We are deeply appreciative of the reviewers for their feedback amidst these trying circumstances. We are glad that the reviewers appreciate the scalability challenge addressed by our work, and the neatness of our proposed algorithm. We would like to emphasize that we compared with over 1056 benchmarks arising from the domain of neural network verification.

Reviewer 3

Comparison with WeightMC We indeed considered evaluating these benchmarks with WeightMC [1], as the reviewer suggests, and found WeightMC unsuitable for our benchmarks for the following reason. WeightMC can only handle benchmarks with a small *tilt*, which is the ratio of the largest weight assignment to the smallest weight assignment. In particular, WeightMC requires an upper bound r on the tilt to be provided as input; the running time of WeightMC is then linear in r (see Theorem 2 of [1]). In particular, the runtime is lower bounded by the minimum of tilt and 2^n, where n is the size of sampling set. The bound on tilt, however, is extremely large for the benchmarks that we consider. For example, in Experiment 1 a benchmark with a sampling set of n variables has tilt (a priori) upper bounded by $\frac{(2/3)^n}{(1/3)^n} = 2^n$; the bound on tilt for our benchmarks is never smaller than 2^{66}. Therefore, WeightMC can not handle any of the benchmarks. To summarize, DeWeight is indeed the state of the art technique for benchmarks with large tilt. To the best of our knowledge, we are not aware of any practical applications of discrete integration that have small tilt.

Projection on few variables Due to the performance of the underlying approximate model counter, our methods is currently limited to work well only for problems where the problem can be projected on 100 or fewer variables (note however that our method can handle many – on the order 10^5 – variables that are not in the sampling set). Even with this limitation, DeWeight is the only technique that is able to handle such benchmarks with large tilt. Furthermore, we trust that improvements in the underlying approximate model counter, which is itself a highly active research area, will allow for larger sampling sets in the future.

Other Weight Settings As we showed in Theorem 1, the performance of our method depends heavily on the total number of bits required for the weight description. Thus one of the limitations our approach is that this total number of bits should be relatively small (say, for up to 3 decimal places). As our experiments indicate, the prior state of the art approaches are unable to handle weights with even one decimal place.

Limitations We will make it clear that we focus on benchmarks arising from the verification of binarized neural networks. Please note that our benchmark suite comprises of over 1000 formulas, a fairly large dataset comprising of three different applications: robustness, trojan attack effectiveness, and fairness. Furthermore, since the underlying approximate counter, ApproxMC, has been shown to be used in wide variety of benchmarks and given DeWeight's runtime is dictated by ApproxMC's runtime performance, we expect DeWeight to be useful in other contexts as well.

Reviewer 4

As mentioned on line 219, we tested our tool on 1056 formulas arising from the domain of neural network verification. These formulas evaluate robustness, trojan attack effectiveness, and fairness of a binarized neural network. Our experiments showed how on these benchmarks we outperform state-of-the-art tools. Please note that we have compared the tool with the state of the art techniques (as mentioned, WISH and WeightMC can not handle these instances, and therefore, we did not add plots corresponding to WISH and WeightMC).

References

[1] S. Chakraborty, D. J. Fremont, K. S. Meel, S. A. Seshia, and M. Y. Vardi. Distribution-aware sampling and weighted model counting for SAT. In *Proc. of AAAI*, pages 1722–1730, 2014.