WHEN LOWER RECONSTRUCTION LOSS HURTS: DISTRIBUTIONALLY ROBUST REFINEMENT FOR LOW-BIT LLM QUANTIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Weight-only post-training quantization (PTQ) relies heavily on reconstruction loss minimization to preserve model quality at low precision. We show that the weights favored by minimizing this loss need not yield better model performance on new tasks. In fact, we find that lower reconstruction loss can even degrade model performance on the same calibration data. Our analysis further shows that weights with lower reconstruction loss on calibration data can have higher loss than other weights when the distribution of input activations changes. Motivated by these observations and our analysis, we propose Distributionally Robust Quantization (DRQ), a post-hoc refinement process that minimizes worst-case reconstruction loss over a constrained set of input activation distributions. DRQ refines the integer codes representing quantized weights within the existing quantization grid, keeping quantization parameters and inference operators unchanged. Extensive experiments show that DRQ improves models quantized by six representative PTQ methods, including AWQ, GPTQ, and ParoQuant, and delivers gains across both dense and mixture-of-experts large language models. These results establish DRQ as a general post-hoc refinement framework for weight-only PTQ, achieving better downstream performance without adding inference overhead.

1 INTRODUCTION

Large language models (LLMs) have demonstrated strong capabilities in language understanding, reasoning, and code generation (OpenAI et al., 2023; Grattafiori et al., 2024; Chen et al., 2021; DeepSeek-AI, 2026). Their increasing memory footprint and bandwidth demands, however, make deployment costly on resource-constrained devices. Weight-only post-training quantization (PTQ) addresses this problem by compressing pretrained weights into low-bit formats such as INT4 while retaining the original activation precision. It reduces weight storage and memory traffic to accelerate memory-bound autoregressive decoding (Frantar et al., 2023; Lin et al., 2024; Ashkboos et al., 2024).

Most weight-only PTQ methods rely on minimizing reconstruction loss on calibration data to preserve model quality (Frantar et al., 2023; Lin et al., 2024; Shao et al., 2024; Tseng et al., 2024; Li et al., 2025; Sun et al., 2025; Liang et al., 2026). For a linear layer, this means matching the output of the quantized weights to that of the full-precision weights on the calibration data,

$$\min_{\widehat{\mathbf{W}} \in \mathcal{Q}} \mathcal{L}(\widehat{\mathbf{W}}), \qquad \mathcal{L}(\widehat{\mathbf{W}}) = \frac{1}{n} \left\| \widehat{\mathbf{W}}\mathbf{X} - \mathbf{W}\mathbf{X} \right\|_{\mathrm{F}}^{2}, \tag{1}$$

where $\mathbf{W}$ and $\widehat{\mathbf{W}}$ denote the full-precision and quantized weights, respectively, the quantization grid $\mathcal{Q}$ denotes the set of all possible quantized weights, and $\mathbf{X}$ denotes the associated input activations of the calibration data. Methods differ in how they minimize this loss. For example, GPTQ compensates for rounding errors (Frantar et al., 2023), AWQ searches for scaling and clipping parameters (Lin et al., 2024), and ParoQuant learns transformations (Liang et al., 2026).

We show that reducing reconstruction loss on calibration data can degrade the performance of the quantized model. We first quantize Llama-3.2-1B-Instruct with 3-bit AWQ using WikiText-2 (WT2)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

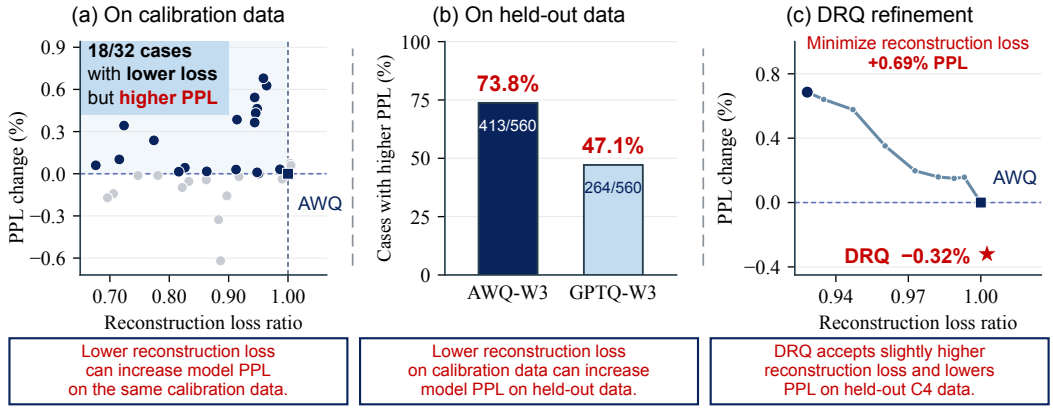


Figure 1: Lower reconstruction loss need not improve quantized model performance. Results are from Llama-3.2-1B-Instruct with 3-bit quantization. (a) Among 32 modifications to AWQ-quantized weights, 18 reduce reconstruction loss but increase model PPL on the same WT2 calibration data. (b) Among 560 modified weight matrices per method that reduce reconstruction loss on calibration data, 413 (73.8%) for AWQ and 264 (47.1%) for GPTQ increase model PPL on held-out WT2 data. (c) Further minimizing reconstruction loss increases held-out C4 PPL by 0.69%, whereas DRQ accepts slightly higher reconstruction loss on calibration data and lowers C4 PPL by 0.32%. Each evaluation modifies one linear layer, keeping all other weights fixed. In (a,c), reconstruction loss is divided by its initial value, and PPL changes are relative to the initial quantized model.

(Merity et al., 2017) as the calibration dataset. Starting from the weights quantized by AWQ, we further optimize one linear layer at a time to reduce reconstruction loss on the calibration data. Specifically, we search over the integer codes representing its weights, while keeping the scales, zero points, and all other layers' weights fixed. We repeat this procedure at 16 layers of the model using two different calibration sets sampled from WT2, producing 32 modified weight matrices. In 18 of 32 cases, the modified weights yield lower reconstruction loss but higher model perplexity (PPL) on the same calibration data (Figure 1a). This mismatch also extends to held-out data. In a separate experiment, we examine all 112 linear layers of Llama-3.2-1B-Instruct quantized with 3-bit AWQ. For each layer, we try replacing the integer codes of the quantized weights while keeping the scales and zero points fixed. We evaluate these replacements using reconstruction loss on calibration data. For each row of a layer's weight matrix, we select the integer-code change that most reduces reconstruction loss on calibration data, while keeping the scales and zero points fixed. We sort these changes from the largest loss reduction to the smallest. Starting from the original quantized weights each time, we create five modified weight matrices by applying the top $1/256$, $1/128$, $1/64$, $1/32$, and $1/16$ of the changes in this list, respectively. This finally gives 560 different matrices, all with lower reconstruction loss on the calibration data than the corresponding original quantized weights. We then evaluate each matrix by replacing only the corresponding layer's weights and measuring model PPL on WT2 data reserved for evaluation. Of these 560 replacements, 413 (73.8%) increase PPL relative to the initial quantized model.

These results expose a mismatch between reducing reconstruction loss on calibration data and improving the performance of the quantized model. Ideally, minimizing reconstruction loss brings the outputs of quantized layers closer to those of their full-precision counterparts on calibration data. Yet this closer match need not generalize to other datasets or improve model performance even on the same calibration data. Optimizing reconstruction loss alone can therefore degrade the performance of the quantized model.

To figure out this problem, we study how this loss depends on the input activations observed during calibration. For example, when an input channel of a linear layer stays close to zero, even large differences between the quantized and full-precision weights in the corresponding column can have little effect on the layer outputs. Yet new data could produce larger activations in this channel, so the same weight differences can cause larger output differences and thus higher reconstruction loss. Our analysis identifies when a quantized weight matrix with lower reconstruction loss on calibration data can have higher loss than another under constrained changes in the distribution of input activations (Theorem 3.1).

Motivated by this analysis, we propose *Distributionally Robust Quantization (DRQ)*, a general refinement framework for weight-only PTQ. DRQ minimizes the worst-case reconstruction loss over a set of input activation distributions, with constraints on how these distributions differ from the calibration distribution. This objective combines reconstruction loss on calibration data with its largest possible increase under these constraints. A real illustrative example in Figure 1c shows that further minimizing reconstruction loss on calibration data increases model PPL on held-out C4. DRQ instead accepts a small increase in reconstruction loss on calibration data and lowers PPL on held-out C4 relative to the initial model quantized by AWQ. The resulting quantized weights therefore perform better on data not used for calibration. DRQ updates the integer codes of quantized weights while keeping the bit width, grouping, scales, zero points, and inference operators unchanged. All refinement is offline, so DRQ adds no inference overhead.

Contributions. We uncover a counterintuitive phenomenon in weight-only PTQ. Lower reconstruction loss on calibration data does not necessarily lead to better performance of the quantized model, even on the same data. Motivated by this finding, we analyze how changes in input activation distributions affect reconstruction loss. This analysis leads to DRQ, which refines quantized weights by minimizing the worst-case reconstruction loss over a constrained set of input activation distributions. DRQ updates the integer codes representing quantized weights while preserving the quantization format and inference operators, so these improvements add no inference overhead. Experiments show that DRQ improves models quantized by six representative PTQ methods, with gains across both dense and mixture-of-experts (MoE) architectures.

2 Preliminaries

2.1 Weight-only quantization on a fixed grid

Quantization approximates high-precision values using a finite set of values that can be stored with fewer bits. Weight-only post-training quantization applies this to pretrained weights while retaining the original activation precision. Consider a linear layer with weights $\mathbf{W} \in \mathbb{R}^{m\times d}$, where d and m are the input and output dimensions. A b-bit uniform quantizer represents each weight w by an integer code $q \in \mathcal{Z}_b = \{0, \dots, 2^b - 1\}$. With round-to-nearest (RTN), we compute the code q and its corresponding quantized weight $\hat{w}$ as

$$q = \mathrm{clip}\big(\mathrm{round}(w/s) + z,\, 0,\, 2^b - 1\big)\,, \qquad \hat{w} = s(q - z).$$

Here $s > 0$ is the scale, which sets the spacing between adjacent quantized values, and the zero point z is the integer code corresponding to zero. The rounding operation selects the nearest integer, and clipping keeps the code between 0 and $2^b - 1$. Following common practice in most PTQ work, we use group-wise quantization (Dettmers & Zettlemoyer, 2023; Liang et al., 2026), which divides each row of $\mathbf{W}$ into groups of g consecutive weights along the input dimension. Each group uses its own scale s and zero point z. Each group uses its own scale, so large weights in one group do not force other groups to use a larger quantization step. This helps represent smaller weights more accurately. Quantized weights in the same group share a scale s and a zero point z. With s and z fixed, $\hat{w} = s(q - z)$ shows that each quantized weight is determined by its integer code q. DRQ therefore refines the weights by updating only their integer codes.

2.2 Reconstruction on calibration data

Calibration uses sample text to collect the input activations reaching each layer. Let the input activation matrix $\mathbf{X} = [x_1, \dots, x_n] \in \mathbb{R}^{d\times n}$ contain these vectors, with one token per column. The full-precision and quantized outputs are $\mathbf{WX}$ and $\widehat{\mathbf{W}}\mathbf{X}$. Their squared difference, summed over output channels and averaged over tokens, is the reconstruction loss

$$\mathcal{L}(\widehat{\mathbf{W}}) = \frac{1}{n}\left\|\widehat{\mathbf{W}}\mathbf{X} - \mathbf{W}\mathbf{X}\right\|_{\mathrm{F}}^2 = \mathrm{tr}\Big((\widehat{\mathbf{W}} - \mathbf{W})\mathbf{H}(\widehat{\mathbf{W}} - \mathbf{W})^\top\Big)\,, \tag{2}$$

where $\mathbf{H} = n^{-1}\mathbf{X}\mathbf{X}^\top$ is the activation second-moment matrix. The squared Frobenius norm sums the squared differences between the two output matrices, and tr sums matrix diagonal entries. Most PTQ methods such as GPTQ, AWQ, and OmniQuant (Frantar et al., 2023; Lin et al., 2024; Shao et al., 2024) minimize reconstruction loss in different ways to obtain quantized weights $\widehat{\mathbf{W}}$.

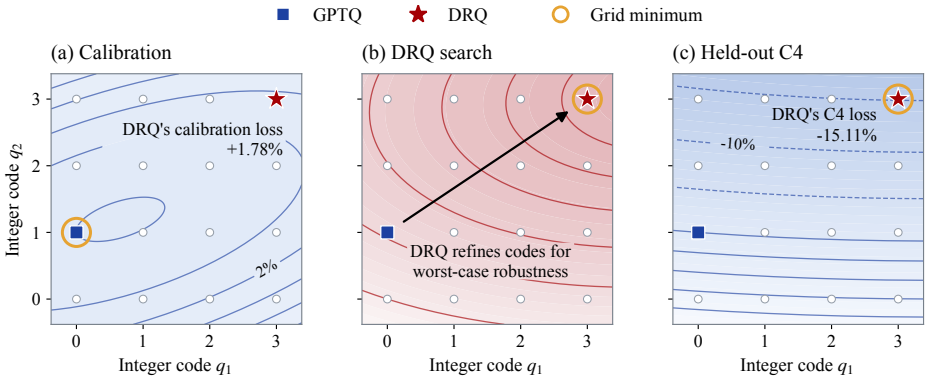


Figure 2: An illustrative example of integer-code refinement with DRQ on Llama-3.2-1B-Instruct quantized with 2-bit GPTQ. We select two integer codes, q_1 and q_2, from the quantized weights and plot the loss contours over different pairs of code values. The percentages indicate changes in reconstruction loss relative to the initial quantized weights. Panel (a) shows reconstruction loss on the WikiText-2 (WT2) calibration data, panel (b) illustrates how DRQ refines the two integer codes, and panel (c) shows reconstruction loss on C4. Although DRQ increases reconstruction loss on the calibration data by 1.78%, it reduces reconstruction loss on C4 by 15.11%, both relative to the initial quantized weights.

3 DRQ: Distributionally Robust Quantization

In this section, we first analyze how changes in the distribution of input activations affect reconstruction loss (Section 3.1). Based on this analysis, we formulate a worst-case reconstruction objective and derive an equivalent form for optimizing quantized weights (Section 3.2). Finally, we describe how DRQ optimizes this objective by updating the integer codes of weights produced by an existing PTQ method, while keeping the quantization parameters fixed (Section 3.3).

