# DartQuant

## Preliminaries

LLM里的Linear layer可以表示为 $Y=XW^T$，其中 $X\in \mathbb{R}^{T\times C_\text{in}}$ 为输入激活，$W\in \mathbb{R}^{C_\text{out}\times C_\text{in}}$ 为权重矩阵。引入正交矩阵 $R\in \mathbb{R}^{C_\text{in}\times C_\text{in}}$（满足 $RR^\top=I$），可得等价变换：

$$Y=(XR)(R^\top W^\top)$$

### 经典Transformer Block (LLaMA)

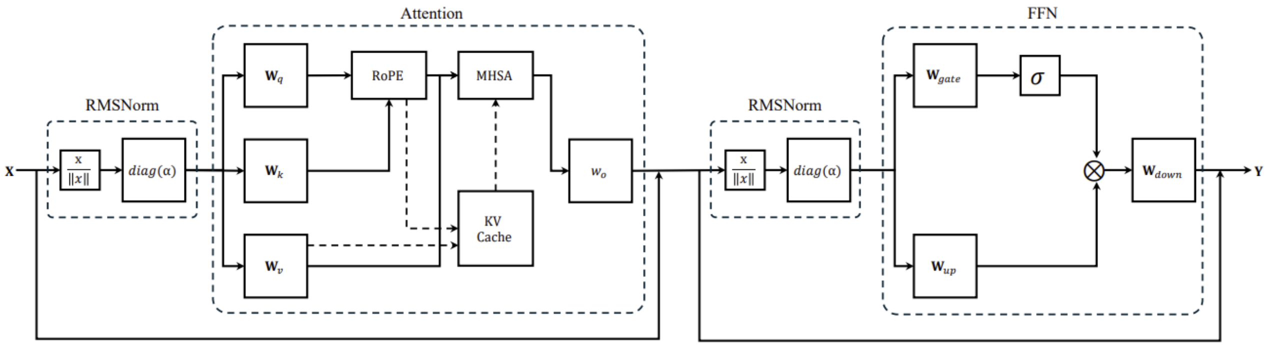


Figure 8: Flowchart of the transformer block used in most language models, including pre-RMSNorm, multi-head self-attention (MHSA), and the gated feedforward network (FFN). The solid arrows represent the data flow during training, pre-filling, and inference for each token. In RMSNorm, the input signal is normalized by its norm and rescaled by the parameter $\sigma$. In MHSA, the RoPE module computes the relative position embeddings, and the dashed arrows indicate access to the KV-Cache during generation. In the FFN, the activation function $\sigma$ is applied to the gated signal, and the two signals are combined element-wise.

