Training Binarized Neural Networks the Easy Way

UNIVERSITY OF OXFORD

Alasdair Paren *aparen@robots.ox.ac.uk*, Rudra P. K. Poudel *rudra.poudel@crl.toshiba.co.uk*

TOSHIBA

Binarized Neural Networks

Binarized Neural Networks (BNN) are an extreme form of quantised neural networks. Specifically, where the majority of weights and activations are constrained to the set $\{-1, 1\}$. BNN have the following advantages:

- Arithmetic operations can be replaced with faster bit-wise alternatives
- Lower computational and energy cost at inference
- Can be used on lightweight mobile hardware
- Up to 32 time less memory storage requirement
- Up to 58 time faster on the CPU [5]
- Around 5 time faster on the GPU [3]

Existing Training Methods

Notation. Vector of all parameters $\boldsymbol{w} \in \mathbb{R}^d$. Vector of parameters to take binary values $\boldsymbol{w}_t^b \in \mathbb{R}^p$. Vector of parameters to retain real values $\boldsymbol{w}_t^r \in \mathbb{R}^{d-p}$. $\tilde{\boldsymbol{w}}^r$ typically contains weights in the first and last layers, batch norm layers, biases, and bottleneck layers.

Straight Through Estimator Method [3] (STE)

$$\boldsymbol{w}_t^b = \mathrm{sign}(\tilde{\boldsymbol{w}}_t^b),$$
$$\tilde{\boldsymbol{w}}_{t+1}^b = \Pi(\tilde{\boldsymbol{w}}_t^b - \eta_t \nabla \ell_{z_t}(\boldsymbol{w}_t^b)), .$$

where η_t is the learning rate and Π is projection onto the interval $[-1, 1]$, and in practice Adam is used.

Mirror Descent View for BNN [1] (BMD)

$$\boldsymbol{w}_t^b = \tanh(\beta_t \tilde{\boldsymbol{w}}_t^b),$$
$$\tilde{\boldsymbol{w}}_{t+1}^b = \tilde{\boldsymbol{w}}_t^b - \eta_t \nabla \ell_{z_t}(\boldsymbol{w}_t^b), .$$

Again the Adam update is used in practice.

Binary Optimiser [2] (BOP)

$$\boldsymbol{m}_t^b = (1 - \eta_t)\boldsymbol{m}_{t-1}^b - \eta_t \nabla \ell_{z_t}(\boldsymbol{w}_t^b), \quad \boldsymbol{m}_0^b = 0$$
$$\forall w^b \in \boldsymbol{w}^b : \textbf{if } |m_{t+1}^b| > \tau \textbf{ and } \mathrm{sign}(m_t^b) = \mathrm{sign}(w_t^b):$$
$$w_{t+1}^b = -w_t^b$$