3.1 Reconstruction under changing input distributions

New data can produce input activations that differ from those observed during calibration. Given the quantized weights $\widehat{\mathbf{W}}$, Equation (2) shows that the effect of input activations on reconstruction loss is determined by the second-moment matrix $\mathbf{H}$. We can therefore study how new data affects this loss by examining changes in that matrix. To describe these changes, we introduce a perturbation matrix $\mathbf{\Delta}$ and write the second-moment matrix of the new input activations as $\mathbf{H} + \mathbf{\Delta}$. Here, $\mathbf{H}$ is computed from the calibration data, and $\mathbf{\Delta}$ is the difference in second-moment matrices between the new and the calibration data.

To examine how changes in the input distribution can increase reconstruction loss, we restrict $\mathbf{\Delta}$ to be positive semidefinite, written $\mathbf{\Delta} \succeq 0$. Under this constraint, the expected reconstruction loss of any fixed quantized weights can increase or remain unchanged. One example is adding noise of the form εu to an input activation vector x, where ε has mean zero and is independent of x. We introduce an *input direction* $u \in \mathbb{R}^d$ to describe how the input channels change and measure the resulting increase in reconstruction loss. In the noise example above, its entries specify the sign and relative size of the noise added to each channel. For two input channels, $u = (1, 1)^\top$ means adding ε to both channels, so they increase or decrease together. With $u = (1, -1)^\top$, we add ε to the first channel and subtract it from the second, so one increases while the other decreases. In this example, the second-moment matrix changes by $\mathbf{\Delta} = \mathbb{E}[\varepsilon^2] uu^\top \succeq 0$. For fixed quantized weights, this change increases the expected reconstruction loss by $\mathbb{E}[\varepsilon^2] \left\| (\widehat{\mathbf{W}} - \mathbf{W})u \right\|_2^2$. Comparing input directions under the same distribution-change budget therefore lets us identify the changes that increase this loss the most.

The loss increase in the example above depends on both the input direction and the size of the added noise. Without a limit on the noise, reconstruction loss can grow arbitrarily large. We therefore impose $\mathrm{tr}(\mathbf{C}\mathbf{\Delta}) \leq \alpha$, where $\alpha \geq 0$ is the distribution-change budget. The positive-definite matrix $\mathbf{C}$ determines how much change is permitted along each input direction. For a change $\mathbf{\Delta} = tuu^\top$ with

$t \geq 0$, the constraint becomes $t\, u^\top \mathbf{C} u \leq \alpha$. Thus, a larger $u^\top \mathbf{C} u$ permits a smaller t under the same budget. To compare all directions at the same budget, we rescale each u to satisfy $u^\top \mathbf{C} u = 1$. Then $\mathbf{\Delta} = \alpha u u^\top$ uses the full budget α.

Panels (a) and (c) of Figure 2 illustrate how weights with lower reconstruction loss on calibration data can have higher loss on new inputs. Consider two quantized weight matrices with $\mathcal{L}(\widehat{\mathbf{W}}_1) < \mathcal{L}(\widehat{\mathbf{W}}_2)$. Input changes can increase both losses, but increase the loss of $\widehat{\mathbf{W}}_1$ more. For its loss to remain no greater than that of $\widehat{\mathbf{W}}_2$, the difference between these increases must not exceed the original loss difference $\mathcal{L}(\widehat{\mathbf{W}}_2) - \mathcal{L}(\widehat{\mathbf{W}}_1)$. The following theorem gives the exact condition.

Theorem 3.1 (Reconstruction loss under distribution changes)**.** *Let $\widehat{\mathbf{W}}_1$ and $\widehat{\mathbf{W}}_2$ satisfy $\mathcal{L}(\widehat{\mathbf{W}}_1) < \mathcal{L}(\widehat{\mathbf{W}}_2)$. Consider input activation distributions with second-moment matrix $\mathbf{H} + \mathbf{\Delta}$, where $\mathbf{\Delta} \succeq 0$ and $\operatorname{tr}(\mathbf{C}\mathbf{\Delta}) \leq \alpha$. The expected reconstruction loss of $\widehat{\mathbf{W}}_1$ is no greater than that of $\widehat{\mathbf{W}}_2$ for every such distribution if and only if every direction u normalized by $u^\top \mathbf{C} u = 1$ satisfies*

$$\alpha\left(\left\|(\widehat{\mathbf{W}}_1 - \mathbf{W})u\right\|_2^2 - \left\|(\widehat{\mathbf{W}}_2 - \mathbf{W})u\right\|_2^2\right) \leq \mathcal{L}(\widehat{\mathbf{W}}_2) - \mathcal{L}(\widehat{\mathbf{W}}_1). \tag{3}$$

If the inequality fails for a direction u with $u^\top \mathbf{C} u = 1$, the change $\mathbf{\Delta} = \alpha u u^\top$ satisfies both constraints and makes $\widehat{\mathbf{W}}_1$ have higher expected reconstruction loss than $\widehat{\mathbf{W}}_2$.

See Appendix B.3 for the proof. Theorem 3.1 motivates a refinement objective that accounts for both reconstruction loss on calibration data and its largest possible increase under the distribution constraints above. DRQ, introduced next, follows this approach. In the schematic in Figure 2, panel (b) illustrates how DRQ refines codes to accept slightly higher reconstruction loss on calibration data in panel (a) and achieve lower loss on held-out inputs in panel (c).

3.2 Robust reconstruction objective

The preceding analysis compares two weight matrices under the same distribution changes. For refinement, we need an objective for evaluating each weight matrix. The proposed method, Distributionally Robust Quantization (DRQ), therefore considers the largest reconstruction loss that each matrix can have under the stated constraints and seeks weights that make this loss as small as possible. Using the same distribution-change budget α, we minimize

$$\min_{\widehat{\mathbf{W}} \in \mathcal{Q}} \max_{\mathbf{\Delta} \succeq 0,\, \operatorname{tr}(\mathbf{C}\mathbf{\Delta}) \leq \alpha} \operatorname{tr}\left((\widehat{\mathbf{W}} - \mathbf{W})(\mathbf{H} + \mathbf{\Delta})(\widehat{\mathbf{W}} - \mathbf{W})^\top\right), \tag{4}$$

where $\mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}) := \max_{\mathbf{\Delta} \succeq 0,\, \operatorname{tr}(\mathbf{C}\mathbf{\Delta}) \leq \alpha} \operatorname{tr}((\widehat{\mathbf{W}} - \mathbf{W})(\mathbf{H} + \mathbf{\Delta})(\widehat{\mathbf{W}} - \mathbf{W})^\top)$. The outer minimization selects quantized weights on the existing quantization grid that minimize this loss. To evaluate $\mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}})$, the following theorem reduces the inner maximization to finding the input direction u along which reconstruction loss increases the most under the same distribution-change budget. The resulting expression is reconstruction loss on calibration data plus this worst-case increase.

Theorem 3.2 (Worst-case reconstruction loss)**.** *For any $\widehat{\mathbf{W}} \in \mathbb{R}^{m \times d}$, $\mathbf{C} \succ 0$, and $\alpha \geq 0$, the worst-case loss in Equation (4) satisfies*

$$\mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}) = \mathcal{L}(\widehat{\mathbf{W}}) + \alpha \mathcal{S}_{\mathbf{C}}(\widehat{\mathbf{W}}), \tag{5}$$

where the worst-case directional error *is*

$$\mathcal{S}_{\mathbf{C}}(\widehat{\mathbf{W}}) = \max_{u^\top \mathbf{C} u = 1} \left\|(\widehat{\mathbf{W}} - \mathbf{W})u\right\|_2^2. \tag{6}$$

For an input vector u, $(\widehat{\mathbf{W}} - \mathbf{W})u = \widehat{\mathbf{W}}u - \mathbf{W}u$ is the difference between the quantized and full-precision layer outputs. Among directions rescaled to satisfy $u^\top \mathbf{C} u = 1$, the *worst-case input direction* gives the largest squared output difference, $\mathcal{S}_{\mathbf{C}}(\widehat{\mathbf{W}})$. A distribution change $\mathbf{\Delta} = \alpha u u^\top$ along this direction uses the full budget and increases reconstruction loss by $\alpha \mathcal{S}_{\mathbf{C}}(\widehat{\mathbf{W}})$. This is the largest increase among all distribution changes satisfying the constraints.

In Equation (4), α sets the budget for changes in the input activation distribution. In the equivalent objective in Equation (5), it weights the worst-case directional error relative to reconstruction loss on calibration data. DRQ can accept slightly higher reconstruction loss on calibration data when this reduces the largest possible increase enough to lower the total worst-case loss. Appendix B.5 gives the proof and a distribution change that attains this loss.

We introduce $\mathbf{C}$ to account for input activations that remain small during calibration. When an input channel stays near zero, even large differences between the quantized and full-precision weights in the corresponding column can have little effect on the layer outputs. We therefore consider larger possible increases in these activations when evaluating the worst-case loss. To apply the same idea to combinations of channels, we set $\mathbf{C} = (1-\beta)\mathbf{H} + \beta\mathbf{I}$, with $0 < \beta \leq 1$. For directions with the same Euclidean norm, a smaller $u^\top \mathbf{H} u$ gives a smaller $u^\top \mathbf{C} u$ when $\beta < 1$, permitting a larger increase under the same budget. At $\beta = 1$, all directions with the same Euclidean norm are treated equally. The identity term keeps $\mathbf{C}$ positive definite, so changes remain bounded in every direction.

3.3 Refinement on a fixed quantization grid

DRQ processes one Transformer block at a time and refines each linear layer after the base PTQ method quantizes its weights. It changes only integer codes, keeping the bit width, grouping, scales, and zero points fixed.

Algorithm 1 summarizes how we optimize Equation (5), using numerical estimates of the worst-case directions and losses. The set $\mathcal{P}$ contains proposed weight matrices. To preserve the fit to calibration data, reconstruction loss may increase only by a preset percentage relative to the initial weights, on both the search subset and separate checking subsets. All refinement is offline and leaves inference operators unchanged, so DRQ adds no inference overhead. Appendix C gives the search procedure and parameter settings, and Appendix D.3 covers sparsely activated experts.

Algorithm 1 DRQ refinement of one linear layer

Require: Full-precision weights $\mathbf{W}$, initial quantized weights $\widehat{\mathbf{W}}$, calibration activations $\mathbf{X}$, α, β

$\mathbf{H} \leftarrow \mathbf{X}\mathbf{X}^\top/n, \quad \mathbf{C} \leftarrow (1-\beta)\mathbf{H} + \beta\mathbf{I}$
for up to three rounds **do**
  $u \leftarrow \arg\max_{u^\top \mathbf{C} u = 1} \left\| (\widehat{\mathbf{W}} - \mathbf{W})u \right\|_2^2$
  Propose $\mathcal{P} \subset \mathcal{Q}$ by reducing $\mathcal{L}(\widehat{\mathbf{W}}) + \alpha \left\| (\widehat{\mathbf{W}} - \mathbf{W})u \right\|_2^2$
  $\mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}_k) \leftarrow \mathcal{L}(\widehat{\mathbf{W}}_k) + \alpha \max_{u^\top \mathbf{C} u = 1} \left\| (\widehat{\mathbf{W}}_k - \mathbf{W})u \right\|_2^2, \quad \forall \widehat{\mathbf{W}}_k \in \mathcal{P}$
  $\mathcal{P} \leftarrow \{\widehat{\mathbf{W}}_k \in \mathcal{P} : \mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}_k) < \mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}),$ checks pass$\}$
  if $\mathcal{P} = \varnothing$ **then break**
  $\widehat{\mathbf{W}} \leftarrow \arg\min_{\widehat{\mathbf{W}}_k \in \mathcal{P}} \mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}_k)$
end for
return the initial or accepted weights with the lowest $\mathcal{L}_{\mathrm{DRQ}}$

4 Experimental Evaluation

4.1 Experimental setup

Models and tasks. We apply DRQ to both dense and mixture-of-experts (MoE) models, including Llama-3.2-Instruct (1B, 3B), Llama-3.1-Instruct (8B, 70B) (Grattafiori et al., 2024), and Qwen3-30B-A3B (Yang et al., 2025). We evaluate model quality after quantization using two types of metrics: (1) perplexity on WikiText-2 (WT2) (Merity et al., 2017) and C4 (Raffel et al., 2020); (2) zero-shot accuracy on seven downstream tasks, including PIQA (Bisk et al., 2020), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2019), ARC-Easy and ARC-Challenge (Clark et al., 2018), BoolQ (Clark et al., 2019), and MMLU (Hendrycks et al., 2021).

Implementation. We evaluate DRQ at W4A16, W3A16, and W2A16 to study how well it preserves model quality across different weight bit widths. All three settings use group-wise weight quantization with a group size of 128 (Frantar et al., 2023; Lin et al., 2024). Unless otherwise specified, we calibrate with 128 WikiText-2 training sequences of 2,048 tokens and select DRQ hyperparameters using 64 validation samples from Pile (Gao et al., 2020). Experiments run on NVIDIA A800 80GB GPUs unless otherwise specified. Further implementation details are provided in Appendix C.

Baselines. We use GPTQ (Frantar et al., 2023) and AWQ (Lin et al., 2024), two widely used weight-only PTQ methods, as the main baselines for most experiments. To demonstrate that DRQ generalizes across quantization methods, we also include round-to-nearest (RTN), OmniQuant, GPTAQ, and ParoQuant.

Table 1: Perplexity and average task accuracy (%) of dense Llama models across model sizes and weight bit widths. FP denotes the unquantized full-precision model.