### DartQuant后的Transformer Block (LLaMA)

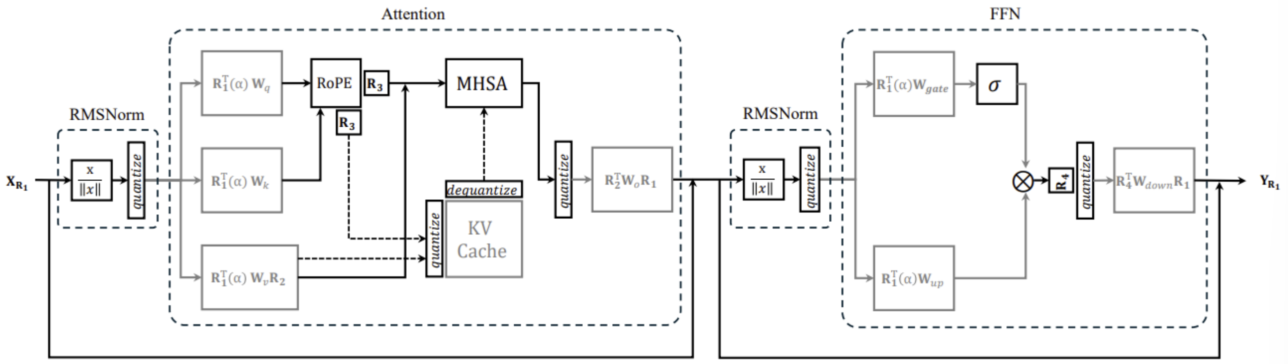


Figure 9: Transformer applied in DartQuant. The RMSNorm scaling factor ($\sigma$) has been absorbed into the weight matrices. The black section represents the flow in FP16 format, while the gray section indicates the flow in INT4 format, and the dashed line shows the flow in and out of the KV buffer. The hidden state $X$ has been rotated by $R_1$, which is offset by $R_1^\top$ and absorbed into the weight matrices $W_q, W_k, W_v, W_{up}, W_{gate}$. $R_1$ is also incorporated into $W_o$ and $W_{down}$ to ensure correct residual calculation. $R_2$ in $W_v$ cancels with $R_2^\top$ in $W_o$. $R_3$ and $R_4$ are random Hadamard matrices computed online: $R_3$ cancels out during the attention computation, and $R_4$ cancels with $R_3^\top$ absorbed in $W_{down}$. All weights are stored in INT4 format, and all activations prior to the weights are also quantized to INT4. The result of the matrix multiplication between INT4 weights and INT4 activations on TensorCore is INT32, which is immediately converted (and scaled) to FP16.

---

## 创新点

1. **Rotational Distribution Calibration**：提出高效的分布感知旋转校准方法，通过约束旋转后激活分布来降低优化复杂度
2. **Whip Loss**：优化激活分布使其趋向均匀分布，减少outliers影响
3. **QR-Orth Optimization**：用QR分解替代昂贵的流形优化（Cayley SGD），显著提升效率
4. **资源效率**：首次在单张3090 GPU上完成70B模型的旋转校准（约3小时），实现47×加速和10×显存节省

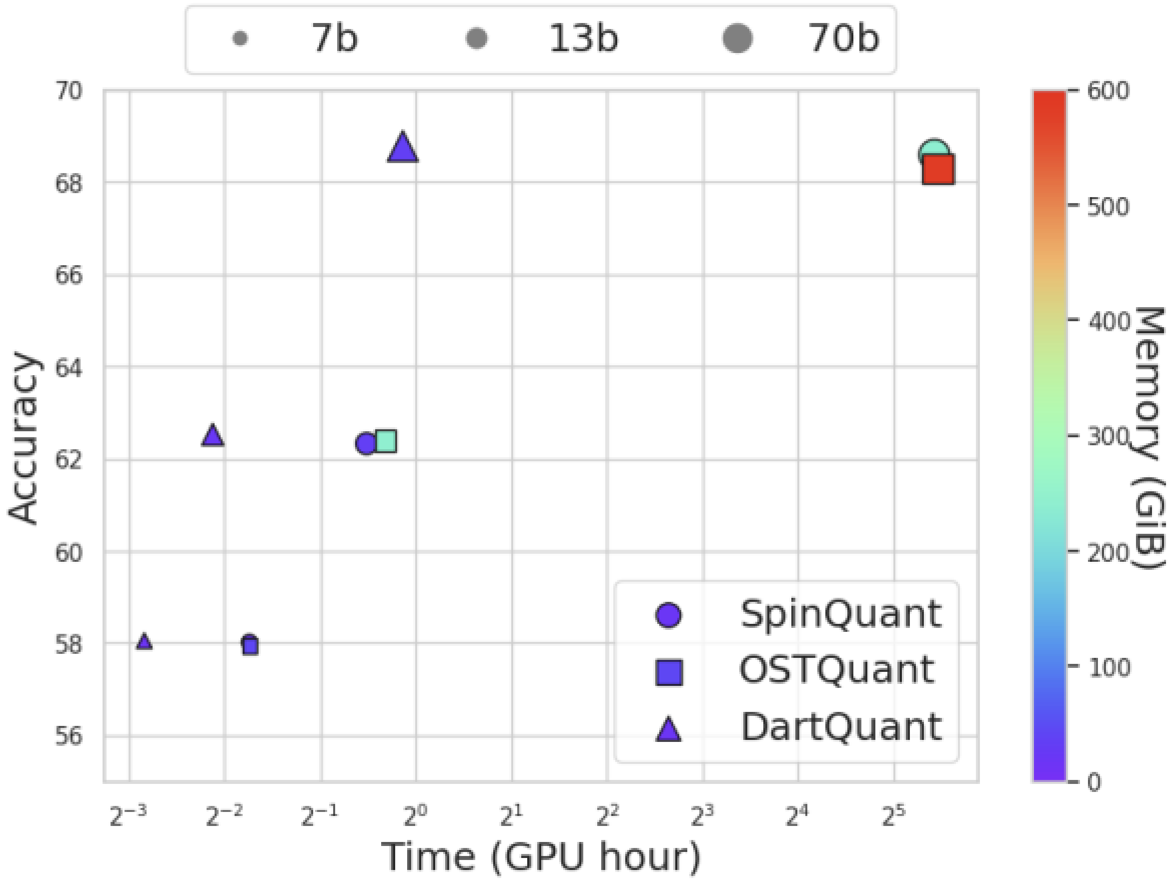


Figure 1: Comparison of computational costs across different rotation optimization methods.

