A. Lower bound for rate distortion function

In this section, we finish the proof of the lower bound and achievability in Section 4. Our approach is based on the water-filling approach (McDonald & Schultheiss, 1964).

A.1. General lower bound

First, we establish establishes a lower bound of the rate distortion function, which works for general models..

Lemma 1 *The rate-distortion function $R(D) \geq \underline{R}(D) = h(W) - C$, where C is the optimal value of the following optimization problem.*

$$\begin{aligned} \max_{P_{\hat{W}|W}} \quad & \sum_{i=1}^{m} \min\left\{h(W_i), \frac{1}{2}\log(2\pi e \mathbb{E}_{W,\hat{W}}[(W_i - \hat{W}_i)^2])\right\} \\ \text{s.t.} \quad & E_{W,\hat{W}}\left[d(W,\hat{W})\right] \leq D. \end{aligned}$$

where $h(W) = -\int_{w\in\mathcal{W}} P_W(w)\log P_W(w)dw$ is the differential entropy of W and $h(W_i)$ is the differential entropy of the i-th entry of W.

A.1.1. Proof of Lemma 1

Recall that the rate distortion function for model compression is defined as $R(D) = \min_{P_{\hat{W}|W}:\mathbb{E}_{W,\hat{W}}[d(W,\hat{W})]\leq D} I(W;\hat{W})$. Now we lower bound the mutual information $I(W,\hat{W})$ by

$$\begin{aligned} I(W;\hat{W}) &= h(W) - h(W\,|\,\hat{W}), \\ &= h(W) - \sum_{i=1}^{m} h(W_i\,|\,W_1,\ldots,W_{i-1},\hat{W}_i,\ldots,\hat{W}_m) \\ &\geq h(W) - \sum_{i=1}^{m} h(W_i\,|\,\hat{W}_i). \end{aligned}$$

Here the last inequality comes from the fact that conditioning does not increase entropy. Notice that the first term $h(W)$ does not depend on the compressor. For the last term, we upper bound each term $h(W_i\,|\,\hat{W}_i)$ in two ways. On one hand, $h(W_i\,|\,\hat{W}_i)$ is upper bounded by $h(W_i)$ because conditioning does not increase entropy. On the other hand, $h(W_i\,|\,\hat{W}_i) = h(W_i - \hat{W}_i\,|\,\hat{W}_i) \leq h(W_i - \hat{W}_i)$, and by Cover & Thomas (2012, Theorem 8.6.5), differential entropy is maximized by Gaussian distribution, for given second moment. We then have:

$$\begin{aligned} h(W_i\,|\,\hat{W}_i) &\leq \min\left\{h(W_i), h(W_i - \hat{W}_i)\right\} \\ &\leq \min\left\{h(W_i), \frac{1}{2}\log\left(2\pi e \mathbb{E}_{W,\hat{W}}[(W_i - \hat{W}_i)^2]\right)\right\} \\ &= \min\left\{h(W_i), \frac{1}{2}\log(2\pi e \mathbb{E}_{W,\hat{W}}[(W_i - \hat{W}_i)^2])\right\}. \end{aligned}$$

Therefore, the lower bound of the mutual information is given by,

$$I(W;\hat{W}) \geq h(W) - \sum_{i=1}^{m} \min\left\{h(W_i), \frac{1}{2}\log(2\pi e \mathbb{E}_{W,\hat{W}}[(W_i - \hat{W}_i)^2])\right\}.$$

A.2. Lower bound for linear model

For complex models, the general lower bound in Lemma 1 is difficult to evaluate, due to the large dimension of parameters. It was shown by Jiao et al. (2017) that the sample complexity to estimate differential entropy is exponential to the dimension.

It's even harder to design an algorithm to achieve the lower bound. But for linear model, the lower bound can be simplified. For $f_w(x) = w^T x$, the distortion function $d(w, \hat{w})$ can be written as

$$\begin{aligned} d(w,\hat{w}) &= \mathbb{E}_X\left[(f_w(X) - f_{\hat{w}}(X))^2\right] = \mathbb{E}_X\left[(w^T X - \hat{w}^T X)^2\right] \\ &= \mathbb{E}_X\left[(w-\hat{w})^T X X^T (w-\hat{w})\right] = (w-\hat{w})^T \mathbb{E}_X[XX^T](w-\hat{w}). \end{aligned}$$