Quantization		**Llama-3.2-1B**			**Llama-3.2-3B**			**Llama-3.1-8B**			**Llama-3.1-70B**		
Bits	Method	WT2	C4	ACC	WT2	C4	ACC	WT2	C4	ACC	WT2	C4	ACC
FP	Unquantized	13.16	20.82	58.68	11.05	16.19	66.48	7.22	11.24	74.40	3.78	8.24	80.64
W2	GPTQ	1719.88	1.03e4	34.75	141.54	645.35	34.58	139.90	681.48	35.49	13.92	31.77	48.23
	+ DRQ	**1402.87**	**7233.14**	**35.93**	**130.02**	**585.86**	**35.03**	**98.67**	**390.36**	**36.64**	**12.48**	**27.11**	**54.56**
	AWQ	1.08e5	1.22e5	35.39	2.42e5	5.57e5	35.38	1.63e6	1.84e6	37.80	1.42e6	1.25e6	37.99
	+ DRQ	**1.31e4**	**1.80e4**	**36.51**	**1.85e4**	**1.03e5**	**38.38**	**7.76e5**	**5.66e5**	**38.01**	**1.31e6**	**1.17e6**	**38.01**
W3	GPTQ	18.65	28.84	50.50	15.74	30.34	55.83	8.60	15.39	68.46	5.41	10.01	78.67
	+ DRQ	**17.64**	**28.42**	**51.68**	**14.63**	**22.11**	**60.24**	8.61	**14.91**	**69.48**	**5.39**	**9.92**	**79.03**
	AWQ	21.12	32.73	52.65	15.19	20.80	61.69	9.01	14.43	69.90	5.38	9.77	79.02
	+ DRQ	**20.89**	**31.82**	**53.08**	**14.90**	**20.52**	**62.54**	**9.00**	**14.43**	**70.10**	**5.37**	9.77	**79.23**
W4	GPTQ	14.27	22.24	57.47	11.96	17.57	65.26	7.48	12.07	73.11	4.17	8.59	80.22
	+ DRQ	**14.21**	**22.12**	**58.13**	**11.64**	**17.17**	**65.96**	**7.46**	**12.02**	**74.00**	**4.14**	**8.57**	**80.55**
	AWQ	14.55	22.63	57.27	11.56	16.89	66.18	7.55	12.03	73.49	4.18	8.62	80.38
	+ DRQ	**14.41**	**22.54**	**57.43**	**11.52**	**16.81**	**66.70**	**7.55**	**12.02**	**73.62**	4.19	**8.61**	**80.52**

4.2 Dense-model results across sizes and bit widths

Table 1 shows that DRQ improves average task accuracy across both GPTQ and AWQ at every evaluated model size and bit width, while reducing perplexity in most cases. The gains are particularly pronounced when aggressive quantization causes substantial quality loss. For Llama-3.1-70B with GPTQ W2, DRQ lowers C4 PPL from 31.77 to 27.11 and raises accuracy from 48.23% to 54.56%. Similarly, for Llama-3.2-3B with GPTQ W3, C4 PPL falls from 30.34 to 22.11 and accuracy increases from 55.83% to 60.24%. These results show that DRQ's refinement can recover part of the quality lost at low precision, although severe AWQ-W2 degradation persists. Its benefits also extend to stronger initial quantized models. At W4, where the baselines already retain much of their full-precision accuracy, DRQ improves accuracy in all eight comparisons. Appendix D.1 provides the detailed results.

4.3 Extension to mixture-of-experts models

Table 2 extends the evaluation to Qwen3-30B-A3B at W3A16. With many experts and a limited calibration set, some experts may receive few or no calibration tokens, which makes it harder to quantize these experts accurately. DRQ lowers perplexity on both WT2 and C4 and improves accuracy on five of the seven downstream tasks. Together with the dense-model results, these improvements demonstrate DRQ's applicability across dense and MoE architectures. Further details on MoE quantization and expert refinement are provided in Appendix D.3.

Table 2: Perplexity and seven-task accuracy (%) for Qwen3-30B-A3B W3A16.

Quantization		Perplexity		Accuracy (%)							
Bits	Method	WT2	C4	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
W3	GPTQ	9.29	15.01	78.24	75.85	72.16	86.15	49.57	73.99	68.90	72.12
	+ DRQ	**9.20**	**14.90**	**78.78**	**76.00**	**73.07**	85.87	**52.99**	**76.05**	67.56	**72.90**

4.4 VERSATILITY ACROSS SIX PTQ METHODS

DRQ improves models quantized by all six PTQ methods, reducing both WT2 and C4 perplexity and increasing average accuracy on Llama-3.2-3B W3 (Figure 3). ParoQuant achieves the best results before refinement and retains its lead on all three metrics after applying DRQ, showing that even a strong initial quantized model can benefit from refinement. Detailed task-level results and computation costs are provided in Appendices D.2 and E.1.

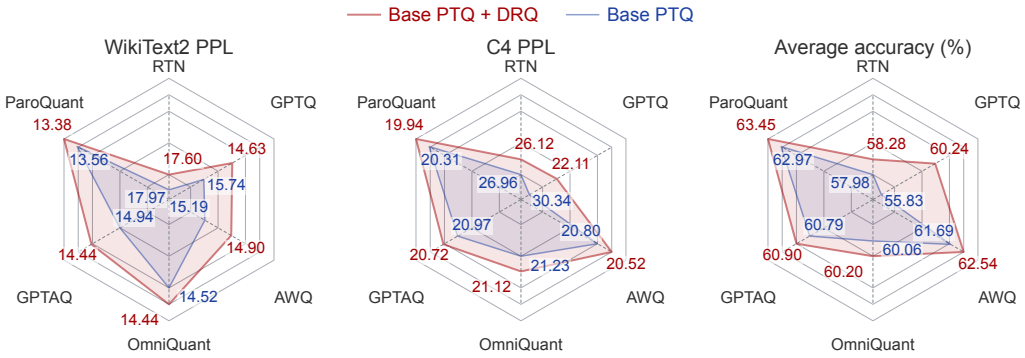


Figure 3: Six-method comparison on Llama-3.2-3B W3.

4.5 CALIBRATION SIZE AND THE CHOICE OF OBJECTIVE

We vary calibration size from 16 to 256 sequences of 2,048 tokens from WT2 or C4 for Llama-3.2-3B-Instruct with GPTQ W4. Reconstruction-only refinement (rec-only) uses the same code search as DRQ with $\alpha = 0$. The first column of Figure 4 shows larger relative loss reductions from rec-only as calibration size grows. This further reduction is possible because GPTQ does not fully minimize reconstruction loss over the larger calibration sets. Yet even as rec-only reduces the loss further relative to GPTQ, model performance worsens, with PPL increasing by orders of magnitude. DRQ instead keeps reconstruction loss close to GPTQ's while achieving lower PPL on both datasets. These results show that more calibration data need not resolve this mismatch, as also established theoretically in Theorem B.2. Figure 7 in Appendix D.4 studies the effect of α and β.

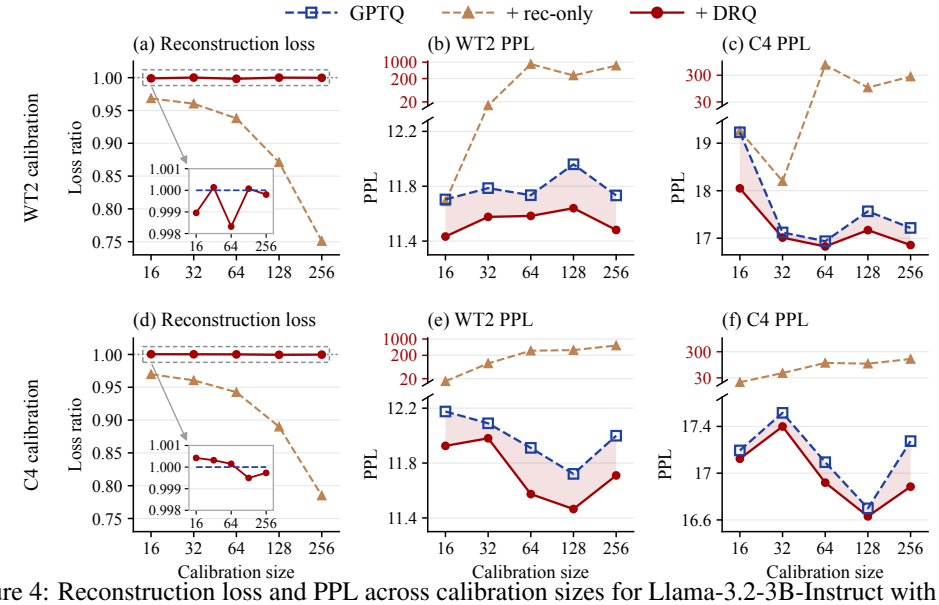


Figure 4: Reconstruction loss and PPL across calibration sizes for Llama-3.2-3B-Instruct with GPTQ W4. Loss ratios divide the reconstruction loss summed across layers after refinement by that before refinement, using each run's own calibration activations.

4.6 Efficiency of DRQ

Figure 5 shows the offline cost of DRQ on Llama-3.1-8B at W3. We time the loop that quantizes successive Transformer blocks, including forward passes on calibration data. Across 20 configurations, refinement takes almost the same time with GPTQ and AWQ, averaging 6.01 and 5.96 minutes, respectively. It is because DRQ applies the same code refinement within the existing quantization grid in both cases. Further efficiency analysis appears in Appendix E.

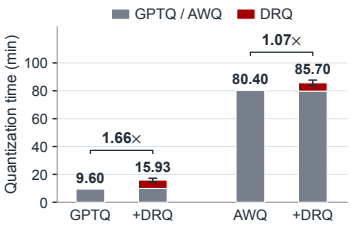


Figure 5: Quantization time for Llama-3.1-8B W3.

5 Related Work

Post-training quantization of LLMs. PTQ methods improve low-bit approximation by reducing the effect of outliers and optimizing quantized weights. Scaling and clipping control channel magnitudes and quantization ranges (Lin et al., 2024; Xiao et al., 2023; Shao et al., 2024), while rotations and affine transformations redistribute values across channels (Ashkboos et al., 2024; Liu et al., 2025; Liang et al., 2026; Sun et al., 2025). QuIP and QTIP combine randomized transformations with adaptive rounding and trellis coding, respectively (Chee et al., 2023; Tseng et al., 2024). For weight optimization, GPTQ uses second-order error compensation during rounding (Frantar et al., 2023), while ReQuant further reduces reconstruction loss by revisiting integer codes on a fixed grid (Ma et al., 2026). Complementary work modifies the reconstruction objective to better preserve model performance. GPTAQ accounts for errors from preceding quantized layers when matching full-precision layer outputs, and CoreQ adjusts the strength of this correction across layers (Li et al., 2025; Cha et al., 2026). GuidedQuant uses gradients of the model's final loss to weight reconstruction loss (Kim et al., 2025), while LFQ matches the model's token probabilities to those of the full-precision model (Lee et al., 2026b). SchurQuant refines both integer codes and quantization parameters and augments its objective (Lee et al., 2026a).

Distributionally robust optimization. Distributionally robust optimization (DRO) minimizes the worst expected loss over an uncertainty set of plausible distributions. Moment constraints, divergence bounds, and transport distances provide different ways to define this set (Delage & Ye, 2010; Mohajerin Esfahani & Kuhn, 2018; Levy et al., 2020). Other approaches learn or constrain how an adversary reweights individual samples (Michel et al., 2022; Setlur et al., 2023). The choice of uncertainty set determines which distributional changes are covered, but overly broad sets can yield conservative solutions or amplify the influence of outliers. Geometry-aware approaches use data structure to restrict these changes or calibrate robust risks (Liu et al., 2022; 2024), while outlier-resistant formulations address data contamination (Zhai et al., 2021). Connections to regularization explain how uncertainty sets can induce penalties on the learning objective (Husain, 2020; Cranko et al., 2021). Complementary work establishes population-risk guarantees for Wasserstein DRO (Azizian et al., 2023; Le & Malick, 2025) and develops scalable solvers (Haddadpour et al., 2022; Mehta et al., 2024; Zhang et al., 2025).

6 Conclusion

The weights that minimize the reconstruction loss on calibration data need not minimize reconstruction loss on new inputs or yield the best model performance. We establish this gap between calibration fit and performance at deployment and show why exact optimization and more calibration data need not close it both theoretically and empirically. Our analysis characterizes how input changes reverse reconstruction rankings and motivates a distributionally robust objective that accounts for those changes. The proposed method, DRQ, combines reconstruction loss on calibration data with a worst-case directional error term to refine existing integer codes of quantized weights. This improves the selection criterion while preserving the quantization grid, scales, and zero points. Experiments across multiple PTQ methods and dense and mixture-of-experts models demonstrate better perplexity and average task accuracy. By retaining the original quantization format and inference operators, DRQ turns robust reconstruction into better downstream performance without extra inference overhead.

AI USE STATEMENT

An OpenAI coding assistant assisted with manuscript organization, language editing, mathematical formulation, and LaTeX validation. The authors are responsible for the final methods, claims, references, and disclosures.

REPRODUCIBILITY STATEMENT

The appendix provides proofs, notation, the refinement procedure, parameter settings, and evaluation protocols. The source package includes the LaTeX, figure assets, plotting data, and scripts for reproducing the candidate comparisons and selection statistics.

REFERENCES

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-free 4-bit inference in rotated LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 100213–100240. Curran Associates, Inc., 2024. doi: 10.52202/079017-3180. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/b5b939436789f76f08b9d0da5e81af7c-Abstract-Conference.html.

Waïss Azizian, Franck Iutzeler, and Jérôme Malick. Exact generalization guarantees for (regularized) Wasserstein distributionally robust models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 14584–14596, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/2f060912eacace9ce61ef339205ec54c-Abstract-Conference.html.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. URL https://arxiv.org/abs/1911.11641.

Seohyeon Cha, Huancheng Chen, Dongjun Kim, Haoran Zhang, Kevin Chan, Gustavo de Veciana, and Haris Vikalo. CoreQ: Learning-free mismatch correction and successive rounding for quantization, 2026. URL https://arxiv.org/abs/2602.05902.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. QuIP: 2-bit quantization of large language models with guarantees. In *Advances in Neural Information Processing Systems*, volume 36, pp. 4396–4429. Curran Associates, Inc., 2023. URL https://arxiv.org/abs/2307.13304.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, et al. Evaluating large language models trained on code, 2021.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pp. 2924–2936, 2019. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Zac Cranko, Zhan Shi, Xinhua Zhang, Richard Nock, and Simon Kornblith. Generalised Lipschitz regularisation equals distributional robustness. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2178–2188, 2021. URL https://proceedings.mlr.press/v139/cranko21a.html.

DeepSeek-AI. DeepSeek-V4.1-Flash: Pushing the limits of KV cache compression, 2026. URL https://arxiv.org/abs/2609.19969.

Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612, 2010. doi: 10.1287/opre.1090.0741. URL https://pubsonline.informs.org/doi/10.1287/opre.1090.0741.

Tim Dettmers and Luke Zettlemoyer. The case for 4-bit precision: k-bit inference scaling laws. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7750–7774. PMLR, 2023. URL https://proceedings.mlr.press/v202/dettmers23a.html.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers, 2023. URL https://arxiv.org/abs/2210.17323.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Farzin Haddadpour, Mohammad Mahdi Kamani, Mehrdad Mahdavi, and Amin Karbasi. Learning distributionally robust models at scale via composite optimization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=To-R742x7se.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.03300.