---

## 问题背景

### 激活Outliers问题

LLM后训练量化中，激活值比权重更难处理，因为存在**极端outliers**会严重降低量化精度。

> *Ma et al. (AffineQuant, ICLR 2024)*：旋转矩阵和仿射变换能有效减少激活outliers，显著提升量化性能。

### 当前旋转方法的局限

**旋转矩阵的优势**：

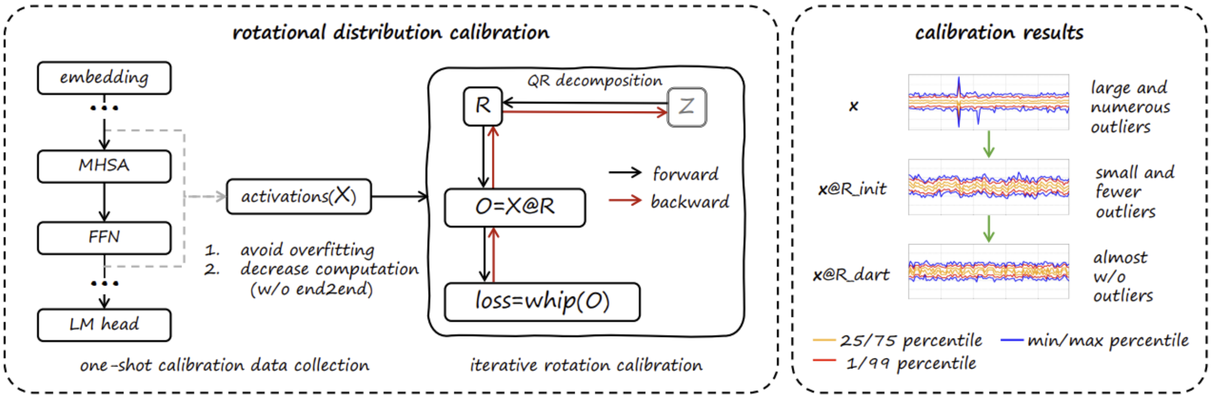
- 可逆、保范数（$|Rx|_2 = |x|_2$）
- 可无缝融入模型架构，不增加推理开销

**现有方法问题**：

- SpinQuant/OSTQuant将旋转矩阵作为网络参数进行端到端微调
- 需要Cayley SGD等流形优化器，计算开销约 $O(6n^3)$
- 小样本微调容易过拟合（见Table 1）
- 70B模型需数百GiB显存和数十GPU小时

---

# Method



> Figure 4: Left: The DartQuant implementation process, with $Z$ representing the latent parameters in QR-orth and $R$ as the applied rotation matrix. Right: The change in rotation matrix before and after calibration.

## 4.1 Rotational Distribution Calibration

**核心思想**：不做端到端微调，而是直接优化旋转后激活的分布，使其更适合量化。

**优化目标**：最小化outliers数量 $$\min_R\sum_{i=1}^{C_\text{in}}\mathbb{I}(|(Rx)_i|>\tau)$$

其中 $\mathbb{I}(\cdot)$ 是指示函数，$\tau$ 是outlier阈值。

**为什么不用方差/峰度?**

- **方差不可行**：由于激活对称分布，方差 $\propto |x|_2^2$。而旋转保范数，即 $|Rx|_2 = |x|_2$，导致方差在旋转前后不变，梯度信号极弱
- **峰度收敛慢**：旋转后激活已接近Gaussian，峰度优化速度慢（见Figure 7a）

