Optimization for inference

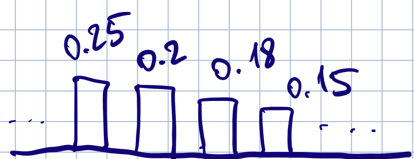
- Calibration & uncertainty estimation
- Distillation
- Pruning
- Quantization

Calibration

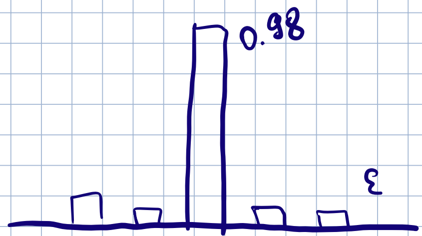
$$p_i = \frac{\exp(x_i)}{\sum_{k=1}^{C} \exp(x_k)}$$

x – logits
p – probabilities

confident prediction

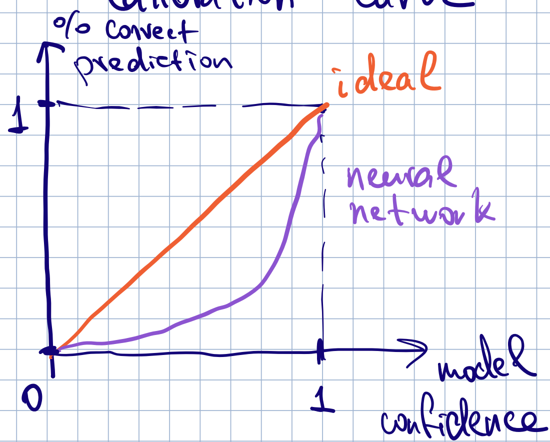


unconfident prediction



reality

Calibration curve



- Temperature softmax $(\tau > 0)$

$$p_i^{\tau} = \frac{\exp(x_i/\tau)}{\sum_{k=1}^{C} \exp(x_k/\tau)}$$

$i = \arg\max x$

1) $\tau \to 0 \qquad p^{\tau} \to \delta(\arg\max x) = (0, 0, .., 0, 1, 0, .., 0)$

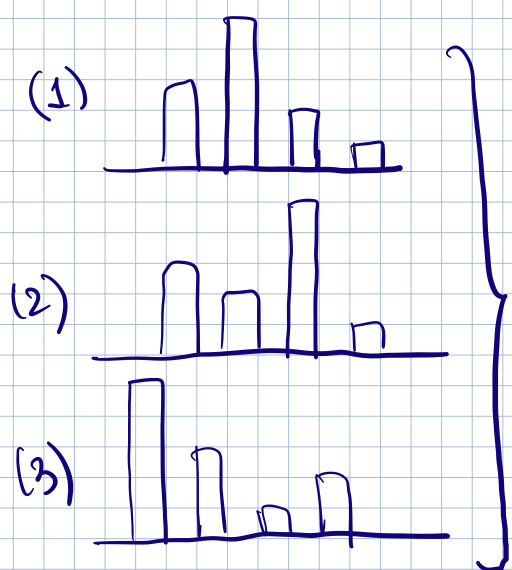
2) $\tau \to \infty \qquad p^{\tau} \to \mathcal{U}(\{1, \ldots, C\})$

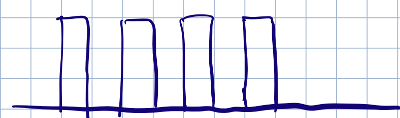
$\tau > 1$ usually in practice

$$\tau^* = \arg\min_{\tau} \underbrace{\left(- \sum_{n=1}^{N} \sum_{k=1}^{C} [y_n = k] \log p_{nk}^{\tau} \right)}_{\text{on validation set}}$$

- Ensemble (Deep Ensemble)

$$p_i^{ens} = \frac{1}{R} \sum_{r=1}^{R} p_i^r \qquad (R = 3)$$





(Knowledge) Distillation

p^T_{nk} – teacher probability
p^S_{nk} – student probability

$\Big\}$ with temperature

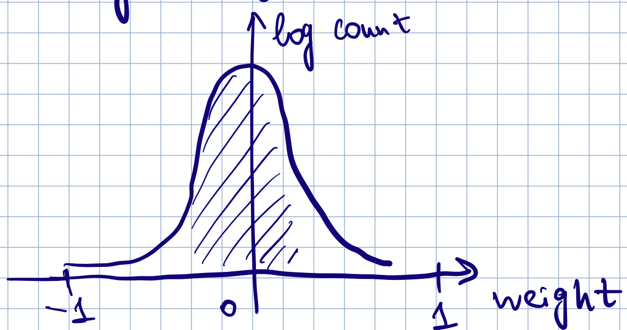
CE loss : $\mathcal{L}_{CE} = -\sum_{n=1}^{N} \sum_{k=1}^{C} [y_n = k] \log p^S_{nk}$