Hisham Husain. Distributional robustness with IPMs and links to regularization and GANs. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11816–11827, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/8929c70f8d710e412d38da624b21c3c8-Abstract.html.

Jinuk Kim, Marwa El Halabi, Wonpyo Park, Clemens J. S. Schaefer, Deokjae Lee, Yeonhong Park, Jae W. Lee, and Hyun Oh Song. GuidedQuant: Large language model quantization via exploiting end loss guidance. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30011–30037. PMLR, 2025. doi: 10.48550/arXiv.2505.07004. URL https://proceedings.mlr.press/v267/kim25d.html.

Tam Le and Jérôme Malick. Universal generalization guarantees for Wasserstein distributionally robust models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/35b5c175e139bff5f22a5361270fce87-Abstract-Conference.html.

Gunjun Lee, Sehwan Son, Younjoo Lee, Byungjun Kim, and Jung Ho Ahn. SchurQuant: Groupwise discrete optimization for layer-wise LLM quantization, 2026a. URL https://arxiv.org/abs/2608.15567.

Jung Hyun Lee, June Yong Yang, Jungwook Choi, and Eunho Yang. LFQ: Logit-aware final-block quantization for boosting the generation quality of low-bit quantized LLMs. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026b. URL https://openreview.net/forum?id=ykdc70h1ND.

Daniel Levy, Yair Carmon, John C. Duchi, and Aaron Sidford. Large-scale methods for distributionally robust optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 8847–8860, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/64986d86a17424eeac96b08a6d519059-Abstract.html.

Yuhang Li, Ruokai Yin, Donghyun Lee, Shiting Xiao, and Priyadarshini Panda. GPTAQ: Efficient finetuning-free quantization for asymmetric calibration. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 36690–36706. PMLR, 2025. URL https://proceedings.mlr.press/v267/li25dn.html.

Yesheng Liang, Haisheng Chen, Zihan Zhang, Song Han, and Zhijian Liu. ParoQuant: Pairwise rotation quantization for efficient reasoning LLM inference. In *International Conference on Learning Representations*, 2026. URL https://github.com/z-lab/paroquant.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. In *Proceedings of Machine Learning and Systems*, volume 6, pp. 87–100, 2024. URL https://proceedings.mlsys.org/paper_files/paper/2024/file/42a452cbafa9dd64e9ba4aa95cc1ef21-Paper-Conference.pdf.

Jiashuo Liu, Jiayun Wu, Bo Li, and Peng Cui. Distributionally robust optimization with data geometry. In *Advances in Neural Information Processing Systems*, volume 35, pp. 33689–33701, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/da535999561b932f56efdd559498282e-Abstract-Conference.html.

Jiashuo Liu, Jiayun Wu, Tianyu Wang, Hao Zou, Bo Li, and Peng Cui. Geometry-calibrated DRO: Combating over-pessimism with free energy implications. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32184–32200, 2024. URL https://proceedings.mlr.press/v235/liu24br.html.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations. In *The Thirteenth International Conference on Learning Representations*, 2025. doi: 10.48550/arXiv.2405.16406. URL https://openreview.net/forum?id=ogO6DGE6FZ.

Yongge Ma, Guoan Wang, Feiyu Wang, Yaoming Li, Qian Zhang, Zihan Yan, Yinjun Han, and Tong Yang. ReQuant: Fixed-grid discrete refinement for post-training quantization, 2026. URL https://arxiv.org/abs/2608.07019.

Ronak Mehta, Jelena Diakonikolas, and Zaid Harchaoui. Drago: Primal-dual coupled variance reduction for faster distributionally robust optimization. In *Advances in Neural Information Processing Systems*, volume 37, pp. 134770–134825, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/f2c5a0d8e445cc359eb446521f438ee3-Abstract-Conference.html.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Paul Michel, Tatsunori Hashimoto, and Graham Neubig. Distributionally robust models with parametric likelihood ratios. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=a34GrNaYEcS.

Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1–2):115–166, 2018. doi: 10.1007/s10107-017-1172-1. URL https://link.springer.com/article/10.1007/s10107-017-1172-1.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, et al. GPT-4 technical report, 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL https://jmlr.org/papers/v21/20-074.html.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale, 2019. URL https://arxiv.org/abs/1907.10641.

Amrith Setlur, Don Kurian Dennis, Benjamin Eysenbach, Aditi Raghunathan, Chelsea Finn, Virginia Smith, and Sergey Levine. Bitrate-constrained DRO: Beyond worst case robustness to unknown group shifts. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=2QzNuaRHn4Z.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=8Wuvhh0LYW.

Yuxuan Sun, Ruikang Liu, Haoli Bai, Han Bao, Kang Zhao, Yuening Li, Jiaxin Hu, Xianzhi Yu, Lu Hou, Chun Yuan, Xin Jiang, Wulong Liu, and Jun Yao. FlatQuant: Flatness matters for LLM quantization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 57587–57613. PMLR, 2025. URL https://proceedings.mlr.press/v267/sun25l.html.

Albert Tseng, Qingyao Sun, David Hou, and Christopher De Sa. QTIP: Quantization with trellises and incoherence processing. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/6de2e84b8da47bb2eb5e2ac96c63d2b0-Abstract-Conference.html.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 38087–38099, 2023. URL https://proceedings.mlr.press/v202/xiao23c.html.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Runtian Zhai, Chen Dan, Zico Kolter, and Pradeep Ravikumar. DORO: Distributional and outlier robust optimization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12345–12355, 2021. URL https://proceedings.mlr.press/v139/zhai21a.html.

Qi Zhang, Yi Zhou, Simon Khan, Ashley Prater-Bennette, Lixin Shen, and Shaofeng Zou. Revisiting large-scale non-convex distributionally robust optimization. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/bdbb5a9dd174d3f8230b33269a9ccf56-Abstract-Conference.html.

A NOTATION

Table 3: Summary of notation.

Symbol	Meaning
$\mathbf{W}, \widehat{\mathbf{W}}(\mathbf{q})$	Full-precision and quantized weights in $\mathbb{R}^{m\times d}$, after any scaling or rotation by the base PTQ method
$\mathbf{q}, \mathcal{Q}$	Integer codes representing quantized weights and the grid defined by fixed bit width, grouping, scales, and zero points
$\widehat{\mathbf{W}}_0$	Initial quantized weights produced by the base PTQ method, used to limit increases in reconstruction loss
$\mathbf{X}$	Calibration input activation matrix in $\mathbb{R}^{d\times n}$, with one token per column
$\mathbf{H}$	Activation second-moment matrix $\mathbf{X}\mathbf{X}^\top/n$
$\mathbf{X}_\mathrm{d}, \mathbf{X}_\mathrm{c}^{(h)}$	Input activations in the search and checking subsets, with n_d and $n_\mathrm{c}^{(h)}$ columns. DRQ computes $\mathbf{H}$ and $\mathcal{L}$ from the search subset
$\mathcal{L}(\widehat{\mathbf{W}})$	Layer-output reconstruction loss $\left\|\widehat{\mathbf{W}}\mathbf{X} - \mathbf{W}\mathbf{X}\right\|_F^2/n$
$\mathcal{J}(\widehat{\mathbf{W}})$	Fixed evaluation loss used in Theorem B.2, such as model token NLL after replacing one layer's weights
$e_{12}(\mathbf{C})$	Largest increase in the reconstruction loss of $\widehat{\mathbf{W}}_1$ minus that of $\widehat{\mathbf{W}}_2$ under $\mathbf{\Delta}\succeq 0$ and $\mathrm{tr}(\mathbf{C}\mathbf{\Delta})\le 1$
$\mathbf{\Delta}$	Change in the input activation second-moment matrix from $\mathbf{H}$ to $\mathbf{H}+\mathbf{\Delta}$, with $\mathbf{\Delta}\succeq 0$ and $\mathrm{tr}(\mathbf{C}\mathbf{\Delta})\le\alpha$
$\mathbf{C}$	Positive-definite matrix that weights input changes in the budget $\mathrm{tr}(\mathbf{C}\mathbf{\Delta})\le\alpha$. DRQ uses $(1-\beta)\mathbf{H}+\beta\mathbf{I}$
$\mathcal{S}_\mathbf{C}(\widehat{\mathbf{W}})$	Worst-case directional error $\max_{u^\top\mathbf{C}u=1}\left\|(\widehat{\mathbf{W}}-\mathbf{W})u\right\|_2^2$
α, β	Distribution-change budget (also the weight on worst-case directional error) and the weight on the identity matrix in $\mathbf{C}$
$\mathcal{L}_\mathrm{DRQ}(\widehat{\mathbf{W}})$	Worst-case reconstruction loss under $\mathbf{\Delta}\succeq 0$ and $\mathrm{tr}(\mathbf{C}\mathbf{\Delta})\le\alpha$, equal to $\mathcal{L}(\widehat{\mathbf{W}})+\alpha\mathcal{S}_\mathbf{C}(\widehat{\mathbf{W}})$

B THEORETICAL RESULTS AND PROOFS

This appendix supports the analysis and objective in Section 3. We first explain how a fixed quantization grid can make reconstruction loss on calibration data favor weights with higher loss on new inputs, and why more calibration data need not resolve this difference. We then prove the results for changes in the input activation distribution and explain the guarantees of the DRQ objective. Throughout, the full-precision weights and quantization grid remain fixed. The matrix $\mathbf{H}$ is the second-moment matrix of the calibration input activations, computed from samples or from their distribution as specified below.

B.1 THE ROLE OF THE QUANTIZATION GRID

Without a quantization constraint, retaining the full-precision weights gives zero reconstruction loss for any input distribution. Quantization usually removes this option. The following result makes this distinction explicit.

Proposition B.1 (Reconstruction when full-precision weights are representable)**.** *For every activation second-moment matrix $\mathbf{H}\succeq 0$, the unconstrained problem $\min_{\widehat{\mathbf{W}}\in\mathbb{R}^{m\times d}}\left\|(\widehat{\mathbf{W}}-\mathbf{W})\mathbf{H}^{1/2}\right\|_\mathrm{F}^2$ has $\mathbf{W}$ as a global minimizer. This minimizer is unique when $\mathbf{H}\succ 0$. If $\mathbf{W}\in\mathcal{Q}$, it also minimizes reconstruction loss over the quantization grid for every input distribution with a finite second moment. Thus two positive-definite second-moment matrices can select different unique minimizers only if the grid excludes $\mathbf{W}$.*

Proof of Proposition B.1. For any $\widehat{\mathbf{W}}$ and $\mathbf{H} \succeq 0$, $\left\|(\widehat{\mathbf{W}} - \mathbf{W})\mathbf{H}^{1/2}\right\|_{\mathrm{F}}^2 \geq 0$. Setting $\widehat{\mathbf{W}} = \mathbf{W}$ gives zero loss. If $\mathbf{H} \succ 0$, equality requires $\widehat{\mathbf{W}} = \mathbf{W}$, so the minimizer is unique. The same argument applies when optimization is restricted to a grid containing $\mathbf{W}$. □

Excluding $\mathbf{W}$ makes different choices possible, but does not force them. The fixed-grid example in Theorem B.3 shows that the weights minimizing reconstruction loss on calibration data can have larger loss on a new input.

B.2 Why more calibration data may not resolve the mismatch

More calibration data can estimate reconstruction loss more accurately. This does not necessarily improve the selected weights if minimizing the expected reconstruction loss already favors weights with a higher evaluation loss. The following result separates this mismatch between objectives from error due to a limited calibration sample.

Let $\mathcal{J}(\widehat{\mathbf{W}})$ be a fixed, finite evaluation loss, with smaller values indicating better performance. It may be reconstruction loss on new input activations or the model's token negative log-likelihood (NLL) after replacing one layer's weights while keeping all other weights fixed. The evaluation rule and distribution of evaluation data remain unchanged as the number of calibration samples increases.

Theorem B.2 (More calibration data need not improve the selected weights)**.** *Let $\mathcal{Q}$ be a fixed finite quantization grid, and let $x_1, x_2, \ldots$ be independent samples from the same calibration input activation distribution with $\mathbb{E}\,\|x\|_2^2 < \infty$. Suppose $\widehat{\mathbf{W}} \in \mathcal{Q}$ uniquely minimizes $\mathbb{E}\left\|(\widehat{\mathbf{W}} - \mathbf{W})x\right\|_2^2$ over this grid, while another matrix $\widehat{\mathbf{W}}' \in \mathcal{Q}$ satisfies $\mathcal{J}(\widehat{\mathbf{W}}') < \mathcal{J}(\widehat{\mathbf{W}})$. With probability one, every exact minimizer of reconstruction loss on the first n samples equals $\widehat{\mathbf{W}}$ for all sufficiently large n. Its evaluation loss therefore remains worse than that of $\widehat{\mathbf{W}}'$ by the fixed positive amount $\mathcal{J}(\widehat{\mathbf{W}}) - \mathcal{J}(\widehat{\mathbf{W}}')$.*

Proof of Theorem B.2. For each quantized weight matrix $\mathbf{V} \in \mathcal{Q}$, write the reconstruction loss on the first n calibration inputs as

$$\widehat{\mathcal{L}}_n(\mathbf{V}) = \frac{1}{n}\sum_{k=1}^{n} \|(\mathbf{V} - \mathbf{W})x_k\|_2^2 . \tag{7}$$

The finite-second-moment assumption ensures that this squared loss has a finite expectation. The strong law of large numbers gives $\widehat{\mathcal{L}}_n(\mathbf{V}) \to \mathbb{E}\,\|(\mathbf{V} - \mathbf{W})x\|_2^2$ almost surely. Because the grid is finite, the largest absolute difference between empirical and expected loss over all matrices in $\mathcal{Q}$ also converges to zero.

The unique minimizer $\widehat{\mathbf{W}}$ of expected reconstruction loss is separated from every other matrix by a positive gap

$$\gamma = \min_{\mathbf{V}\in\mathcal{Q},\, \mathbf{V}\neq\widehat{\mathbf{W}}} \left\{ \mathbb{E}\,\|(\mathbf{V} - \mathbf{W})x\|_2^2 - \mathbb{E}\left\|(\widehat{\mathbf{W}} - \mathbf{W})x\right\|_2^2 \right\} > 0. \tag{8}$$

With probability one, all empirical losses eventually lie within $\gamma/3$ of their expectations. Every other matrix then has empirical loss at least $\gamma/3$ larger than that of $\widehat{\mathbf{W}}$. The exact empirical minimizer therefore equals $\widehat{\mathbf{W}}$ for all sufficiently large n.

