On the Impact of Weight Quantization on Deep Neural Network Uncertainty

Shuang Liang, Xun Lu, Zi-Ang Liu, Ming-Liang Wang, Yan Lyu✉, and Shao-Qun Zhang✉
LAMDA Group, Nanjing University, China

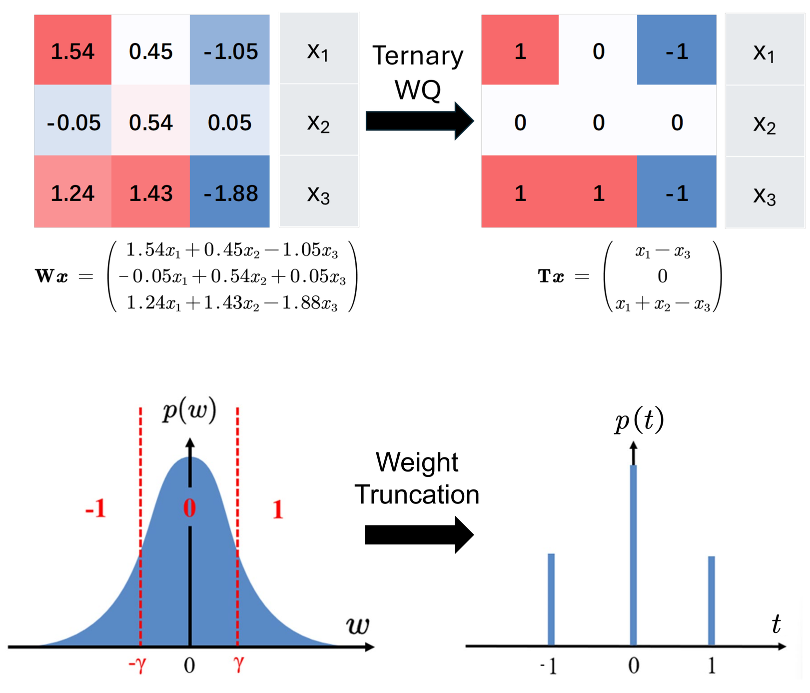
Contact
liangs@lamda.nju.edu.cn
zhangsq@lamda.nju.edu.cn

TL;DR

- We find that Weight Quantization (WQ) significantly increase DNN uncertainty via the novel EMP estimator, and propose the MOMA to reduce the uncertainty.

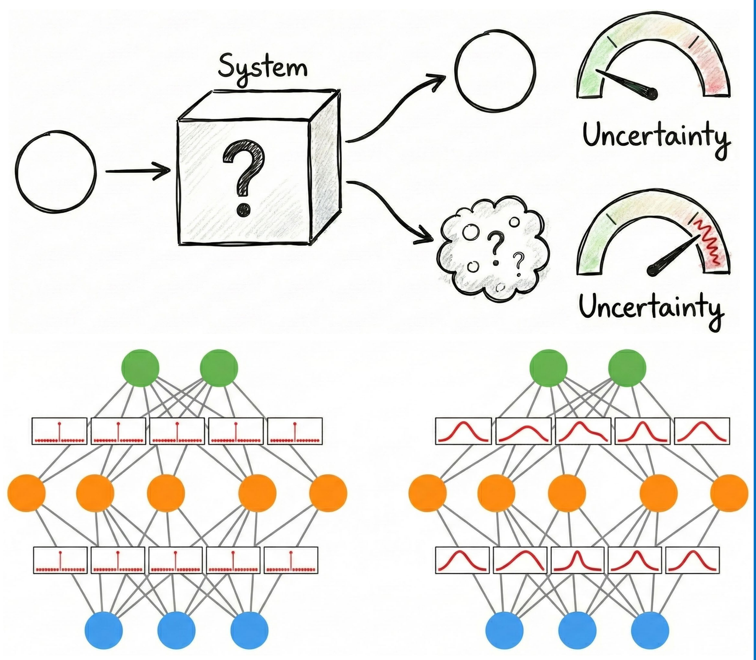
Weight Quantization (WQ)

- WQ is a popular lightweight method that accelerates inference with accuracy comparable to full-precision counterparts by encoding model weights w into n bits.
- Examples:
  - $n=1, w\in\{-1,1\}$ (BWN)
  - $n=1.67, w\in\{-1,0,1\}$ (TWN)
- Ternary WQ via truncation is standard, where the truncation threshold can be set as the mean of weight matrix.