Since we assumed that $\mathbb{E}[X] = 0$, $\mathbb{E}[X_i^2] = \lambda_{x,i} > 0$ and $\mathbb{E}[X_i X_j] = 0$, so the constraint in Lemma 1 is given by

$$\begin{aligned} D &\geq \mathbb{E}_{W,\hat{W}}\left[(W-\hat{W})^T \mathbb{E}_X[XX^T](W-\hat{W})\right] \\ &= \sum_{i=1}^m \lambda_{x,i} \underbrace{\mathbb{E}_{W,\hat{W}}\left[(W_i - \hat{W}_i)^2\right]}_{D_i}. \end{aligned}$$

Then the optimization problem in Lemma 1 can be written as follows

$$\begin{aligned} \max_{p(\hat{w}|w)} \quad & \sum_{i=1}^m \min\{h(W_i), \frac{1}{2}\log(2\pi e D_i)\} \\ \text{s.t.} \quad & \sum_{i=1}^m \lambda_{x,i} D_i \leq D. \end{aligned}$$

Here W_i is a Gaussian random variable, so $h(W_i) = \frac{1}{2}\log(2\pi e \mathbb{E}[W_i^2])$. The Lagrangian function of the problem is given by

$$\begin{aligned} & \mathcal{L}(D_1, \ldots, D_m, \mu) \\ = & \sum_{i=1}^m \Big(\min\{\frac{1}{2}\log \mathbb{E}[W_i^2], \frac{1}{2}\log D_i\} + \frac{1}{2}\log(2\pi e) - \mu \lambda_{x,i} D_i \Big). \end{aligned}$$

By setting the derivative w.r.t. D_i to 0, we have

$$0 = \frac{\partial \mathcal{L}}{\partial D_i} = \frac{1}{2D_i} - \mu \lambda_{x,i}.$$

for all D_i such that $D_i < \mathbb{E}[W_i^2]$. So the optimal D_i should satisfy that $D_i \lambda_{x,i}$ is constant, for all D_i such that $D_i < \mathbb{E}[W_i^2]$. Also the optimal D_i is at most $\mathbb{E}[W_i^2]$. Also, since $h(W) = \frac{m}{2}\log(2\pi e) + \frac{1}{2}\log\det(\Sigma_W)$ the lower bound is given by

$$R(D) \geq \frac{1}{2}\log\det(\Sigma_W) - \sum_{i=1}^m \frac{1}{2}\log(D_i),$$

where

$$D_i = \begin{cases} \mu/\lambda_{x,i} & \text{if} \mu < \lambda_{x,i}\mathbb{E}_W[W_i^2]\,, \\ \mathbb{E}_W[W_i^2] & \text{if} \mu \geq \lambda_{x,i}\mathbb{E}_W[W_i^2]\,, \end{cases}$$

where μ is chosen that $\sum_{i=1}^m \lambda_{x,i} D_i = D$.

This lower bound gives rise to a "weighted water-filling", which differs from the classical "water-filling" for rate-distortion of colored Gaussian source in Cover & Thomas (2012, Figure 13.7), since the water level's D_i are proportional to $1/\lambda_{x,i}$, which is related to the input of the model rather than the parameters to be compressed. To illustrate the "weighted water-filling" process, we choose a simple example where $\Sigma_W = \Sigma_X = \text{diag}[3, 2, 1]$. In Figure 6, the widths of each rectangle are proportional to $\lambda_{x,i}$, and the heights are proportional to $\Sigma_W = [3, 2, 1]$. The water level in each rectangle is D_i and the volume of water is μ. As D starts to increase from 0, each rectangle is filled with same volume of water (μ is the same), but the water level D_i's increase with speed $1/\lambda_{x,i}$ respectively (Figure 6.(a)). This gives segment (a) of the rate distortion curve in Figure 6.(d). If D is large enough such that the third rectangle is full, then D_3 is fixed to be $\mathbb{E}[W_3^2] = 1$, whereas D_1 and D_2 continuously increase (Figure 6.(b)). This gives segment (b) in Figure 6.(d). Keep increasing D until the second rectangle is also full, then D_2 is fixed to be $\mathbb{E}[W_2^2] = 2$ and D_1 continuous increasing (Figure 6 (c)). This gives segment (c) in Figure 6.(d). The entire rate-distortion function is shown in Figure 6(d), where the first red dot corresponds to the moment that the third rectangle is exactly full, and the second red dot corresponds to moment that the second rectangle is exactly full.