The selected weights then retain the evaluation-loss gap

$$\mathcal{J}(\widehat{\mathbf{W}}) - \mathcal{J}(\widehat{\mathbf{W}}') > 0, \tag{9}$$

which proves the claim. □

The theorem assumes that the expected reconstruction loss and evaluation loss favor different weights. It shows that more samples from the same distribution cannot remove this existing mismatch simply by estimating reconstruction loss more accurately. It does not require evaluation on a different dataset, since $\mathcal{J}$ can also measure model performance on the calibration distribution. Theorem B.3 gives a concrete instance with reconstruction loss on a new input as $\mathcal{J}$.

B.3 When input changes reverse the reconstruction loss comparison

We now compare two quantized weight matrices with $\mathcal{L}(\widehat{\mathbf{W}}_1) < \mathcal{L}(\widehat{\mathbf{W}}_2)$. As in Section 3.1, new input activations have second-moment matrix $\mathbf{H} + \mathbf{\Delta}$, where $\mathbf{\Delta} \succeq 0$ and $\mathrm{tr}(\mathbf{C}\mathbf{\Delta}) \leq \alpha$, with fixed $\mathbf{C} \succ 0$. The proof of Theorem 3.1 finds the largest possible increase in the loss of $\widehat{\mathbf{W}}_1$ relative to that of $\widehat{\mathbf{W}}_2$ under these constraints.

For the proof, we use two auxiliary quantities to express the difference between the reconstruction losses of $\widehat{\mathbf{W}}_1$ and $\widehat{\mathbf{W}}_2$,

$$\begin{aligned} \mathbf{B}_{12} &= (\widehat{\mathbf{W}}_1 - \mathbf{W})^\top(\widehat{\mathbf{W}}_1 - \mathbf{W}) - (\widehat{\mathbf{W}}_2 - \mathbf{W})^\top(\widehat{\mathbf{W}}_2 - \mathbf{W}), \\ e_{12}(\mathbf{C}) &= \left[\lambda_{\max}\left(\mathbf{C}^{-1/2}\mathbf{B}_{12}\mathbf{C}^{-1/2}\right)\right]_+ . \end{aligned} \tag{10}$$

For an input vector u, each squared output error measures the difference between a quantized layer's output and the full-precision output $\mathbf{W}u$. The quantity $u^\top \mathbf{B}_{12} u$ subtracts this error for $\widehat{\mathbf{W}}_2$ from that for $\widehat{\mathbf{W}}_1$. The shorthand $e_{12}(\mathbf{C})$ gives the largest increase in their reconstruction-loss difference under a distribution-change budget of one. Here $[a]_+ = \max\{a, 0\}$, since $\mathbf{\Delta} = 0$ satisfies the constraints and leaves the difference unchanged. We will show that the loss of $\widehat{\mathbf{W}}_1$ remains no larger than that of $\widehat{\mathbf{W}}_2$ exactly when $\mathcal{L}(\widehat{\mathbf{W}}_2) - \mathcal{L}(\widehat{\mathbf{W}}_1) \geq \alpha e_{12}(\mathbf{C})$.

Proof of Theorem 3.1. Under the second moment $\mathbf{H} + \mathbf{\Delta}$, the difference between the two weight matrices' reconstruction losses is

$$\begin{aligned} &\mathrm{tr}\Big[(\widehat{\mathbf{W}}_1 - \mathbf{W})(\mathbf{H} + \mathbf{\Delta})(\widehat{\mathbf{W}}_1 - \mathbf{W})^\top\Big] \\ &\quad - \mathrm{tr}\Big[(\widehat{\mathbf{W}}_2 - \mathbf{W})(\mathbf{H} + \mathbf{\Delta})(\widehat{\mathbf{W}}_2 - \mathbf{W})^\top\Big] \\ &\quad = \mathcal{L}(\widehat{\mathbf{W}}_1) - \mathcal{L}(\widehat{\mathbf{W}}_2) + \mathrm{tr}(\mathbf{B}_{12}\mathbf{\Delta}), \end{aligned} \tag{11}$$

The first term is the loss difference on calibration data. The second is the change in that difference caused by $\mathbf{\Delta}$.

Set $\mathbf{E} = \mathbf{C}^{1/2}\mathbf{\Delta}\mathbf{C}^{1/2}$ and $\mathbf{D} = \mathbf{C}^{-1/2}\mathbf{B}_{12}\mathbf{C}^{-1/2}$. The constraints on $\mathbf{\Delta}$ are equivalent to $\mathbf{E} \succeq 0$ and $\mathrm{tr}(\mathbf{E}) \leq \alpha$. Hence

$$\begin{aligned} \max_{\mathbf{\Delta} \succeq 0,\, \mathrm{tr}(\mathbf{C}\mathbf{\Delta}) \leq \alpha} \mathrm{tr}(\mathbf{B}_{12}\mathbf{\Delta}) &= \max_{\mathbf{E} \succeq 0,\, \mathrm{tr}(\mathbf{E}) \leq \alpha} \mathrm{tr}(\mathbf{D}\mathbf{E}) \\ &= \alpha[\lambda_{\max}(\mathbf{D})]_+ . \end{aligned} \tag{12}$$

The last equality follows by setting $\mathbf{E} = \alpha v v^\top$ for a unit leading eigenvector v when its eigenvalue is positive, and setting $\mathbf{E} = 0$ otherwise. The largest loss difference in Equation (11) is therefore $\mathcal{L}(\widehat{\mathbf{W}}_1) - \mathcal{L}(\widehat{\mathbf{W}}_2) + \alpha e_{12}(\mathbf{C})$. The loss of $\widehat{\mathbf{W}}_1$ remains no larger under every change satisfying the two constraints exactly when this expression is nonpositive. For $\mathbf{\Delta} = \alpha u u^\top$ with $u^\top \mathbf{C} u = 1$, the change in the loss difference is $\alpha u^\top \mathbf{B}_{12} u$. Requiring it to be at most $\mathcal{L}(\widehat{\mathbf{W}}_2) - \mathcal{L}(\widehat{\mathbf{W}}_1)$ for every such u gives Equation (3). Since $\mathcal{L}(\widehat{\mathbf{W}}_2) - \mathcal{L}(\widehat{\mathbf{W}}_1) > 0$ and a maximizing $\mathbf{\Delta}$ has rank at most one, the directional condition is equivalent to the condition over all matrices $\mathbf{\Delta}$ satisfying the constraints.

If the inequality fails, take a unit eigenvector v corresponding to the largest eigenvalue of $\mathbf{D}$ and set $\mathbf{\Delta}^* = \alpha \mathbf{C}^{-1/2} v v^\top \mathbf{C}^{-1/2}$. This rank-one change satisfies both constraints and makes $\widehat{\mathbf{W}}_1$ have higher expected reconstruction loss than $\widehat{\mathbf{W}}_2$ on the new inputs. □

When $e_{12}(\mathbf{C}) > 0$, $\big(\mathcal{L}(\widehat{\mathbf{W}}_2) - \mathcal{L}(\widehat{\mathbf{W}}_1)\big)/e_{12}(\mathbf{C})$ is the largest distribution-change budget for which $\widehat{\mathbf{W}}_1$ is guaranteed to have no larger reconstruction loss than $\widehat{\mathbf{W}}_2$. If $e_{12}(\mathbf{C}) = 0$, no positive semidefinite change in the second-moment matrix can give $\widehat{\mathbf{W}}_1$ higher reconstruction loss than $\widehat{\mathbf{W}}_2$. The result compares layer-output reconstruction losses, not model PPL.

B.4 A FIXED-GRID EXAMPLE OF HIGHER LOSS ON A NEW INPUT

The following example gives two quantized weight vectors with arbitrarily small reconstruction losses on calibration data. The vector with the smaller loss nevertheless has a larger squared output error on a new input. Both vectors use the same fixed scale, and the new input has unit length, so the difference does not arise from changing the scale or increasing the input magnitude.

Theorem B.3 (Lower reconstruction loss on calibration data need not generalize)**.** *Consider a linear layer with two inputs and one output, with reconstruction loss* $\mathcal{L}(\widehat{\mathbf{w}}) = (\widehat{\mathbf{w}} - \mathbf{w})\mathbf{H}(\widehat{\mathbf{w}} - \mathbf{w})^\top$. *Let* $\mathcal{Z}_b$ *be any finite set of integer codes containing* 0 *and* 1*, and fix the quantization scale* $s > 0$ *in* $\mathcal{Q}_b(s) = \{s(k_1, k_2) : k_1, k_2 \in \mathcal{Z}_b\}$*. For every* $\varepsilon > 0$*, there exist a full-precision weight vector* $\mathbf{w} \notin \mathcal{Q}_b(s)$*, a calibration second-moment matrix* $\mathbf{H} \succ 0$ *with* $\mathrm{tr}(\mathbf{H}) = 1$*, and two quantized weight vectors* $\widehat{\mathbf{w}}_{\mathrm{rec}}, \widehat{\mathbf{w}}_{\mathrm{alt}} \in \mathcal{Q}_b(s)$ *such that* $\widehat{\mathbf{w}}_{\mathrm{rec}}$ *uniquely minimizes reconstruction loss over the grid and* $0 < \mathcal{L}(\widehat{\mathbf{w}}_{\mathrm{rec}}) < \mathcal{L}(\widehat{\mathbf{w}}_{\mathrm{alt}}) \le \varepsilon$*. On the unit-length input* $\mathbf{x}_* = (0, 1)^\top$*, their squared output errors are* s^2 *and* 0*, respectively. The full-precision weight vector* $\mathbf{w}$ *is the unique unconstrained minimizer of the same reconstruction loss.*

Corollary B.4 (Reconstruction-only refinement can increase loss on a new input)**.** *In the example of Theorem B.3, take* $\widehat{\mathbf{w}}_{\mathrm{alt}}$ *as the initial quantized weights. Exactly minimizing reconstruction loss over the same grid returns* $\widehat{\mathbf{w}}_{\mathrm{rec}}$*. This reduces the loss on calibration data but increases the squared output error on* $\mathbf{x}_*$ *by exactly* s^2*.*

Proof of Theorem B.3. Fix a finite set of integer codes $\mathcal{Z}_b$ containing 0 and 1. Choose an irrational $\delta \in (0, 1)$ such that

$$k_1 + \delta k_2 \neq 0 \quad \text{for all } (k_1, k_2) \in \mathcal{Z}_b^2 \setminus \{(0, 0)\}. \tag{13}$$

An irrational δ ensures that the expression above is nonzero for every nonzero integer pair in $\mathcal{Z}_b^2$.

Define $\mathbf{w} = (-\delta s, s)$, $\mathbf{x} = (1, \delta)^\top$, and $\widetilde{\mathbf{H}}_\eta = \mathbf{x}\mathbf{x}^\top + \eta \mathbf{I}_2$. Let $Z_\eta = 1 + \delta^2 + 2\eta$ and $\mathbf{H}_\eta = \widetilde{\mathbf{H}}_\eta / Z_\eta$, so $\mathrm{tr}(\mathbf{H}_\eta) = 1$. We first use the unnormalized second-moment matrix $\widetilde{\mathbf{H}}_\eta$, since dividing by the positive scalar Z_η preserves the loss comparison between all weight vectors in the grid. Because $\mathbf{w}\mathbf{x} = 0$, the reconstruction loss of $\widehat{\mathbf{w}} = s(k_1, k_2)$ becomes

$$L_\eta(\widehat{\mathbf{w}}) = s^2(k_1 + \delta k_2)^2 + \eta \left\| \widehat{\mathbf{w}} - \mathbf{w} \right\|_2^2. \tag{14}$$

At $\eta = 0$, Equation (13) makes $\widehat{\mathbf{w}}_{\mathrm{rec}} = (0, 0)$ the only quantized weight vector with zero loss. The finite grid gives a positive reconstruction-loss gap

$$r = \min_{\widehat{\mathbf{w}} \in \mathcal{Q}_b(s),\, \widehat{\mathbf{w}} \neq \widehat{\mathbf{w}}_{\mathrm{rec}}} L_0(\widehat{\mathbf{w}}) > 0. \tag{15}$$

Let $D = \max_{\widehat{\mathbf{w}} \in \mathcal{Q}_b(s),\, \widehat{\mathbf{w}} \neq \widehat{\mathbf{w}}_{\mathrm{rec}}} \left| \left\| \widehat{\mathbf{w}} - \mathbf{w} \right\|_2^2 - \left\| \mathbf{w} \right\|_2^2 \right|$. Set $\eta_0 = 1$ if $D = 0$ and $\eta_0 = r/(2D)$ otherwise. For every nonzero quantized weight vector and $0 < \eta < \eta_0$,

$$L_\eta(\widehat{\mathbf{w}}) - L_\eta(\widehat{\mathbf{w}}_{\mathrm{rec}}) = L_0(\widehat{\mathbf{w}}) + \eta \left(\left\| \widehat{\mathbf{w}} - \mathbf{w} \right\|_2^2 - \left\| \mathbf{w} \right\|_2^2 \right) \tag{16}$$

$$\ge r - \eta D > r/2 > 0. \tag{17}$$

Thus $\widehat{\mathbf{w}}_{\mathrm{rec}}$ remains the unique minimizer of reconstruction loss over the grid when the second-moment matrix is positive definite.

Take $\widehat{\mathbf{w}}_{\mathrm{alt}} = (0, s)$. The two weight vectors differ from $\mathbf{w}$ by $(\delta s, -s)$ and $(\delta s, 0)$, respectively. Substitution gives

$$L_\eta(\widehat{\mathbf{w}}_{\mathrm{rec}}) = \eta s^2 (1 + \delta^2), \qquad L_\eta(\widehat{\mathbf{w}}_{\mathrm{alt}}) = \delta^2 s^2 (1 + \eta), \tag{18}$$

whose difference is $s^2(\delta^2 - \eta)$. The reconstruction minimizer has strictly smaller loss on calibration data whenever $\eta < \delta^2$.

