POST-TRAINING QUANTIZATION FOR NEURAL NETWORKS WITH PROVABLE GUARANTEES

JINJIE ZHANG, YIXUAN ZHOU, AND RAYAN SAAB

Abstract. While neural networks have been remarkably successful in a wide array of applications, implementing them in resource-constrained hardware remains an area of intense research. By replacing the weights of a neural network with quantized (e.g., 4-bit, or binary) counterparts, massive savings in computation cost, memory, and power consumption are attained. To that end, we generalize a post-training neural-network quantization method, GPFQ, that is based on a greedy path-following mechanism. Among other things, we propose modifications to promote sparsity of the weights, and rigorously analyze the associated error. Additionally, our error analysis expands the results of previous work on GPFQ to handle general quantization alphabets, showing that for quantizing a single-layer network, the relative square error essentially decays linearly in the number of weights – i.e., level of over-parametrization. Our result holds across a range of input distributions and for both fully-connected and convolutional architectures thereby also extending previous results. To empirically evaluate the method, we quantize several common architectures with few bits per weight, and test them on ImageNet, showing only minor loss of accuracy compared to unquantized models. We also demonstrate that standard modifications, such as bias correction and mixed precision quantization, further improve accuracy.

1. Introduction

Over the past decade, deep neural networks (DNNs) have achieved great success in many challenging tasks, such as computer vision, natural language processing, and autonomous vehicles. Nevertheless, over-parameterized DNNs are computationally expensive to train, memory intensive to store, and energy consuming to apply. This hinders the deployment of DNNs to resource-limited applications. Therefore, model compression without significant performance degradation is an important active area of deep learning research [11, 6, 10]. One prominent approach to compression is *quantization*. Here, rather than adopt a 32-bit floating point format for the model parameters, one uses significantly fewer bits for representing weights, activations, and even gradients. Since the floating-point operations are substituted by more efficient low-bit operations, quantization can reduce inference time and power consumption.

Following [16], we can classify quantization methods into two categories: *quantization-aware training* and *post-training quantization*. The fundamental difficulty in quantization-aware training stems from the fact that it reduces to an integer programming problem with a non-convex loss function, making it NP-hard in general. Nevertheless, many well-performing heuristic methods exist, e.g., [4, 12, 35, 15, 33, 21, 31]. Here one, for example, either modifies the training procedure to produce quantized weights, or successively quantizes each layer and then retrains the subsequent layers. Retraining is a powerful, albeit computationally

intensive way to compensate for the accuracy loss resulting from quantization and it remains generally difficult to analyze rigorously.

Hence, much attention has recently been dedicated to post-training quantization schemes, which directly quantize pretrained DNNs having real-valued weights, without retraining. These quantization methods either rely on a small amount of data [1, 3, 34, 24, 14, 30, 19, 22] or can be implemented without accessing training data, i.e. data-free compression [23, 2, 32, 20].

1.1. **Related Work.** We now summarize some prior work on post-training quantization methods. The majority of these methods aim to reduce quantization error by minimizing a mean squared error (MSE) objective, e.g. $\min_{\alpha>0}\left\|W-\alpha\left\lfloor\frac{W}{\alpha}\right\rceil\right\|_F$, where W is a weight matrix and $\lfloor\cdot\rceil$ is a round-off operator that represents a map from the set of real numbers to the low-bit alphabet. Generally $\lfloor\cdot\rceil$ simply assigns numbers in different intervals or "bins" to different elements of the alphabet. Algorithms in the literature differ in their choice of $\lfloor\cdot\rceil$, as they use different strategies for determining the quantization bins. However, they share the property that once the quantization bins are selected, weights are quantized independently of each other. For example, Banner et al. [1] (see also [34]) choose the thresholds to minimize a MSE metric. Their numerical results also show that for convolutional networks using different quantization thresholds "per-channel" and bias correction can improve the accuracy of quantized models. Choukroun et al. [3] solve a minimum mean squared error (MMSE) problem for both weights and activations quantization. Based on a small calibration data set, Hubara et al. [14] suggest a per-layer optimization method followed by integer programming to determine the bit-width of different layers. A bit-split and stitching technique is used by [30] that "splits" integers into multiple bits, then optimizes each bit, and finally stitches all bits back to integers. Li et al. [19] leverage the basic building blocks in DNNs and reconstructs them one-by-one. As for data-free model quantization, there are different strategies, such as weight equalization [23], reconstructing calibration data samples according to batch normalization statistics (BNS) [2, 32], and adversarial learning [20].

1.2. **Contribution.** In spite of reasonable heuristic explanations and empirical results, all quantization methods mentioned in Section 1.1 lack rigorous theoretical guarantees. Recently, Lybrand and Saab [22] proposed and analyzed a method for quantizing the weights of pretrained DNNs called *greedy path following quantization* (GPFQ), see Section 2.2 for details. In this paper, we substantially improve GPFQ's theoretical analysis, propose a modification to handle convolutional layers, and propose a sparsity promoting version to encourage the algorithm to set many of the weights to zero. We demonstrate that the performance of our quantization methods is not only good in experimental settings, but, equally importantly, has favorable and rigorous error guarantees. Specifically, the contributions of this paper are threefold:

1. We generalize the results of [22] in several directions. Indeed, the results of [22] apply only to alphabets, $\mathcal{A}$, of the form $\mathcal{A}=\{0,\pm 1\}$ and standard Gaussian input because the proof technique in [22] relies heavily on properties of Gaussians and case-work over elements of the alphabet. It also requires the assumption that floating point weights are ϵ-away from

alphabet elements. In contrast, by using a different and more natural proof technique, our results avoid this assumption and extend to general alphabets like $\mathcal{A}$ in (3) and make the main result in [22] a special case of our Theorem 3.2, which in turn follows from Theorem 3.1. Moreover, we extend the class of input vectors for which the theory applies. For example, in Section 3, we show that if the input data $X \in \mathbb{R}^{m\times N_0}$ is either bounded or drawn from a mixture of Gaussians, then the relative square error of quantizing a neuron $w \in \mathbb{R}^{N_0}$ satisfies the following inequality with high probability:

$$\frac{\|Xw - Xq\|_2^2}{\|Xw\|_2^2} \lesssim \frac{m \log N_0}{N_0} \tag{1}$$

where $q \in \mathcal{A}^{N_0}$ is the quantized weights. A mixture of Gaussians is a reasonable model for the output of some of the deeper layers in neural networks that focus on classification, thus our results are relevant in those contexts. Further, to handle convolutional neural networks (CNNs), we introduce a modification to GPFQ in Section 5.1 that relies on random subsampling to make quantizing DNNs practically feasible with large batch size m. This also allows us to obtain quantization error bounds that resemble (1), for single-layer CNNs in Section 3.3.

2. In order to reduce the storage, computational, and power requirements of DNNs one complimentary approach to quantization is to sparsify the weights, i.e., set many of them to zero. In Section 4, we propose modifications to GPFQ that leverage soft and hard thresholding to increase sparsity of the weights of the *quantized* neural networks. We present error bounds, similar to the ones in Theorem 3.1, and provide their proofs in Appendix E.

3. We provide extensive numerical experiments to illustrate the performance of GPFQ and its proposed modifications on common computer vision DNNs. First, we provide comparisons with other post-training quantization approaches (Section 5) and show that GPFQ achieves near-original model performance using 4 bits and that the results for 5 bits are competitive with state-of-the-art methods. Our experiments also demonstrate that GPFQ is compatible with various ad-hoc performance enhancing modifications such as bias correction [1], unquantizing the last layer [35, 19], and mixed precision [7, 2]. To illustrate the effects of sparsity, we further explore the interactions among prediction accuracy, sparsity of the weights, and regularization strength in our numerical experiments. Our results show that one can achieve near-original model performance even when half the weights (or more) are quantized to zero.

2. Preliminaries

In this section, we first introduce the notation that will be used throughout this paper and then recall the original GPFQ algorithm in [22].

2.1. Notation.

Various positive absolute constants are denoted by C, c. We use $a \lesssim b$ as shorthand for $a \leq Cb$, and $a \gtrsim b$ for $a \geq Cb$. Let $S \subseteq \mathbb{R}^n$ be a Borel set. $\mathrm{Unif}(S)$ denotes the uniform distribution over S. An L-layer multi-layer perceptron, Φ, acts on a vector $x \in \mathbb{R}^{N_0}$ via

$$\Phi(x) := \varphi^{(L)} \circ A^{(L)} \circ \cdots \circ \varphi^{(1)} \circ A^{(1)}(x) \tag{2}$$

where $\varphi^{(i)} : \mathbb{R}^{N_i} \to \mathbb{R}^{N_i}$ is an activation function acting entrywise, and $A^{(i)} : \mathbb{R}^{N_{i-1}} \to \mathbb{R}^{N_i}$ is an affine map given by $A^{(i)}(z) := W^{(i)\top} z + b^{(i)}$. Here, $W^{(i)} \in \mathbb{R}^{N_{i-1} \times N_i}$ is a weight matrix and $b^{(i)} \in \mathbb{R}^{N_i}$ is a bias vector. Since $w^\top x + b = \langle (w, b), (x, 1) \rangle$, the bias term $b^{(i)}$ can be treated as an extra row to the weight matrix $W^{(i)}$, so we will henceforth ignore it. We focus on *midtread* alphabets

$$\mathcal{A} = \mathcal{A}_K^\delta := \{\pm k\delta : 0 \le k \le K, k \in \mathbb{Z}\} \tag{3}$$

and their variants

$$\widetilde{\mathcal{A}} = \mathcal{A}_K^{\delta,\lambda} := \{0\} \cup \{\pm(\lambda + k\delta) : 0 \le k \le K, k \in \mathbb{Z}\} \tag{4}$$

where $\delta > 0$ denotes the quantization step size and $\lambda > 0$ is a threshold. For example, $\mathcal{A}_1^1 = \{0, \pm 1\}$ is a ternary alphabet. Moreover, for alphabet $\mathcal{A} = \mathcal{A}_K^\delta$, we define the associated *memoryless scalar quantizer* (MSQ) $\mathcal{Q} : \mathbb{R} \to \mathcal{A}$ by

$$\mathcal{Q}(z) := \operatorname{argmin}_{p \in \mathcal{A}} |z - p| = \delta \operatorname{sign}(z) \min\left\{ \left| \left\lfloor \frac{z}{\delta} + \frac{1}{2} \right\rfloor \right|, K \right\}. \tag{5}$$

Further, the MSQ over $\widetilde{\mathcal{A}} = \mathcal{A}_K^{\delta,\lambda}$ is given by

$$\begin{aligned} \widetilde{\mathcal{Q}}(z) : &= \begin{cases} 0 & \text{if } |z| \le \lambda, \\ \arg\min\limits_{p \in \widetilde{\mathcal{A}}} |z - p| & \text{otherwise} \end{cases} \\ &= \mathbb{1}_{\{|z| > \lambda\}} \operatorname{sign}(z) \left(\lambda + \delta \min\left\{ \left| \left\lfloor \frac{s_\lambda(z)}{\delta} + \frac{1}{2} \right\rfloor \right|, K \right\} \right). \end{aligned} \tag{6}$$

Here, $s_\lambda(z) := \operatorname{sign}(z) \max\{|z| - \lambda, 0\}$ is the *soft thresholding* function and its counterpart, *hard thresholding* function, is defined by

$$h_\lambda(z) := z \mathbb{1}_{\{|z| > \lambda\}} = \begin{cases} z & \text{if } |z| > \lambda, \\ 0 & \text{otherwise.} \end{cases}$$

2.2. **GPFQ.** Given a data set $X \in \mathbb{R}^{m \times N_0}$ with vectorized data stored as rows and a trained neural network Φ with weight matrices $W^{(i)}$, the GPFQ algorithm [22] is a map $W^{(i)} \to Q^{(i)} \in \mathcal{A}^{N_{i-1} \times N_i}$, giving a new quantized neural network $\widetilde{\Phi}$ with $\widetilde{\Phi}(X) \approx \Phi(X)$. The matrices $W^{(1)}, \dots, W^{(L)}$ are quantized sequentially and in each layer every neuron (a column of $W^{(i)}$) is quantized independently of other neurons, which allows parallel quantization across neurons in a layer.

Thus, GPFQ can be implemented recursively. Let $\Phi^{(i)}$, $\widetilde{\Phi}^{(i)}$ denote the original and quantized neural networks up to layer i respectively. Assume the first $i-1$ layers have been quantized and define $X^{(i-1)} := \Phi^{(i-1)}(X)$, $\widetilde{X}^{(i-1)} := \widetilde{\Phi}^{(i-1)}(X) \in \mathbb{R}^{m \times N_{i-1}}$. Then each neuron $w \in \mathbb{R}^{N_{i-1}}$ in layer i is quantized by constructing $q \in \mathcal{A}^{N_{i-1}}$ such that

$$\widetilde{X}^{(i-1)} q = \sum_{t=1}^{N_{i-1}} q_t \widetilde{X}_t^{(i-1)} \approx \sum_{t=1}^{N_{i-1}} w_t X_t^{(i-1)} = X^{(i-1)} w$$

Algorithm 1: Using GPFQ to quantize MLPs

Input: A L-layer MLP Φ with weight matrices $W^{(i)} \in \mathbb{R}^{N_{i-1}\times N_i}$, input mini-batches $\{X_i\}_{i=1}^{L} \subset \mathbb{R}^{m\times N_0}$

for $i = 1$ **to** L **do**
  Phase I: Forward propagation
  Generate $X^{(i-1)} = \Phi^{(i-1)}(X_i) \in \mathbb{R}^{m\times N_{i-1}}$ and $\widetilde{X}^{(i-1)} = \widetilde{\Phi}^{(i-1)}(X_i) \in \mathbb{R}^{m\times N_{i-1}}$
  Phase II: Parallel quantization for $W^{(i)}$
  repeat
  Pick a column (neuron) $w \in \mathbb{R}^{N_{i-1}}$ of $W^{(i)}$ and set $u_0 = 0 \in \mathbb{R}^m$
  for $t = 1$ **to** N_{i-1} **do**
    Implement (9) and $u_t = u_{t-1} + w_t X_t^{(i-1)} - q_t \widetilde{X}_t^{(i-1)}$
  until All columns of $W^{(i)}$ are quantized
  Obtain quantized i-th layer $Q^{(i)} \in \mathcal{A}^{N_{i-1}\times N_i}$

Output: Quantized neural network $\widetilde{\Phi}$

where $X_t^{(i-1)}$, $\widetilde{X}_t^{(i-1)}$ are the t-th columns of $X^{(i-1)}$, $\widetilde{X}^{(i-1)}$. This is done by selecting q_t, for $t = 1, 2, \ldots, N_{i-1}$, so the running sum $\sum_{j=1}^{t} q_j \widetilde{X}_j^{(i-1)}$ tracks its analog $\sum_{j=1}^{t} w_j X_j^{(i-1)}$ as well as possible in an ℓ_2 sense. So,

$$q_t = \arg\min_{p\in\mathcal{A}} \Big\| \sum_{j=1}^{t} w_j X_j^{(i-1)} - \sum_{j=1}^{t-1} q_j \widetilde{X}_j^{(i-1)} - p\widetilde{X}_t^{(i-1)} \Big\|_2^2. \tag{7}$$

This is equivalent to the following iteration, which facilitates the analysis of the approximation error:

$$\begin{cases} u_0 = 0 \in \mathbb{R}^m, \\ q_t = \operatorname{argmin}_{p\in\mathcal{A}} \big\| u_{t-1} + w_t X_t^{(i-1)} - p\widetilde{X}_t^{(i-1)} \big\|_2^2, \\ u_t = u_{t-1} + w_t X_t^{(i-1)} - q_t \widetilde{X}_t^{(i-1)}. \end{cases} \tag{8}$$

By induction, one can verify that $u_t = \sum_{j=1}^{t} (w_j X_j^{(i-1)} - q_j \widetilde{X}_j^{(i-1)})$ for $t = 0, 1, \ldots, N_{i-1}$, and thus $\|u_{N_{i-1}}\|_2 = \|X^{(i-1)} w - \widetilde{X}^{(i-1)} q\|_2$. Moreover, one can derive a closed-form expression of q_t in (8) as

$$q_t = \mathcal{Q}\Big(\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1} + w_t X_t^{(i-1)} \rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2} \Big), \tag{9}$$

which is proved in Lemma A.1. The whole algorithm for quantizing multilayer perceptrons (MLPs) is summarized in Algorithm 1. For the i-th layer, this parallelizable algorithm has run time complexity $O(mN_{i-1})$ per neuron. Note that in order to quantize convolutional neural networks (CNNs), one can simply vectorize the sliding (convolutional) kernels and unfold, i.e., vectorize, the corresponding image patches. Then, taking the usual inner product on vectors, one can reduce to the case of MLPs, also see Section 3.3.

