Supplementary Materials for the Submission: Estimating Model Uncertainty of Neural Networks in Sparse Information Form

Jongseok Lee [1] **Matthias Humt** [1] **Jianxiang Feng** [1] **Rudolph Triebel** [1 2]

1. Organization of the document

This document is organized as follows. Firstly, the mathematical derivations can be found in section 2. Then, we present the theoretical analysis (section 3) followed by their proofs (section 4). Implementation details and further results are presented in section 5 and 6 respectively. In particular, following additional results can be found.

- Empirical evidence on spectral sparsity of information matrix (section 6.1).
- Effects of low rank approximation on approximation quality of information matrix (section 6.2).
- Effects of diagonal correction, data-set size, and critical review on KFAC on toy data (section 6.3).
- Effects of hyperparameters N and τ (section 6.4).
- Different architectures on ImageNet data-set and a time complexity analysis (section 6.5).

2. Mathematical Derivations

2.1. Problem Statement

Let us assume that DNNs parameter posterior is estimated with MND layer-wise. Without diagonal approximation or Kronecker factorization of the covariance matrix (or similarly IM), the computational complexity of several operations namely storage, inversion and Cholesky decomposition becomes intractable. For example, if there exists a layer with 1 million parameters, the storage of covariance alone scales quadratic. If we use the simple formula for back-of-the envelope computations: total RAM for an $N \times N$ double precision matrix requires $N^2 * \frac{8}{10^9}$ gigabytes, storing a covariance matrix for $N = 1000000$ requires 8000 gigabytes, which is computationally intractable for most of modern computers. Consequently, cubic in cost operations such as inversion or Cholesky decomposition is not feasible in a naive strategy. This is a reason why the current approaches use diagonal approximation or Kronecker factorization of the covariance matrix if MND is the chosen posterior family.

Table 1. **Bounds on space complexity when compared to a naive strategy.** We compare the several operations with a naive strategy to our presented derivations, which is the main results of our work. Here, $L << N$ where L can be chosen with fidelity vs cost trade-off. This shows the sparse information form as a scalable Gaussian posterior family, providing alternatives to the diagonal covariance assumption or matrix normal distribution.

Operations	**Space Complexity** *Naive Strategy*	*Ours*
Storage	$O(N^2)$	$O(L + na + mg)$
Inversion	$O(N^3)$	$O(L^3)$
Cholesky Decomposition	$O(N^3)$	$O(L^3)$

Following this statement, we list the memory-wise infeasible operations in detail: (i) naively extracting diagonal elements of $(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T$ and $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$, naively storing, inverting and Cholesky decomposing $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D$ (iii) naively storing $(U_A \otimes U_G)$ and performing LRA on $(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T$. Our solution to the first two points are derived in this section while we have proposed an algorithm in the main manuscript to tackle the challenges of the last point.

Table 1 shows the main result on space complexity, where the proposed sparse information form of DNNs posterior is also analyzed. Without resorting to mean-field approximations or matrix normal distribution, we show an alternative form of MND is also possible. Mathematical derivations we present below make this possible. Note that the inversion scheme considered is Gauss–Jordan elimination.

2.2. Diagonal Correction without Full Evaluation

Directly evaluating $U_A \otimes U_G$ may not be computationally feasible for modern DNNs. Therefore, we derive the analytical form of the diagonal elements for $(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T$ without having to fully evaluate the Kronecker product. Let $U_A \in \mathbb{R}^{n \times n}$ and $U_G \in \mathbb{R}^{m \times m}$ be the square matrices. $\Lambda \in \mathbb{R}^{mn \times mn}$ is a diagonal matrix by construction. $V = U_A \otimes U_G \in \mathbb{R}^{mn \times mn}$ is a Kronecker product with ele-

[1]Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Wessling, Germany [2]Computer Vision Group, Technical University of Munich (TU Munich), Garching, Germany. Correspondence to: Jongseok Lee <jongseok.lee@dlr.de>.

Proceedings of the 37th International Conference on Machine Learning, Online, PMLR 119, 2020.

ments $v_{i,j}$ with $i = m(\alpha-1)+\gamma$ and $j = m(\beta-1)+\zeta$ (from definition of Kronecker product). Then, the diagonal entries of $(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T$ can be computed as follows:

$$\Big[(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T\Big]_{ii} = \sum_{j=1}^{nm}(v_{i,j}\sqrt{\Lambda_j})^2 \qquad (1)$$

Derivation: As a first step of the derivation, we express $(A \otimes B)\Lambda(A \otimes B)^T$ in the following form:

$$\begin{aligned}(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T &= (U_A \otimes U_G)\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}(U_A \otimes U_G)^T \\ &= \Big[(U_A \otimes U_G)\Lambda^{\frac{1}{2}}\Big]\Big[(U_A \otimes U_G)\Lambda^{\frac{1}{2}}\Big]^T \\ &= UU^T\end{aligned}$$