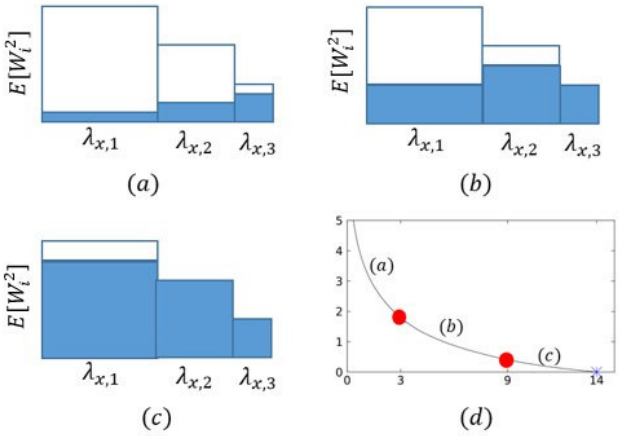

Figure 6. Illustration of "weighted water-filling" process.

A.3. Achievability

We prove that this lower bound is achievable. To achieve the lower bound, we construct the compression algorithm in Algorithm 1,

Algorithm 1 Optimal compression algorithm for linear regression

Input: distortion D, covariance matrix of parameters Σ_W, covariance matrix of data $\Sigma_X = \mathrm{diag}[\lambda_{x,1}, \ldots, \lambda_{x,m}]$.
Choose D_i's such that

$$D_i = \begin{cases} \mu/\lambda_{x,i} & \text{if} \mu < \lambda_{x,i}\mathbb{E}_W[W_i^2] \ , \\ \mathbb{E}_W[W_i^2] & \text{if} \mu \geq \lambda_{x,i}\mathbb{E}_W[W_i^2] \ , \end{cases}$$

where $\sum_{i=1}^m \lambda_{x,i} D_i = D$.
for $i = 1$ to m **do**
  if $D_i = \mu/\lambda_{x,i}$ **then**
    Choose $\hat{W}_i = 0$
  else
    Choose a conditional distribution $P_{\hat{W}_i|W_i}$ such that $W_i = \hat{W} + Z_i$ where $Z_i \sim \mathcal{N}(0, D_i)$, $\hat{W}_i \sim \mathcal{N}(0, \mathbb{E}_W[W_i^2] - D_i)$ and $\hat{W}_i$ is independent of Z_i.
  end if
end for
Combine the conditional probability distributions by $P_{\hat{W}|W} = \prod_{i=1}^m P_{\hat{W}_i|W_i}$.

Intuitively, the optimal compressor does the following: (1) Find the optimal water levels D_i for "weighted water filling". (2) For the entries where the corresponding rectangles are full, simply discard the entries; (3) for the entries where the corresponding rectangles are not full, add a noise which is independent of $\hat{W}_i$ and has a variance proportional to the water level. That is possible since W is Gaussian. (4) Combine the conditional probabilities.

To see that this compressor is optimal, we will check that the compressor makes all the inequalities become equality. Here is all the inequalities used in the proof.

- $h(W_i \,|\, W_1, \ldots, W_{i-1}, \hat{W}_i, \ldots, \hat{W}_m) \leq h(W_i|\hat{W}_i)$ for all $i = 1...m$. It becomes equality by $P_{\hat{W}|W} = \prod_{i=1}^m P_{\hat{W}_i|W}$.
- Either
  - $h(W_i|\hat{W}_i) \leq h(W_i)$. It becomes equality for those $\hat{W}_i = 0$.

- $h(W_i - \hat{W}_i|\hat{W}_i) \le h(W_i - \hat{W}_i) \le \frac{1}{2}\log(2\pi e \mathbb{E}_{W,\hat{W}}[(W_i - \hat{W})^2])$. It becomes equality for those $\hat{W}_i$'s such that $W_i - \hat{W}_i$ is independent of $\hat{W}_i$ and $W_i - \hat{W}_i$ is Gaussian.

- The "water levels" D_i. It becomes equality by choosing the D_i's according to Lagrangian conditions.

Therefore, Algorithm 1 gives a compressor $P^{(D)}_{\hat{W}|W}$ such that $\mathbb{E}_{P_W \circ P^{(D)}_{\hat{W}|W}}[d(W,\hat{W})] = D$ and $I(W;\hat{W}) = \underline{R}(D)$, hence the lower bound is tight.