3. New Theoretical Results for GPFQ

In this section, we present error bounds for GPFQ with single-layer networks Φ in (2) with $L = 1$. Since the error bounds associated with the sparse GPFQ in (27) and (28) are very similar to the one we have for (9), we focus on original GPFQ here and leave the theoretical analysis for sparse GPFQ to Appendix E.

In the single-layer case, we quantize the weight matrix $W := W^{(1)} \in \mathbb{R}^{N_0 \times N_1}$ and implement (8) and (9) using $i = 1$. Defining the input data $X := X^{(0)} = \widetilde{X}^{(0)} \in \mathbb{R}^{m \times N_0}$, the iteration can be expressed as

$$\begin{cases} u_0 = 0 \in \mathbb{R}^m, \\ q_t = \mathcal{Q}\big(w_t + \frac{X_t^\top u_{t-1}}{\|X_t\|_2^2}\big), \\ u_t = u_{t-1} + w_t X_t - q_t X_t. \end{cases} \tag{10}$$

Moreover, we have $u_t = \sum_{j=1}^t (w_j X_j - q_j X_j)$ for $t = 1, 2 \ldots, N_0$. Clearly, our goal is to control $\|u_t\|_2$. In particular, given $t = N_0$, we recover the ℓ_2 distance between full-precision and quantized pre-activations: $\|u_{N_0}\|_2 = \|Xw - Xq\|_2$.

3.1. Bounded Input Data.

We start with a quantization error bound where the feature vectors, i.e. columns, of the input data matrix $X \in \mathbb{R}^{m \times N_0}$ are bounded. This general result is then applied to data drawn uniformly from a Euclidean ball, and to Bernoulli random data, showing that the resulting relative square error due to quantization decays linearly with the width N_0 of the network.

Theorem 3.1 (Bounded input data)**.** *Suppose that the columns X_t of $X \in \mathbb{R}^{m \times N_0}$ are drawn independently from a probability distribution for which there exists $s \in (0, 1)$ and $r > 0$ such that $\|X_t\|_2 \le r$ almost surely, and such that for all unit vector $u \in \mathbb{S}^{m-1}$ we have*

$$\mathbb{E}\frac{\langle X_t, u\rangle^2}{\|X_t\|_2^2} \ge s^2. \tag{11}$$

Let $\mathcal{A}$ be the alphabet in (3) with step size $\delta > 0$, and the largest element $q_{\max}$. Let $w \in \mathbb{R}^{N_0}$ be the weights associated with a neuron with $\|w\|_\infty \le q_{\max}$. Quantizing w using (10), we have

$$\mathrm{P}\left(\|Xw - Xq\|_2^2 \le \frac{r^2\delta^2}{s^2}\log N_0\right) \ge 1 - \frac{1}{N_0^2}\left(2 + \frac{1}{\sqrt{1-s^2}}\right), \tag{12}$$

and

$$\mathrm{P}\left(\max_{1\le t\le N_0} \|u_t\|_2^2 \le \frac{r^2\delta^2}{s^2}\log N_0\right) \ge 1 - \frac{1}{N_0}\left(2 + \frac{1}{\sqrt{1-s^2}}\right). \tag{13}$$

Furthermore, if the activation function $\varphi : \mathbb{R} \to \mathbb{R}$ is ξ-Lipschitz continuous, that is, $|\varphi(x) - \varphi(y)| \le \xi|x - y|$ for all $x, y \in \mathbb{R}$, then we have

$$\mathrm{P}\left(\|\varphi(Xw) - \varphi(Xq)\|_2^2 \le \frac{r^2\delta^2\xi^2}{s^2}\log N_0\right) \ge 1 - \frac{1}{N_0^2}\left(2 + \frac{1}{\sqrt{1-s^2}}\right). \tag{14}$$

Proof. Let $\alpha > 0$ and $\eta > 0$. In the t-th step, by Markov's inequality, one can get

$$\mathrm{P}(\|u_t\|_2^2 \ge \alpha) = \mathrm{P}(e^{\eta\|u_t\|_2^2} \ge e^{\eta\alpha}) \le e^{-\eta\alpha}\mathbb{E}e^{\eta\|u_t\|_2^2}. \tag{15}$$

According to Lemma A.5,

$$\mathbb{E}e^{\eta\|u_t\|_2^2} \le \max\Big\{\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2}e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}), \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\Big\}. \tag{16}$$

Moreover, observing that $\|X_t\|_2^2 \le r^2$ a.s., then applying the law of total expectation, Lemma A.5 (2) with $\beta = 1$, and assumption (11) sequentially, we obtain

$$\begin{aligned}
\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2}e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}) &\le e^{\eta r^2\delta^2/4}\mathbb{E}e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \\
&= e^{\eta r^2\delta^2/4}\mathbb{E}(\mathbb{E}(e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1})) \\
&\le e^{\eta r^2\delta^2/4}\mathbb{E}\Big(-\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1})(e^{\eta\|u_{t-1}\|_2^2}-1) + e^{\eta\|u_{t-1}\|_2^2}\Big) \\
&\le e^{\eta r^2\delta^2/4}\mathbb{E}(-s^2(e^{\eta\|u_{t-1}\|_2^2}-1) + e^{\eta\|u_{t-1}\|_2^2}) \\
&= (1-s^2)e^{\eta r^2\delta^2/4}\mathbb{E}e^{\eta\|u_{t-1}\|_2^2} + s^2e^{\eta r^2\delta^2/4}
\end{aligned}$$

Hence, for each t, inequality (16) becomes

$$\mathbb{E}e^{\eta\|u_t\|_2^2} \le \max\Big\{a\mathbb{E}e^{\eta\|u_{t-1}\|_2^2} + b, \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\Big\}. \tag{17}$$

where $a := (1-s^2)e^{\eta r^2\delta^2/4}$ and $b := s^2e^{\eta r^2\delta^2/4}$. Let $t_0 = |\{1 \le i \le t : \mathbb{E}e^{\eta\|u_{i-1}\|_2^2} \le a\mathbb{E}e^{\eta\|u_{i-1}\|_2^2} + b\}|$. Then, noting that $u_0 = 0$, the following inequality follows from (17),

$$\mathbb{E}e^{\eta\|u_t\|_2^2} \le a^{t_0}\mathbb{E}e^{\eta\|u_0\|_2^2} + b(1 + a + \ldots + a^{t_0-1}) = a^{t_0} + \frac{b(1-a^{t_0})}{1-a} \le 1 + \frac{b}{1-a} \tag{18}$$

where the last inequality holds provided that $a = (1-s^2)e^{\eta r^2\delta^2/4} < 1$. Since the result above hold for all $\eta > 0$ such that $(1-s^2)e^{\eta r^2\delta^2/4} < 1$, we can choose $\eta = \frac{-2\log(1-s^2)}{r^2\delta^2}$. Then we get $a = (1-s^2)^{1/2}$ and $b = s^2(1-s^2)^{-1/2}$. It follows from (15) and (18) that

$$\begin{aligned}
\mathrm{P}(\|u_t\|_2^2 \ge \alpha) &\le e^{-\eta\alpha}\Big(1 + \frac{b}{1-a}\Big) = \exp\Big(\frac{2\alpha\log(1-s^2)}{r^2\delta^2}\Big)\Big(1 + \frac{s^2(1-s^2)^{-1/2}}{1-(1-s^2)^{1/2}}\Big) \\
&= \exp\Big(\frac{2\alpha\log(1-s^2)}{r^2\delta^2}\Big)\Big(1 + (1-s^2)^{-1/2}(1 + (1-s^2)^{1/2})\Big) \\
&= \exp\Big(\frac{2\alpha\log(1-s^2)}{r^2\delta^2}\Big)\Big(2 + \frac{1}{\sqrt{1-s^2}}\Big) \\
&\le \exp\Big(\frac{-2\alpha s^2}{r^2\delta^2}\Big)\Big(2 + \frac{1}{\sqrt{1-s^2}}\Big).
\end{aligned}$$

The last inequality can be obtained using the fact $\log(1+x) \le x$ for all $x > -1$. Picking $\alpha = \frac{r^2\delta^2\log N_0}{s^2}$, we get

$$\mathrm{P}\Big(\|u_t\|_2^2 \ge \frac{r^2\delta^2}{s^2}\log N_0\Big) \le \frac{1}{N_0^2}\Big(2 + \frac{1}{\sqrt{1-s^2}}\Big). \tag{19}$$

From (19) we can first deduce (12), by setting $t = N_0$ and using the fact $u_{N_0} = Xw - Xq$. If the activation function φ is ξ-Lipschitz, then $\|\varphi(Xw) - \varphi(Xq)\|_2 \le \xi\|Xw - Xq\|_2$ and (12) implies (14). Moreover, applying a union bound over t to (19), one can get (13). □

Next, we illustrate how Theorem 3.1 can be applied to obtain error bounds associated with uniformly distributed and Bernoulli distributed input data.

3.1.1. *Uniformly Distributed Data.* Let $B_r \subset \mathbb{R}^m$ be the closed ball with center 0 and radius $r > 0$. Suppose that columns X_t of $X \in \mathbb{R}^{m\times N_0}$ are drawn i.i.d. from $\mathrm{Unif}(B_r)$. Then $\|X_t\|_2 \le r$ and $Z := X_t/\|X_t\|_2 \sim \mathrm{Unif}(\mathbb{S}^{m-1})$. Since Z is rotation invariant, for any unit vector $u \in \mathbb{S}^{m-1}$, we have $\mathbb{E}\left\langle \frac{X_t}{\|X_t\|_2}, u\right\rangle^2 = \mathbb{E}\langle Z, u\rangle^2 = \mathbb{E}\langle Z, e_1\rangle^2 = \mathbb{E}Z_1^2 = \frac{1}{m}$. The last equality holds because $\|Z\|_2 = 1$ and $\mathbb{E}Z_1^2 = \mathbb{E}Z_2^2 = \ldots = \mathbb{E}Z_m^2 = \frac{1}{m}\mathbb{E}\left(\sum_{i=1}^m Z_i^2\right) = \frac{1}{m}$. So Theorem 3.1 implies that, with high probability

$$\|Xw - Xq\|_2^2 \le mr^2\delta^2 \log N_0. \tag{20}$$

Moreover, by Lemma A.3, $\mathbb{E}\|X_t\|_2^2 = \frac{mr^2}{m+2}$. It follows that $\mathbb{E}(X^\top X) = \mathbb{E}\|X_1\|_2^2 I_{N_0} = \frac{mr^2}{m+2} I_{N_0}$ and thus $\mathbb{E}\|Xw\|_2^2 = w^\top \mathbb{E}(X^\top X)w = \frac{mr^2}{m+2}\|w\|_2^2$. If the weight vector $w \in \mathbb{R}^{N_0}$ is *generic* in the sense that $\|w\|_2^2 \gtrsim N_0$, then

$$\mathbb{E}\|Xw\|_2^2 \gtrsim \frac{mN_0r^2}{m+2}. \tag{21}$$

Combining (20) with (21), the relative error satisfies $\frac{\|Xw-Xq\|_2^2}{\|Xw\|_2^2} \lesssim \frac{m\delta^2 \log N_0}{N_0}$.

3.1.2. *Data from a Symmetric Bernoulli Distribution.* We say that a random vector $Z = (Z_1, Z_2, \ldots, Z_m)$ is *symmetric Bernoulli* if the coordinates Z_i are independent and $\mathrm{P}(Z_i = 1) = \mathrm{P}(Z_i = -1) = \frac{1}{2}$. Now assume that columns X_t of $X \in \mathbb{R}^{m\times N_0}$ are independent and subject to symmetric Bernoulli distribution. Clearly, $\|X_t\|_2 = \sqrt{m}$. If $u \in \mathbb{R}^m$ is a unit vector, then $\mathbb{E}\frac{\langle X_t,u\rangle^2}{\|X_t\|_2^2} = \frac{u^\top \mathbb{E}(X_tX_t^\top)u}{m} = \frac{\|u\|_2^2}{m} = \frac{1}{m}$. Hence, by Theorem 3.1,

$$\|Xw - Xq\|_2^2 \le m^2\delta^2 \log N_0 \tag{22}$$

holds with high probability. Again, a generic $w \in \mathbb{R}^{N_0}$ with $\|w\|_2^2 \gtrsim N_0$ satisfies $\mathbb{E}\|Xw\|_2^2 = w^\top \mathbb{E}(X^\top X)w = m\|w\|_2^2 \gtrsim mN_0$ and therefore $\frac{\|Xw-Xq\|_2^2}{\|Xw\|_2^2} \lesssim \frac{m\delta^2 \log N_0}{N_0}$.

3.2. **Gaussian Clusters.** Here, we consider data drawn from Gaussian clusters, which unlike the previously considered models, are unbounded. One reason for considering Gaussian clusters is that they are a reasonable model for the activations in deeper layers of networks designed for classification. Specifically, suppose our samples are drawn from d normally distributed clusters $\mathcal{K}_i := \mathcal{N}(z^{(i)}, \sigma^2 I_{N_0})$ with fixed centers $z^{(i)} \in \mathbb{R}^{N_0}$ and $\sigma > 0$. Suppose, for simplicity, that we independently draw n samples from each cluster and vertically stack them in order as rows of X (this ordering does not affect our results in Theorem 3.2). Let $m := nd$. So, for $1 \le i \le d$, the row indices of X ranging from $(i-1)n+1$ to in come from cluster $\mathcal{K}_i$. Then the t-th column of X is of the form

$$X_t = [Y_t^{(1)}, Y_t^{(2)}, \ldots, Y_t^{(d)}]^\top \in \mathbb{R}^m \tag{23}$$

where $Y_t^{(i)} \sim \mathcal{N}(z_t^{(i)}\mathbb{1}_n, \sigma^2 I_n)$.

Theorem 3.2 (Gaussian clusters). *Let $X \in \mathbb{R}^{m\times N_0}$ be as in (23) and let $\mathcal{A}$ be as in (3), with step size $\delta > 0$ and the largest element $q_{\max}$. Let $p \in \mathbb{N}$, $K := 1 + \sigma^{-2}\max_{1\le i\le d}\|z^{(i)}\|_\infty^2$, and $w \in \mathbb{R}^{N_0}$ be the weights associated with a neuron, with $\|w\|_\infty \le q_{\max}$. Quantizing w using (10), we have*

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}, \quad \text{and}$$

$$\mathrm{P}\Big(\max_{1\le t\le N_0}\|u_t\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^{p-1}}.$$

If the activation function φ is ξ-Lipschitz continuous, then

$$\mathrm{P}\Big(\|\varphi(Xw) - \varphi(Xq)\|_2^2 \ge 4pm^2K^2\xi^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}.$$

The proof of Theorem 3.2 can be found in Appendix D.1.

3.2.1. *Normally Distributed Data.* As a special case of (23), let $X \in \mathbb{R}^{m\times N_0}$ be a Gaussian matrix with $X_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0,\sigma^2)$ corresponding to $d = 1$, $n = m$, and $z^{(1)} = 0$. Theorem 3.2 implies that $K = 1$ and

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \ge 4pm^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{m}}{N_0^p}. \tag{24}$$

Further, suppose that $w \in \mathbb{R}^{N_0}$ is generic, i.e. $\|w\|_2^2 \gtrsim N_0$. In this case, $\mathbb{E}\|Xw\|_2^2 = m\sigma^2\|w\|_2^2 \gtrsim m\sigma^2 N_0$. So, with high probability, the relative error in our quantization satisfies

$$\frac{\|Xw - Xq\|_2^2}{\|Xw\|_2^2} \lesssim \frac{m\delta^2\log N_0}{N_0}. \tag{25}$$

Thus, here again, the relative square error for quantizing a single-layer MLP decays linearly (up to a log factor) in the number of neurons N_0. Note that (25), for ternary alphabets, is the main result given by [22], which we now obtain as a special case of Theorem 3.2.