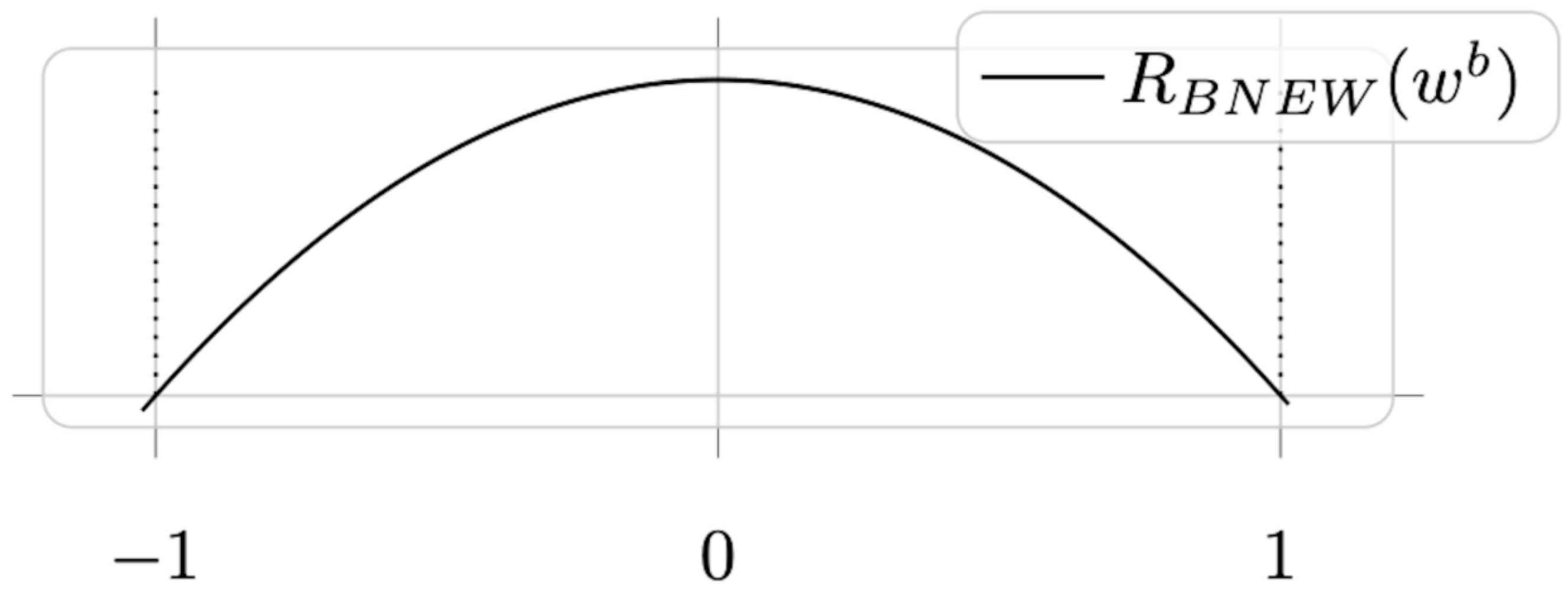
Objective

Unlike most existing methods for optimising BNN we relax the constraint on the binary parameters to its convex hull and add a regulation function R_{BNEW} that penalises solutions with $\boldsymbol{w}_\star^b \notin \{-1, 1\}^p$

$$\boldsymbol{w}_\star \in \underset{\boldsymbol{w} \in \Omega}{\mathrm{argmin}}\, F_\lambda \triangleq f(\boldsymbol{w}) + \lambda R_{BNEW}(\boldsymbol{w}),$$

where $\Omega \triangleq [-1, 1]^p \cup \mathbb{R}^{d-p}$.

Regularisation



We use the regularisation function $R_{BNEW}(\boldsymbol{w}) \triangleq -\|\boldsymbol{w}^b\|^2 + p$. This regularisation can be seen at negative weight decay.

Update

The resulting algorithm which we name BNEW (Binarized Networks the Easy Way) uses the update:

$$\boldsymbol{w}_{t+1}^b = \Pi\left((1 + 2\lambda_t \eta_t)(\boldsymbol{w}_t^b - \eta_t \nabla \ell_{z_t}(\boldsymbol{w}_t^b))\right).$$

Again, the Adam update is used. η_t follows a linearly decaying schedule, that is reset after step 1 below.

Training Procedure

We use the following training procedure:

1. Train the network to have binary activations and real valued parameters ($\lambda_t = 0$)
2. Slowly binarize the parameters $\boldsymbol{w}^b$ by linearly increasing $\lambda_t = \beta \cdot t$
3. Project $\boldsymbol{w}_{t=t_{lock}}^b$ onto $\{-1, 1\}^p$, set $\nabla \boldsymbol{w}_{t>t_{lock}}^b = 0$
4. Fine-tune only the real valued parameters $\tilde{\boldsymbol{w}}^r$.

Results

Results training a small BNN architecture on CIFAR:

Data Set	CIFAR-10		CIFAR-100	
Distillation	No	Yes	No	Yes
Real Valued	$91.4\sigma0.3$	-	$67.1\sigma0.7$	-
STE	$84.3\sigma.3$	$85.1\sigma.4$	$55.0\sigma.5$	$56.8\sigma.3$
BMD	$84.0\sigma.4$	$84.9\sigma.3$	$54.8\sigma.5$	$56.8\sigma.5$
BOP	$84.5\sigma.2$	$85.2\sigma.4$	$55.3\sigma.4$	$57.7\sigma.4$
BNEW	$84.5\sigma.3$	$85.1\sigma.4$	$55.0\sigma.3$	$57.5\sigma.3$

Training a ReActNet architecture [4] on ImageNet:

Optimiser	Accuracy
STE	69.4
BNEW	69.7

Properties of BNEW

- Very simple
- Strong empirical results
- Strong theoretical justification
- Poor estimate of performance during training
- Best results require long training time (large T)
- The optimal value of β depends on T

References

[1] Thalaiyasingam Ajanthan, Kartik Gupta, Philip Torr, Richad Hartley, and Puneet Dokania. Mirror descent view for neural network quantization. *International Conference on Artificial Intelligence and Statistics*, 2021.

[2] Koen Helwegen, James Widdicombe, Lukas Geiger, Zechun Liu, Kwang-Ting Cheng, and Roeland Nusselder. Latent weights do not exist: Rethinking binarized neural network optimization. *Neural Information Processing Systems*, 2019.

[3] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. *Neural Information Processing Systems*, 2016.

[4] Zechun Liu, Zhiqiang Shen, Marios Savvides, and Kwang-Ting Cheng. Reactnet: Towards precise binary neural network with generalized activation functions. *European Conference on Computer Vision*, 2020.

[5] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. *European Conference on Computer Vision*, 2016.