We would like to thank the reviewers for their comments, and take the opportunity to answer their questions below.

R1: (1) We thank the reviewer for the relevant [Amari et al., 2000] reference, which we will cite and discuss. To our knowledge, the first reference to suggest a similar procedure for approximating IHVP is (Hassibi & Stork, 1993), cited as [10], in the context of pruning. We repeatedly acknowledged (e.g. lines 49-51, 195-197) that our contribution is **not** in introducing this technique, but in considering its applicability, accuracy, and efficiency in the context of pruning modern deep networks, for which we show state-of-the-art results. ([10] considered pruning single-layer neural networks with at most 65 parameters on small datasets. Similarly, [Amari et al., 2000] considers single-layer networks with < 50 parameters.) Further, we examined the method's accuracy relative to recent techniques, and extended it to account for first-order information. **(2)** We are open to changing the term "WoodFisher" which we used as a mnemonic and to simplify the differentiation between variants (e.g. WoodFisher -> WoodTaylor). We are confident that changing the name and discussing [Amari et al., 2000] would not require a significant revision. We gently ask the reviewer to reconsider their score in view of this. **(3)** By applicability, we mean if the methods can be applied to any network type (see Appendix S2 for more). Also, we will provide additional technical details in Section 4 and address the other comments.

(a) (b)

Figure 1: (a) WoodFisher vs K-FAC (MLPNET, MNIST) (b) Effect of λ: WoodFisher (RESNET-20, CIFAR10) (avg over 4 seeds)

R2: (1) Regarding λ, we selected a small value so that the Hessian is not dominated by the dampening. We note that the algorithm is largely insensitive to this dampening value, Fig 1b. We did not perform an exhaustive hyperparameter search for λ: we identified a small value (1e-5) which worked, and adopted it across all our experiments without further tuning. Indeed, huge λ would approximate global magnitude pruning. But, it is not exactly the same as simply tuning this λ alone, as that would only control the diagonal. **(2)** Fisher minibatch size: This is a practical trick which allows us to utilize more samples at the same cost. Please see Appendix S5 for ablation studies. **(3)** We only use multiple Hessian inverses for the one-shot pruning experiment on ImageNet. We did not use it in gradual pruning, so our comparisons are completely fair. **(4)** Comparison to K-FAC: that's a great catch. To address this comparison issue, we have implemented K-FAC-based pruning, and found that it is clearly outperformed by our method, even on a simple MLP example (see Figure 1a). For convolutional layers, K-FAC needs to make additional approximations, so we can expect the results to further improve. **(5)** For simplicity, we consider the scaling constant as 1 here. Fig 3 illustrates the possible cases where the quadratic model $\{$over, under, closely$\}$ - estimates the loss based on λ. Note, pruning is independent of this scaling constant. **(6)** We believe our self-tuning is fairly limited: we simply adopted the hyperparameter combination which works best for magnitude pruning (as followed in the literature) and plugged in WoodFisher as the estimator. In fact, post-submission, we noticed that some hyperparameter values (e.g. weight decay) can be further tuned to give an additional boost to our method (at 89% sparsity, MOBILENETV1 results increase from $63.87 \rightarrow 64.59$, etc.). Plus, we do not use "additional tricks" such as label smoothing or cosine LR schedules, which can further help accuracy (see STR [26]). **(7)** We will carefully follow your notes on presentation.

R3: Thanks for the suggestions, we will correct the font sizes & the broken references. (Lines 522-524 in the Appendix contain the corrected references.) Please see the K-FAC results in Fig 1a, which we will add in the paper too.

R4: (1) Theoretical guarantees: When the model and data distribution match, the (true) Fisher and Hessian are indeed equivalent. Our visual analysis of the Hessian and empirical Fisher serves to showcase the approximation quality under relaxed conditions that arise in practice. Further, in the loss analysis on ResNet-20, we observe that empirical Fisher can estimate the loss in a local neighbourhood remarkably well. We are very interested in developing proper theoretical guarantees to the approximation quality in future work; the WoodTaylor variant in fact came out of this effort. **(2)** Our Hessian approximation wouldn't make sense in the interpolation setting, where the gradient at each sample is already zero. But this is an artificial example, which is unlikely to arise in practice. **(3)** Block size is calculated based on model size and CPU memory. We always seek to maximize block size; c.f. Fig. S10 for the ablation study. **(4)** Relative to OBS, we apply a similar method to modern-day networks such as ResNet50, while [10] was applied only to a single hidden layer MLP. This is made possible by the combination of techniques: block-wise chunking, mini-batch gradients in empirical Fisher to utilize more samples, and proper book-keeping to enable this implementation in PyTorch. In addition, we incorporate the first-order term ignored by [10] and most of the literature (WoodTaylor), but can be vital in the dynamic pruning setting (see results in Fig S14). Lastly, in Fig S1, we show that we outperform L-OBS [32] which is a recent attempt to make [10] practically viable by defining separate layer-wise losses.