Remark 3.3. In Section 3.1 and Section 3.2, we have shown that if the columns of $X \in \mathbb{R}^{m\times N_0}$ are drawn from proper distributions, then the relative error for quantization is small when $m \ll N_0$. Now consider the case where the feature vectors $\{X_t\}_{t=1}^{N_0}$ live in a l-dimensional subspace with $l < m$. In this case, $X = VF$ where $V \in \mathbb{R}^{m\times l}$ satisfies $V^\top V = I$, and the columns F_t of $F \in \mathbb{R}^{l\times N_0}$ are drawn i.i.d. from a distribution $\mathcal{P}$. Suppose, for example, that $\mathcal{P} = \mathrm{Unif}(B_r)$. Due to $X = VF$, one can express any unit vector in the range of X as $u = Vv$ with $v \in \mathbb{R}^l$. Then we have $1 = \|u\|_2 = \|Vv\|_2 = \|v\|_2$, $\|X_t\|_2 = \|VF_t\|_2 = \|F_t\|_2 \le r$, and $\mathbb{E}\frac{\langle X_t,u\rangle^2}{\|X_t\|_2^2} = \mathbb{E}\frac{\langle VF_t,Vv\rangle^2}{\|VF_t\|_2^2} = \mathbb{E}\frac{\langle F_t,v\rangle^2}{\|F_t\|_2^2} = l^{-1}$ by our assumption for $\mathcal{P}$. Because u_t in Theorem 3.1 is a linear combination of X_j, the proof of Theorem 3.1 remains unchanged if (11) holds for all unit vectors u in the range of X. It follows that Theorem 3.1 holds for X with $s^2 = l^{-1}$ and thus the relative error for quantizing the data in a l-dimensional subspace is improved to $\frac{\|Xw-Xq\|_2^2}{\|Xw\|_2^2} \le \frac{l\delta^2\log N_0}{N_0}$. Applying a similar argument to $\mathcal{P}$ representing either a symmetric Bernoulli distribution or Gaussian distribution, one can replace m in their corresponding

relative errors by l. In short, the relative error depends not on the number of training samples m but on the intrinsic dimension of the features l.

3.3. **Convolutional Neural Networks.** In this section, we derive error bounds for single-layer CNNs. Let $Z \in \mathbb{R}^{B\times C_{\text{in}}\times S_1\times S_2}$ be a mini-batch of images with batch size B, input channels C_{in}, height S_1, and width S_2. Suppose that all entries of Z are i.i.d. drawn from $\mathcal{N}(0,1)$ and suppose we have C_{out} convolutional kernels $\{w_i\}_{i=1}^{C_{\text{out}}} \subseteq \mathbb{R}^{C_{\text{in}}\times k_1\times k_2}$. Let these kernels "slide" over Z with fixed stride (k_1, k_2) such that sliding local blocks generated by moving w_i on Z are disjoint. Additionally, if T is the number of randomly selected sliding local blocks (in $\mathbb{R}^{C_{\text{in}}\times k_1\times k_2}$) from each image, then one can vectorize all BT local blocks and stack them together to obtain a single data matrix $X \in \mathbb{R}^{BT\times C_{\text{in}}k_1k_2}$. Moreover, each kernel w_i can be viewed as a column vector in $\mathbb{R}^{C_{\text{in}}k_1k_2}$ and thus $W = [w_1, w_2, \dots, w_{C_{\text{out}}}] \in \mathbb{R}^{C_{\text{in}}k_1k_2\times C_{\text{out}}}$ is the weight matrix to be quantized. Thus, we need to convert W to $Q = [q_1, q_2, \dots, q_{C_{\text{out}}}] \in \mathcal{A}^{C_{\text{in}}k_1k_2\times C_{\text{out}}}$ with $XQ \approx XW$, as before. Since extracted local blocks from Z are disjoint, columns of X are independent and subject to $\mathcal{N}(0, I_{BT})$. Hence, one can apply (24) with $m = BT$, $N_0 = C_{\text{in}}k_1k_2$, $\sigma = 1$, and any $p \in \mathbb{N}$. Specifically, for $1 \le i \le C_{\text{out}}$, we get $\mathrm{P}\Big(\|Xw_i - Xq_i\|_2^2 \ge 4pB^2T^2\delta^2\log(C_{\text{in}}k_1k_2)\Big) \lesssim \frac{\sqrt{BT}}{(C_{\text{in}}k_1k_2)^p}$. By a union bound, $\mathrm{P}\Big(\max_{1\le i\le C_{\text{out}}}\|Xw_i - Xq_i\|_2^2 \ge 4pB^2T^2\delta^2\log(C_{\text{in}}k_1k_2)\Big) \lesssim \frac{C_{\text{out}}\sqrt{BT}}{(C_{\text{in}}k_1k_2)^p}$.

4. Sparse GPFQ and Error Analysis

Having extended the results pertaining to GPFQ to cover multiple distributions of the input data, as well as general alphabets, we now propose modifications to produce quantized weights that are also sparse, i.e., that have a large fraction of coefficients being 0. Our sparse quantization schemes result from adding a regularization term to (8). Specifically, in order to generate sparse $q \in \mathcal{A}^{N_{i-1}}$, we compute q_t via

$$q_t = \arg\min_{p\in\mathcal{A}}\left(\frac{1}{2}\left\|u_{t-1} + w_tX_t^{(i-1)} - p\widetilde{X}_t^{(i-1)}\right\|_2^2 + \lambda|p|\|\widetilde{X}_t^{(i-1)}\|_2^2\right) \tag{26}$$

where $\lambda > 0$ is a regularization parameter. Conveniently, Lemma A.2 shows that the solution of (26) is given by

$$q_t = \mathcal{Q}\circ s_\lambda\left(\frac{\langle\widetilde{X}_t^{(i-1)}, u_{t-1} + w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\right) \tag{27}$$

where s_λ denotes soft thresholding. It is then natural to consider a variant of (27) replacing s_λ with hard thresholding, h_λ. Since $h_\lambda(z)$ has jump discontinuities at $z = \pm\lambda$, the corresponding alphabet and quantizer should be adapted to this change. Thus, we use $\widetilde{\mathcal{Q}}(z)$ over $\widetilde{\mathcal{A}} = \mathcal{A}_K^{\delta,\lambda}$ as in (6) and $q_t \in \widetilde{\mathcal{A}}$ is obtained via

$$q_t = \widetilde{\mathcal{Q}}\circ h_\lambda\left(\frac{\langle\widetilde{X}_t^{(i-1)}, u_{t-1} + w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\right). \tag{28}$$

In both cases, we update the error vector via $u_t = u_{t-1} + w_tX_t^{(i-1)} - q_t\widetilde{X}_t^{(i-1)}$, as before. In summary, for quantizing a single-layer network, similar to (10) the two sparse GPFQ schemes related to soft and hard thresholding are given by

$$\text{(29)}\quad \begin{cases} u_0 = 0 \in \mathbb{R}^m, \\ q_t = \mathcal{Q} \circ s_\lambda\big(w_t + \frac{X_t^\top u_{t-1}}{\|X_t\|_2^2}\big), \\ u_t = u_{t-1} + w_t X_t - q_t X_t. \end{cases} \qquad \text{(30)}\quad \begin{cases} u_0 = 0 \in \mathbb{R}^m, \\ q_t = \widetilde{\mathcal{Q}} \circ h_\lambda\big(w_t + \frac{X_t^\top u_{t-1}}{\|X_t\|_2^2}\big), \\ u_t = u_{t-1} + w_t X_t - q_t X_t. \end{cases}$$

Interesting, with these sparsity promoting modifications, one can prove similar error bounds to GPFQ. To illustrate with bounded or Gaussian clustered data, we show that sparse GPFQ admits similar error bounds as in Theorem 3.1 and Theorem 3.2. The following results are proved in Appendix E.

Theorem 4.1 (Sparse GPFQ with bounded input data)**.** *Under the conditions of Theorem 3.1, we have the following.*

(*a*) *Quantizing w using (29) with the alphabet $\mathcal{A}$ in (3), we have*

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \le \frac{r^2(2\lambda+\delta)^2}{s^2}\log N_0\Big) \ge 1 - \frac{1}{N_0^2}\Big(2 + \frac{1}{\sqrt{1-s^2}}\Big).$$

(*b*) *Quantizing w using (30) with the alphabet $\widetilde{\mathcal{A}}$ in (4), we have*

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \le \frac{r^2\max\{2\lambda,\delta\}^2}{s^2}\log N_0\Big) \ge 1 - \frac{1}{N_0^2}\Big(2 + \frac{1}{\sqrt{1-s^2}}\Big).$$

Theorem 4.2 (Sparse GPFQ for Gaussian clusters)**.** *Under the assumptions of Theorem 3.2, the followings inequalities hold.*

(*a*) *Quantizing w using (29) with the alphabet $\mathcal{A}$ in (3), we have*

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \ge 4pm^2K^2(2\lambda+\delta)^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}.$$

(*b*) *Quantizing w using (30) with the alphabet $\widetilde{\mathcal{A}}$ in (4), we have*

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \ge 4pm^2K^2\max\{2\lambda,\delta\}^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}.$$

Note that the sparsity regularization term λ only appears in the error bounds, making them slightly worse than those where no sparsity is enforced. In Section 5.2.4, we will numerically explore the impact of λ on the sparsity and accuracy of quantized neural networks.

5. Experiments

To evaluate the performance of our method and compare it with the approaches reviewed in Section 1.1, we test our modified GPFQ on the ImageNet classification task [1]. In particular, we focus on ILSVRC-2012 [5], a 1000-category dataset with over 1.2 million training images and 50 thousand validation images. All images in ILSVRC-2012 are preprocessed in a standard manner before they are fed into neural networks: we resize each image to 256×256 and use the normalized 224×224 center crop. The evaluation metrics we use are top-1 and top-5 accuracy of the quantized models on the validation dataset.

[1]Our code for experiments is available: https : //github.com/YixuanSeanZhou/Quantized_Neural_Nets.git

5.1. **Experimental Setup.** For reproducibility and fairness of comparison, we use the pretrained 32-bit floating point neural networks provided by torchvision[2] in PyTorch [25]. We test several well-known neural network architectures including: AlexNet [17], VGG-16 [26], GoogLeNet [27], ResNet-18, ResNet-50 [13], and EfficeintNet-B1 [28]. In the following experiments, we will focus on quantizing the weights of fully-connected and convolutional layers of the above architectures, as our theory applies specifically to these types of layers[3].

Let $b \in \mathbb{N}$ denote the number of bits used for quantization. Here, we fix b for all the layers. In our experiments with GPFQ, we adopt the midtread alphabets $\mathcal{A}_K^\delta$ in (3) with

$$K = 2^{b-1}, \quad \delta = \frac{R}{2^{b-1}}, \tag{31}$$

where $R > 0$ is a hyper-paramter. Indeed, according to (3), $\mathcal{A}_K^\delta$ is symmetric with maximal element $q_{\max} = K\delta = R$. Since b is fixed, all that remains is to select R in (31) based on the distribution of weights. To that end, suppose we are quantizing the i-th layer of a neural network with weight matrix $W^{(i)} \in \mathbb{R}^{N_{i-1}\times N_i}$. Then, Theorem 3.1 and Theorem 3.2 require that $R = q_{\max} \geq \max_{k,j} |W_{k,j}^{(i)}|$, and yield error bounds that favor a smaller step size $\delta \propto R$. In practice, however, the weights may have outliers with large magnitudes, which would entail unnecessarily using a large R. Thus, rather than choosing $R = \max_{k,j} |W_{k,j}^{(i)}|$, we will consider the average infinity norm of weights across all neurons w, i.e. columns of $W^{(i)}$. That is $R \propto \frac{1}{N_i} \sum_{1\leq j\leq N_i} \|W_j^{(i)}\|_\infty$. Then, by (31), the step size used for quantizing the i-th layer is given by

$$\delta^{(i)} := \frac{C}{2^{b-1}N_i} \sum_{1\leq j\leq N_i} \|W_j^{(i)}\|_\infty. \tag{32}$$

Here, $C \geq 1$ is independent of i and fixed across layers, batch-sizes, and bit widths. To obtain a good choice of C, we perform a grid search with cross-validation over the interval $[1, 2]$, albeit on a small batch size $m \leq 128$. So the tuning of C takes very little time compared to the quantization with the full training data. Note that the tuning and quantization scale linearly in the size of the data set and the number of parameters of the network. This means that this entire process's computational complexity is dominated by the original training of the network and there is no problem with its scaling to large networks. Moreover, by choosing the maximal element in our alphabet, i.e. $q_{\max} = 2^{b-1}\delta^{(i)}$, to be a constant $C \in [1, 2]$ times the average ℓ_∞ norm of all the neurons, we are selecting a number that is effectively larger than most of the weights and thereby corresponding perfectly with the theory for most of the neurons. For the remaining neurons, the vast majority of the weights will be below this threshold, and only the outlier weights, in general, will exceed it. In Appendix C, we present a theoretical analysis of the expected error when a few weights exceed $q_{\max}$. We not only show that the proposed algorithm is still effective in this scenario, but also that in some

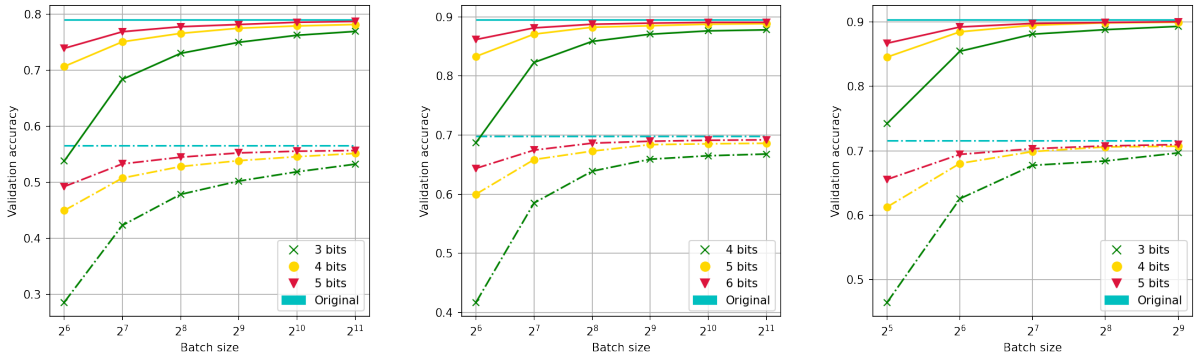
[2] https://pytorch.org/vision/stable/models.html

[3] Batch normalization layers, while not explicitly covered by our methods in the preceeding sections, are easy to handle. Indeed, in Appendix B, we show that our approach can effectively quantize batch normalization layers by merging them with their preceding convolutional layers before quantization, and we demonstrate experimentally that this does not negatively impact performance.

cases, it may be beneficial to choose δ small enough such that some weights exceed $q_{\max}$. The analysis in Appendix C is consistent with, and helps explain the experimental results in this section. Further, we comment that a more thorough search for an optimal C depending on these individual parameters, e.g. b, may improve performance.

TABLE 1. Top-1/Top-5 accuracy drop using $b = 5$ bits.

Model	C	m	Acc Drop (%)	Model	C	m	Acc Drop (%)
AlexNet	1.1	2048	0.85/0.33	GoogLeNet	1.41	2048	0.60/0.46
VGG-16	1.0	512	0.63/0.32	EfficientNet-B1	1.6	2048	0.45/0.18
ResNet-18	1.16	4096	0.49/0.23	ResNet-50	1.81	2048	0.62/0.11



(A) AlexNet (B) GoogLeNet (C) VGG-16

(D) ResNet-18 (E) ResNet-50 (F) EfficientNet-B1

FIGURE 1. Top-1 (dashed lines) and Top-5 (solid lines) accuracy for original and quantized models on ImageNet.