B. Proof of Theorem 4

In this section, we provide the proof of Theorem 4. For simplicity let $\sigma(t) = t\mathbb{I}\{t \ge 0\}$ denotes the ReLU activation function. First we deal with the objective of the compression algorithm,

$$\begin{aligned}
(w-\hat{w})^T I_w (w-\hat{w}) &= (w-\hat{w})^T \mathbb{E}_X\left[\nabla_w f_w(x) \nabla_w f_w(x)^T\right](w-\hat{w}) \\
&= (w-\hat{w})^T \mathbb{E}_X\left[\nabla_w \sigma(w^T x) \nabla_w \sigma(w^T x)^T\right](w-\hat{w}) \\
&= (w-\hat{w})^T \mathbb{E}_X\left[x^T (\sigma'(w^T x))^2 x\right](w-\hat{w}) \\
&= \mathbb{E}_X\left[\mathbb{I}\{w^T x \ge 0\}((w-\hat{w})^T x)^2\right]
\end{aligned}$$

Notice that x is jointly Gaussian random variable with zero mean and non-degenerate variance, so the distribution of x is equivalent to the distribution of $-x$. Therefore,

$$\begin{aligned}
&\mathbb{E}_X[\mathbb{I}\{w^T x \ge 0\}((w-\hat{w})^T x)^2] = \int_{x: w^T x \ge 0} ((w-\hat{w}^T)x)^2 dx \\
= \;& \frac{1}{2}\left(\int_{x: w^T x \ge 0} ((w-\hat{w}^T)x)^2 dx + \int_{x: w^T x \le 0} ((w-\hat{w}^T)x)^2 dx\right) \\
= \;& \frac{1}{2}\int_{x \in \mathbb{R}}^{d} ((w-\hat{w}^T)x)^2 dx = \frac{1}{2}(w-\hat{w})^T \Sigma_X (w-\hat{w})
\end{aligned}$$

So minimizing the gradient-squared based loss is equivalent to minimizing $(w-\hat{w})^T \Sigma_X (w-\hat{w})$. Similarly, the condition $\hat{w} I_w (w-\hat{w}) = 0$ is equivalent to $\hat{w}\Sigma_X(w-\hat{w}) = 0$. Now we deal with the MSE loss function $\mathbb{E}[(f_w(x) - f_{\hat{w}}(x))^2]$. We utilize the Hermite polynomials and Fourier analysis on Gaussian space. We use the following key lemma,

Lemma 2 *(Ge et al. (2017, Claim 4.3)) Let f, g be two functions from $\mathbb{R}$ to $\mathbb{R}$ such that $f^2, g^2 \in L^2(\mathbb{R}, e^{-x^2/2})$. The for any unit vectors u, v, we have that*

$$\mathbb{E}_{x \in \mathcal{N}(0, I_{d\times d})}[f(u^T x) g(v^T x)] = \sum_{p=0}^{\infty} \hat{f}_p \hat{g}_p (u^T v)^p$$

where $\hat{f}_p = \mathbb{E}_{x \in \mathcal{N}(0,1)}[f(x) h_p(x)]$ is the p-th order coefficient of f, where h_p is the p-th order probabilists' Hermite polynomial.

Please see Section 4.1 in Ge et al. (2017) for more backgrounds of the Hermite polynomials and Fourier analysis on Gaussian space. For ReLU function, the coefficients are given by $\hat{\sigma}_0 = \frac{1}{\sqrt{2\pi}}$, $\hat{\sigma}_1 = \frac{1}{2}$. For $p \ge 2$ and even, $\hat{\sigma}_p = \frac{((p-3)!!)^2}{\sqrt{2\pi p!}}$. For $p \ge 2$ and odd, $\hat{\sigma}_p = 0$. Since $X \sim \mathcal{N}(0, \Sigma_X)$, we can write $x = \Sigma_X^{1/2} z$, where $z \sim \mathcal{N}(0, I_d)$. So for any compressed weight $\hat{w}$,