Uncertainty Quantification (UQ)

- UQ is a technique to quantitatively characterize and estimate the variability in system output, which originates from the intrinsic stochasticity of the system.
- Given $Y=F(X;\Theta)$,
  - Stochastic F induces stochastic Y, e.g., stochastic sampling.
  - Stochastic X induces stochastic Y, e.g., noisy instances.
  - Stochastic Θ induces stochastic Y, e.g., modeling weights as random variables.



Research Questions and Contributions

- Q1: How to quantify WQ's impact on DNN uncertainty?
- C1: To ensure a fair comparison between the uncertainty before and after WQ, we propose the EMP, a unified uncertainty estimator compatible with both full-precision and ternary weights. The EMP provides an exact solution to the output covariance under very mild assumptions.
- Q2: Whether and to what extend does the WQ increase DNN uncertainty? Moreover, if the WQ increases DNN uncertainty, how can it be reduced?
- C2: Through the EMP uncertainty estimator, we observe that WQ significantly increases DNN uncertainty. Based on this quantification, we propose the MOMA to reduce the uncertainty increase caused by WQ.

EMP Algorithm for Uncertainty Quantification

$\boldsymbol{h}=\sigma(\boldsymbol{a})$ — $\boldsymbol{\mu}_{\boldsymbol{h}}, \boldsymbol{\Sigma}_{\boldsymbol{h}}$

Nonlinear Activation — Induce → Nonlinear Computation of EMP

Assume $\boldsymbol{a}\sim\mathcal{N}$

$$\mathbb{E}[h_i]=\sigma_i\left(r_i\Phi(r_i)+\phi(r_i)\right),$$
$$\mathbb{V}(h_i)=\sigma_i^2\left[(1+r_i^2)\Phi(r_i)+r_i\phi(r_i)\right]-\mathbb{E}[h_i]^2,$$
$$\mathrm{Cov}(h_1,h_2)=\sigma_1\sigma_2\big[r_2\phi(r_1)\Phi(c_2)+r_1\phi(r_2)\Phi(c_1)+(r_1r_2+\rho)\Phi_2(r_1,r_2;\rho)+\sqrt{1-\rho^2}\,\phi(r_2)\phi(c_1)\big]-\mathbb{E}[Z_1]\mathbb{E}[Z_2].$$

$\boldsymbol{a}=\mathbf{W}\boldsymbol{h}'+\boldsymbol{b}$ — $\boldsymbol{\mu}_{\boldsymbol{a}}, \boldsymbol{\Sigma}_{\boldsymbol{a}}$

Linear Transformation — Induce → Linear Computation of EMP

$$\mathbb{E}[a_i]=\boldsymbol{\mu}_{\mathbf{w}_i}^\top\boldsymbol{\mu}_{\boldsymbol{h}'}+b_i,$$
$$\mathbb{V}(a_i)=\boldsymbol{\mu}_{\boldsymbol{h}'}^\top\boldsymbol{\Sigma}_{\mathbf{W}_i}\boldsymbol{\mu}_{\boldsymbol{h}'}+\mathrm{tr}\left(\boldsymbol{\Sigma}_{\mathbf{W}_i}\boldsymbol{\Sigma}_{\boldsymbol{h}'}\right)+\boldsymbol{\mu}_{\mathbf{w}_i}^\top\boldsymbol{\Sigma}_{\boldsymbol{h}'}\boldsymbol{\mu}_{\mathbf{W}_i},$$
$$\mathrm{Cov}(a_1,a_2)=\mathrm{tr}\left(\mathrm{Cov}(\mathbf{W}_1,\mathbf{W}_2)\boldsymbol{\Sigma}_{\boldsymbol{h}'}\right)+\boldsymbol{\mu}_{\mathbf{W}_1}^\top\boldsymbol{\Sigma}_{\boldsymbol{h}'}\boldsymbol{\mu}_{\mathbf{W}_2}+\boldsymbol{\mu}_{\boldsymbol{h}'}^\top\mathrm{Cov}(\mathbf{W}_1,\mathbf{W}_2)\boldsymbol{\mu}_{\boldsymbol{h}'}.$$

$\boldsymbol{h}'$ — $\boldsymbol{\mu}_{\boldsymbol{h}'}, \boldsymbol{\Sigma}_{\boldsymbol{h}'}, \boldsymbol{\mu}_{\mathbf{W}}, \boldsymbol{\Sigma}_{\mathbf{W}}$

Data propagation — **Moment propagation**

Exactly propagate the **moments**, i.e., mean and covariance, of the input layer by layer.