As mentioned in Section 3.3, we introduce a sampling probability $p \in (0, 1]$, associated with GPFQ for convolutional layers. This is motivated, in part, by decreasing the computational cost associated with quantizing such layers. Indeed, a batched input tensor of a convolutional layer can be unfolded as a stack of vectorized sliding local blocks, i.e., a matrix. Since,

additionally, the kernel can be reshaped into a column vector, matrix-vector multiplication followed by reshaping gives the output of this convolutional layer. On the other hand, due to potentially large overlaps between sliding blocks, the associated matrices have large row size and thus the computational complexity is high. To accelerate our computations, we extract the data used for quantization by setting the stride (which defines the step size of the kernel when sliding through the image) equal to the kernel size and choosing $p = 0.25$. This choice gives a good trade-off between accuracy and computational complexity, which both increase with p. Recall that the batch size $m \in \mathbb{N}$ denotes the number of samples used for quantizing each layer of a neural network. In all experiments, b is chosen from $\{3, 4, 5, 6\}$.

5.2. **Results on ImageNet.**

5.2.1. ***Impact of b and m.*** The first experiment is designed to explore the effect of the batch size m, as well as bit-width b, on the accuracy of the quantized models. We compute the validation accuracy of quantized networks with respect to different choices of b and m. In particular, Table 1 shows that, using $b = 5$ bits, all quantized models achieve less than 1% loss in top-1 and top-5 accuracy. Moreover, we illustrate the relationship between the quantization accuracy and the batch size m in Figure 1, where the horizontal lines in cyan, obtained directly from the original validation accuracy of unquantized models, are used for comparison against our quantization method. We observe that (1) all curves with distinct b quickly approach an accuracy ceiling while curves with high b eventually reach a higher ceiling; (2) Quantization with $b \geq 4$ attains near-original model performance with sufficiently large m; (3) one can expect to obtain higher quantization accuracy by taking larger m but the extra improvement that results from increasing the batch size rapidly diminishes.

5.2.2. ***Comparisons with Baselines.*** Next, we compare GPFQ against other post-training quantization schemes discussed in Section 1.1 on various architectures. We note, however, that for a fixed architecture each post-training quantization method starts with a potentially different set of parameters (weights and biases), and these parameters are not available to us. As such, we simply report other methods' accuracies as they appear in their associated papers. Due to this, a perfect comparison between methods is not possible. Another factor that impacts the comparison is that following DoReFa-Net [36], many baseline quantization schemes [34, 14, 19] leave the first and the last layers of DNNs unquantized to alleviate accuracy degradation. On the other hand, we quantize *all* layers of the model. Table 2 displays the number of bits and the method used to quantize each network. It also contains the accuracy of quantized and full-precision models respectively, as well as their difference, i.e. accuracy drop. We report the results of GPFQ (without the † superscript) for all models with $b = 3, 4, 5$. The important observation here is that our method is competitive across architectures and bit-widths, and shows the best performance on a number of them.

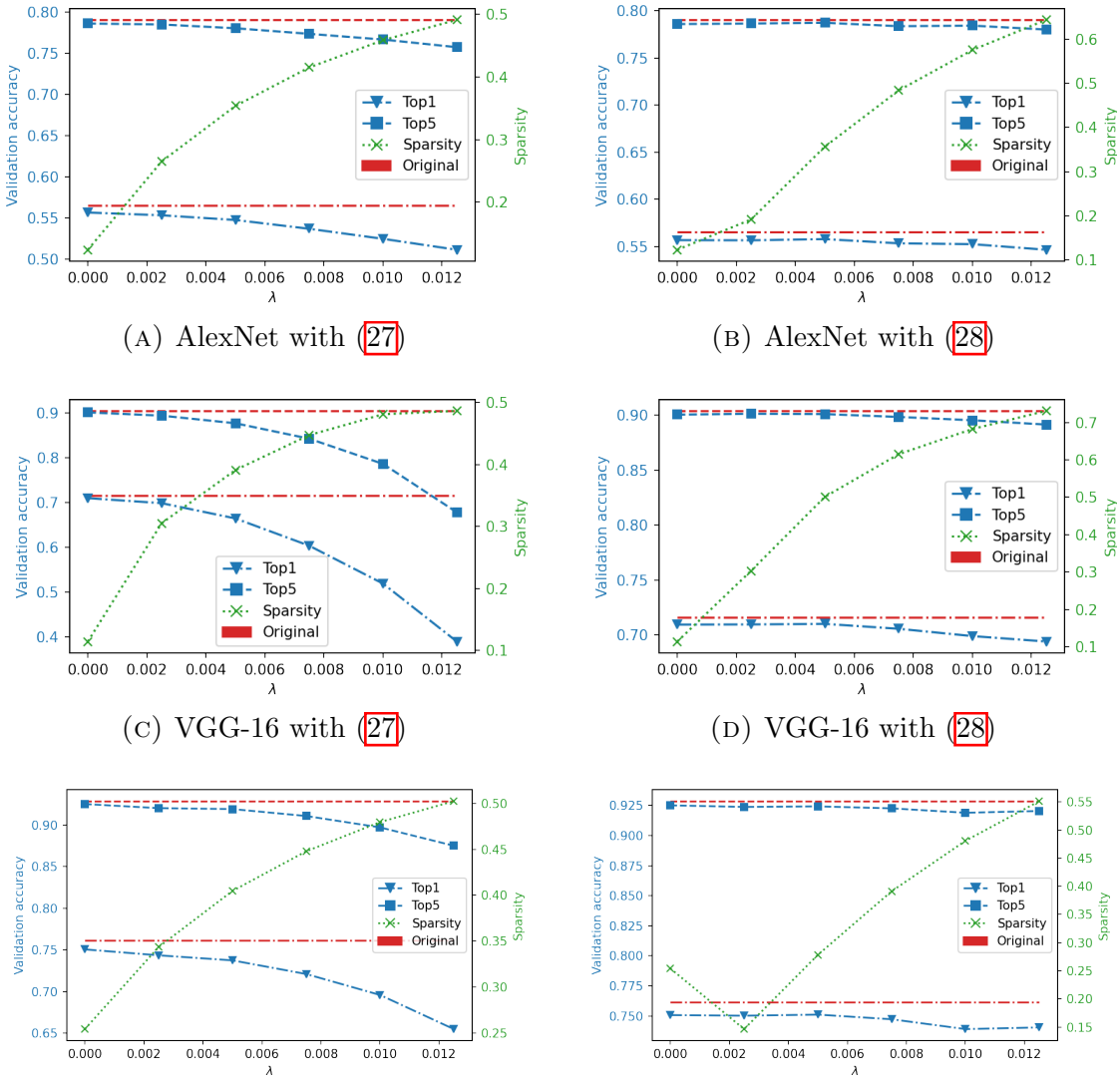
5.2.3. ***Further Improvement of GPFQ.*** In this section, we show that the validation accuracy of the proposed approach can be further improved by incorporating the following modifications used by prior work: (1) mixing precision for quantization, such as using different bit-widths to quantize fully-connected and convolutional layers respectively [2] or leaving

TABLE 2. ImageNet Top-1 accuracy with weight quantization.

Model	Bits	Method	Quant Acc (%)	Ref Acc (%)	Acc Drop (%)
Alexnet	3	GPFQ (Ours)	53.22	56.52	3.30
		GPFQ (Ours)†	54.77	56.52	1.75
	4	OMSE [3]	55.52	56.62	1.10
		GPFQ (Ours)	55.15	56.52	1.37
		GPFQ (Ours)†	55.51	56.52	1.01
	5	GPFQ (Ours)	55.67	56.52	0.85
		GPFQ (Ours)†	55.94	56.52	0.58
	8	DoReFa [36]	53.00	55.90	2.90
VGG-16	3	GPFQ (Ours)	69.67	71.59	1.92
		GPFQ (Ours)†	70.24	71.59	1.35
	4	MSE [1]	70.50	71.60	1.10
		OMSE [3]	71.48	73.48	2.00
		GPFQ (Ours)	70.70	71.59	0.89
		GPFQ (Ours)†	70.90	71.59	0.69
	5	GPFQ (Ours)	70.96	71.59	0.63
		GPFQ (Ours)†	71.05	71.59	0.54
	8	Lee et al. [18]	68.05	68.34	0.29
ResNet-18	3	GPFQ (Ours)	66.55	69.76	3.21
		GPFQ (Ours)†	67.63	69.76	2.13
	4	MSE [1]	67.00	69.70	2.70
		OMSE [3]	68.38	69.64	1.26
		S-AdaQuant [14]	69.40	71.97	2.57
		AdaRound [24]	68.71	69.68	0.97
		BRECQ [19]	70.70	71.08	0.38
		GPFQ (Ours)	68.55	69.76	1.21
		GPFQ (Ours)†	68.81	69.76	0.95
	5	RQ [21]	65.10	69.54	4.44
		GPFQ (Ours)	69.27	69.76	0.49
		GPFQ (Ours)†	69.50	69.76	0.26
	6	DFQ [23]	66.30	70.50	4.20
		RQ [21]	68.65	69.54	0.89
ResNet-50	3	GPFQ (Ours)	71.80	76.13	4.33
		GPFQ (Ours)†	72.18	76.13	3.95
	4	MSE [1]	73.80	76.10	2.30
		OMSE [3]	73.39	76.01	2.62
		OCS + Clip [34]	69.30	76.10	6.80
		PWLQ [8]	73.70	76.10	2.40
		AdaRound [24]	75.23	76.07	0.84
		S-AdaQuant [14]	75.10	77.20	2.10
		BRECQ [19]	76.29	77.00	0.71
		GPFQ (Ours)	75.10	76.13	1.03
		GPFQ (Ours)†	75.30	76.13	0.83
	5	OCS + Clip [34]	73.40	76.10	2.70
		GPFQ (Ours)	75.51	76.13	0.62
		GPFQ (Ours)†	75.66	76.13	0.47
	8	IAOI [15]	74.90	76.40	1.50

the last fully-connected layer unquantized [36]; (2) applying bias correction [1, 23] to the last layer, that is, subtracting the average quantization error from the layer's bias term. In Table 2, we examine some of these empirical rules by leaving the last layer intact and

performing bias correction to remove the noise due to quantization. This variant of GPFQ is highlighted by a † symbol. By using the enhanced GPFQ, the average increment of accuracy exceeds 0.2% for $b = 4, 5$ bits, and is greater than 0.7% for $b = 3$ bits. This demonstrates, empirically, that GPFQ can be easily adapted to incorporate heuristic modifications that improve performance.



(A) AlexNet with (27)

(B) AlexNet with (28)

(C) VGG-16 with (27)

(D) VGG-16 with (28)

(E) ResNet-50 with (27)

(F) ResNet-50 with (28)

FIGURE 2. (1) Left y-axis: Top-1 (dashed-dotted lines) and Top-5 (dash lines) accuracy for original (in red) and quantized (in blue) models on ImageNet. (2) Right y-axis: The sparsity of quantized models plotted by dotted green lines.

5.2.4. ***Sparse Quantization.*** For our final experiment, we illustrate the effects of sparsity via the sparse quantization introduced in Section 4. Recall that the sparse GPFQ with soft thresholding in (27) uses alphabets $\mathcal{A}_K^\delta$ as in (3) while the version of hard thresholding, see

(28), relies on alphabets $\mathcal{A}_K^{\delta,\lambda}$ as in Equation (4). In the setting of our experiment, both K and δ are still defined and computed as in Section 5.1, where the number of bits $b = 5$ and the corresponding scalar $C > 0$ and batch size $m \in \mathbb{N}$ for each neural network is provided by Table 1. Moreover, the *sparsity* of a given neural network is defined as the proportion of zeros in the weights. According to Equation (27) and Equation (28), in general, the sparsity of DNNs is boosted as λ increases. Hence, we treat $\lambda > 0$ as a variable to control sparsity and explore its impact on validation accuracy of different DNNs. As shown in Figure 2, we quantize AlexNet, VGG-16, and ResNet-50 using both (27) and (28), with $\lambda \in \{0, 0.0025, 0.005, 0.0075, 0.01, 0.0125\}$. Curves for validation accuracy and sparsity are plotted against λ. We note that, for all tested models, sparse GPFQ with hard thresholding, i.e. (28), outperforms soft thresholding, achieving significantly higher sparsity and better accuracy. For example, by quantizing AlexNet and VGG-16 with (28), one can maintain near-original model accuracy when half the weights are quantized to zero, which implies a remarkable compression rate $\frac{0.5b}{32} = \frac{2.5}{32} \approx 7.8\%$. Similarly, Figure 2f and Figure 2e show that ResNet-50 can attain 40% sparsity with subtle decrement in accuracy. Additionally, in all cases, one can expect to get higher sparsity by increasing λ while the validation accuracy tends to drop gracefully. Moreover, in Figure 2e, we observe that the sparsity of quantized ResNet50 with $\lambda = 0.0025$ is even lower than the result when thresholding functions are not used, that is, $\lambda = 0$. A possible reason is given as follows. In contrast with $\mathcal{A}_K^{\delta}$, the alphabet $\mathcal{A}_K^{\delta,\lambda}$ has only one element 0 between $-\lambda$ and λ. Thus, to compensate for the lack of small alphabet elements and also reduce the path following error, sparse GPFQ in (28) converts more weights to nonzero entries of $\mathcal{A}_K^{\delta,\lambda}$, which in turn dampens the upward trend in sparsity.

Acknowledgements

This work was supported in part by National Science Foundation Grant DMS-2012546. The authors thank Eric Lybrand for stimulating discussions on the topics of this paper.

References

[1] R. Banner, Y. Nahshan, E. Hoffer, and D. Soudry. Post-training 4-bit quantization of convolution networks for rapid-deployment. *arXiv preprint arXiv:1810.05723*, 2018.

[2] Y. Cai, Z. Yao, Z. Dong, A. Gholami, M. W. Mahoney, and K. Keutzer. Zeroq: A novel zero shot quantization framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13169–13178, 2020.

[3] Y. Choukroun, E. Kravchik, F. Yang, and P. Kisilev. Low-bit quantization of neural networks for efficient inference. In *ICCV Workshops*, pages 3009–3018, 2019.

[4] M. Courbariaux, Y. Bengio, and J.-P. David. Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in neural information processing systems*, pages 3123–3131, 2015.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[6] L. Deng, G. Li, S. Han, L. Shi, and Y. Xie. Model compression and hardware acceleration for neural networks: A comprehensive survey. *Proceedings of the IEEE*, 108(4):485–532, 2020.

[7] Z. Dong, Z. Yao, A. Gholami, M. W. Mahoney, and K. Keutzer. Hawq: Hessian aware quantization of neural networks with mixed-precision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 293–302, 2019.

[8] J. Fang, A. Shafiee, H. Abdel-Aziz, D. Thorsley, G. Georgiadis, and J. H. Hassoun. Post-training piecewise linear quantization for deep neural networks. In *European Conference on Computer Vision*, pages 69–86. Springer, 2020.

[9] S. Foucart and H. Rauhut. An invitation to compressive sensing. In *A mathematical introduction to compressive sensing*, pages 1–39. Springer, 2013.

[10] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer. A survey of quantization methods for efficient neural network inference. *arXiv preprint arXiv:2103.13630*, 2021.

[11] Y. Guo. A survey on methods and theories of quantized neural networks. *arXiv preprint arXiv:1808.04752*, 2018.

[12] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] I. Hubara, Y. Nahshan, Y. Hanani, R. Banner, and D. Soudry. Improving post training neural quantization: Layer-wise calibration and integer programming. *arXiv preprint arXiv:2006.10518*, 2020.

[15] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713, 2018.

[16] R. Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*, 2018.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25: 1097–1105, 2012.

[18] J. H. Lee, S. Ha, S. Choi, W.-J. Lee, and S. Lee. Quantization for rapid deployment of deep neural networks. *arXiv preprint arXiv:1810.05488*, 2018.

[19] Y. Li, R. Gong, X. Tan, Y. Yang, P. Hu, Q. Zhang, F. Yu, W. Wang, and S. Gu. Brecq: Pushing the limit of post-training quantization by block reconstruction. *arXiv preprint arXiv:2102.05426*, 2021.

