{pmatrix}\right)$.*

Lemma 2 (Scalar multiplication) *Let $u = (u_1, U_2)$ be a function represented by a 2-layer neural network. Then, the function αu is represented by $(\alpha u_1, U_2)$.*

These operations define a linear space. A two-layer network f is preserved when scaling the ith row of W_2 by $\alpha > 0$ and the ith entry of w_1 by α^{-1}. We therefore assume that each row of W_2 is scaled to have unit norm, removing any rows of W_2 that consist entirely of zero entries.[1] Now, we define an inner product as follows:

Definition 1 (An inner product between 2-layer networks) *Let k be a characteristic kernel [7] [2] over $\mathbb{R}^d$. Let u and v be two-layer networks represented by $(u_1, U_2) \in \mathbb{R}^{m_u} \times \mathbb{R}^{m_u \times d}$ and $(v_1, V_2) \in \mathbb{R}^{m_v} \times \mathbb{R}^{m_v \times d}$, respectively, where no row of U_2 or V_2 is a zero vector, and each row has unit norm. Define*

$$\langle u, v \rangle_{\mathcal{H}} := \sum_{i=1}^{m_u} \sum_{j=1}^{m_v} [u_1]_i [v_1]_j k([U_2]_{i,:}, [V_2]_{i,:}), \tag{2}$$

where $[M]_{i,:}$ denotes the ith row of M, which induces the norm $\|f\|_{\mathcal{H}} := \sqrt{\langle f, f \rangle_{\mathcal{H}}}$.

We note that k must be characteristic to guarantee the property of a norm that $\|f\|_{\mathcal{H}} = 0 \iff f = \mathbf{0}$.

This inner product inherits the structure of the linear space defined above. Using the addition (Lemma 1) and scalar multiplication (Lemma 2) operations, verifying that Equation (2) satisfies the properties of an inner product is now a basic exercise.

We may take Equation (2) as the basis of a constructive proof that two-layer networks have polynomial time computable norm. To summarize, to compute such a norm, we need to:

1. Normalize $w = (w_1, W_2)$ so W_2 has rows with unit norm, and no row is a zero vector, which takes $\mathcal{O}(md)$ time;
2. Compute $\langle w, w \rangle$ according to Equation (2), which is quadratic in m times the complexity of $k(x, x')$;
3. Compute $\sqrt{\langle w, w \rangle}$.

Therefore, assuming we allow square roots as operations, the constructed norm can be computed in a quadratic time in the cost of the evaluation of k. For example, for a Gaussian kernel $k(x, x') := \exp(-\gamma \|x - x'\|^2)$, and allowing exp as operation, the cost of the kernel evaluation is linear in the input dimension and the cost of the constructed norm is quadratic in the number of hidden units.

*The first workshop on Statistical Deep Learning for Computer Vision, in *Seoul, Korea*, 2019.

†This author is currently affiliated with Deepmind.

[1]The choice of vector norm is not particularly important. For concreteness we may assume it be L_1 normalized, which when considering rational weights with bounded coefficients, preserves polynomial boundedness after normalization.

[2]See main paper, Definition 1

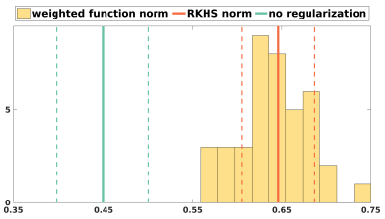

Figure 1: Histogram of accuracies with weighted function norm on the Oxford Flowers dataset over 10 trials with 4 different regularization sample sizes, compared to the mean and standard deviation of RKHS norm performance, and the mean and standard deviation of the accuracy obtained without regularization.

2. Additional experiments

2.1. Oxford Flowers classification with kernelized logistic regression

This experiment shows that the approximate L_2 norm and the RKHS norm have similar behavior on the test data. We consider the 17 classes Oxford Flower Dataset, composed of 80 images per class, and precomputed kernels that have been shown to give good performance on a classification task [5, 6]. We have taken the mean of Gaussian kernels as described in [1]. To test the effect of the regularization, we train the logistic regression on a subset of 10% of the data, and test on 20% of the samples. The remaining 70% are used as potential samples for regularization. For both regularizers, the regularization parameter is selected by a 3-fold cross validation. For the approximate norm regularization, we used a 4 different sample sizes ranging from 20% to 70% of the data. This procedure is repeated on 10 different splits of the data for a better estimate. The optimization is performed by quasi-Newton gradient descent, which is guaranteed to converge due to the convexity of the objective. Figure 1 shows the means and standard deviations of the accuracy on the test set obtained without regularization, and with regularization using the RKHS norm, along with the histogram of accuracies obtained with the weighted norm regularization with the different sample sizes and across the 10 trials. This figure demonstrates the equivalent effect of both regularizers.

The use of the weighted function norm is more useful for DNNs, where we showed that polynomial time exact norms do not exist. The next experiment shows the efficiency of the proposed regularization strategy other regularization strategies: Weight decay [4], dropout [2] and batch normalization [3].

References

[1] Peter V. Gehler and Sebastian Nowozin. On feature combination for multiclass object classification. In *International Conference on Computer Vision*, pages 221–228, 2009.

[2] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv:1207.0580*, 2012.

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 2015.

[4] Anders Krogh and John A Hertz. A simple weight decay can improve generalization. *Advances in Neural Information Processing Systems*, 4:950–957, 1995.

[5] Maria-Elena Nilsback and Andrew Zisserman. A visual vocabulary for flower classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1447–1454, 2006.

[6] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

[7] Bharath K Sriperumbudur, Kenji Fukumizu, and Gert RG Lanckriet. Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, 12(Jul):2389–2410, 2011.