## 4.2 Whip Loss

**目标**：将Laplace分布的激活转换为均匀分布，从而减少outliers。

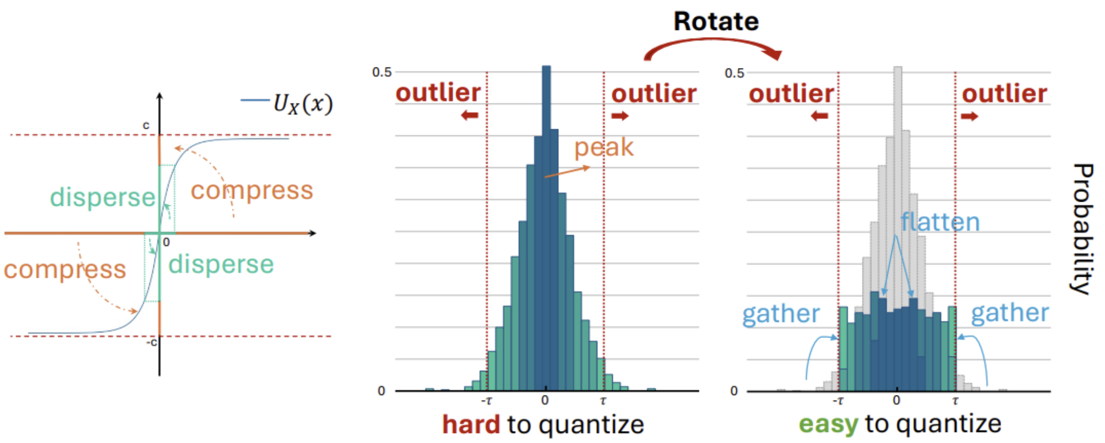


Figure 5: Intuition behind the distribution transformation: $U_X(x)$ transforms the Laplace distribution into a uniform distribution by flattening the peak and aggregating the outliers.

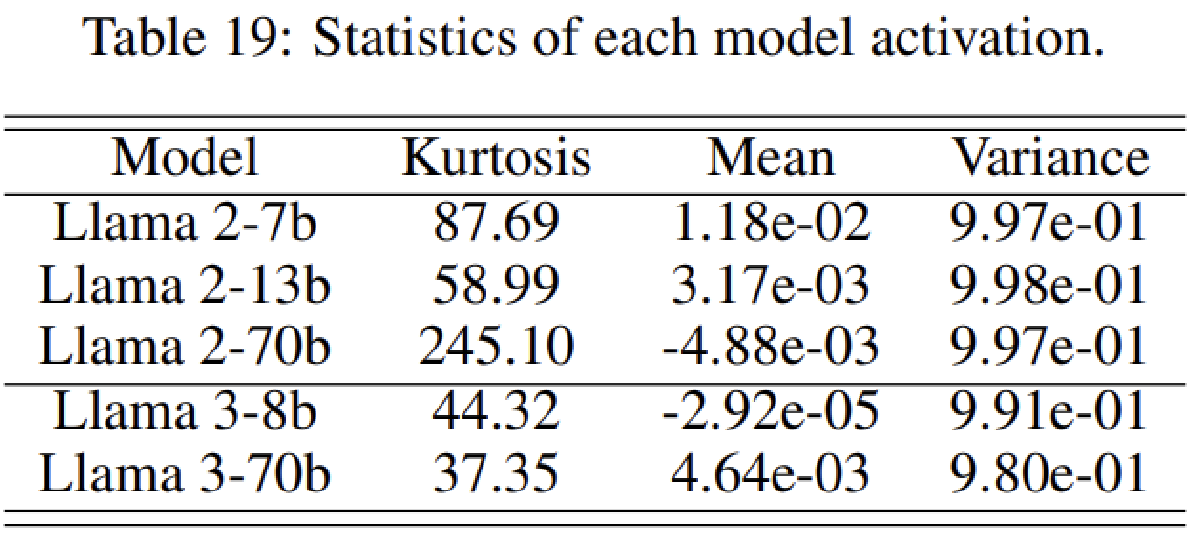


Table 19: Statistics of each model activation.

| Model | Kurtosis | Mean | Variance |
|---|---|---|---|
| Llama 2-7b | 87.69 | 1.18e-02 | 9.97e-01 |
| Llama 2-13b | 58.99 | 3.17e-03 | 9.98e-01 |
| Llama 2-70b | 245.10 | -4.88e-03 | 9.97e-01 |
| Llama 3-8b | 44.32 | -2.92e-05 | 9.91e-01 |
| Llama 3-70b | 37.35 | 4.64e-03 | 9.80e-01 |