[20] Y. Liu, W. Zhang, and J. Wang. Zero-shot adversarial quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1512–1521, 2021.

[21] C. Louizos, M. Reisser, T. Blankevoort, E. Gavves, and M. Welling. Relaxed quantization for discretized neural networks. In *International Conference on Learning Representations*, 2019.

[22] E. Lybrand and R. Saab. A greedy algorithm for quantizing neural networks. *Journal of Machine Learning Research*, 22(156):1–38, 2021.

[23] M. Nagel, M. v. Baalen, T. Blankevoort, and M. Welling. Data-free quantization through weight equalization and bias correction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1325–1334, 2019.

[24] M. Nagel, R. A. Amjad, M. Van Baalen, C. Louizos, and T. Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International Conference on Machine Learning*, pages 7197–7206. PMLR, 2020.

[25] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[27] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[28] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.

[29] R. Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[30] P. Wang, Q. Chen, X. He, and J. Cheng. Towards accurate post-training network quantization via bit-split and stitching. In *International Conference on Machine Learning*, pages 9847–9856. PMLR, 2020.

[31] P. Wang, X. He, G. Li, T. Zhao, and J. Cheng. Sparsity-inducing binarized neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12192–12199, 2020.

[32] S. Xu, H. Li, B. Zhuang, J. Liu, J. Cao, C. Liang, and M. Tan. Generative low-bitwidth data free quantization. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020.

[33] D. Zhang, J. Yang, D. Ye, and G. Hua. Lq-nets: Learned quantization for highly accurate and compact deep neural networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 365–382, 2018.

[34] R. Zhao, Y. Hu, J. Dotzel, C. De Sa, and Z. Zhang. Improving neural network quantization without retraining using outlier channel splitting. In *International conference on machine learning*, pages 7543–7552. PMLR, 2019.

[35] A. Zhou, A. Yao, Y. Guo, L. Xu, and Y. Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. *arXiv preprint arXiv:1702.03044*, 2017.

[36] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen, and Y. Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016.

Appendix A. Useful Lemmata

Lemma A.1. *In the context of (8), we have* $q_t = \mathcal{Q}\Big(\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\Big)$. *Here, we suppose* $\widetilde{X}_t^{(i-1)} \neq 0$.

Proof. According to (8), $q_t = \arg\min_{p\in\mathcal{A}} \big\|u_{t-1} + w_tX_t^{(i-1)} - p\widetilde{X}_t^{(i-1)}\big\|_2^2$. Expanding the square and removing the terms irrelevant to p, we obtain

$$\begin{aligned}
q_t &= \arg\min_{p\in\mathcal{A}}\Big(p^2\|\widetilde{X}_t^{(i-1)}\|_2^2 - 2p\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle\Big)\\
&= \arg\min_{p\in\mathcal{A}}\Big(p^2 - 2p\cdot\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\Big)\\
&= \arg\min_{p\in\mathcal{A}}\Big(p - \frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\Big)^2\\
&= \arg\min_{p\in\mathcal{A}}\Big|p - \frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\Big|\\
&= \mathcal{Q}\Big(\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\Big).
\end{aligned}$$

In the last equality, we used the definition of (5). □

Lemma A.2. *Suppose* $\widetilde{X}_t^{(i-1)} \neq 0$. *The closed-form expression of* q_t *in (26) is given by* $q_t = \mathcal{Q}\circ s_\lambda\Big(\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\Big)$. *Here,* $s_\lambda(x) := \operatorname{sign}(x)\max\{|x|-\lambda, 0\}$ *is the soft thresholding function.*

Proof. Expanding the square and removing the terms irrelevant to p, we obtain

$$\begin{aligned}
q_t &= \arg\min_{p\in\mathcal{A}}\Big(\frac{p^2}{2}\|\widetilde{X}_t^{(i-1)}\|_2^2 - p\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle + \lambda|p|\|\widetilde{X}_t^{(i-1)}\|_2^2\Big)\\
&= \arg\min_{p\in\mathcal{A}}\Big(\frac{p^2}{2} - p\cdot\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2} + \lambda|p|\Big)\\
&= \arg\min_{p\in\mathcal{A}}\Big(\frac{p^2}{2} - \alpha_t p + \lambda|p|\Big) \qquad (33)
\end{aligned}$$

where $\alpha_t := \frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1}+w_tX_t^{(i-1)}\rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}$. Define $g_t(p) := \frac{1}{2}p^2 - \alpha_t p + \lambda|p|$ for $p\in\mathbb{R}$. By (3), we have $q_t = \arg\min_{p\in\mathcal{A}} g_t(p) = \arg\min_{\substack{|k|\le K\\ k\in\mathbb{Z}}} g_t(k\delta)$. Now we analyze two cases $\alpha_t \ge 0$ and $\alpha_t < 0$. The idea is to investigate the behaviour of $g_t(k\delta)$ over $k\in\{-K, ..., K\}$.

(I) Assume $\alpha_t \ge 0$. Since $g_t(k\delta) > g_t(0) = 0$ for all $-K \le k \le -1$, then $g_t(k\delta)$ is minimized at some $k \ge 0$. Note that $g_t(p)$ is a convex function passing through the origin. So, for $1 \le k \le K-1$, $g_t(k\delta)$ is the minimum if and only if $g_t(k\delta) \le \min\{g_t((k+1)\delta), g_t((k-1)\delta)\}$.

It is easy to verify that the condition above is equivalent to

$$\Big(k-\frac{1}{2}\Big)\delta + \lambda \le \alpha_t \le \Big(k+\frac{1}{2}\Big)\delta + \lambda. \qquad (34)$$

It only remains to check $k = 0$ and $k = K$. For $k = 0$, note that when $\alpha_t \in [0, \delta/2 + \lambda]$, we have

$$g_t(\delta) \geq g_t(0) = 0, \tag{35}$$

and if $\alpha_t \geq (K - \frac{1}{2})\delta + \lambda$, then

$$g_t(K\delta) \leq g_t((K-1)\delta). \tag{36}$$

Combining (34), (35), and (36), we conclude that

$$q_t = \arg\min_{\substack{|k|\leq K \\ k\in\mathbb{Z}}} g_t(k\delta) = \begin{cases} 0 & \text{if } 0 \leq \alpha_t < \frac{\delta}{2} + \lambda, \\ k\delta & \text{if } |\alpha_t - \lambda - k\delta| \leq \frac{\delta}{2} \text{ and } 1 \leq k \leq K-1, \\ K\delta & \text{if } \alpha_t \geq \lambda + \frac{\delta}{2} + (K-1)\delta. \end{cases} \tag{37}$$

(II) In the opposite case where $\alpha_t < 0$, it suffices to minimize $g_t(k\delta)$ with $k \leq 0$ because $g_t(k\delta) > 0$ for all $k \geq 1$. Again, notice that $g_t(p)$ is a convex function on $[-\infty, 0]$ satisfying $g_t(0) = 0$. Applying a similar argument as in the case $\alpha_t \geq 0$, one can get

$$q_t = \arg\min_{\substack{|k|\leq K \\ k\in\mathbb{Z}}} g_t(k\delta) = \begin{cases} 0 & \text{if } -\frac{\delta}{2} - \lambda < \alpha_t < 0, \\ k\delta & \text{if } |\alpha_t + \lambda - k\delta| \leq \frac{\delta}{2} \text{ and } -(K-1) \leq k \leq -1, \\ -K\delta & \text{if } \alpha_t \leq -\lambda - \frac{\delta}{2} - (K-1)\delta. \end{cases} \tag{38}$$

It follows from (37) and (38) that $q_t = \mathcal{Q}(s_\lambda(\alpha_t)) = \mathcal{Q} \circ s_\lambda\left(\frac{\langle \widetilde{X}_t^{(i-1)}, u_{t-1} + w_t X_t^{(i-1)} \rangle}{\|\widetilde{X}_t^{(i-1)}\|_2^2}\right)$ where $s_\lambda(x) := \operatorname{sign}(x)\max\{|x| - \lambda, 0\}$ is the soft thresholding function. □

Lemma A.3. *Let* $\mathrm{Unif}(B_r)$ *denote the uniform distribution on the closed ball* $B_r \subset \mathbb{R}^m$ *with center at the origin and radius* $r > 0$. *Suppose that the random vector* $X \in \mathbb{R}^m$ *is drawn from* $\mathrm{Unif}(B_r)$. *Then we have* $\mathbb{E}\|X\|_2^2 = \frac{mr^2}{m+2}$.

Proof. Note that the density function of $\mathrm{Unif}(B_r)$ is given by $f(x) = \frac{1}{\mathrm{vol}(B_r)}\mathbb{1}_{B_r}(x)$ where $\mathrm{vol}(B_r) = r^m\pi^{\frac{m}{2}}/\Gamma(\frac{m}{2}+1)$ is the volume of B_r. Moreover, by integration in spherical coordinates, one can get

$$\begin{aligned} \mathbb{E}\|X\|_2^2 &= \int_{\mathbb{R}^m} \|x\|_2^2 f(x)\, dx = \int_0^\infty \int_{\mathbb{S}^{m-1}} z^{m-1}\|zx\|_2^2 f(zx)\, d\sigma(x)\, dz \\ &= \int_0^r \int_{\mathbb{S}^{m-1}} \frac{z^{m+1}}{\mathrm{vol}(B_r)}\, d\sigma(x)\, dz = \frac{\sigma(\mathbb{S}^{m-1})}{\mathrm{vol}(B_r)} \int_0^r z^{m+1}\, dz = \frac{mr^2}{m+2}. \end{aligned}$$

Here, $\sigma(\mathbb{S}^{m-1}) = 2\pi^{\frac{m}{2}}/\Gamma(\frac{m}{2})$ is the spherical measure (area) of the unit sphere $\mathbb{S}^{m-1} \subset \mathbb{R}^m$. □

Orthogonal Projections. Given a closed subspace $S \subseteq \mathbb{R}^m$, we denote the orthogonal projection onto S by P_S. In particular, if $z \in \mathbb{R}^m$ is a vector, then we use P_z and $P_{z^\perp}$ to represent orthogonal projections onto $\mathrm{span}(z)$ and $\mathrm{span}(z)^\perp$ respectively. Hence, for any $x \in \mathbb{R}^m$, we have

$$P_z(x) = \frac{\langle z, x\rangle z}{\|z\|_2^2}, \quad x = P_z(x) + P_{z^\perp}(x), \quad \text{and} \quad \|x\|_2^2 = \|P_z(x)\|_2^2 + \|P_{z^\perp}(x)\|_2^2. \tag{39}$$

Lemma A.4. *Let $\mathcal{A}$ be as in (3) with step size $\delta > 0$, and largest element q_{max}. Suppose that $w \in \mathbb{R}^{N_0}$ satisfies $\|w\|_\infty \le q_{\max}$, and consider the quantization scheme given by (10). Let $\theta_t := \angle(X_t, u_{t-1})$ be the angle between X_t and u_{t-1}. Then, for $t = 1, 2, \dots, N_0$, we have*

$$\|u_t\|_2^2 - \|u_{t-1}\|_2^2 \le \begin{cases} \frac{\delta^2}{4}\|X_t\|_2^2 - \|u_{t-1}\|_2^2 \cos^2\theta_t & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{max}, \\ 0 & \text{otherwise.} \end{cases} \tag{40}$$

Proof. By applying (39) and (10), we get

$$\begin{aligned} \|P_{X_t}(u_t)\|_2^2 &= \frac{(X_t^\top u_t)^2}{\|X_t\|_2^2} = \frac{(X_t^\top u_{t-1} + (w_t - q_t)\|X_t\|_2^2)^2}{\|X_t\|_2^4}\|X_t\|_2^2 \\ &= \left(w_t + \frac{X_t^\top u_{t-1}}{\|X_t\|_2^2} - q_t\right)^2 \|X_t\|_2^2 = \left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\right)^2 \|X_t\|_2^2. \end{aligned} \tag{41}$$

The last equation holds because $X_t^\top u_{t-1} = \|X_t\|_2\|u_{t-1}\|_2\cos\theta_t$. Note that

$$\left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\right)^2 - \left(\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right)^2 = \underbrace{\left(w_t + \frac{2\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\right)}_{\text{(I)}}\underbrace{(w_t - q_t)}_{\text{(II)}},$$

$|w_t| \le q_{\max}$, and $q_t = \mathcal{Q}\left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right)$. If $\left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right) > q_{\max}$, then $q_t = q_{\max}$ and thus $0 \le q_t - w_t \le \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t$. So (I) $\ge w_t + 2(q_t - w_t) - q_t = q_t - w_t \ge 0$ and (II) ≤ 0. Moreover, if $\left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right) < -q_{\max}$, then $q_t = -q_{\max}$ and $\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t \le q_t - w_t \le 0$. Hence, (I) $\le w_t + 2(q_t - w_t) - q_t = q_t - w_t \le 0$ and (II) ≥ 0. It follows that

$$\left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\right)^2 \le \left(\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right)^2 \tag{42}$$

when $\left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max}$. Now, assume that $\left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}$. In this case, since the argument of $\mathcal{Q}$ lies in the active range of $\mathcal{A}$, we obtain

$$\left(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\right)^2 \le \frac{\delta^2}{4}. \tag{43}$$

Applying (42) and (43) to (41), one can get

$$\|P_{X_t}(u_t)\|_2^2 \le \begin{cases} \frac{\delta^2}{4}\|X_t\|_2^2 & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}, \\ \|u_{t-1}\|_2^2\cos^2\theta_t & \text{otherwise.} \end{cases} \tag{44}$$

Further, we have

$$P_{X_t^\perp}(u_t) = P_{X_t^\perp}(u_{t-1} + w_t X_t - q_t X_t) = P_{X_t^\perp}(u_{t-1}). \tag{45}$$

It follows that

$$\begin{aligned}\|u_t\|_2^2 - \|u_{t-1}\|_2^2 &= \|P_{X_t}(u_t)\|_2^2 + \|P_{X_t^\perp}(u_t)\|_2^2 - \|u_{t-1}\|_2^2 \\ &= \|P_{X_t}(u_t)\|_2^2 + \|P_{X_t^\perp}(u_{t-1})\|_2^2 - \|u_{t-1}\|_2^2 && \text{(by (45))} \\ &= \|P_{X_t}(u_t)\|_2^2 - \|P_{X_t}(u_{t-1})\|_2^2 && \text{(using (39))} \\ &= \|P_{X_t}(u_t)\|_2^2 - \|u_{t-1}\|_2^2 \cos^2\theta_t.\end{aligned}$$

Substituting $\|P_{X_t}(u_t)\|_2^2$ with its upper bounds in (44), we obtain (40). □

Lemma A.5. *Let $\mathcal{A}$ be as in (3) with step size $\delta > 0$, and largest element $q_{\max}$. Suppose that $w \in \mathbb{R}^{N_0}$ satisfies $\|w\|_\infty \le q_{\max}$, and consider the quantization scheme given by (10). Additionally, denote the information of the first $t-1$ quantization steps by a σ-algebra $\mathcal{F}_{t-1}$, and let $\beta, \eta > 0$, $s^2 \in (0,1)$. Then the following results hold for $t = 1, 2, \dots, N_0$.*

(1) $\mathbb{E}e^{\eta\|u_t\|_2^2} \le \max\left\{\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}}), \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\right\}$.

(2) $\mathbb{E}(e^{\eta\beta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1}) \le -\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1})(e^{\eta\beta\|u_{t-1}\|_2^2} - 1) + e^{\eta\beta\|u_{t-1}\|_2^2}$.

Here, θ_t is the angle between X_t and u_{t-1}.