Distill. loss : $\mathcal{L}_D = -\sum_{n=1}^{N} \sum_{k=1}^{C} p^T_{nk} \log p^S_{nk}$

$$\mathcal{L} = \alpha \cdot \mathcal{L}_D + (1-\alpha)\mathcal{L}_{CE}, \quad \alpha \in (0,1)$$

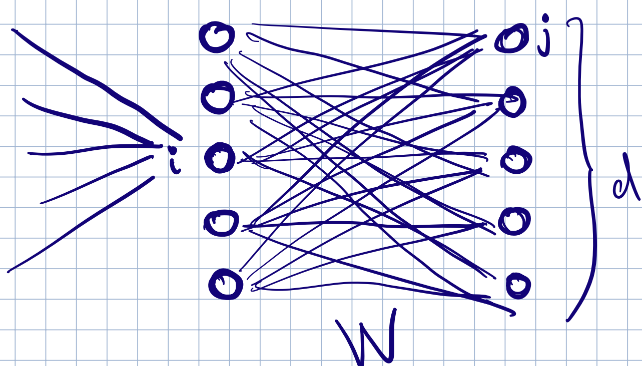
Pruning

Weights distribution



drop α% weights with lowest magnitude
– unstructured pruning

– structured pruning



$$\rho^\alpha_i = \sqrt[\alpha]{\sum_{j=1}^{d} |w_{ij}|^\alpha}$$

Several pruning stages:

1. Drop $\alpha\%$ weight
2. Fine-tune pruned network

SVD of linear layers

$W \in \mathbb{R}^{d_1 \times d_2}$

$W = U \Sigma V$ - SVD of weight matrix

$U \in \mathbb{R}^{d_1 \times s}$ $\quad \Sigma \in \mathbb{R}^{s \times s}$

$V \in \mathbb{R}^{s \times d_2}$

Quantization

float 32 $\to$ int8

$W \in \mathbb{R}^{d_1 \times d_2}$ – weight matrix

w – one scalar weight (float 32)

$$w = s_w (w^q - z_w)$$

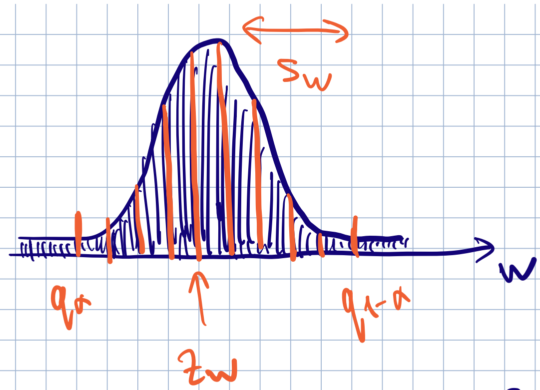
s_w – float 32 – scale parameter

z_w – int 8 – zero parameter

w^q – int 8 – quantized weight

$$w^q = z_w + \frac{w}{s_w}$$

$$w^q = \text{clip}\left(\text{round}\left(\frac{w}{s_w}\right) + z_w, -128, 127\right)$$



$q_\alpha \to -128$

$q_{1-\alpha} \to 127$

$X \in \mathbb{R}^{B\times d_1} \qquad Y \in \mathbb{R}^{B\times d_2} \qquad W \in \mathbb{R}^{d_1\times d_2}$

$Y = XW \qquad Y_{ij} = \sum_{k=1}^{d_1} X_{ik}\cdot W_{kj}$

$\boxed{S_x, Z_x \;;\; S_y, Z_y \;;\; S_w, Z_w}$

$$Y_{ij} = \sum_{k=1}^{d_1} S_x\left(X^q_{ik} - Z_x\right)\cdot S_w\left(W^q_{kj} - Z_w\right) =$$

$$= S_x S_w \Bigg[\underbrace{\sum_{k=1}^{d_1} X^q_{ik}\cdot W^q_{ik}}_{= \, S_y\left(Y^q_{ij} - Z_y\right)} - Z_x \sum_{k=1}^{d_1} W^q_{kj} - Z_w \sum_{k=1}^{d_1} X^q_{ik} + d_1 Z_x Z_w \Bigg]$$

matrix mult in int8