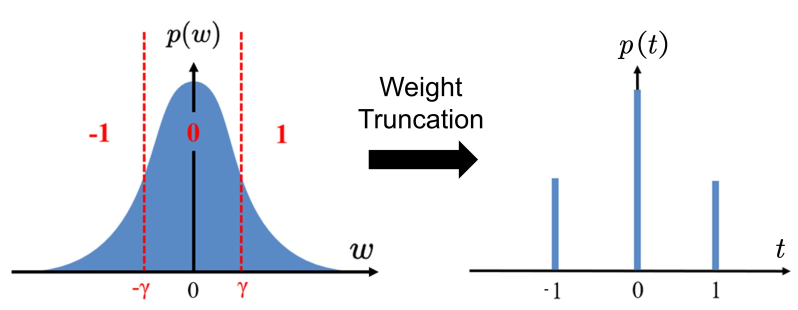
Experimental Results of EMP

- Performance evaluation: We treat the classical Monte Carlo (MC) estimator as the ground truth, with other theoretical estimators (E) being evaluated based on their closeness to it.
- Metric: $\mathbb{E}[r_o(\boldsymbol{x})]\pm\sqrt{\mathbb{V}[r_o(\boldsymbol{x})]}$, where $r_o(\boldsymbol{x})=\mathbb{V}_o^{\mathrm{MC}}(\boldsymbol{x})/\ \mathbb{V}_o^{\mathrm{E}}(\boldsymbol{x})$.
- Optimal: 1 ± 0.
- Partial results:

Datasets	o	FP-LeNet-5			T-LeNet-5		
		EMP	MP	PL-DNN	EMP	MP	PL-DNN
MNIST	0	1.0007 ± 0.0101	1.1809 ± 0.0217	1.2772 ± 0.0153	1.0023 ± 0.0109	–	–
	1	0.9999 ± 0.0103	1.1793 ± 0.0238	1.2759 ± 0.0139	1.0023 ± 0.0121	–	–
	2	0.9990 ± 0.0118	1.1794 ± 0.0239	1.2748 ± 0.0170	1.0021 ± 0.0117	–	–
	3	1.0008 ± 0.0104	1.1808 ± 0.0215	1.2770 ± 0.0153	1.0022 ± 0.0108	–	–
	4	1.0011 ± 0.0096	1.1807 ± 0.0239	1.2775 ± 0.0136	1.0023 ± 0.0107	–	–
	5	1.0010 ± 0.0106	1.1815 ± 0.0237	1.2773 ± 0.0149	1.0022 ± 0.0109	–	–
	6	1.0011 ± 0.0096	1.1818 ± 0.0246	1.2774 ± 0.0130	1.0023 ± 0.0105	–	–
	7	0.9996 ± 0.0099	1.1803 ± 0.0243	1.2755 ± 0.0143	1.0021 ± 0.0101	–	–
	8	1.0014 ± 0.0098	1.1814 ± 0.0218	1.2422 ± 0.0140	1.0021 ± 0.0107	–	–
	9	0.9999 ± 0.0100	1.1801 ± 0.0225	1.2759 ± 0.0142	1.0019 ± 0.0106	–	–
CIFAR-10	0	1.0098 ± 0.0099	1.5029 ± 0.0177	1.6167 ± 0.0334	1.0141 ± 0.0113	–	–
	1	1.0092 ± 0.0105	1.5020 ± 0.0207	1.6156 ± 0.0299	1.0136 ± 0.0115	–	–
	2	1.0090 ± 0.0127	1.5017 ± 0.0218	1.6154 ± 0.0358	1.0141 ± 0.0110	–	–
	3	1.0100 ± 0.0110	1.5031 ± 0.0189	1.6169 ± 0.0338	1.0141 ± 0.0115	–	–
	4	1.0106 ± 0.0098	1.5041 ± 0.0179	1.6179 ± 0.0307	1.0137 ± 0.0116	–	–
	5	1.0099 ± 0.0109	1.5030 ± 0.0200	1.6167 ± 0.0311	1.0139 ± 0.0118	–	–
	6	1.0103 ± 0.0094	1.5036 ± 0.0185	1.6174 ± 0.0282	1.0139 ± 0.0109	–	–
	7	1.0090 ± 0.0099	1.5017 ± 0.0184	1.6153 ± 0.0307	1.0139 ± 0.0120	–	–
	8	1.0103 ± 0.0093	1.5036 ± 0.0179	1.6173 ± 0.0312	1.0140 ± 0.0118	–	–
	9	1.0095 ± 0.0100	1.5025 ± 0.0179	1.6162 ± 0.0314	1.0141 ± 0.0109	–	–

- Near-ideal uncertainty estimation performance.
- Surpass the MP (ICLR'19) and the PL-DNN (CVPR'18).