Proof. (1) In the t-th step, by Lemma A.4, we have

$$\|u_t\|_2^2 - \|u_{t-1}\|_2^2 \le \begin{cases} \frac{\delta^2}{4}\|X_t\|_2^2 - \|u_{t-1}\|_2^2\cos^2\theta_t & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}, \\ 0 & \text{otherwise,}\end{cases}$$

where $\theta_t = \angle(X_t, u_{t-1})$ is the angle between X_t and u_{t-1}. On the one hand, if $\left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}$, we obtain

$$\mathbb{E}e^{\eta\|u_t\|_2^2} = \mathbb{E}(e^{\eta(\|u_t\|_2^2 - \|u_{t-1}\|_2^2)}e^{\eta\|u_{t-1}\|_2^2}) \le \mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}}) \tag{46}$$

On the other hand, if $\left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max}$, we get

$$\mathbb{E}e^{\eta\|u_t\|_2^2} = \mathbb{E}(e^{\eta(\|u_t\|_2^2 - \|u_{t-1}\|_2^2)}e^{\eta\|u_{t-1}\|_2^2}) \le \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}. \tag{47}$$

Combining (46) and (47), we conclude that

$$\mathbb{E}e^{\eta\|u_t\|_2^2} \le \max\left\{\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}}), \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\right\}.$$

(2) Conditioning on $\mathcal{F}_{t-1}$, the function $f(x) = e^{\eta\beta x\|u_{t-1}\|_2^2}$ is convex. It follows that

$$\begin{aligned}\mathbb{E}(e^{\eta\beta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1}) &= \mathbb{E}(f(\cos^2\theta_t\cdot 0 + (1-\cos^2\theta_t)\cdot 1) \mid \mathcal{F}_{t-1}) \\ &\le \mathbb{E}(\cos^2\theta_t + (1-\cos^2\theta_t)e^{\eta\beta\|u_{t-1}\|_2^2} \mid \mathcal{F}_{t-1}) \\ &\le \mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1}) + (1 - \mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1}))e^{\eta\beta\|u_{t-1}\|_2^2} \\ &= -\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1})(e^{\eta\beta\|u_{t-1}\|_2^2} - 1) + e^{\eta\beta\|u_{t-1}\|_2^2}.\end{aligned}$$

□

Appendix B. Fusing Convolution and Batch Normalization Layers

For many neural networks, e.g. MobileNets and ResNets, a convolutional layer is usually followed by a batch normalization (BN) layer to normalize the output. Here, we show how our quantization approach admits a simple modification that takes into account such BN layers. Specifically, denote the convolution operator by * and suppose that a convolutional layer

$$f_{\text{conv}}(x) := w_{\text{conv}} * x + b_{\text{conv}} \tag{48}$$

is followed by a BN layer given by

$$f_{\text{bn}}(x) := \frac{x - \hat{\mu}}{\sqrt{\hat{\sigma}^2 + \epsilon}} \cdot w_{\text{bn}} + b_{\text{bn}}. \tag{49}$$

Here, w_{conv}, w_{bn}, b_{conv}, and b_{bn} are learned parameters and $\hat{\mu}$, $\hat{\sigma}$ are the running mean and standard-deviation respectively while $\epsilon > 0$ is to keep the denominator bounded away from 0. Note that the parameters in both Equation (48) and Equation (49) are calculated per-channel over the mini-batches during training, but fixed thereafter.

Table 3. Top-1 accuracy drop for ResNet-18 and ResNet-50.

Model	b	m	Unfused		Fused	
			C	Acc Drop (%)	C	Acc Drop (%)
ResNet-18	4	2048	1.16	1.63	1.29	1.72
	4	4096		1.21		1.18
	5	2048		0.71		0.72
	5	4096		0.49		0.51
ResNet-50	5	512	1.81	0.97	1.82	1.03
	5	1024		0.90		0.81
	5	2048		0.62		0.64

Thus, to quantize the convolutional and subsequent BN layers simultaneously, we first observe that we can write

$$f_{\text{bn}} \circ f_{\text{conv}}(x) = w_{\text{new}} * x + b_{\text{new}} \tag{50}$$

with

$$w_{\text{new}} := \frac{w_{\text{conv}} w_{\text{bn}}}{\sqrt{\hat{\sigma}^2 + \epsilon}}, \quad b_{\text{new}} := \frac{(b_{\text{conv}} - \hat{\mu}) w_{\text{bn}}}{\sqrt{\hat{\sigma}^2 + \epsilon}} + b_{\text{bn}}.$$

As a result, to quantize the convolutional and subsequent BN layer simulatenously, we can simply quantize the parameters $w_{\text{new}}, b_{\text{new}}$ in (50) using our methods. Although BN layers are not quantized in our experiments in Section 5, we will show here that the proposed algorithm GPFQ is robust to neural network fusion as described above. In Table 3, we compare the Top-1 quantization accuracy between fused ResNets and unfused ResNets when quantized using our methods with different bits and batch sizes. Note that the scalar C for unfused networks remains the same as in Table 1 while C for fused networks is selected using

the procedure after Equation (32). We observe that the performance of GPFQ for fused ResNet-18 and ResNet-50 is quite similar to that for unfused networks.

Appendix C. Quantizing Large Weights

In this section, we demonstrate that the proposed quantization algorithm (10) is still effective for weights with magnitudes that exceed the largest element, $q_{\max} = K\delta$, in the alphabet set $\mathcal{A}$.

Specifically, we prove Theorem C.2, bounding the expected error when $n := n(\delta)$ entries of w are greater than $K\delta$. In turn, Theorem C.2 suggests that in some cases, choosing δ such that $n(\delta) > 0$ may be advantageous, a finding that is consistent with our experiments in Section 5. We begin with the following lemma needed to prove Theorem C.2.

Lemma C.1. *Let $\mathcal{A}$ be as in (3) with step size $\delta > 0$, and largest element q_{max}. Suppose that $w \in \mathbb{R}^{N_0}$ satisfies $\|w\|_\infty \le kq_{\max}$ for some $k > 1$, and consider the quantization scheme given by (10). Let $\theta_t := \angle(X_t, u_{t-1})$ be the angle between X_t and u_{t-1}. Then*

(51)
$$\|u_t\|_2^2 \le \begin{cases} \frac{\delta^2}{4}\|X_t\|_2^2 + \|u_{t-1}\|_2^2(1-\cos^2\theta_t) & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}, \\ \|u_{t-1}\|_2^2 & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max} \text{ and } |w_t| \le q_{\max}, \\ (\|u_{t-1}\|_2 + (k-1)q_{\max}\|X_t\|_2)^2 & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max} \text{ and } |w_t| > q_{\max} \end{cases}$$

holds for $t = 1, 2, \dots, N_0$.

Proof. The first two cases in (51) are covered by Lemma A.4. So it remains to consider the case where $\left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max}$ and $|w_t| > q_{\max}$. As in the proof of Lemma A.4, we have

$$\|u_t\|_2^2 = (v_t - q_t)^2\|X_t\|_2^2 + (1-\cos^2\theta_t)\|u_{t-1}\|_2^2$$

where $v_t := w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t$. Since $q_t = \mathcal{Q}(v_t)$ and $|v_t| > q_{\max}$, we get $q_t = \operatorname{sign}(v_t)q_{\max}$. It follows that

$$\begin{aligned} \|u_t\|_2^2 &= (v_t - \operatorname{sign}(v_t)q_{\max})^2\|X_t\|_2^2 + (1-\cos^2\theta_t)\|u_{t-1}\|_2^2 \\ &= (|v_t| - q_{\max})^2\|X_t\|_2^2 + (1-\cos^2\theta_t)\|u_{t-1}\|_2^2. \end{aligned} \tag{52}$$

By symmetry, we can assume without loss of generality that $v_t > q_{\max}$. In this case, since $|w_t| \le \|w\|_\infty \le kq_{\max}$,

$$|v_t| - q_{\max} = v_t - q_{\max} = w_t - q_{\max} + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t \le (k-1)q_{\max} + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t.$$

Then (52) becomes

$$\begin{aligned} \|u_t\|_2^2 &\le \left((k-1)q_{\max} + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right)^2\|X_t\|_2^2 + (1-\cos^2\theta_t)\|u_{t-1}\|_2^2 \\ &= (k-1)^2q_{\max}^2\|X_t\|_2^2 + \|u_{t-1}\|_2^2 + 2(k-1)q_{\max}\langle X_t, u_{t-1}\rangle \\ &= \|(k-1)q_{\max}X_t + u_{t-1}\|_2^2 \\ &\le (\|u_{t-1}\|_2 + (k-1)q_{\max}\|X_t\|_2)^2. \end{aligned}$$

This completes the proof. □

We are now ready to bound the expected quantization error in the case when some weights have magnitude greater than $q_{\max}$.

Theorem C.2. *Suppose that the columns X_t of $X \in \mathbb{R}^{m\times N_0}$ are drawn independently from a probability distribution for which there exists $s \in (0,1)$ and $r > 0$ such that $\|X_t\|_2 \le r$ almost surely, and such that for all unit vector $u \in \mathbb{S}^{m-1}$ we have*

$$\mathbb{E}\frac{\langle X_t, u\rangle^2}{\|X_t\|_2^2} \ge s^2. \tag{53}$$

Let $\mathcal{A}$ be the alphabet in (3) with step size $\delta > 0$, and the largest element $q_{\max}$. Let $w \in \mathbb{R}^{N_0}$ be the weights associated with a neuron such that $\|w\|_\infty \le kq_{\max}$ for some $k > 1$. Let $n = |\{t : |w_t| > q_{\max}\}|$ be the number of weights with magnitude greater than $q_{\max}$. Quantizing w using (10), we have

$$\mathbb{E}\|Xw - Xq\|_2^2 \le \left(nr(k-1)q_{\max} + \frac{\delta r}{2s}\right)^2. \tag{54}$$

Proof. Let θ_t be the angle between X_t and u_{t-1}. It follows from (53) that

$$\mathbb{E}(\cos^2\theta_t \mid u_{t-1}) = \mathbb{E}\Big(\frac{\langle X_t, u_{t-1}\rangle^2}{\|X_t\|_2^2\|u_{t-1}\|_2^2} \;\Big|\; u_{t-1}\Big) \ge s^2.$$

Since $\|X_t\|_2 \le r$ almost surely and $\mathbb{E}(\cos^2\theta_t \mid u_{t-1}) \ge s^2$, by Lemma C.1, we obtain

$$\mathbb{E}(\|u_t\|_2^2 \mid u_{t-1}) \le \tag{55}$$
$$\begin{cases} a\|u_{t-1}\|_2^2 + b & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}, \\ \|u_{t-1}\|_2^2 & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max} \text{ and } |w_t| \le q_{\max}, \\ (\|u_{t-1}\|_2 + c)^2 & \text{if } \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| > q_{\max} \text{ and } |w_t| > q_{\max} \end{cases}$$

where $a := (1-s^2)$, $b := \frac{\delta^2}{4}r^2$, and $c := (k-1)rq_{\max}$. Define the indices $t_0 := 0 < t_1 < \ldots < t_n < t_{n+1} := N_0 + 1$ where $|w_{t_j}| > q_{\max}$ for $1 \le j \le n$ and let

$$m_j := \left|\{t_{j-1} < t < t_j : \left|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right| \le q_{\max}\}\right|, \quad 1 \le j \le n+1.$$

We first consider the case where $n = 1$. Applying the law of total expectation to the first two cases in (55), one obtains

$$\mathbb{E}\|u_{t_1-1}\|_2^2 \le a^{m_1}\mathbb{E}\|u_0\|_2^2 + b(1 + a + \ldots + a^{m_1-1}) = b(1 + a + \ldots + a^{m_1-1}). \tag{56}$$

In the last equation, we used the fact $u_0 = 0$. Next, the last case in (55) can be used to bound $\mathbb{E}\|u_{t_1}\|_2^2$. Specifically, we have

$$\begin{aligned}
\mathbb{E}\|u_{t_1}\|_2^2 &= \mathbb{E}(\mathbb{E}(\|u_{t_1}\|_2^2 \mid u_{t_1-1}) \\
&\leq \mathbb{E}\|u_{t_1-1}\|_2^2 + 2c\mathbb{E}\|u_{t_1-1}\|_2 + c^2 && \text{(using (55))} \\
&\leq \mathbb{E}\|u_{t_1-1}\|_2^2 + 2c(\mathbb{E}\|u_{t_1-1}\|_2^2)^{\frac{1}{2}} + c^2 && \text{(by Jensen's inequality)} \\
&= ((\mathbb{E}\|u_{t_1-1}\|_2^2)^{\frac{1}{2}} + c)^2 \\
&\leq \left(c + \sqrt{b(1 + a + \ldots + a^{m_1-1})}\right)^2 && \text{(using (56)).} \qquad (57)
\end{aligned}$$

Since $|w_t| \leq q_{\max}$ for $t_1 < t < t_2 = N_0 + 1$, using (55), we can derive

$$\begin{aligned}
\mathbb{E}\|u_{t_2-1}\|_2^2 &\leq a^{m_2}\mathbb{E}\|u_{t_1}\|_2^2 + b(1 + a + \ldots + a^{m_2-1}) \\
&\leq a^{m_2}\left(c + \sqrt{b \cdot \frac{1 - a^{m_1}}{1 - a}}\right)^2 + b \cdot \frac{1 - a^{m_2}}{1 - a} && \text{(using (57))} \\
&= a^{m_2}c^2 + b \cdot \frac{1 - a^{m_1+m_2}}{1 - a} + 2a^{m_2}c\sqrt{\frac{b(1 - a^{m_1})}{1 - a}} \\
&\leq a^{m_2}c^2 + b \cdot \frac{1 - a^{m_1+m_2}}{1 - a} + 2a^{m_2/2}c\sqrt{\frac{b(1 - a^{m_1+m_2})}{1 - a}} && \text{(since } 0 < a < 1) \\
&\leq \left(c + \sqrt{\frac{b(1 - a^{m_1+m_2})}{1 - a}}\right)^2. && (58)
\end{aligned}$$

Hence, we obtain $\mathbb{E}\|u_{N_0}\|_2^2 \leq \left(c + \sqrt{\frac{b}{1-a}}\right)^2$ when $n = 1$. Proceeding by induction on n, we obtain

$$\mathbb{E}\|u_{N_0}\|_2^2 \leq \left(nc + \sqrt{\frac{b}{1 - a}}\right)^2 = \left(nr(k-1)q_{\max} + \frac{\delta r}{2s}\right)^2. \qquad (59)$$

Since $u_{N_0} = Xw - Xq$, we have $\mathbb{E}\|Xw - Xq\|_2^2 \leq \left(nr(k-1)q_{\max} + \frac{\delta r}{2s}\right)^2$. □

Our numerical experiments in Section 5 demonstrated that choosing our alphabet with $q_{\max} < \|w\|_\infty$ can yield better results than if we strictly conformed to choosing $\mathcal{A}$ with $q_{\max} \geq \|w\|_\infty$. Let us now see how Theorem C.2 can help explain these experimental results. First, recall from (3) that $q_{\max} = K\delta = 2^{b-1}\delta$ where b is the number of bits, and observe that the condition $\|w\|_\infty \leq kq_{\max}$ in Theorem C.2 implies that we can set $k = \|w\|_\infty / q_{\max}$. Thus (54), coupled with Jensen's inequality, yields

$$\mathbb{E}\|Xw - Xq\|_2 \leq nr(\|w\|_\infty - q_{\max}) + \frac{\delta r}{2s} = nr(\|w\|_\infty - 2^{b-1}\delta) + \frac{\delta r}{2s}. \qquad (60)$$

Now, note that s, r are fixed parameters that only depend on the input data distribution so for a fixed b, $n = n(\delta) = |\{t : |w_t| > 2^{b-1}\delta\}|$ is a decreasing function of δ. In other words, the right hand side of (60) is the sum of an increasing function of δ and a decreasing function of δ. This means that there exists an optimal value of δ^* that minimizes the bound. In

particular, it may not always be optimal to choose a large δ such that $\|w\|_\infty = 2^{b-1}\delta$. This gives a theoretical justification for why the simple grid search we used in Section 5 yielded better results.

Appendix D. Theoretical Analysis for Gaussian Clusters

In this section, we will prove Theorem 3.2, which we first restate here for convenience.