**假设**：激活服从 Laplace$(0, b)$ 分布，PDF为： $$f(x) = \frac{1}{2b}\exp\left(-\frac{|x|}{b}\right)$$

**分布变换函数**：用CDF将 $x \sim \text{Laplace}(0,b)$ 映射到均匀分布 $[-\tau, \tau]$： $$U_X(x) = 2\tau\left[\int_{-\infty}^x \frac{1}{2b}\exp\left(-\frac{|t|}{b}\right)dt - \frac{1}{2}\right] = \begin{cases} \tau[\exp(\frac{x}{b})-1], & x\le 0\ \tau[1-\exp(-\frac{x}{b})], & x > 0 \end{cases}$$

**Whip Loss设计**： $$\mathcal{L}{\text{Whip}} = \sum{i=1}^{C_\text{in}}\exp(-|x_i|)$$

**为什么Whip有效?**

1. $\exp(-|x|)$ 在零点附近梯度大，将小值"推离"零点
2. 由于旋转保范数（$|Rx|_2 = |x|_2$），中间值被放大时，outliers必须被压缩
3. 最终效果：峰值被平滑，outliers被聚拢，分布趋向均匀

> **直觉**：4维向量 $[x_1, x_2, x_3, x_4]$，若 $x_1,x_2,x_3$ 小而 $x_4$ 是outlier。范数守恒下放大 $x_1,x_2,x_3$ 必然压缩 $x_4$。

### 4.3 QR-Orth Optimization

**问题**：直接梯度下降会破坏正交性 $$R' = R - \eta\frac{\partial \mathcal{L}}{\partial R} \quad \Rightarrow \quad R'R'^{\top} \neq I$$

**传统方案**：Cayley SGD在Stiefel流形上优化，额外计算开销 $O(6n^3)$

**QR-Orth方案**：引入无约束隐变量 $Z \in \mathbb{R}^{n\times n}$

**算法流程**：

1. 初始化：$Z_0$ 为随机Hadamard矩阵
2. QR分解：$Z = RU$，取正交矩阵 $R$，丢弃上三角 $U$
3. 前向传播：计算 $O = XR$，$\mathcal{L} = \text{Whip}(O)$
4. 梯度更新：$Z \leftarrow Z - \eta \frac{\partial \mathcal{L}}{\partial Z}$
5. 重复步骤2-4直到收敛

**关键点**：

- 我们更新的是 $Z$（无约束），通过QR分解间接得到正交的 $R$
- 最终只保留 $R$，$Z$ 是辅助变量
- QR分解复杂度 $O(\frac{4}{3}n^3)$，比Cayley SGD的 $O(6n^3)$ 额外开销低得多
- 实测加速 1.4×（相同迭代数），由于收敛更快，总体加速可达 41×

### 4.4 四个旋转矩阵的用途

| 矩阵 | 位置 | 优化方式 | 推理开销 |
|---|---|---|---|
| $R_1$ | 层间（$W_q, W_k, W_v, W_{up}, W_{gate}$ 前） | DartQuant学习 | **无**（融入权重） |
| $R_2$ | Attention内（$W_v$ 和 $W_o$ 之间） | DartQuant学习 | **无**（融入权重） |
| $R_3$ | RoPE后的Q、K之间 | 随机Hadamard | 有（online计算，用fast kernel） |
| $R_4$ | $W_{down}$ 前 | 随机Hadamard | 有（online计算，用fast kernel） |

### 4.5 后续量化流程

旋转校准完成后：

1. 将 $R_1, R_2$ 融入相邻权重矩阵
2. 使用 **GPTQ** 对权重进行重建（128样本，序列长度2048）
3. 激活使用 per-token 非对称量化

---

## 实验结果

### 主要结果（W4A4KV16）

| 模型 | 方法 | PPL ↓ | 0-shot Acc ↑ |
|---|---|---|---|

| 模型 | 方法 | PPL ↓ | 0-shot Acc ↑ |
|---|---|---|---|
| LLaMA-2 70B | SpinQuant | 11.70 | 68.59 |
| LLaMA-2 70B | OSTQuant | 11.98 | 68.29 |
| LLaMA-2 70B | **DartQuant** | **11.51** | **69.02** |
| LLaMA-3 70B | SpinQuant | 9.61 | 66.06 |
| LLaMA-3 70B | OSTQuant | 7.67 | 67.94 |
| LLaMA-3 70B | **DartQuant** | 7.99 | **69.39** |