MOMA Algorithm for Uncertainty Reduction



Theorem 9. *Under Assumptions on the Gaussian pre-activation variables and the ternary-valued weights, if $q-p=0$ and $p+q-(q-p)^2=0$ hold for the ternary-valued distribution parameters (p,q) of each layer, then the output covariance satisfies $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{y}}=\mathbf{0}$.*

$\gamma=\mathrm{mean}(\mathbf{W})$ (Li et al. 2016)

↓ Modify

$$\begin{cases}\tilde{\gamma}=\underset{\gamma}{\mathrm{argmin}}\,|g_3(\gamma,\mathbf{W})|+\lambda|g_4(\gamma,\mathbf{W})|,\\ \text{s.t.}\quad \gamma\in((1-c)\gamma,(1+c)\gamma),\end{cases}\quad\text{where}$$

Inspire →

$$\begin{cases}g_1(\gamma,\mathbf{W})=\sum_{i,j}\chi_{(-\infty,-\gamma)}(\mathbf{W}_{ij})/\mathrm{card}(\mathbf{W}),\\ g_2(\gamma,\mathbf{W})=\sum_{i,j}\chi_{(\gamma,+\infty)}(\mathbf{W}_{ij})/\mathrm{card}(\mathbf{W}),\\ g_3(\gamma,\mathbf{W})=g_2(\gamma,\mathbf{W})-g_1(\gamma,\mathbf{W}),\\ g_4(r,\mathbf{W})=g_1(\gamma,\mathbf{W})+g_2(\gamma,\mathbf{W})-(g_3(\gamma,\mathbf{W}))^2,\end{cases}$$

Optimize the output covariance toward zero by **aligning** the first and second central **moments** of the ternary-valued distribution to zero when searching the truncation threshold.

Experimental Results of MOMA

- Performance evaluation: We average the trace of the output covariance across test samples, lower is better.
- Metric: $\mathbb{E}_{\boldsymbol{x}}\left[\mathrm{tr}\left(\boldsymbol{\Sigma}_{\hat{\boldsymbol{y}}(\boldsymbol{x})}\right)\right]$.
- Optimal: 0.
- Partial results:

Models	Accuracy	Uncertainty	Reduction
FP-MLP-2	$0.9761_{0.0002}$	$1.4697_{0.0082}\times10^{3}$	$68.08_{0.1389}\%$
T-MLP-2	$0.9752_{0.0005}$	$1.2572_{0.0236}\times10^{5}$	
T-MLP-2-MOMA	$0.9751_{0.0006}$	$4.0129_{0.0039}\times10^{4}$	
FP-LeNet-5	$0.9911_{0.0001}$	$1.7728_{0.0232}\times10^{3}$	$98.55_{0.0025}\%$
T-LeNet-5	$0.9903_{0.0003}$	$2.8113_{0.0490}\times10^{5}$	
T-LeNet-5-MOMA	$0.9885_{0.0006}$	$4.0906_{0.1433}\times10^{3}$	
FP-VGG-13	$0.9985_{0.0001}$	$4.6029_{0.2302}\times10^{-1}$	$20.86_{2.5366}\%$
T-VGG-13	$0.9954_{0.0005}$	$1.4500_{0.0210}\times10^{1}$	
T-VGG-13-MOMA	$0.9957_{0.0005}$	$1.1463_{0.0198}\times10^{1}$	
FP-ResNet-18	$0.9963_{0.0001}$	$6.9757_{0.3743}\times10^{-3}$	$44.27_{4.4212}\%$
T-ResNet-18	$0.9938_{0.0002}$	$4.2808_{0.1772}\times10^{-1}$	
T-ResNet-18-MOMA	$0.9951_{0.0002}$	$2.4067_{0.0501}\times10^{-1}$	

- Significant uncertainty reduction.
- Preserve the accuracy.