Theorem 3.2: Let $X \in \mathbb{R}^{m\times N_0}$ be as in (23) and let $\mathcal{A}$ be as in (3), with step size $\delta > 0$ and the largest element $q_{\max}$. Let $p \in \mathbb{N}$, $K := 1 + \sigma^{-2}\max_{1\le i\le d}\|z^{(i)}\|_\infty^2$, and $w \in \mathbb{R}^{N_0}$ be the weights associated with a neuron, with $\|w\|_\infty \le q_{\max}$. Quantizing w using (10), we have

$$\mathrm{P}\Big(\|Xw - Xq\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}, \quad \text{and}$$

$$\mathrm{P}\Big(\max_{1\le t\le N_0}\|u_t\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^{p-1}}.$$

If the activation function φ is ξ-Lipschitz continuous, then

$$\mathrm{P}\Big(\|\varphi(Xw) - \varphi(Xq)\|_2^2 \ge 4pm^2K^2\xi^2\delta^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}.$$

D.1. **Proof of Theorem 3.2.** Due to $\|X_t\|_2^2 = \sum_{i=1}^d \|Y_t^{(i)}\|_2^2$,

$$\mathbb{E}\|X_t\|_2^2 = \sum_{i=1}^d \mathbb{E}\|Y_t^{(i)}\|_2^2 = \sum_{i=1}^d (n\sigma^2 + n(z_t^{(i)})^2) = m\sigma^2 + n\sum_{i=1}^d (z_t^{(i)})^2 \tag{61}$$

Additionally, given a unit vector $u = (u^{(1)}, u^{(2)}, \dots, u^{(d)}) \in \mathbb{R}^m$ with $u^{(i)} \in \mathbb{R}^n$, we have $\langle X_t, u\rangle = \sum_{i=1}^d \langle Y_t^{(i)}, u^{(i)}\rangle \sim \mathcal{N}\Big(\sum_{i=1}^d z_t^{(i)} u^{(i)\top}\mathbb{1}_n, \sigma^2\Big)$. In fact, once we get the lower bound of $\mathbb{E}\frac{\langle X_t,u\rangle^2}{\|X_t\|_2^2}$ as in (11), the quantization error for unbounded data (23) can be derived similarly to the proof of Theorem 3.1, albeit using different techniques. It follows from the Cauchy-Schwarz inequality that

$$\mathbb{E}\frac{\langle X_t, u\rangle^2}{\|X_t\|_2^2} \ge \frac{(\mathbb{E}|\langle X_t, u\rangle|)^2}{\mathbb{E}\|X_t\|_2^2}. \tag{62}$$

$\mathbb{E}\|X_t\|_2^2$ is given by (61) while $\mathbb{E}|\langle X_t, u\rangle|$ can be evaluated by the following results.

Lemma D.1. *Let $Z \sim \mathcal{N}(\mu, \sigma^2)$ be a normally distributed random variable. Then*

$$\mathbb{E}|Z| \ge \sigma\sqrt{\frac{2}{\pi}}\Big(1 - \frac{4}{27\pi}\Big). \tag{63}$$

Proof. Let $\Psi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^x e^{-t^2/2}\,dt$ be the normal cumulative distribution function. Due to $Z \sim \mathcal{N}(\mu, \sigma^2)$, the folded normal distribution $|Z|$ has mean $\mathbb{E}|Z| = \sigma\sqrt{\frac{2}{\pi}}e^{-\mu^2/2\sigma^2} + \mu(1 - 2\Psi(-\frac{\mu}{\sigma}))$. A well-known result [9, 29] that can be used to bound $\Psi(x)$ is

$$\int_x^\infty e^{-t^2/2}\,dt \le \min\Big(\sqrt{\frac{\pi}{2}}, \frac{1}{x}\Big)e^{-x^2/2}, \quad \text{for } x > 0. \tag{64}$$

Additionally, in order to evaluate $\mathbb{E}|Z|$, it suffices to analyze the case $\mu \geq 0$ because one can replace Z by $-Z$ without changing $|Z|$ when $\mu < 0$. So we suppose $\mu \geq 0$.

By (64), we obtain

$$\mathbb{E}|Z| = \sigma\sqrt{\frac{2}{\pi}}e^{-\mu^2/2\sigma^2} + \mu - 2\mu\Psi(-\mu/\sigma) = \sigma\sqrt{\frac{2}{\pi}}e^{-\mu^2/2\sigma^2} + \mu - \mu\sqrt{\frac{2}{\pi}}\int_{\mu/\sigma}^{\infty} e^{-t^2/2}\,dt$$
$$\geq \sigma\sqrt{\frac{2}{\pi}}e^{-\mu^2/2\sigma^2} + \mu - \min\left(\mu, \sigma\sqrt{\frac{2}{\pi}}\right)e^{-\mu^2/2\sigma^2}.$$

If $\mu \geq \sigma\sqrt{\frac{2}{\pi}}$, then one can easily get $\mathbb{E}|Z| \geq \mu \geq \sigma\sqrt{\frac{2}{\pi}}$. Further, if $0 \leq \mu < \sigma\sqrt{\frac{2}{\pi}}$, then $\mathbb{E}|Z| \geq (\sigma\sqrt{2/\pi} - \mu)e^{-\mu^2/2\sigma^2} + \mu$. Due to $e^x \geq 1 + x$ for all $x \in \mathbb{R}$, one can get

$$\mathbb{E}|Z| \geq (\sigma\sqrt{2/\pi} - \mu)(1 - \mu^2/2\sigma^2) + \mu = \frac{1}{2\sigma^2}\mu^3 - \frac{1}{\sigma\sqrt{2\pi}}\mu^2 + \sigma\sqrt{\frac{2}{\pi}} \geq \sigma\sqrt{\frac{2}{\pi}}\left(1 - \frac{4}{27\pi}\right).$$

In the last inequality, we optimized in $\mu \in (0, \sigma\sqrt{2/\pi})$ and thus chose $\mu = \frac{2}{3}\cdot\sigma\sqrt{\frac{2}{\pi}}$. □

Lemma D.2. *Let clustered data $X = [X_1, X_2, \ldots, X_{N_0}] \in \mathbb{R}^{m\times N_0}$ be defined as in (23) and $u \in \mathbb{R}^m$ be a unit vector. Then, for $1 \leq t \leq N_0$, we have*

$$\mathbb{E}\frac{\langle X_t, u\rangle^2}{\|X_t\|_2^2} \geq \frac{5}{9}\cdot\frac{\sigma^2}{m(\sigma^2 + \max_{1\leq i\leq d}\|z^{(i)}\|_\infty^2)}. \tag{65}$$

Proof. Since $\langle X_t, u\rangle$ is normally distributed with variance σ^2, (63) implies $\mathbb{E}|\langle X_t, u\rangle| \geq \sigma\sqrt{\frac{2}{\pi}}\left(1 - \frac{4}{27\pi}\right)$. Plugging the inequality above and (61) into (62), we obtain

$$\mathbb{E}\frac{\langle X_t, u\rangle^2}{\|X_t\|_2^2} \geq \frac{(\mathbb{E}|\langle X_t, u\rangle|)^2}{\mathbb{E}\|X_t\|_2^2} \geq \frac{2(1 - \frac{4}{27\pi})^2}{\pi}\cdot\frac{\sigma^2}{m\sigma^2 + n\sum_{i=1}^d (z_t^{(i)})^2} \geq \frac{5}{9}\cdot\frac{\sigma^2}{m\sigma^2 + n\sum_{i=1}^d (z_t^{(i)})^2}.$$

Therefore, (65) holds due to $(z_t^{(i)})^2 \leq \|z^{(i)}\|_\infty^2 \leq \max_{1\leq i\leq d}\|z^{(i)}\|_\infty^2$ and $m = nd$. □

Now we are ready to prove Theorem 3.2.

Proof. Let $\alpha > 0$ and $\eta > 0$. By using exactly the same argument as in (15), at the t-th step of (10), we have

$$\mathrm{P}(\|u_t\|_2^2 \geq \alpha) \leq e^{-\eta\alpha}\mathbb{E}e^{\eta\|u_t\|_2^2}. \tag{66}$$

Moreover, Lemma A.5 implies

$$\mathbb{E}e^{\eta\|u_t\|_2^2} \leq \max\left\{\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2}e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}), \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\right\} \tag{67}$$

Until now our analysis here has been quite similar to what we did for bounded input data in Theorem 3.1. Nevertheless, unlike Theorem 3.1, we will control the moment generating function of $\|X_t\|_2^2$ because $\|X_t\|_2^2$ is unbounded. Specifically, applying the Cauchy-Schwarz inequality and Lemma A.5 (2) with $\beta = 2$, we obtain

$$\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2}e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1}) \leq \left(\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}\right)^{\frac{1}{2}}\left(\mathbb{E}(e^{2\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1})\right)^{\frac{1}{2}}$$
$$\leq \left(\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}\right)^{\frac{1}{2}}\left(-\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1})(e^{2\eta\|u_{t-1}\|_2^2} - 1) + e^{2\eta\|u_{t-1}\|_2^2}\right)^{\frac{1}{2}} \tag{68}$$

In the first step, we also used the fact that X_t is independent of $\mathcal{F}_{t-1}$. By (65), we have

$$\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1}) = \mathbb{E}\Big(\frac{\langle X_t, u_{t-1}\rangle^2}{\|X_t\|_2^2\|u_{t-1}\|_2^2}\,\Big|\,\mathcal{F}_{t-1}\Big) \geq \frac{5}{9mK} =: s^2.$$

Plugging the inequality above into (68), we get

$$\begin{aligned}
\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}} \mid \mathcal{F}_{t-1}) &\leq \big(\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}\big)^{\frac{1}{2}}\Big(-s^2(e^{2\eta\|u_{t-1}\|_2^2}-1)+e^{2\eta\|u_{t-1}\|_2^2}\Big)^{\frac{1}{2}} \\
&= \big(\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}\big)^{\frac{1}{2}}\Big(e^{2\eta\|u_{t-1}\|_2^2}(1-s^2)+s^2\Big)^{\frac{1}{2}} \\
&\leq \big(\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}\big)^{\frac{1}{2}}(e^{\eta\|u_{t-1}\|_2^2}(1-s^2)^{\frac{1}{2}}+s) \\
&\leq \big(\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}\big)^{\frac{1}{2}}(e^{\eta\|u_{t-1}\|_2^2}(1-\frac{1}{2}s^2)+s) \qquad (69)
\end{aligned}$$

where the last two inequalities hold due to $(x^2+y^2)^{\frac{1}{2}} \leq |x|+|y|$ for all $x, y \in \mathbb{R}$, and $(1-x)^{\frac{1}{2}} \leq 1-\frac{1}{2}x$ whenever $x \leq 1$.

Now we evaluate $\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2}$ and note that

$$\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2} = \mathbb{E}\exp\Big(\frac{\eta\delta^2}{2}\sum_{i=1}^{d}\|Y_t^{(i)}\|_2^2\Big) = \prod_{i=1}^{d}\mathbb{E}\exp\Big(\frac{\eta\delta^2}{2}\|Y_t^{(i)}\|_2^2\Big). \qquad (70)$$

Since $Y_t^{(i)} \sim \mathcal{N}(z_t^{(i)}\mathbb{1}_n, \sigma^2 I_n)$, we have

$$\begin{aligned}
\mathbb{E}\exp\Big(\frac{\eta\delta^2}{2}\|Y_t^{(i)}\|_2^2\Big) &= \Big[\frac{1}{\sigma\sqrt{2\pi}}\int_{\mathbb{R}}\exp\Big(-\frac{(x-z_t^{(i)})^2}{2\sigma^2}+\frac{\eta\delta^2x^2}{2}\Big)\,dx\Big]^n \\
&= \Big\{\frac{1}{\sigma\sqrt{2\pi}}\cdot\exp\Big(\frac{\eta\delta^2(z_t^{(i)})^2}{2-2\eta\delta^2\sigma^2}\Big)\int_{\mathbb{R}}\exp\Big[-\frac{1-\eta\delta^2\sigma^2}{2\sigma^2}\Big(x-\frac{z_t^{(i)}}{1-\eta\delta^2\sigma^2}\Big)^2\Big]\,dx\Big\}^n \\
&= \Big[(1-\eta\delta^2\sigma^2)^{-\frac{1}{2}}\exp\Big(\frac{\eta\delta^2(z_t^{(i)})^2}{2-2\eta\delta^2\sigma^2}\Big)\Big]^n
\end{aligned}$$

where the last equality holds if $\eta\delta^2\sigma^2 < 1$ and we use the integral of the normal density function:

$$\Big(\frac{1-\eta\delta^2\sigma^2}{2\pi\sigma^2}\Big)^{\frac{1}{2}}\int_{\mathbb{R}}\exp\Big[-\frac{1-\eta\delta^2\sigma^2}{2\sigma^2}\Big(x-\frac{z_t^{(i)}}{1-\eta\delta^2\sigma^2}\Big)^2\Big]\,dx = 1.$$

Notice that $\frac{1}{1-x} \leq 1+2x$ for $x \in [0, \frac{1}{2}]$ and $1+x \leq e^x$ for all $x \in \mathbb{R}$. Now, we suppose $\eta\delta^2\sigma^2 \leq \frac{1}{2}$ and thus $(1-\eta\delta^2\sigma^2)^{-\frac{1}{2}} = \Big(\frac{1}{1-\eta\delta^2\sigma^2}\Big)^{\frac{1}{2}} \leq (1+2\eta\delta^2\sigma^2)^{\frac{1}{2}} \leq e^{\eta\delta^2\sigma^2}$. It follows that

$$\begin{aligned}
\mathbb{E}\exp\Big(\frac{\eta\delta^2}{2}\|Y_t^{(i)}\|_2^2\Big) &\leq \Big[\exp\Big(\eta\delta^2\sigma^2+\frac{\eta\delta^2(z_t^{(i)})^2}{2-2\eta\delta^2\sigma^2}\Big)\Big]^n \leq \Big[\exp\Big(\eta\delta^2\sigma^2+\eta\delta^2(z_t^{(i)})^2\Big)\Big]^n \\
&\leq \exp\Big(n\eta\delta^2\sigma^2\Big(1+\frac{\|z^{(i)}\|_\infty^2}{\sigma^2}\Big)\Big) \\
&\leq \exp(nK\eta\delta^2\sigma^2) \qquad (71)
\end{aligned}$$

Substituting (71) into (70), we get

$$\mathbb{E}e^{\frac{\eta\delta^2}{2}\|X_t\|_2^2} \le e^{ndK\eta\delta^2\sigma^2} = e^{mK\eta\delta^2\sigma^2}. \tag{72}$$