### 资源消耗对比（70B模型）

| 方法 | 时间 (GPU hour) | 显存 (GiB) |
|---|---|---|
| SpinQuant | 42.90 | 238.89 |
| OSTQuant | 44.00 | 583.86 |
| **DartQuant** | **0.91** | **23.47** |
| DartQuant (3090) | 2.90 | 23.47 |

## Appendix 要点

### A. Laplace假设的验证

Table 19显示LLaMA系列激活统计特性：

- Mean ≈ $10^{-2}$ 量级（接近0）
- Variance ≈ 1
- Kurtosis >> 0（重尾特性，Gaussian的kurtosis=0）

**局限**：若mean显著偏离0，Whip Loss效果可能下降。

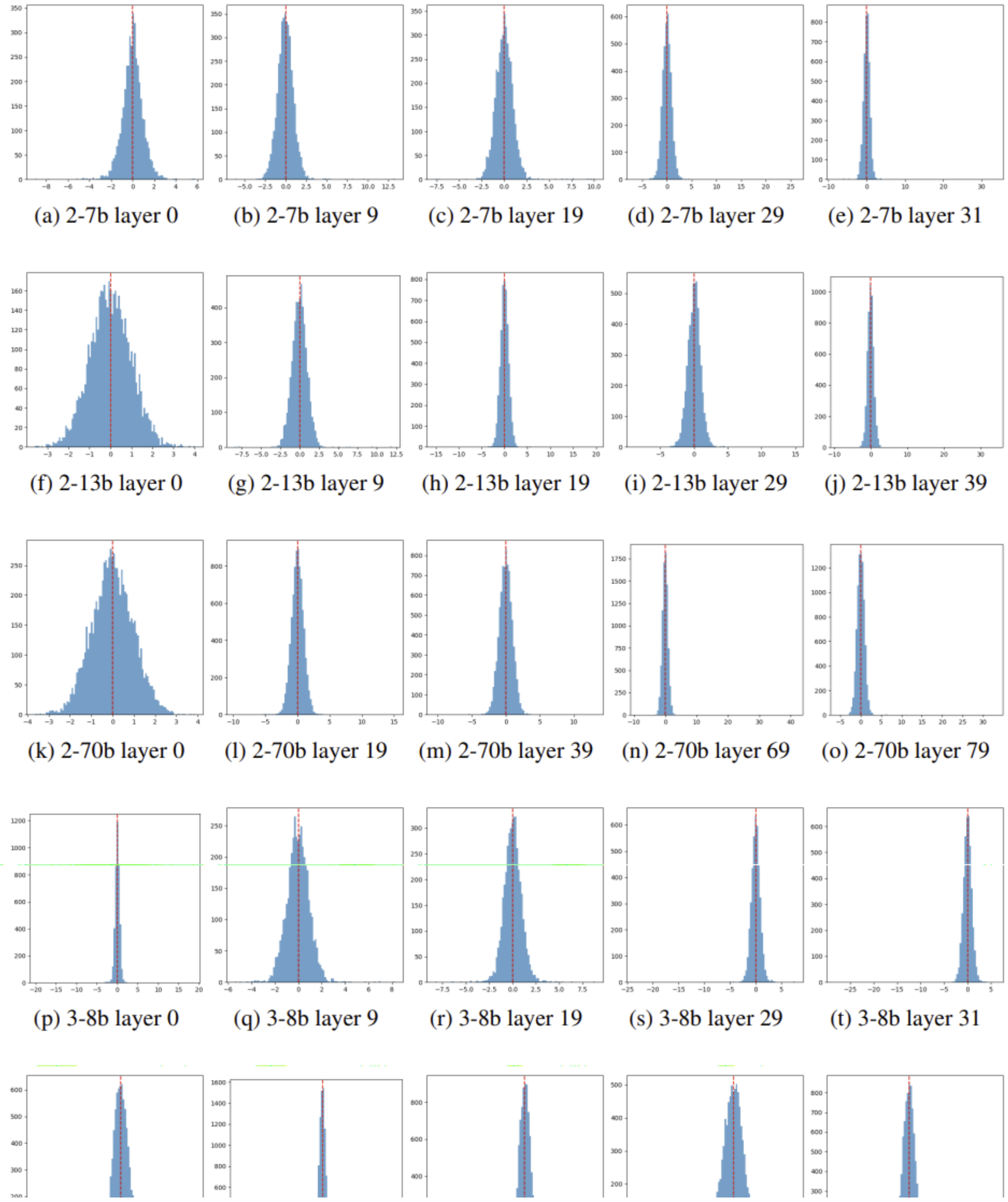
# G Activation Distribution

We conducted an in-depth investigation into the distribution characteristics of activations in LLMs. First, we randomly sampled 1,000 activation samples from each model and computed their mean, variance and kurtosis. The detailed statistical results are presented in Table 19. As observed, the mean of activations is close to zero, the variance is approximately 1, and the activations exhibit high kurtosis (whereas the kurtosis of a Gaussian distribution is 0). These statistics indicate that most activation values are concentrated around zero and exhibit significant heavy-tailed properties.

Figure 11 presents the activation distribution histograms across different layers of various models. As shown, apart from a few outliers, the overall activation distribution is symmetric around zero. These distribution characteristics closely align with those of a Laplacian distribution. Therefore, we model the activation distribution as a simplified Laplacian distribution.