Then, $\text{diag}(UU^T)_i = \Big[UU^T\Big]_{ii} = \sum_{j=1}^{nm} u_{ij}^2$ by definition. Now, we let $(U_A \otimes U_G)\Lambda^{\frac{1}{2}} = V\Lambda^{\frac{1}{2}}$ with $\Lambda^{\frac{1}{2}}$ being again a diagonal matrix. Therefore, $u_{ij} = v_{i,j}\sqrt{\Lambda_j}$ due to the multiplication with a diagonal matrix from a right hand side. Substituting back these results in $\Big[(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T\Big]_{ii} = \sum_{j=1}^{nm}(v_{i,j}\sqrt{\Lambda_j})^2$ which completes the derivation. Formulating equation 1 for the non-square matrices (which results after a low rank approximation) such as $U_a \in \mathbb{R}^{n\times a}$ and $U_g \in \mathbb{R}^{m\times g}$ and paralleling this operation are rather trivial and hence, we omit this part of the derivation.

2.3. Low Rank Sampler - Analytical Form

For a full Bayesian analysis which is approximated by a Monte Carlo integration, sampling is a crucial operation for predicting uncertainty. We start by stating the problem.

Problem statement: Consider drawing samples $\theta_t^s = \text{vec}(W_t^s) \in \mathbb{R}^{nm}$ from the proposed sparse information form:

$$\theta_t^s \sim \mathcal{N}^{-1}(\theta_{\text{MAP}}^{IV}, (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D) \qquad (2)$$

Drawing such samples from a covariance form of MND requires finding a symmetrical factor of the covariance matrix (e.g. Chloesky decomposition) which is cubic in cost $O(N^3)$. Even worse, when represented in an information form as in (2), it requires first an inversion of information matrix and then the computation of a symmetrical factor which overall constitutes two operations of cost $O(N^3)$. Clearly, if N lies in a high dimension such as 1 million, even storing is obvious not feasible, let alone the sampling computations. Therefore, we need a sampling computation that (a) keeps the Kronecker structure while sampling so that first, the storage is memory-wise feasible, and then (b) the operations that require cubic cost such as inversion, must be performed in the dimensions of low rank L instead of full parameter dimensions N. We provide the solution below.

Analytical solution: Let us define $X^l \in \mathbb{R}^{mn}$ and $X^s \in \mathbb{R}^{m\times n}$ as the samples from a standard Multivariate Normal Distribution in (3) where we denote the followings: $0_{nm} \in \mathbb{R}^{nm}$, $I_{mn} \in \mathbb{R}^{mn\times mn}$, $0_{n\times m} \in \mathbb{R}^{n\times m}$, $I_n \in \mathbb{R}^{n\times n}$ and $I_m \in \mathbb{R}^{m\times m}$. Note that these sampling operations are cheap.

$$X^l \sim N(0_{nm}, I_{nm}) \;\text{ or }\; X^s \sim \mathcal{MN}(0_{n\times m}, I_n, I_m). \qquad (3)$$

Furthermore, we denote $\theta_t^s = vec(W_t^s) \in \mathbb{R}^{mn}$, $\theta_{\text{MAP}} = vec(W_{\text{MAP}}) \in \mathbb{R}^{mn}$ as a sample from equation 2 and its mean as a vector respectively. We also note that $\Lambda_{1:L} \in \mathbb{R}^{L\times L}$ and $D \in \mathbb{R}^{mn\times mn}$ are the low ranked form of the re-scaled eigen-values and the diagonal correction term as previously defined. $U_a \in \mathbb{R}^{n\times a}$ and $U_g \in \mathbb{R}^{m\times g}$ are the eigenvectors of low ranked eigen-basis so that $n \geq a$, $m \geq g$ and $L = ag$. Then, the samples of 2 can be computed analytically as[1]:

$$\begin{aligned}&\theta_t^s = \theta_{\text{MAP}} + F^c X^l \text{ where,} \\ &F^c = D^{-\frac{1}{2}}\Big(I_{nm} - D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}} \\ &\quad (C^{-1} + V_s^T V_s)^{-1}\Lambda_{1:L}^{\frac{1}{2}}(U_a \otimes U_g)^T D^{-\frac{1}{2}}\Big).\end{aligned} \qquad (4)$$

Firstly, the symmetrical factor $F^c \in \mathbb{R}^{mn\times mn}$ in (4) is a function of matrices that are feasible to store as they involve diagonal matrices or small matrices in a Kronecker structure. Furthermore,

$$\begin{aligned}&V_s = D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}} \\ &C = A_c^{-T}(B_c - I_L)A_c^{-1} \text{ with } A_c \text{ and } B_c\end{aligned} \qquad (5)$$

being the Cholesky decomposed matrices of $V_s^T V_s \in \mathbb{R}^{L\times L}$ and $V_s^T V_s + I_L \in \mathbb{R}^{L\times L}$ such that:

$$\begin{aligned}A_c A_c^T &= V_s^T V_s \text{ and} \\ B_c B_c^T &= V_s^T V_s + I_L.\end{aligned} \qquad (6)$$

Consequently, the matrices in (4) are defined as $C \in \mathbb{R}^{L\times L}$, $(C^{-1} + V_s^T V_s) \in \mathbb{R}^{L\times L}$ and $I_L \in \mathbb{R}^{L\times L}$. In this way, the two operations namely Cholesky decomposition and inversion that are cubic in cost $O(N^3)$ are reduced to the low rank dimension L with complexity $O(L^3)$.