Combining (69) and (72), if $\eta\delta^2\sigma^2 \le \frac{1}{2}$, then

$$\begin{aligned}
\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}}) &= \mathbb{E}\Big(\mathbb{E}(e^{\frac{\eta\delta^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}} \mid \mathcal{F}_{t-1})\Big) \\
&\le \mathbb{E}\Big(e^{\frac{1}{2}mK\eta\delta^2\sigma^2}(e^{\eta\|u_{t-1}\|_2^2}(1-\frac{1}{2}s^2)+s)\Big) \\
&= e^{\frac{1}{2}mK\eta\delta^2\sigma^2}(1-\frac{1}{2}s^2)\mathbb{E}e^{\eta\|u_{t-1}\|_2^2} + se^{\frac{1}{2}mK\eta\delta^2\sigma^2} \\
&=: a\mathbb{E}e^{\eta\|u_{t-1}\|_2^2} + b \qquad (73)
\end{aligned}$$

with $a := (1-s^2/2)e^{\frac{1}{2}mK\eta\delta^2\sigma^2}$ and $b := se^{\frac{1}{2}mK\eta\delta^2\sigma^2}$. Plugging (73) into (67), we have $\mathbb{E}e^{\eta\|u_t\|_2^2} \le \max\{a\mathbb{E}e^{\eta\|u_{t-1}\|_2^2}+b, \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\}$. Next, similar to the argument in (18), iterating expectations yields $\mathbb{E}e^{\eta\|u_t\|_2^2} \le a^{t_0}\mathbb{E}(e^{\eta\|u_0\|_2^2}) + b(1+a+\ldots+a^{t_0}) = a^{t_0} + \frac{b(1-a^{t_0})}{1-a} \le 1 + \frac{b}{1-a}$ where the last inequality holds if $a := (1-s^2/2)e^{mK\eta\delta^2\sigma^2/2} < 1$. So we can now choose $\eta = \frac{-\log(1-s^2/2)}{mK\delta^2\sigma^2}$, which satisfies $\eta\delta^2\sigma^2 \in [0, 1/2]$ as required from before. Indeed, due to $m, K \ge 1$ and $s^2 = \frac{5}{9Km} \le \frac{5}{9}$, we have $\eta\delta^2\sigma^2 = \frac{-\log(1-s^2/2)}{mK} \le -\log(1-\frac{5}{18}) < \frac{1}{2}$. Then we get $a = (1-\frac{1}{2}s^2)^{1/2}$ and $b = s(1-\frac{1}{2}s^2)^{-1/2}$. It follows from (66) and $s^2 = \frac{5}{9mK}$ that

$$\begin{aligned}
\mathrm{P}(\|u_t\|_2^2 \ge \alpha) &\le e^{-\eta\alpha}\Big(1+\frac{b}{1-a}\Big) = \exp\Big(\frac{\alpha\log(1-s^2/2)}{mK\delta^2\sigma^2}\Big)\Big(1+\frac{s(1-\frac{1}{2}s^2)^{-1/2}}{1-\sqrt{1-s^2/2}}\Big) \\
&\le \exp\Big(\frac{-\alpha s^2}{2mK\delta^2\sigma^2}\Big)\Big(1+\frac{s(1-\frac{1}{2}s^2)^{-1/2}+s}{s^2/2}\Big) \qquad (\text{since } \log(1+x)\le x) \\
&= \exp\Big(\frac{-\alpha s^2}{2mK\delta^2\sigma^2}\Big)\Big(1+2\frac{(1-\frac{1}{2}s^2)^{-1/2}+1}{s}\Big) \\
&= \exp\Big(-\frac{5\alpha}{18m^2K^2\delta^2\sigma^2}\Big)\Big[1+6\sqrt{\frac{mK}{5}}\Big(1-\frac{5}{18mK}\Big)^{-1/2}+6\sqrt{\frac{mK}{5}}\Big] \\
&\le c\sqrt{mK}\exp\Big(-\frac{\alpha}{4m^2K^2\delta^2\sigma^2}\Big)
\end{aligned}$$

where $c > 0$ is an absolute constant. Pick $\alpha = 4m^2K^2\delta^2\sigma^2\log(N_0^p)$ to get

$$\mathrm{P}\Big(\|u_t\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \le c\sqrt{mK}N_0^{-p}. \tag{74}$$

From (74) we can first conclude, by setting $t = N_0$ and using the fact $u_{N_0} = Xw - Xq$, that

$$\mathrm{P}\Big(\|Xw-Xq\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \le \frac{c\sqrt{mK}}{N_0^p}.$$

If the activation function φ is ξ-Lipschitz, then $\|\varphi(Xw)-\varphi(Xq)\|_2 \le \xi\|Xw-Xq\|_2$ and thus

$$\mathrm{P}\Big(\|\varphi(Xw)-\varphi(Xq)\|_2^2 \ge 4pm^2K^2\xi^2\delta^2\sigma^2\log N_0\Big) \le \frac{c\sqrt{mK}}{N_0^p}.$$

Moreover, applying a union bound over t, yields

$$\mathrm{P}\Big(\max_{1\le t\le N_0}\|u_t\|_2^2 \ge 4pm^2K^2\delta^2\sigma^2\log N_0\Big) \le \frac{c\sqrt{mK}}{N_0^{p-1}}.$$

□

Appendix E. Theoretical Analysis for Sparse GPFQ

In this section, we will show that Theorem 4.1 and Theorem 4.2 (restated here for convenience) hold.

Theorem 4.1: Under the conditions of Theorem 3.1, we have the following.
(a) Quantizing w using (29) with the alphabet $\mathcal{A}$ in (3), we have

$$\mathrm{P}\Big(\|Xw-Xq\|_2^2 \le \frac{r^2(2\lambda+\delta)^2}{s^2}\log N_0\Big) \ge 1-\frac{1}{N_0^2}\Big(2+\frac{1}{\sqrt{1-s^2}}\Big).$$

(b) Quantizing w using (30) with the alphabet $\widetilde{\mathcal{A}}$ in (4), we have

$$\mathrm{P}\Big(\|Xw-Xq\|_2^2 \le \frac{r^2\max\{2\lambda,\delta\}^2}{s^2}\log N_0\Big) \ge 1-\frac{1}{N_0^2}\Big(2+\frac{1}{\sqrt{1-s^2}}\Big).$$

Theorem 4.2: Under the assumptions of Theorem 3.2, the followings inequalities hold.
(a) Quantizing w using (29) with the alphabet $\mathcal{A}$ in (3), we have

$$\mathrm{P}\Big(\|Xw-Xq\|_2^2 \ge 4pm^2K^2(2\lambda+\delta)^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}.$$

(b) Quantizing w using (30) with the alphabet $\widetilde{\mathcal{A}}$ in (4), we have

$$\mathrm{P}\Big(\|Xw-Xq\|_2^2 \ge 4pm^2K^2\max\{2\lambda,\delta\}^2\sigma^2\log N_0\Big) \lesssim \frac{\sqrt{mK}}{N_0^p}.$$

Note that the difference between the sparse GPFQ and the GPFQ in (10) is the usage of thresholding functions. So the key point is to adapt Lemma A.4 and Lemma A.5 for those changes.

E.1. Sparse GPFQ with Soft Thresholding.

We first focus on the error analysis for (29) which needs the following lemmata.

Lemma E.1. *Let $\mathcal{A}$ be one of the alphabets defined in (3) with step size $\delta>0$, and the largest element q_{max}. Let $\theta_t := \angle(X_t, u_{t-1})$ be the angle between X_t and u_{t-1}. Suppose that $w\in\mathbb{R}^{N_0}$ satisfies $\|w\|_\infty \le q_{\max}$, and consider the quantization scheme given by (29). Then, for $t=1,2,\dots,N_0$, we have*

$$\|u_t\|_2^2-\|u_{t-1}\|_2^2 \le \begin{cases} \frac{(2\lambda+\delta)^2}{4}\|X_t\|_2^2-\|u_{t-1}\|_2^2\cos^2\theta_t & \text{if } \Big|w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big| \le q_{max}+\lambda,\\ 0 & \text{otherwise.}\end{cases} \tag{75}$$

Proof. By applying exactly the same argument as in Lemma A.4, one can get

$$\|P_{X_t}(u_t)\|_2^2 = \Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\Big)^2 \|X_t\|_2^2. \tag{76}$$

and

$$\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\Big)^2 - \Big(\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big)^2 = \underbrace{\Big(w_t + \frac{2\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\Big)}_{\text{(I)}}\underbrace{(w_t - q_t)}_{\text{(II)}},$$

where $|w_t| \le q_{\max}$ and $q_t = \mathcal{Q}\circ s_\lambda\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big)$. We proceed by going through the cases.

First, if $\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big) > q_{\max} + \lambda$, then $q_t = q_{\max}$ and thus $\lambda \le q_t - w_t + \lambda < \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t$. So (I) $> w_t + 2(q_t - w_t + \lambda) - q_t = q_t - w_t + 2\lambda \ge 2\lambda$ and (II) ≤ 0. Moreover, if $\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big) < -q_{\max} - \lambda$, then $q_t = -q_{\max}$ and $\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t < q_t - w_t - \lambda \le -\lambda$. Hence, (I) $< w_t + 2(q_t - w_t - \lambda) - q_t = q_t - w_t - 2\lambda \le -2\lambda$ and (II) ≥ 0. It follows that

$$\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\Big)^2 \le \Big(\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big)^2 \tag{77}$$

when $\Big|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big| > q_{\max} + \lambda$.

Now, assume that $\Big|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big| \le q_{\max} + \lambda$. In this case, let $v_t := s_\lambda\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big)$. Then $|v_t| \le q_{\max}$ and $\Big|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - v_t\Big| \le \lambda$. Since $q_t = \mathcal{Q}(v_t)$, we obtain

$$\begin{aligned}\Big(w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - q_t\Big)^2 &= \Big|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - v_t + v_t - q_t\Big|^2 \\ &\le \Big(\Big|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t - v_t\Big| + |v_t - q_t|\Big)^2 \\ &\le \Big(\lambda + \frac{\delta}{2}\Big)^2. \end{aligned} \tag{78}$$

Applying (77) and (78) to (76), one can get

$$\|P_{X_t}(u_t)\|_2^2 \le \begin{cases} \frac{(2\lambda+\delta)^2}{4}\|X_t\|_2^2 & \text{if } \Big|w_t + \frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big| \le q_{\max} + \lambda, \\ \|u_{t-1}\|_2^2\cos^2\theta_t & \text{otherwise.}\end{cases} \tag{79}$$

Again, by the same discussion after (45) in Lemma A.4, we have $\|u_t\|_2^2 - \|u_{t-1}\|_2^2 = \|P_{X_t}(u_t)\|_2^2 - \|u_{t-1}\|_2^2\cos^2\theta_t$. Replacing $\|P_{X_t}(u_t)\|_2^2$ with its upper bounds in (79), we obtain (75). ☐

Lemma E.2. *Let $\mathcal{A}$ be one of the alphabets defined in (3) with step size $\delta > 0$, and the largest element $q_{\max}$. Suppose that $w \in \mathbb{R}^{N_0}$ satisfies $\|w\|_\infty \le q_{\max}$, and consider the quantization scheme given by (29). Additionally, denote the information of the first $t-1$ quantization steps by a σ-algebra $\mathcal{F}_{t-1}$, and let $\beta, \eta > 0$, $s^2 \in (0,1)$. Then the following results hold for $t = 1, 2, \dots, N_0$.*

(1) $\mathbb{E}e^{\eta\|u_t\|_2^2} \le \max\Big\{\mathbb{E}(e^{\frac{\eta(2\lambda+\delta)^2}{4}\|X_t\|_2^2 e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}}), \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\Big\}$.

(2) $\mathbb{E}(e^{\eta\beta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1}) \leq -\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1})(e^{\eta\beta\|u_{t-1}\|_2^2}-1)+e^{\eta\beta\|u_{t-1}\|_2^2}$.

Here, θ_t is the angle between X_t and u_{t-1}.

Proof. Similar to Lemma A.5, the inequality (1) follows immediately from (75). The proof of part (2) is identical with the one in Lemma A.5. □

Now we are ready to prove Theorem 4.1 as follows.

Proof. The only difference between Lemma A.5 and its analogue Lemma E.2 is that δ^2 in Lemma A.5 is replaced by $(2\lambda+\delta)^2$. Note that Lemma A.5 was used in the proof of both Theorem 3.1 and Theorem 3.2 in which δ^2 serves as a coefficient. Hence, by substituting δ^2 with $(2\lambda+\delta)^2$, every step in the proof still works and thus Theorem 4.1 holds. □

E.2. **Sparse GPFQ with Hard Thresholding.** Now we navigate to the error analysis for (30). Again, Lemma A.4 and Lemma A.5 are altered as follows.

Lemma E.3. *Let $\widetilde{\mathcal{A}}$ be one of the alphabets defined in (4) with step size $\delta>0$, the largest element q_{max}, and threshold $\lambda\in(0,q_{\max})$. Let $\theta_t:=\angle(X_t,u_{t-1})$ be the angle between X_t and u_{t-1}. Suppose that $w\in\mathbb{R}^{N_0}$ satisfies $\|w\|_\infty\leq q_{\max}$, and consider the quantization scheme given by (30). Then, for $t=1,2,\dots,N_0$, we have*

$$\|u_t\|_2^2-\|u_{t-1}\|_2^2\leq\begin{cases}\frac{\max\{2\lambda,\delta\}^2}{4}\|X_t\|_2^2-\|u_{t-1}\|_2^2\cos^2\theta_t & \text{if } \left|w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right|\leq q_{max},\\ 0 & \text{otherwise.}\end{cases}\tag{80}$$

Proof. By applying exactly the same argument as in Lemma A.4, we obtain

$$\|P_{X_t}(u_t)\|_2^2=\left(w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t-q_t\right)^2\|X_t\|_2^2.\tag{81}$$

where $|w_t|\leq q_{\max}$ and $q_t=\mathcal{Q}\circ h_\lambda\Big(w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\Big)$. Due to $\lambda\in(0,q_{\max})$, we have $\mathcal{Q}\circ h_\lambda(z)=\mathcal{Q}(z)$ for $|z|>q_{\max}$. Thus, it follows from the discussion in Lemma A.4 that

$$\left(w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t-q_t\right)^2\leq\left(\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right)^2\tag{82}$$

when $\left|w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right|>q_{\max}$.

Now, assume that $\left|w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right|\leq q_{\max}$. In this case, because the argument of $\mathcal{Q}$ lies in the active range of $\mathcal{A}$, we obtain

$$\left(w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t-q_t\right)^2\leq\max\left\{\lambda,\frac{\delta}{2}\right\}^2.\tag{83}$$

Applying (82) and (83) to (81), one can get

$$\|P_{X_t}(u_t)\|_2^2\leq\begin{cases}\frac{\max\{2\lambda,\delta\}^2}{4}\|X_t\|_2^2 & \text{if } \left|w_t+\frac{\|u_{t-1}\|_2}{\|X_t\|_2}\cos\theta_t\right|\leq q_{\max},\\ \|u_{t-1}\|_2^2\cos^2\theta_t & \text{otherwise.}\end{cases}\tag{84}$$

Again, by the same discussion after (45) in Lemma A.4, we have $\|u_t\|_2^2-\|u_{t-1}\|_2^2=\|P_{X_t}(u_t)\|_2^2-\|u_{t-1}\|_2^2\cos^2\theta_t$. Replacing $\|P_{X_t}(u_t)\|_2^2$ with its upper bounds in (84), we obtain (80). □

Lemma E.4. *Let $\widetilde{\mathcal{A}}$ be one of the alphabets defined in (4) with step size $\delta > 0$, the largest element $q_{\max}$ and $\lambda \in (0, q_{\max})$. Suppose that $w \in \mathbb{R}^{N_0}$ satisfies $\|w\|_\infty \leq q_{\max}$, and consider the quantization scheme given by (30). Additionally, denote the information of the first $t-1$ quantization steps by a σ-algebra $\mathcal{F}_{t-1}$, and let $\beta, \eta > 0$, $s^2 \in (0,1)$. Then the following results hold for $t = 1, 2, \ldots, N_0$.*

(1) $\mathbb{E}e^{\eta\|u_t\|_2^2} \leq \max\Big\{\mathbb{E}(e^{\frac{\eta \max\{2\lambda,\delta\}^2}{4}\|X_t\|_2^2} e^{\eta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)}), \mathbb{E}e^{\eta\|u_{t-1}\|_2^2}\Big\}$.

(2) $\mathbb{E}(e^{\eta\beta\|u_{t-1}\|_2^2(1-\cos^2\theta_t)} \mid \mathcal{F}_{t-1}) \leq -\mathbb{E}(\cos^2\theta_t \mid \mathcal{F}_{t-1})(e^{\eta\beta\|u_{t-1}\|_2^2} - 1) + e^{\eta\beta\|u_{t-1}\|_2^2}$.

Here, θ_t is the angle between X_t and u_{t-1}.

Proof. Similar to Lemma A.5, the inequality (1) follows immediately from (80). The proof of part (2) is identical with the one in Lemma A.5. □

The proof of Theorem 4.2 is given as follows.

Proof. The only difference between Lemma A.5 and its analogue Lemma E.4 is that δ^2 in Lemma A.5 is replaced by $\max\{2\lambda + \delta\}^2$. Note that Lemma A.5 was used in the proof of both Theorem 3.1 and Theorem 3.2 in which δ^2 serves as a coefficient. Hence, by substituting δ^2 with $\max\{2\lambda + \delta\}^2$, it is not hard to verify that Theorem 4.2 holds. □

Department of Mathematics, University of California San Diego
Email address: jiz003@ucsd.edu

Department of Mathematics, University of California San Diego
Email address: yiz044@ucsd.edu

Department of Mathematics and Halıcıoğlu Data Science Institute, University of California San Diego
Email address: rsaab@ucsd.edu