We thank R1, R2, R3 and R4 for their insightful comments on our paper. We appreciate that the reviewers noted the novelty of our quantization decomposition (R1, R3) and the relevance of combining pruning and quantization (R2, R3, R4). We were encouraged that the reviewers found the description of our method well-written and easy to follow (R1, R3, R4) and we are glad that the experiments were convincing (R3, R4). We will extend our discussion on the broader impact of our work. In the remainder of our response we have grouped the responses to the reviewers appropriately.

Novelty and practical usefulness of Bayesian Bits (BB) R2 argued against the novelty and practical usefulness of BB. We would like to mention that the main novelty of BB is the decomposition of the quantization operation in terms of hardware friendly bit widths. Through this we can introduce learnable stochastic gates and eventually arrive at an objective reminiscent of the one from Louizos et. al (2018) (which is a special case). Furthermore, *BB is a practical and efficient method* for learning mixed-precision networks. Since we use per weight / feature map tensor quantization, we only need 4 extra parameters *per tensor of weights and feature maps*, and one extra parameter for the z_2 gates for each output channel. For example: for LeNet5, the BB parameters constitute only 636 of 583k total parameters.

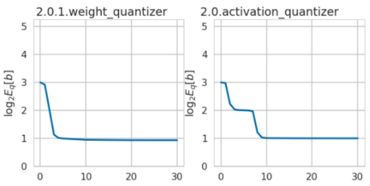
Distribution choices for BB We would like to address questions from R2 and R4 about **(1)** the distribution choices we have in BB, the autoregressive (AR) prior (p) and variational posterior (q), **(2)** designing priors for a target BOP cont and **(3)** the independence of q across layers. For **(1)**, we would like to point out that the higher bit-gates (e.g., z_8) contribute to explaining the data only when the previous ones are switched on (since they interact multiplicatively via $z_2 z_4 z_8$). As a result, if we adopt independent distributions for p, q (and the resulting regularization term) when either z_2 or z_4 is switched off then z_8 only receives a gradient to reduce its associated probability (which, empirically, was an issue). With the AR structure we prevent this over-regularization as, conditioned on an earlier gate being zero, both p and q put zero mass in activating the subsequent gates thus no regularization is happening. As for **(2)**, we concede that it is difficult to target a specific BOP count with the prior. In practice, one would experiment with a range of regularization strengths to generate a Pareto curve, and pick a model that matches the requirements. Finally for **(3)**, notice that while q is independent across layers, as soon as we see data, the choice of the bit width in the learning procedure becomes dependent due to the interactions of the bit widths through the intermediate hidden representations.

Non-doubling bit widths and gating functions R3 asked about extending BB to arbitrary bit widths as well as alternative gating functions. The first is indeed possible, but requires modifications to eqs (1)-(6) in our paper, since the equality under equation (3) does not hold anymore. Binary bit widths could be supported as is but the possible values would be $\{-s_1, 0\}$ for signed values and $\{0, s_1\}$ for unsigned values (which differs from the usual approach to use the sign function). Furthermore, including all possible bit widths would incur a large compute overhead, due to the computation of 32 (instead of 4) residual error tensors. At the request of R3 we also ran the toy experiments with Gumbel-softmax gates. The results are essentially matched (MNIST: 0.38% BOPs / 99.38% acc; CIFAR10: 0.44% BOPs / 93.19% acc), but on CIFAR10 they sometimes diverge (which was not the case with the hard-concrete gates).

BOP metric and hardware timing R1 refers to the Bit OPeration (BOP) metric as confusing, and R4 inquired about timing numbers on real hardware. The BOP metric is commonly used in mixed-precision works (e.g. [33] eq. 12), and serves as an approximation of the number of bit-level operations required to perform a forward pass. We believe the comparison with this metric is fair, as it takes into account hypothetical, hardware agnostic, speed-ups from both pruning and the mixed-precision quantization bit widths on an optimally designed device. Of course, reductions on BOPs do not match 1-to-1 to speedups, as it doesn't take into account data transfer and the specifics of the runtime environment. Taking all of these into consideration is a topic that we aim to tackle in future work.

Evaluation and comparison to literature We respectfully disagree with R1 that the improvements of our method are merely marginal. Note that PACT and LSQ were chosen as *strong and difficult to beat* baselines. Compared to LSQ 4/4 (8 in/out) we achieve a 0.5% increase in accuracy for similar BOPs, while compared to LSQ 4/4 we achieve a 7.5% relative reduction in BOPs at similar accuracy. Furthermore, the PACT (hypothetical, 8 in/out) results serve as a hypothetical upper bound on PACT performance (details in Table 4), yet are still outperformed by our method. Additionally, R2 inquires about ResNet50 on CIFAR10 and comparison to binary nets and pruning. Since the CIFAR10 experiments serve as initial validation, and BB outperforms the baselines, we see no value in ResNet50. Most binary approaches perform similarly on MNIST, but worse on CIFAR10: Peters and Welling (2018) achieve 88.61% on the same architecture, Courbariaux et al. (2015) report 90.1% (vs BB 93.23%) on a larger network and Meng et al. (2020) reach comparable performance to BB but with full precision activations. Pruning FP32 networks gives worse BOPs.

Plots and ablation studies Note that Figure 2(a) in our paper contains the quantization-only ablation study requested by R1 (fixed $z_2 = 1$; 'BB quantization only'). Furthermore, this figure contains two ablation studies in which we fix the bit widths of the network and only apply the pruning aspect of BB. Comparing these results we see that full BB yields improved efficiency vs accuracy trade-offs compared to quantization only or pruning of fixed quantized networks. We will update Table 4 in the appendix to include the results to the ablation studies as well. Figure 10 (supplementary material) contains the plot requested by R4 in which the co-evolution of BOPs and accuracy is shown. Lastly, R3 inquired how bit widths vary during training. In the plot we see two characteristic evolutions of $\log_2 \mathbb{E}_q[\texttt{bit width}]$. The ratio of smooth to abrupt changes depends on regularization strength and network.


2.0.1.weight_quantizer
2.0.activation_quantizer
$\log_2 E_q[b]$
$\log_2 E_q[b]$