Given $\varepsilon > 0$, choose δ satisfying Equation (13) and $\delta^2 s^2 \le \varepsilon/4$. Then choose

$$0 < \eta < \min \left\{ \eta_0, \delta^2/2, 1, \frac{\varepsilon}{4s^2(1 + \delta^2)} \right\}. \tag{19}$$

These choices imply $0 < L_\eta(\widehat{\mathbf{w}}_{\text{rec}}) < L_\eta(\widehat{\mathbf{w}}_{\text{alt}}) \le \varepsilon$ for the unnormalized second moment. Normalizing its trace to one reduces both losses by the common factor Z_η^{-1} and preserves the strict ordering. For the unit input vector $\mathbf{x}_* = (0, 1)^\top$,

$$|(\widehat{\mathbf{w}}_{\text{rec}} - \mathbf{w})\mathbf{x}_*|^2 = s^2, \qquad |(\widehat{\mathbf{w}}_{\text{alt}} - \mathbf{w})\mathbf{x}_*|^2 = 0. \tag{20}$$

Finally, $\mathbf{H}_\eta \succ 0$ because $\eta > 0$. The unconstrained loss is zero only at $\widehat{\mathbf{w}} = \mathbf{w}$, which the grid cannot represent because $-\delta$ is not an integer. This second moment can also be realized by input activations. For example, a distribution that assigns probability $1/2$ to each column of $\mathbf{X} = \sqrt{2}\,\mathbf{H}_\eta^{1/2}$ has second moment $\mathbf{H}_\eta$. This completes the example with a fixed scale, a trace-one second moment, and a unit-length new input. □

Proof of Corollary B.4. By Theorem B.3, $\widehat{\mathbf{w}}_{\text{rec}}$ uniquely minimizes reconstruction loss over the entire grid and has strictly lower loss on calibration data than $\widehat{\mathbf{w}}_{\text{alt}}$. Exact minimization must therefore return $\widehat{\mathbf{w}}_{\text{rec}}$. The squared output errors of these two vectors on $\mathbf{x}_*$ are s^2 and zero, respectively. □

Sampling independently from the two-point distribution above satisfies the assumptions of Theorem B.2. Taking $\mathcal{J}(\widehat{\mathbf{w}}) = |(\widehat{\mathbf{w}} - \mathbf{w})\mathbf{x}_*|^2$, the theorem gives an evaluation-loss gap of s^2 for the exact empirical minimizer for all sufficiently large sample sizes, with probability one. This is a counterexample for linear-layer reconstruction, rather than a bound on model PPL. It permits a general change in input distribution, whereas the following DRQ result uses the positive semidefinite changes specified in Section 3.1.

B.5 Worst-case reconstruction loss

DRQ evaluates each quantized weight matrix by its largest expected reconstruction loss over input activation distributions whose second-moment matrices are $\mathbf{H} + \mathbf{\Delta}$, with $\mathbf{\Delta} \succeq 0$ and $\operatorname{tr}(\mathbf{C}\mathbf{\Delta}) \le \alpha$. We compute $\mathbf{H}$ from the search subset and keep both $\mathbf{H}$ and $\mathbf{C}$ fixed during refinement. Theorem 3.2 shows that a change along one input direction attains the largest loss increase. Finding this direction reduces to an eigenvalue problem, as shown below.

Proof of Theorem 3.2. Set $\mathbf{E} = \mathbf{C}^{1/2}\mathbf{\Delta}\mathbf{C}^{1/2}$ and

$$\mathbf{B} = \mathbf{C}^{-1/2}(\widehat{\mathbf{W}} - \mathbf{W})^\top(\widehat{\mathbf{W}} - \mathbf{W})\mathbf{C}^{-1/2}.$$

This change of variables is invertible, and the constraints become $\mathbf{E} \succeq 0$ and $\operatorname{tr}(\mathbf{E}) \le \alpha$. The expected reconstruction loss under the second-moment matrix $\mathbf{H} + \mathbf{\Delta}$ is

$$\operatorname{tr}\left[(\widehat{\mathbf{W}} - \mathbf{W})(\mathbf{H} + \mathbf{\Delta})(\widehat{\mathbf{W}} - \mathbf{W})^\top\right] = \mathcal{L}(\widehat{\mathbf{W}}) + \operatorname{tr}(\mathbf{B}\mathbf{E}). \tag{21}$$

Since both matrices are positive semidefinite, $\operatorname{tr}(\mathbf{B}\mathbf{E}) \le \lambda_{\max}(\mathbf{B})\operatorname{tr}(\mathbf{E}) \le \alpha\lambda_{\max}(\mathbf{B})$.

Let v be a unit eigenvector for the largest eigenvalue of $\mathbf{B}$, and choose $\mathbf{E}^* = \alpha vv^\top$. Then $\operatorname{tr}(\mathbf{E}^*) = \alpha$ and the upper bound is attained. The resulting change in the second-moment matrix

$$\mathbf{\Delta}^* = \alpha\mathbf{C}^{-1/2}vv^\top\mathbf{C}^{-1/2} \tag{22}$$

satisfies both constraints and uses the full distribution-change budget α. Finally, $\lambda_{\max}(\mathbf{B}) = \mathcal{S}_{\mathbf{C}}(\widehat{\mathbf{W}})$ by setting $u = \mathbf{C}^{-1/2}v$, which gives $u^\top\mathbf{C}u = 1$ and $(\widehat{\mathbf{W}} - \mathbf{W})^\top(\widehat{\mathbf{W}} - \mathbf{W})u = \lambda_{\max}(\mathbf{B})\mathbf{C}u$. This proves Theorem 3.2. □

If $\alpha = 0$ or $\widehat{\mathbf{W}} = \mathbf{W}$, $\mathbf{\Delta} = 0$ attains the maximum. Otherwise the construction above has rank one and can be realized by adding an independent perturbation that equals $\sqrt{\alpha}\,u$ or $-\sqrt{\alpha}\,u$, each with probability $1/2$, to the calibration input activations. Its mean is zero and its second moment is $\alpha uu^\top$, so the resulting distribution has second moment $\mathbf{H} + \mathbf{\Delta}^*$.

The maximizing direction depends on both the weight difference $\widehat{\mathbf{W}} - \mathbf{W}$ and $\mathbf{C}$. Changing integer codes can therefore change the worst-case direction, which is why DRQ estimates it again before accepting proposed weights.

B.6 Comparing refined and initial quantized weights

Two weight matrices can attain their worst-case losses under different input distributions. A smaller worst-case loss therefore does not imply smaller loss under every distribution in the set. To check this stronger property, we compare both matrices under the same $\mathbf{\Delta}$. Write $\widehat{\mathbf{W}}_0$ for the initial quantized weights. Using $e_{12}(\mathbf{C})$ from Equation (10), define

$$\Gamma_\alpha(\widehat{\mathbf{W}}_1, \widehat{\mathbf{W}}_2) = \mathcal{L}(\widehat{\mathbf{W}}_1) - \mathcal{L}(\widehat{\mathbf{W}}_2) + \alpha e_{12}(\mathbf{C}). \tag{23}$$

Proposition B.5 (Comparing reconstruction losses under the same input distribution)**.** *For any $\widehat{\mathbf{W}}_1, \widehat{\mathbf{W}}_2 \in \mathbb{R}^{m\times d}$, $\mathbf{C} \succ 0$, and $\alpha \geq 0$, the quantity in Equation (23) satisfies*

$$\begin{aligned}\Gamma_\alpha(\widehat{\mathbf{W}}_1, \widehat{\mathbf{W}}_2) &= \max_{\mathbf{\Delta}\succeq 0,\, \mathrm{tr}(\mathbf{C}\mathbf{\Delta})\leq\alpha} \mathrm{tr}(\mathbf{B}_{12}(\mathbf{H}+\mathbf{\Delta})) \\ &\leq \mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}_1) - \mathcal{L}(\widehat{\mathbf{W}}_2).\end{aligned} \tag{24}$$

Consequently, $\Gamma_\alpha(\widehat{\mathbf{W}}_1, \widehat{\mathbf{W}}_0) < 0$ certifies that $\widehat{\mathbf{W}}_1$ has strictly smaller reconstruction loss than $\widehat{\mathbf{W}}_0$ for every second moment satisfying these constraints.

Proof. The equality follows from Equation (12). Since $\mathbf{\Delta} \succeq 0$, the reconstruction loss of $\widehat{\mathbf{W}}_2$ is at least $\mathcal{L}(\widehat{\mathbf{W}}_2)$, while that of $\widehat{\mathbf{W}}_1$ is at most $\mathcal{L}_{\mathrm{DRQ}}(\widehat{\mathbf{W}}_1)$. Subtracting these bounds and maximizing proves Equation (24). A negative maximum makes the pairwise difference negative for every second moment satisfying the constraints. □

The guarantee uses the exact value of Γ_α. DRQ estimates this quantity for reporting when its selected weights differ from the initial weights. A negative estimate alone does not establish the guarantee, since a Rayleigh quotient can underestimate the largest eigenvalue. The acceptance rule uses the reconstruction tolerances and a decrease in the estimated worst-case reconstruction loss, as described in Appendix C.

B.7 Equivalent optimization over input directions

For a matrix, $\|\cdot\|_2$ denotes its largest singular value. The identity $\left\|(\widehat{\mathbf{W}} - \mathbf{W})\mathbf{C}^{-1/2}\right\|_2^2 = \max_{u^\top \mathbf{C} u=1} \left\|(\widehat{\mathbf{W}} - \mathbf{W})u\right\|_2^2$ gives the directional penalty in Equation (5). Introducing a scalar t that bounds this error gives an equivalent form of the minimization in Equation (4),

$$\begin{aligned}\min_{\widehat{\mathbf{W}}\in\mathcal{Q}, t} \quad & \mathcal{L}(\widehat{\mathbf{W}}) + \alpha t \\ \text{s.t.} \quad & \left\|(\widehat{\mathbf{W}} - \mathbf{W})u\right\|_2^2 \leq t, \quad \forall u : u^\top \mathbf{C} u = 1.\end{aligned} \tag{25}$$

For fixed weights and $\alpha > 0$, minimizing over t sets it to the largest directional error. At $\alpha = 0$, this value of t is still feasible and optimal, although larger values are also optimal. The formulation therefore preserves the minimum and the minimizing weight matrices for every $\alpha \geq 0$. It is called semi-infinite because it has finitely many optimization variables but one constraint for each of infinitely many input directions.

B.8 A reconstruction bound using Wasserstein distance

Another way to restrict a change in the input activation distribution is to pair original and new activation vectors and bound their average squared difference. Using the weighted 2-Wasserstein distance to make this comparison gives a reconstruction-loss bound involving the same worst-case directional error as DRQ.

Let $\widehat{P}_n$ assign probability $1/n$ to each calibration input vector used for code search, so its second moment is $\mathbf{H}$. For a distribution Q with a finite second moment, define the weighted 2-Wasserstein distance by

$$\mathsf{W}^2_{2,\mathbf{C}}(Q, \widehat{P}_n) = \inf_{\pi\in\Pi(\widehat{P}_n, Q)} \mathbb{E}_{(x,x')\sim\pi} \left\|\mathbf{C}^{1/2}(x' - x)\right\|_2^2.$$

Each joint distribution in $\Pi(\widehat{P}_n, Q)$ pairs an original activation vector x with a new vector x', while preserving their distributions $\widehat{P}_n$ and Q. The squared distance is the smallest average value of $\left\|\mathbf{C}^{1/2}(x'-x)\right\|_2^2$ over these pairings. A change $h = x' - x$ contributes $h^\top \mathbf{C} h$ to this average. For $\varepsilon \geq 0$ and $\zeta > 0$,

$$\begin{aligned} &\sup_{Q:\mathsf{W}_{2,\mathbf{C}}(Q,\widehat{P}_n)\leq\varepsilon} \mathbb{E}_Q \left\|(\widehat{\mathbf{W}} - \mathbf{W})x\right\|_2^2 \\ &\qquad \leq (1+\zeta)\left[\mathcal{L}(\widehat{\mathbf{W}}) + \frac{\varepsilon^2}{\zeta}\left\|(\widehat{\mathbf{W}} - \mathbf{W})\mathbf{C}^{-1/2}\right\|_2^2\right]. \end{aligned} \tag{26}$$

The bound combines reconstruction loss on calibration data with the same worst-case directional error used by DRQ. It is an upper bound for distributions within a Wasserstein distance, whereas Theorem 3.2 gives an equality for DRQ's second-moment constraints. These are two different sets of distributions.

Proof of Equation (26). Fix Q with $\mathsf{W}_{2,\mathbf{C}}(Q, \widehat{P}_n) \leq \varepsilon$ and let (x, x') be drawn from a joint distribution $\pi \in \Pi(\widehat{P}_n, Q)$. Minkowski's inequality in L_2 gives

$$\begin{aligned} &\left(\mathbb{E}_Q \left\|(\widehat{\mathbf{W}} - \mathbf{W})x'\right\|_2^2\right)^{1/2} \\ &\qquad \leq \left(\mathbb{E}_{\widehat{P}_n} \left\|(\widehat{\mathbf{W}} - \mathbf{W})x\right\|_2^2\right)^{1/2} + \left(\mathbb{E}_\pi \left\|(\widehat{\mathbf{W}} - \mathbf{W})(x'-x)\right\|_2^2\right)^{1/2}. \end{aligned} \tag{27}$$

For every displacement h, $\left\|(\widehat{\mathbf{W}} - \mathbf{W})h\right\|_2 \leq \left\|(\widehat{\mathbf{W}} - \mathbf{W})\mathbf{C}^{-1/2}\right\|_2 \left\|\mathbf{C}^{1/2}h\right\|_2$. Taking the infimum over joint distributions π bounds the second term by $\varepsilon \left\|(\widehat{\mathbf{W}} - \mathbf{W})\mathbf{C}^{-1/2}\right\|_2$. The first term is $\mathcal{L}(\widehat{\mathbf{W}})^{1/2}$. Applying $(a+b)^2 \leq (1+\zeta)a^2 + (1+\zeta^{-1})b^2$ and taking the supremum over distributions Q within the specified Wasserstein distance proves Equation (26). □

C Algorithmic Details

Blockwise integration. DRQ follows Algorithm 1, refining each linear layer after the base PTQ method quantizes it. The bit width, grouping, scales, and zero points remain fixed. DRQ uses cached calibration activations and the corresponding full-precision weights after any scaling or rotation by the base method. In GPTQ, refined layers supply inputs for subsequent layer groups within the block. In AWQ, refinement follows the block's scaling and clipping. The refined block then produces calibration inputs for the next block.