Table 19: Statistics of each model activation.

| Model | Kurtosis | Mean | Variance |
|---|---|---|---|
| Llama 2-7b | 87.69 | 1.18e-02 | 9.97e-01 |
| Llama 2-13b | 58.99 | 3.17e-03 | 9.98e-01 |
| Llama 2-70b | 245.10 | -4.88e-03 | 9.97e-01 |
| Llama 3-8b | 44.32 | -2.92e-05 | 9.91e-01 |
| Llama 3-70b | 37.35 | 4.64e-03 | 9.80e-01 |

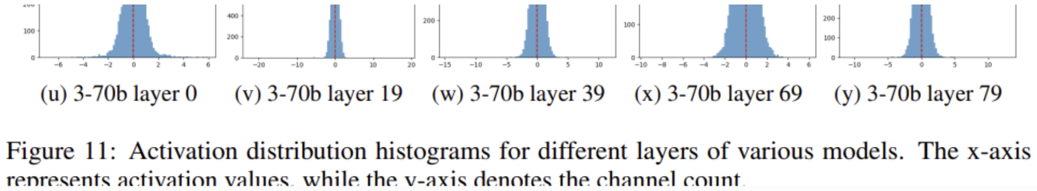


(a) 2-7b layer 0 (b) 2-7b layer 9 (c) 2-7b layer 19 (d) 2-7b layer 29 (e) 2-7b layer 31

(f) 2-13b layer 0 (g) 2-13b layer 9 (h) 2-13b layer 19 (i) 2-13b layer 29 (j) 2-13b layer 39

(k) 2-70b layer 0 (l) 2-70b layer 19 (m) 2-70b layer 39 (n) 2-70b layer 69 (o) 2-70b layer 79

(p) 3-8b layer 0 (q) 3-8b layer 9 (r) 3-8b layer 19 (s) 3-8b layer 29 (t) 3-8b layer 31



(u) 3-70b layer 0 (v) 3-70b layer 19 (w) 3-70b layer 39 (x) 3-70b layer 69 (y) 3-70b layer 79

Figure 11: Activation distribution histograms for different layers of various models. The x-axis represents activation values, while the y-axis denotes the channel count.

### B. 复杂度分析

- QR分解（Householder）：$\frac{4}{3}n^3$
- Cayley SGD额外开销：$6n^3$
- 加速比：$\approx 1.4\times$（单次迭代），$\approx 41\times$（考虑收敛速度）

### C. 抗过拟合验证

Table 1显示端到端微调在特定数据集上过拟合明显（PTB上PPL下降但泛化变差）。DartQuant对校准数据集不敏感（Table 5）。

---

## 总结

**核心贡献**：

1. 用分布校准替代端到端微调，避免过拟合
2. Whip Loss利用旋转保范数特性，通过放大小值来压缩outliers
3. QR-Orth将流形优化转化为无约束优化，大幅降低计算开销
4. 首次实现单张消费级GPU量化70B模型