we have

$$\begin{aligned}
& \mathbb{E}_X\left[(f_w(x)-f_{\hat{w}}(x))^2\right]=\mathbb{E}_X\left[(\sigma(w^Tx)-\sigma(\hat{w}^Tx))^2\right] \\
= \;& \mathbb{E}_{z\in\mathcal{N}(0,I_d)}[(\sigma(w^T\Sigma_X^{1/2}z)-\sigma(\hat{w}^T\Sigma_X^{1/2}z))^2] \\
= \;& \mathbb{E}_{z\in\mathcal{N}(0,I_d)}[\sigma(w^T\Sigma_X^{1/2}z)^2]-2\mathbb{E}_{z\in\mathcal{N}(0,I_d)}[\sigma(w^T\Sigma_X^{1/2}z)\sigma(\hat{w}^T\Sigma_X^{1/2}z)]+\mathbb{E}_{z\in\mathcal{N}(0,I_d)}[\sigma(\hat{w}^T\Sigma_X^{1/2}z)^2] \\
= \;& \sum_{p=0}^{\infty}\hat{\sigma}_p^2(w^T\Sigma_Xw)^p-2\sum_{p=0}^{\infty}\hat{\sigma}_p^2(w^T\Sigma_X\hat{w})^p+\sum_{p=0}^{\infty}\hat{\sigma_p}^2(\hat{w}^T\Sigma_X\hat{w})^p \\
= \;& \sum_{p=0}^{\infty}\hat{\sigma}_p^2\left(\underbrace{(w^T\Sigma_Xw)^p-2(w^T\Sigma_X\hat{w})^p+(\hat{w}^T\Sigma_X\hat{w})^p}_{D_p(w,\hat{w})}\right)
\end{aligned}$$

Now we can see that $D_0(w,\hat{w})=0$. $D_1(w,\hat{w})=w^T\Sigma_Xw-2w^T\Sigma_X\hat{w}+\hat{w}^T\Sigma_Xw=(w-\hat{w})^T\Sigma_X(w-\hat{w})$, is just the objective. The following lemma gives the minimizer of $D_p(w,\hat{w})$ for higher order p.

Lemma 3 *If $\hat{w}^*$ satisfies $\hat{w}^*\Sigma_X(\hat{w}-w)=0$ and*

$$\hat{w}^*=\arg\min_{\hat{s}\in\mathcal{W}}D_1(w,\hat{w})$$

for some constrained set $\mathcal{W}$. Then for any $p\geq 2$ and even, we have

$$\hat{w}^*=\arg\min_{\hat{w}\in\mathcal{W}}D_p(w,\hat{w})$$

Since the coefficients $\hat{\sigma}_p$ is zero for $p\geq 3$ and odd, so if a compressed weight $\hat{w}$ satisfied $\hat{w}\Sigma_X(\hat{w}-w)=0$ and minimizes $D_1(\hat{w},w)=(\hat{w}-w)^T\Sigma_X(\hat{w}-w)$, then it is the minimizer for all $D_p(w,\hat{w})$ for even p, therefore a minimizer of the MSE loss.

B.1. Proof of Lemma 3

For simplicity of notation, define $A=w^T\Sigma_Xw$, $B=\hat{w}^T\Sigma_X(\hat{w}-w)$ and $C=D_1(w,\hat{w})=(\hat{w}-w)^T\Sigma_X(\hat{w}-w)$. For all compressors, we have $C\leq A$. Therefore, $w^T\Sigma_X\hat{w}=A+B-C$ and $\hat{w}^T\Sigma_X\hat{w}=A+2B-C$. So

$$D_p(w,\hat{w}) \quad = \quad A^p-2(A+B-C)^p+(A+2B-C)^p$$

First notice that

$$\frac{\partial D_p(w,\hat{w})}{\partial B}=2p((A+2B-C)^{p-1}-(A+B-C)^{p-1}).$$

For even $p\geq 2$, x^{p-1} is monotonically increasing, so $(A+2B-C)^{p-1}>(A+B-C)^{p-1}$ if $B>0$ and vice versa. Therefore, for fixed A and C, $D_p(w,\hat{w})$ is monotonically increasing for positive B and decreasing for negative B. Therefore, $D_p(w,\hat{w})$ is minimized when $B=0$, and the minimal value is $D_p(w,\hat{w})=A^p-2(A-C)^p+(A-C)^p=A^p-(A-C)^p$, which is monotonically increasing with respect to C. So if $\hat{w}^*$ satisfies $B=0$ and is a minimzer of $C=D_1(w,\hat{w})$, it is also a minimizer for $D_p(w,\hat{w})$ for all $p\geq 2$ and even.

C. Details of the experiments

In this appendix, we give some details of the experiment and additional experiments which are omitted in the main text.

C.1. Additional experiment results

We present the experiment results for CIFAR100 here, due to page limit of the main text.

In Figure 7 and Figure 8, we show the result for unsupervised pruning and quantization, introduced in Section 6.1. We can see that, similar to the experiments of MNIST and CIFAR10, the proposed objectives gives better accuracy and smaller loss than the baseline.