Calibration subsets. We split the selected activation vectors into a search subset $\mathbf{X}_\mathrm{d}$ and two separate checking subsets $\mathbf{X}_\mathrm{c}^{(1)}$ and $\mathbf{X}_\mathrm{c}^{(2)}$. The search subset determines $\mathbf{H}$, $\mathbf{C}$, and reconstruction loss used to score code changes. These subsets and matrices stay fixed during refinement. Appendix D.3 describes the adjustments for experts receiving few calibration tokens.

Proposing code changes. Each round estimates the current weights' worst-case input direction u. Holding u fixed, we score code replacements by their change in $\mathcal{L}(\widehat{\mathbf{W}}) + \alpha \left\|(\widehat{\mathbf{W}} - \mathbf{W})u\right\|_2^2$. In each row, we retain the eight lowest-scoring single-code replacements and test pairs that change two different weights. We keep the single replacement or pair with the largest decrease in this objective, provided the decrease is positive. We then rank these row updates by their decreases and apply different fractions of the list to the current weights, starting with the largest decreases.

Accepting new weights. For each proposed matrix, we estimate its worst-case direction again. Its estimated $\mathcal{L}_\mathrm{DRQ}$ must decrease relative to the current weights. On the search subset and each checking subset, reconstruction loss must also stay within the specified percentage above that of the initial PTQ weights $\widehat{\mathbf{W}}_0$. Proposals that fail these conditions or the numerical checks are rejected.

We continue with the passing matrix having the lowest estimated $\mathcal{L}_{\mathrm{DRQ}}$, or stop if none passes. We return the initial or accepted weights with the lowest estimated loss.

For the dense-model and six-method comparisons, we select α and β using 64 validation samples from Pile (Gao et al., 2020). The settings below use the standard $128 \times 2{,}048$ calibration setup. Other comparisons retain their stated settings.

Parameter	Setting
Activation vectors per layer	512 for search; 2×256 for checking
Distribution-change budget α	$\{0.1, 0.3, 0.5, 1\}$
Identity weight β in $\mathbf{C}$	$\{0.005, 0.1, 0.25, 0.75, 1\}$
Reconstruction-loss increase limit	3% for GPTQ; 1% for other base methods
Single-code replacements retained per row	8
Fractions of ranked row updates tested	$\{0, 1/32, 1/16, 1/8, 1/4, 1/2, 1\}$
Maximum refinement rounds	3

D More Experimental Results

D.1 Detailed dense Llama results

Tables 4 to 7 report accuracy on each of the seven downstream tasks before and after DRQ refinement. Average task accuracy improves in every reported comparison with the initial quantized model. The size of the improvement at lower bit widths depends on the base PTQ method. For the 70B model quantized with GPTQ at W2, DRQ improves all seven task scores and raises average accuracy by 6.33 percentage points.

Table 4: Llama-3.2-1B-Instruct accuracy (%) on each downstream task.

Bits	Method	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
FP	Unquantized	74.16	60.76	45.86	69.42	38.05	63.26	59.27	58.68
W2	GPTQ	51.52	26.47	24.19	42.54	24.66	25.63	48.22	34.75
	+ DRQ	**51.80**	**26.79**	**26.41**	**43.15**	**25.77**	**26.22**	**51.38**	**35.93**
	AWQ	51.74	25.99	23.66	47.65	25.68	23.53	49.49	35.39
	+ DRQ	51.31	**26.93**	**24.86**	**51.25**	25.51	**24.20**	**51.54**	**36.51**
W3	GPTQ	67.90	52.69	28.32	62.26	32.68	53.16	56.51	50.50
	+ DRQ	67.79	**53.60**	**33.75**	**62.94**	**34.22**	**53.75**	55.72	**51.68**
	AWQ	68.44	54.08	35.59	66.42	32.17	55.93	55.88	52.65
	+ DRQ	**68.88**	**54.13**	**36.25**	**66.79**	**34.13**	54.80	**56.59**	**53.08**
W4	GPTQ	73.72	58.99	43.95	67.95	36.86	61.95	58.88	57.47
	+ DRQ	**73.78**	**59.36**	**44.28**	**68.10**	**37.37**	**63.22**	**60.77**	**58.13**
	AWQ	72.63	58.74	43.41	68.72	36.95	61.99	58.48	57.27
	+ DRQ	71.87	**59.02**	43.09	68.59	**38.23**	**62.33**	**58.88**	**57.43**

Table 5: Llama-3.2-3B-Instruct accuracy (%) on each downstream task.

Bits	Method	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
FP	Unquantized	75.68	70.47	60.45	77.74	45.56	67.76	67.72	66.48
W2	GPTQ	51.69	27.74	23.86	40.24	25.09	26.01	47.43	34.58
	+ DRQ	50.60	**29.11**	**24.83**	**40.43**	25.00	25.34	**49.88**	**35.03**
	AWQ	51.31	26.76	26.89	42.66	25.26	26.09	48.70	35.38
	+ DRQ	**51.36**	**28.44**	26.89	**60.67**	24.74	**26.30**	**50.28**	**38.38**
W3	GPTQ	69.15	61.76	50.04	63.61	33.53	52.65	60.06	55.83
	+ DRQ	**71.33**	**65.25**	49.25	**73.33**	**38.57**	**59.60**	**64.33**	**60.24**
	AWQ	72.96	65.02	51.59	73.18	39.85	64.73	64.48	61.69
	+ DRQ	**73.23**	**65.05**	**52.37**	72.72	**41.89**	**66.67**	**65.82**	**62.54**
W4	GPTQ	75.14	69.93	58.26	76.15	44.80	66.92	65.67	65.26
	+ DRQ	**75.57**	69.89	**59.22**	**76.42**	**45.39**	**69.15**	**66.06**	**65.96**
	AWQ	75.03	69.70	59.09	78.72	46.08	67.72	66.93	66.18
	+ DRQ	**75.57**	**69.85**	59.03	**79.14**	**46.16**	**69.19**	**67.96**	**66.70**

Table 6: Llama-3.1-8B-Instruct accuracy (%) on each downstream task.

Bits	Method	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
FP	Unquantized	81.12	79.25	68.05	83.94	55.03	79.59	73.80	74.40
W2	GPTQ	49.95	27.67	24.11	43.03	26.37	27.53	49.80	35.49
	+ DRQ	**51.80**	**29.11**	**24.60**	**47.43**	25.85	26.56	**51.14**	**36.64**
	AWQ	51.03	26.60	26.89	61.93	24.40	25.00	48.78	37.80
	+ DRQ	51.03	**26.67**	26.89	60.28	**24.74**	**25.21**	**51.22**	**38.01**
W3	GPTQ	76.93	75.40	57.01	81.53	45.48	71.25	71.59	68.46
	+ DRQ	**78.02**	74.90	**58.43**	81.19	**47.95**	**74.66**	71.19	**69.48**
	AWQ	78.13	74.53	59.54	81.28	49.91	73.65	72.22	69.90
	+ DRQ	**78.24**	74.44	**59.97**	**81.62**	49.74	**74.03**	**72.69**	**70.10**
W4	GPTQ	79.98	78.39	66.39	82.81	52.82	77.65	73.72	73.11
	+ DRQ	**80.20**	**78.40**	**66.96**	**83.36**	**55.20**	**79.21**	**74.66**	**74.00**
	AWQ	80.90	78.68	65.71	83.30	53.67	79.38	72.77	73.49
	+ DRQ	80.85	78.57	**66.02**	**83.46**	53.67	**79.63**	**73.16**	**73.62**

Table 7: Llama-3.1-70B-Instruct accuracy (%) on each downstream task.

Bits	Method	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
FP	Unquantized	83.73	84.63	82.29	88.01	63.23	83.67	78.93	80.64
W2	GPTQ	63.71	60.43	29.01	58.50	28.67	40.61	56.67	48.23
	+ DRQ	**67.08**	**64.81**	**35.14**	**68.56**	**34.22**	**51.64**	**60.46**	**54.56**
	AWQ	51.52	26.66	24.65	62.17	26.71	24.75	49.49	37.99
	+ DRQ	51.31	26.61	24.65	62.17	26.54	**25.04**	**49.72**	**38.01**
W3	GPTQ	82.70	82.90	79.86	86.57	59.30	81.78	77.58	78.67
	+ DRQ	81.50	**82.99**	79.85	86.54	**61.60**	**83.08**	**77.66**	**79.03**
	AWQ	82.32	82.70	79.84	87.09	60.58	82.87	77.74	79.02
	+ DRQ	**82.86**	82.63	79.60	**87.28**	**61.43**	82.79	**77.98**	**79.23**
W4	GPTQ	83.57	84.21	81.56	87.89	62.80	83.08	78.45	80.22
	+ DRQ	**83.73**	**84.39**	**81.80**	87.74	62.80	**83.46**	**79.95**	**80.55**
	AWQ	83.24	84.26	81.64	87.55	63.31	83.67	79.01	80.38
	+ DRQ	**83.79**	**84.29**	81.43	**87.71**	62.97	**83.92**	**79.56**	**80.52**

D.2 Results across PTQ methods

Table 8 reports WT2 and C4 perplexity and accuracy on each of the seven downstream tasks for the initial and DRQ-refined Llama-3.2-3B-Instruct W3A16 models in Figure 3. The OmniQuant runs use learned weight clipping without learned equivalent transformations. Across the six selected comparisons, DRQ lowers both perplexities and raises average task accuracy. For GPTQ and AWQ, accuracy improves on six of the seven tasks. For RTN, gains on PIQA, BoolQ, and ARC-Easy outweigh declines on the other four tasks.

Table 8: Perplexity and seven-task accuracy (%) for Llama-3.2-3B W3.

	Perplexity		Accuracy (%)							
Method	WT2	C4	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
RTN	17.97	26.96	69.42	62.41	45.21	72.39	37.12	56.78	62.51	57.98
+ DRQ	**17.60**	**26.12**	**70.18**	62.03	45.20	**73.58**	36.95	**57.95**	62.04	**58.28**
GPTQ	15.74	30.34	69.15	61.76	50.04	63.61	33.53	52.65	60.06	55.83
+ DRQ	**14.63**	**22.11**	**71.33**	**65.25**	49.25	**73.33**	**38.57**	**59.60**	**64.33**	**60.24**
AWQ	15.19	20.80	72.96	65.02	51.59	73.18	39.85	64.73	64.48	61.69
+ DRQ	**14.90**	**20.52**	**73.23**	**65.05**	**52.37**	72.72	**41.89**	**66.67**	**65.82**	**62.54**
OmniQuant	14.52	21.23	71.60	63.99	50.28	73.94	37.20	62.21	61.17	60.06
+ DRQ	**14.44**	**21.12**	**71.87**	**64.01**	**50.41**	**74.22**	37.12	**62.84**	60.93	**60.20**
GPTAQ	14.94	20.97	73.78	64.57	46.80	74.10	38.48	62.75	65.04	60.79
+ DRQ	**14.44**	**20.72**	**74.37**	64.49	**47.08**	73.70	**38.74**	**62.88**	65.04	**60.90**
ParoQuant	13.56	20.31	74.10	67.24	54.76	74.50	40.61	64.23	65.35	62.97
+ DRQ	**13.38**	**19.94**	**75.03**	**67.78**	54.54	**75.38**	40.61	**64.73**	**66.06**	**63.45**

QuaRot with GPTQ. We further evaluate DRQ on Llama-3.1-8B-Instruct with QuaRot's Hadamard rotations followed by GPTQ (Ashkboos et al., 2024; Frantar et al., 2023). DRQ refines the integer codes of the quantized weights in the rotated representation while retaining the rotations and quantization parameters. We test W2A16 and W3A16 with a group size of 128. Each calibration set contains 128 WT2 sequences of 2,048 tokens. The reconstruction tolerances on the search and checking subsets are both 0.01. We use $(\alpha, \beta) = (1, 0.005)$ for W2 and $(0.1, 0.005)$ for W3. As is shown in Table 9, at W3, DRQ improves accuracy on all seven tasks, raising mean accuracy from 68.44% to 69.72% while lowering WT2 and C4 PPL. This shows that DRQ can further improve a model quantized with QuaRot and GPTQ.

Table 9: QuaRot + GPTQ with DRQ on Llama-3.1-8B-Instruct.

Bits	Method	Perplexity		Accuracy (%)							
		WT2	C4	PIQA	HellaSwag	MMLU	BoolQ	ARC-C	ARC-E	WinoGrande	Mean
W2	QuaRot + GPTQ	72.62	164.40	53.05	30.91	24.11	44.59	22.95	29.46	49.57	36.38
	+ DRQ	**67.53**	**159.03**	**53.32**	**31.06**	23.33	**46.67**	**24.23**	**29.76**	49.41	**36.82**
W3	QuaRot + GPTQ	8.58	14.86	78.18	74.07	58.60	81.44	46.67	69.95	70.17	68.44
	+ DRQ	**8.24**	**14.23**	**78.35**	**75.41**	**60.18**	**82.63**	**48.81**	**71.25**	**71.43**	**69.72**

D.3 Calibration tokens and expert refinement

We examine how the number of calibration tokens reaching each expert affects DRQ refinement in the Qwen3-30B-A3B run reported in Table 2. We quantize 18,432 expert linear layers and 192 attention linear layers across 48 Transformer blocks, leaving the routers unquantized. Each expert contains three linear layers.

Routing distributes calibration tokens unevenly. Figure 6(a) shows that 4,794 of 18,624 linear layers (25.74%) receive fewer than the 1,024 input activation vectors used for the full search and checking subsets. These include 399 linear layers with one or two routed tokens and 201 with none, all belonging to experts. The router sends different numbers of calibration tokens to different experts. Some experts therefore receive too few tokens to form separate checking subsets.

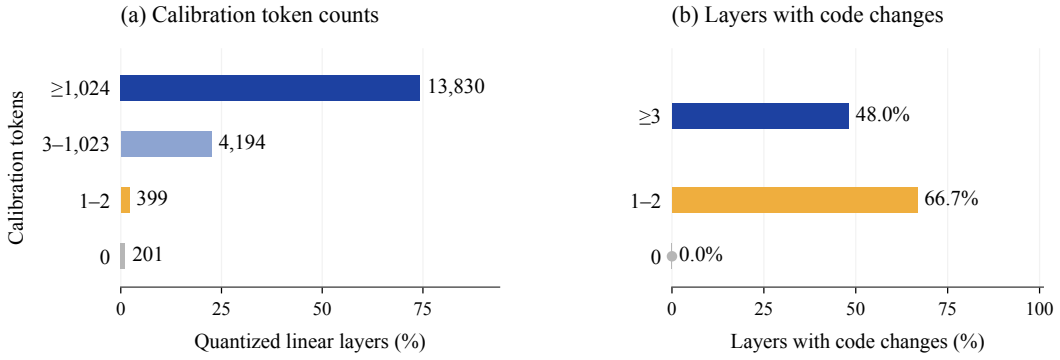


Figure 6: Calibration token counts and integer-code changes in Qwen3-30B-A3B W3A16. (a) The 18,624 quantized linear layers grouped by the number of calibration tokens received, with layer counts shown beside the bars. (b) The fraction of linear layers with at least one changed integer code in each token-count group. The 0–2-token groups contain expert linear layers exclusively, while the ≥ 3 group includes both expert and attention linear layers.

