Low Precision Using Unsigned 8-bit Integer

In this section, we describe the low representation of a generic GEMM in the form of $b = Ax$. We define shifted and scaled elements $\hat{A}$, $\hat{x}$, and $\hat{b}$ as:

$$\begin{aligned}
A &= \sigma_A \hat{A} + \mu_A 1_A \\
x &= \sigma_x \hat{x} + \mu_x 1_x \\
b &= \sigma_b \hat{b} + \mu_b 1_b
\end{aligned}$$

Using this affine transformation we can write $b = Ax$ as:

$$\hat{b} = \frac{\sigma_A \sigma_x}{\sigma_b} \left((\hat{A} + \frac{\mu_A}{\sigma_A} 1_A)(\hat{x} + \frac{\mu_x}{\sigma_x} 1_x) - \frac{\mu_b}{\sigma_A \sigma_x} 1_b \right) \tag{1}$$

So far, everything was exact and there were no approximations. Now we assume that $\hat{A}$, $\hat{x}$, and $\hat{b}$ are represented using unsigned 8-bit integer, and therefore

$$\hat{A}, \hat{x}, \hat{b} \in \{0, 1, \cdots, 255\}.$$

Using this representation, we need to set μs and σs such that the quantization error is minimum. We do not go into details of how to optimize these values here but one obvious choice (but not necessarily optimized) is

$$\begin{aligned}
\mu_A &= \min(A) \\
\sigma_A &= \left(\max(A) - \min(A)\right) / 2^8
\end{aligned}$$

for A. Using these parameters, $\hat{A}$ can be computed as

$$\hat{A} = \text{uint8} \left(\frac{A - \min(A)}{\max(A) - \min(A)} \times 2^8 \right)$$

where uint8 is the cast operation to an unsigned 8-bit integer with proper overflow and underflow.

gemmlowp as well as farm calculate matrix multiplications in the form

$$\hat{b} = \frac{\gamma}{2^e} \left((\hat{A} + \alpha 1_A)(\hat{x} + \zeta 1_x) + \beta 1_b \right) \tag{2}$$

where α, ζ, β, γ, and e are all 32-bit integers. Matching Equations 1 and 2, we get

$$
\begin{aligned}
\frac{\gamma}{2^e} &= \frac{\sigma_A \sigma_x}{\sigma_b} \\
\alpha &= \frac{\mu_A}{\sigma_A} \\
\zeta &= \frac{\mu_x}{\sigma_x} \\
\beta &= \frac{\mu_b}{\sigma_A \sigma_x}
\end{aligned}
$$

The above equations impose a soft constraint on values of μs and σs, i.e., they should be chosen such that α, ζ, β, γ, and e can be represented using 32-bit integer.

Please note that the α, ζ, β, γ, and e here correspond to variables **lhs_offset**, **rhs_offset**, **result_offset**, **result_mult_int**, and **result_shift** in the farm library, respectively.