In Figure 9 and Figure 10, we show the result for supervised pruning and quantization, introduced in Section 6.2. Due to the slow running speed for estimating the Hessian $\nabla^2_{w_i}\mathcal{L}_w(x, y)$, we only compare two objectives — baseline and gradient. It is shown that the gradient objective gives better accuracy and smaller loss.

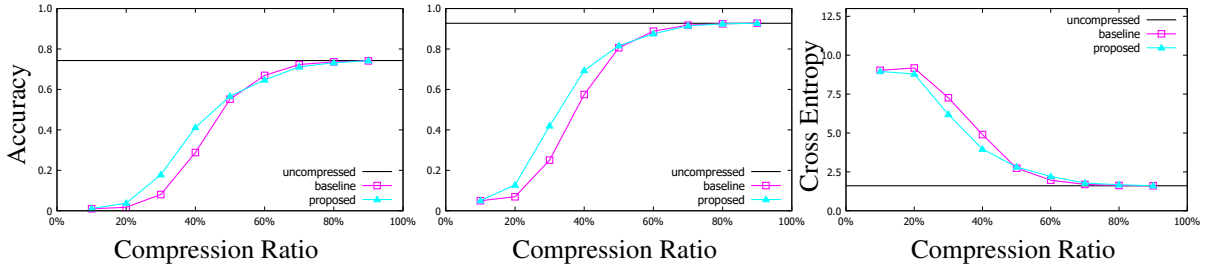

Figure 7. Result for unsupervised pruning experiment for CIFAR 100 experiment. Left: top-1 accuracy. Middle: top-5 accuracy. Right: cross entropy loss.

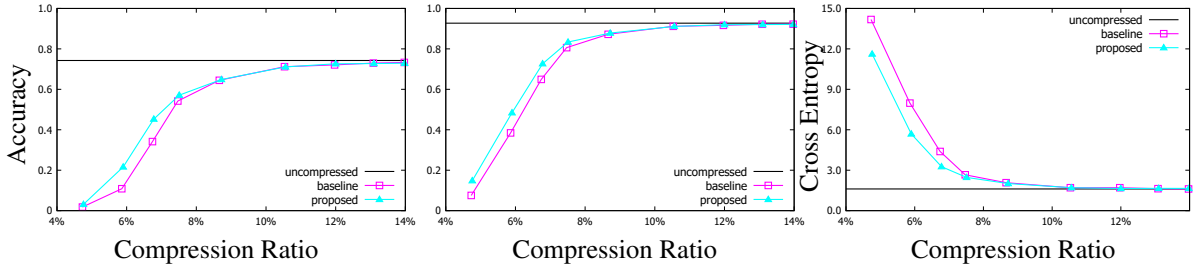

Figure 8. Result for unsupervised quantization experiment for CIFAR 100 experiment. Left: top-1 accuracy. Middle: top-5 accuracy. Right: cross entropy loss.

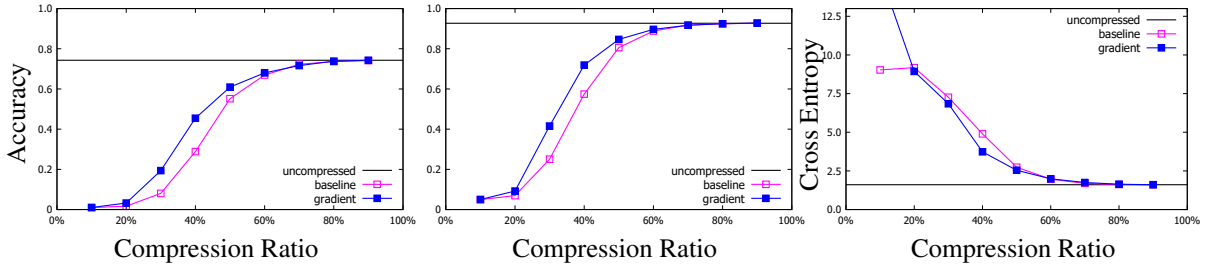

Figure 9. Result for supervised pruning experiment for CIFAR 100 experiment. Left: top-1 accuracy. Middle: top-5 accuracy. Right: cross entropy loss.