Refinement uses the available tokens. DRQ adjusts its reconstruction checks to the number of available tokens. With at least three tokens, it forms three disjoint subsets, one for proposing and scoring integer-code changes and two for checking reconstruction loss. It reduces their sizes when needed. With one or two tokens, all input activation vectors are used for search. Linear layers receiving no tokens use RTN initialization and set $\mathbf{H}$ and $\mathbf{C}$ to the identity matrix. In this case, the reconstruction term is the squared Frobenius distance between the quantized and full-precision weights. Layers with fewer than three tokens have no separate checking subsets. The reconstruction-loss limits on both checking subsets therefore apply to layers with at least three tokens. This procedure uses the activations routed by the model without adding artificial tokens or changing which experts the router selects.

In Figure 6(b), DRQ modifies 266 of the 399 linear layers receiving one or two tokens (66.7%), compared with 8,657 of 18,024 linear layers in the ≥ 3 group (48.0%). Layers receiving one or two tokens are modified more often but contributes only 199,289 of the 5,425,802 integer-code changes (3.67%). All 201 zero-token linear layers retain their initial integer codes. Overall, DRQ changes 0.01815% of the weights included in these counts, compared with the codes before each layer is refined, while retaining scales and zero points. These changes affect many linear layers but a small fraction of their weights, with most code changes occurring in the ≥ 3 group.

D.4 PARAMETER SENSITIVITY

Figure 7 evaluates the roles of α and β on Llama-3.2-3B-Instruct W3. In Equation (5), α weights the worst-case directional error, while β sets the weight on the identity matrix in $\mathbf{C}$. We test all combinations of $\alpha \in \{0.1, 0.3, 0.5, 1\}$ and $\beta \in \{0.005, 0.1, 0.25, 0.75, 1\}$. Among the displayed runs, 17 of 20 GPTQ configurations and 11 of 20 AWQ configurations improve average task accuracy and both WT2 and C4 PPL relative to the initial quantized model. Improvements on all three metrics occur across more GPTQ configurations, while the favorable AWQ settings concentrate around $\alpha = 0.3$ and 0.5.

At $\alpha = 0.3$, both methods improve all three metrics across all five β values. Increasing α to 1 worsens WT2 PPL in three GPTQ and two AWQ configurations. A stronger penalty on worst-case directional error therefore does not consistently improve model performance in this sweep. The objective balances this penalty against reconstruction loss on calibration data, and the parameter settings that improve model performance depend on the base PTQ method.

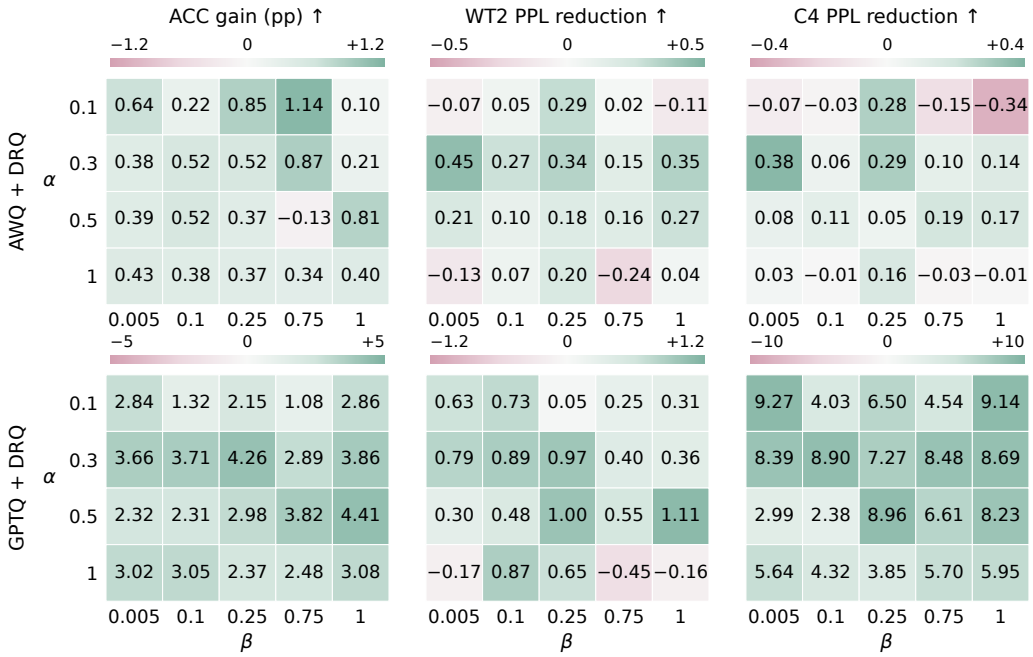


Figure 7: Performance across different α and β combinations on Llama-3.2-3B-Instruct at W3.

E Computational Efficiency

E.1 Refinement time across PTQ methods

Table 10 compares DRQ refinement time for the settings in Appendix D.2. The layer-refinement time sums 196 DRQ calls, one per linear layer. The time including integration also covers the surrounding code that calls DRQ within the base PTQ method or passes stored calibration activations to it. Both columns exclude base-method quantization and hyperparameter search. RTN, GPTQ, AWQ, OmniQuant, and GPTAQ were remeasured on the same A800 80GB GPU. ParoQuant uses its earlier timing measurement.

Table 10: DRQ refinement time for Llama-3.2-3B W3.

Base PTQ method	Layer refinement (s)	Including integration (s)
RTN	556.74	788.04
GPTQ	227.28	233.24
AWQ	482.56	488.99
OmniQuant	520.38	792.13
GPTAQ	296.92	967.00
ParoQuant (earlier run)	220.79	794.04

E.2 Scaling with model size and bit width

Figure 8 compares DRQ refinement time at W2–W4 across 1B–70B Llama models. We sum the time spent refining linear layers. Each 1B–8B point averages the runs for that setting, with 86 runs in total at $(\alpha, \beta) = (0.3, 0.25)$. The 1B–8B models use one A800 80GB GPU, and the 70B model uses two. At W3, refinement takes 2.24, 4.05, 5.95, and 28.05 minutes with GPTQ for 1B, 3B, 8B, and 70B, respectively. The corresponding AWQ times are 2.00, 3.90, 5.40, and 28.96 minutes. The two 70B W3 runs therefore spend similar time on DRQ. At 70B, W2/W4 take 36.55/26.85 minutes with GPTQ and 76.03/28.20 minutes with AWQ. The AWQ W2 and W4 runs use $(\alpha, \beta) = (1, 1)$ and $(0.3, 1)$, respectively. For every 1B–70B model and base PTQ method, W2 takes longer than W3 and W4, including when comparing runs with the same seeds. Refinement time depends on the code replacements tested and the number of search and checking steps performed.

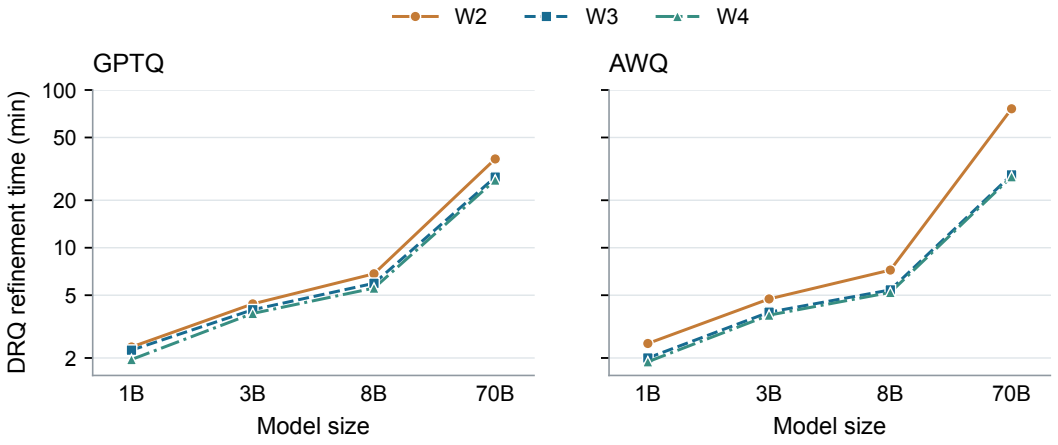


Figure 8: DRQ refinement time across model sizes and weight bit widths. The 1B–8B results are means, while each 70B point represents one run.

E.3 SEARCH COMPLEXITY AND DEPLOYMENT

DRQ limits search cost by retaining a small set of integer-code changes and testing selected combinations. Consider a weight matrix with m rows, one per output channel, and d columns, one per input channel. With n_{d} calibration input vectors in the search subset and $B = |\mathcal{Z}_b|$ available integer codes, scoring all single-code changes costs $O(mdn_{\mathrm{d}} + mdB)$ per round. Only the $k = 8$ shortlisted changes in each output row enter pair scoring, giving at most $\binom{k}{2} = 28$ candidate pairs per row. Computing their activation inner products costs $O(mk^2 n_{\mathrm{d}})$. This shortlist limits the cost of scoring pairs even for wide linear layers while accounting for interactions that single-code scores miss.

DRQ retains at most one update per output row, replacing one or two integer codes. It ranks these row updates by the decrease in the objective with the input direction u held fixed in $O(m \log m)$ time and checks $p \leq 6$ distinct nonzero prefixes. Each prefix applies the first part of this ranked list to the current weights. Increasing its length changes more rows. Direct checks of reconstruction loss between quantized and full-precision layer outputs cost $O(pmd(n_{\mathrm{d}} + n_{\mathrm{c}}))$, where n_{c} counts all input vectors in the calibration subsets reserved for checking reconstruction loss. Wider layers and larger activation subsets make each check more expensive, but the prefix limit bounds how many checks are needed.

We also bound the number of numerical solves used to estimate the worst-case input direction. Let T_{eig} denote the time for one complete numerical solve, including matrix products, linear solves involving $\mathbf{C}$, Krylov basis operations, and Rayleigh–Ritz extraction. Evaluating the initial quantized weights, the current weights at each round, and candidate prefixes costs at most $[1 + \sum_{r=1}^{R}(1 + p_{\mathrm{e},r})]T_{\mathrm{eig}}$ over $R \leq 3$ rounds, where $p_{\mathrm{e},r} \leq p$ counts candidate solves in round r. Even if all six prefixes reach this stage in every round, at most 22 solves are needed per linear layer. Early rejection and stopping reduce this number. Initial reconstruction checks and the optional final comparison with the initial quantized weights in Equation (23) add one-time costs.

All of this computation takes place during offline refinement. DRQ changes the integer codes representing quantized weights while preserving the bit width, quantization grid, dequantization rule, and parameter layout. It therefore adds no inference operators or additional stored quantization parameters. Appendix E.4 measures the resulting latency, throughput, and memory use.

E.4 INFERENCE OVERHEAD

We benchmark Llama-3.1-8B-Instruct at W4A16 with a group size of 128 on one NVIDIA A800 80GB GPU, using vLLM 0.6.3 in eager mode. Batch sizes are $\{1, 4, 8\}$, input lengths are $\{128, 2048\}$, and output length is 128 tokens. We evaluate each initial quantized model and its DRQ-refined counterpart on the same workload and physical GPU, with 10 warm-up runs and 30 timed repeats per round. We time GPU-synchronized engine steps, including scheduling and sampling. Prefill latency covers the first step that processes the input sequence, while decoding latency averages the remaining 127 token-generation steps. Throughput accounts for both prefill and decoding. For latency and throughput, we take the median within each round and average these medians over four retained pairs of benchmark rounds for GPTQ and AWQ. Peak memory is the maximum PyTorch allocated memory across retained rounds.

Table 11 reports all six workloads. Across these workloads, prefill and decoding times change by -0.80% to $+0.82\%$ after refinement, computed from the geometric mean of paired latency ratios across rounds. Model tensors occupy the same amount of memory. DRQ changes the integer codes representing weights while preserving tensor layouts and inference operators, which explains the unchanged memory footprint under matched workloads. These results demonstrate that DRQ adds no extra inference computation or memory overhead.

Table 11: Inference cost across batch sizes and input lengths on Llama-3.1-8B-Instruct, W4A16. Output length is fixed at 128 tokens.

Method	Batch size	Input (tokens)	Prefill (ms)	Decoding (ms/token)	Throughput (tokens/s)	Peak memory (GiB)
GPTQ	1	128	33.61	16.52	60.04	7.86
+ DRQ			33.49	16.49	60.16	7.86
GPTQ	1	2,048	156.58	16.49	56.92	8.01
+ DRQ			157.86	16.59	56.58	8.01
GPTQ	4	128	49.07	17.04	231.35	7.89
+ DRQ			49.05	16.90	233.17	7.89
GPTQ	4	2,048	520.06	17.06	190.56	8.65
+ DRQ			520.16	17.15	189.75	8.65
GPTQ	8	128	77.81	17.24	451.75	7.93
+ DRQ			77.80	17.17	453.57	7.93
GPTQ	8	2,048	1009.35	17.40	318.06	9.56
+ DRQ			1009.24	17.41	318.04	9.56
AWQ	1	128	36.58	18.25	54.38	7.70
+ DRQ			36.56	18.15	54.67	7.70
AWQ	1	2,048	168.47	18.19	51.66	8.01
+ DRQ			168.44	18.20	51.61	8.01
AWQ	4	128	71.86	18.72	209.06	7.89
+ DRQ			71.88	18.74	208.87	7.89
AWQ	4	2,048	557.16	18.75	174.22	8.65
+ DRQ			557.26	18.69	174.71	8.65
AWQ	8	128	103.41	18.92	408.67	7.93
+ DRQ			103.44	18.84	410.28	7.93
AWQ	8	2,048	1074.83	19.02	293.35	9.55
+ DRQ			1074.85	18.96	293.96	9.55