C.2. Algorithm for finding optimal quantization

We present a variation of k-means algorithm which are used to find the optimal quantization for the following objective,

$$\min_{c_1,\ldots,c_k,A\in[k]^m} \sum_{i=1}^{m} \left(I_i(w_i - c_{A_i})^2 + H_i(w_i - c_{A_i})^4\right)$$

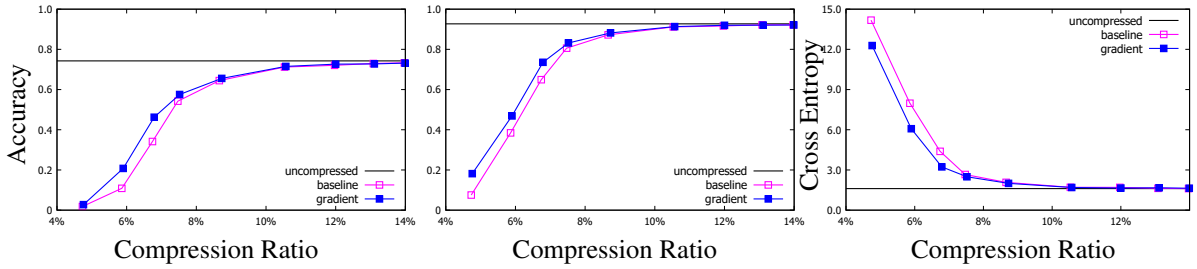

Figure 10. Result for supervised quantization experiment for CIFAR 100 experiment. Left: top-1 accuracy. Middle: top-5 accuracy. Right: cross entropy loss.

where I_i is positive weight importance for quadratic term and H_i is positive weight importance for quartic term. Basic idea of the algorithm is — the assignment step finds the optimal assignment given fixed centroids, and the update step finds the optimal centroids given fixed assignments. This is used for gradient+hessian objective in Section 6.2.

Algorithm 2 Quartic weighted k-means

input Weights $\{w_1, \dots, w_m\}$, weight importances $\{I_1, \dots, I_m\}$, quartic weight importances $\{H_1, \dots, H_m\}$, number of clusters k, iterations T
  Initialize the centroid of k clusters $\{c_1^{(0)}, \dots, c_k^{(0)}\}$
  for $t = 1$ to T **do**
    Assignment step:
    for $i = 1$ to m **do**
      Assign w_i to the nearest cluster centroid, i.e. $A_i^{(t)} = \arg\min_{j \in [k]} (w_i - c_j^{(t-1)})^2$.
    end for
    Update step:
    for $j = 1$ to k **do**
      Find the only real root x^* of the cubic equation

$$(\sum_{i:A_i^{(t)}=j} 4H_i)x^3 - (\sum_{i:A_i^{(t)}=j} 12H_i w_i)x^2 + (\sum_{i:A_i^{(t)}=j} (12H_i w_i^2 + 2I_i))x - (\sum_{i:A_i^{(t)}=j} (4H_i w_i^3 + 2I_i w_i)) = 0$$

      Update the cluster centroids $c_j^{(t)}$ be the real root x^*.
    end for
  end for
output Centroids $\{c_1^{(T)}, \dots, c_k^{(T)}\}$ and assignments $A^{(T)} \in [k]^m$.

Here we show that the cubic equation in Algorithm 2 has only one real root. It was know that if the determinant $\Delta_0 = b^2 - 3ac$ of a cubic equation $ax^3 + bx^2 + cx + d = 0$ is negative, then the cubic equation is strictly increasing or decreasing, hence only have one real root. Now we show that the determinant is negative in this case (we drop the subsripts of the summation for simplicity).

$$\begin{aligned} \Delta_0 &= (\sum 12H_i w_i)^2 - 3(\sum 4H_i)(\sum 12H_i w_i^2 + 2I_i) \\ &= 144\left((\sum H_i w_i)^2 - (\sum H_i)(\sum H_i w_i^2)\right) - 24(\sum H_i)(\sum I_i) \end{aligned}$$

The first term is non-positive because of Cauchy-Schwarz inequality. The second term is negative since H_i's and I_i's are all positive. Hence the determinant is negative.

C.3. Effects of hyperparameters

Here we briefly talk about the hyperparameters used in estimating the gradients $\mathbb{E}[\nabla_{w_i} \mathcal{L}_w(X, Y)]$ and hessians $\mathbb{E}[\nabla^2_{w_i} \mathcal{L}_w(X, Y)]$.

C.3.1. TEMPERATURE SCALING METHOD

The temperature scaling method proposed by (Guo et al., 2017), aims to improve the confidence calibration of a classification model. Denote $z_w(x) \in \mathbb{R}^C$ is the output of the neural network, and classical softmax gives $f_w^{(c)}(x) = \frac{\exp\{z_w^{(c)}(x)\}}{\sum_{c\in C}\exp\{z_w^{(c)}(x)\}}$. The temperature sclaed softmax gives

$$f_w^{(c)}(x) = \frac{\exp\{z_w^{(c)}(x)/T\}}{\sum_{c\in C}\exp\{z_w^{(c)}(x)/T\}}$$

by choosing different T, the prediction of the model does not change, but the cross entropy loss may change. Hence, we can finetune T to get a better model calibration. In our experiment, we found that in MNIST experiment, the model is poorly calibrated. Hence, the variance of estimating gradient and hessian is very large. To solve this, we adopt a temperature $T > 1$ such that the loss from correctly-predicted data can also be backpropagated.

In Figure 11, we show the effect of T for supervised pruning for MNIST. We can see that as T increases from 1, the performance become better at first, then become worse. In our experiment, we choose $T \in \{1.0, 2.0, \ldots, 9.0\}$ which gives best accuracy.

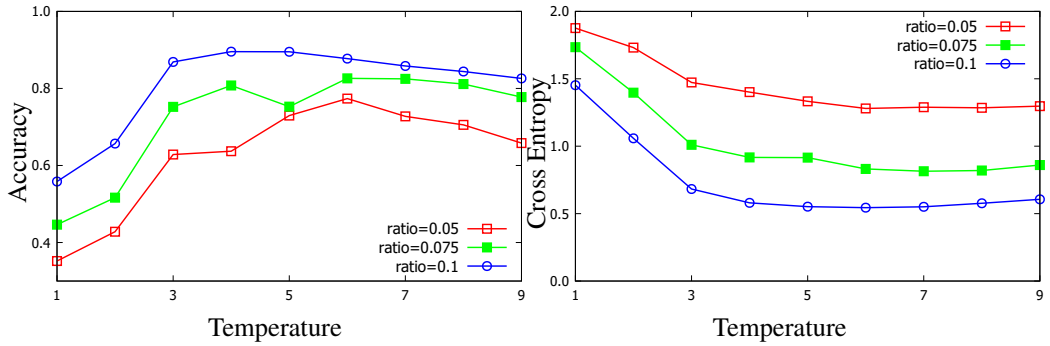

Figure 11. Effect of the temperature T. Left: accuracy of supervised pruning for MNIST. Right: cross entropy of supervised pruning for MNIST. Different lines denote different compression ratio $\in \{0.05, 0.075, 0.1\}$

C.3.2. REGULARIZER OF HESSIAN

In the experiments, we estimate the hessians $\mathbb{E}[\nabla^2_{w_i}\mathcal{L}_w(X, Y)]$ using the curvature propagation algorithm (Martens et al., 2012). However, due to the sparsity introduced by ReLU, there are many zero entries of the estimated hessians, which hurts the performance of the algorithm. Hence, we add a constant $\mu > 0$ to the estimated hessians. In Figure 12, we show that effect of μ for supervised pruning for CIFAR10. We can see that as μ increases from 0, the performance increase first then decrease. We use simple binary search to find the best μ.

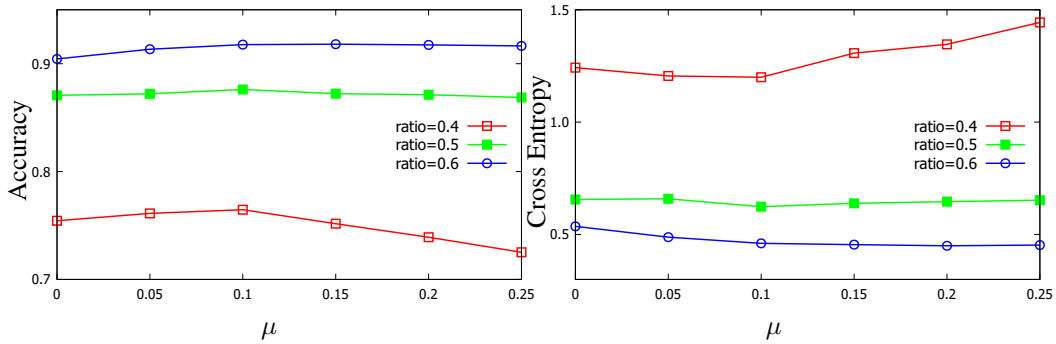

Figure 12. Effect of the regularizer μ. Left: accuracy of supervised pruning for CIFAR10. Right: cross entropy of supervised pruning for CIFAR10. Different lines denote different compression ratio $\in \{0.4, 0.5, 0.6\}$