Derivation: Firstly, note that sampling from a standard multivariate Gaussian for X^l or X^s is computationally cheap (see equation 3). Given a symmetrical factor for the covariance $\Sigma = F^c F^{c^T}$ (e.g. by Cholesky decomposition), samples can be drawn via $\theta_{\text{MAP}} + F^c X^l$ as depicted in (4). Our derivation involves finding such symmetrical factor for the given

[1] We show how the Kronecker structure of F^c can be exploited to compute $F^c X^l$ in the derivation only.

form of covariance matrix while exploring the Kronecker structure for the sampling computations so that the space complexity is bounded to $O(L^3)$ instead of $O(N^3)$.

Let us first reformulate the covariance (inverse of information matrix) as follows.

$$\begin{aligned}\Sigma &= \Big((U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D\Big)^{-1} \\ &= \Big[D^{\frac{1}{2}}(D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}} \\ &\Lambda_{1:L}^{\frac{1}{2}}(U_a \otimes U_g)^T D^{-\frac{1}{2}} + I_{nm})D^{\frac{1}{2}}\Big]^{-1} \\ &= D^{-\frac{1}{2}}\Big[((D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}}) \\ &(D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}})^T + I_{nm})\Big]^{-1} D^{-\frac{1}{2}} \\ &= D^{-\frac{1}{2}}\left[V_s V_s^T + I_{nm}\right]^{-1} D^{-\frac{1}{2}}.\end{aligned} \tag{7}$$

Here, we define: $V_s = D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}}$. Now, a symmetrical factor for $\Sigma = F^c F^{c^T}$ can be found by exploiting the above structure. We let W^c be a symmetrical factor for $V_s V_s^T + I_{nm}$ so that $F^c = D^{-\frac{1}{2}} W^{c^{-1}}$ is the symmetrical factor of Σ. Following the work of Ambikasaran & O'Neil (2014) the symmetrical factor W^c can be found using equations:

$$\begin{aligned}W^c &= I_{nm} + V_s C V_s^T \\ C &= A_c^{-T}(B_c - I_L)A_c^{-1}.\end{aligned} \tag{8}$$

Note that A and B are Cholesky decomposed matrices of $V_s^T V_s \in \mathbb{R}^{L\times L}$ and $V_s^T V_s + I_L \in \mathbb{R}^{L\times L}$ respectively. As a first result, this operation is bounded by complexity $O(L^3)$ instead of the full parameter dimension N. Calculations of $V_s^T V_s$ can also be performed in iterations similar to derivations shown in section 2.2. Now the symmetrical factor for Σ can be expressed as follows.

$$\begin{aligned}F^c &= D^{-\frac{1}{2}} W^{-1} = D^{-\frac{1}{2}}(I_{nm} + V_s C V_s^T)^{-1} \\ &= D^{-\frac{1}{2}}\Big(I_{nm} - V_s(C^{-1} + V_s^T V_s)^{-1} V_s^T\Big).\end{aligned} \tag{9}$$

Woodbury's Identity is used here. Now, by substitution:

$$\begin{aligned}&\theta_t^s = \theta_{\text{MAP}} + F^c X^l \text{ where,} \\ &F^c = D^{-\frac{1}{2}}\Big(I_{nm} - V_s(C^{-1} + V_s^T V_s)^{-1} V_s^T\Big) \\ &= D^{-\frac{1}{2}}\Big(I_{nm} - D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}} \\ &(C^{-1} + V_s^T V_s)^{-1}\Lambda_{1:L}^{\frac{1}{2}}(U_a \otimes U_g)^T D^{-\frac{1}{2}}\Big).\end{aligned} \tag{10}$$

This completes the derivation of (4). As a result, the inversion operation is bounded by complexity $O(L^3)$. Furthermore, the derivation constitutes smaller matrices U_a and U_g or diagonal matrices D and I_{mn} which can be stored as vectors. In short the complexity has significantly reduced.

Now we further derive computations that exploits rules of Kronecker products. Consider:

$$\begin{aligned}F^c X^l = D^{-\frac{1}{2}}\Big(&I_{nm} - D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}} \\ &(C^{-1} + V_s^T V_s)^{-1}\Lambda_{1:L}^{\frac{1}{2}}(U_a \otimes U_g)^T D^{-\frac{1}{2}}\Big)X^l.\end{aligned} \tag{11}$$

Then, it follows by defining inverted matrix $L^c = (C^{-1} + V_s^T V_s)^{-1} \in \mathbb{R}^{L\times L}$ with a cost $O(L^3)$:

$$\begin{aligned}F^c X^l = D^{-\frac{1}{2}}\Big(&I_{nm} - D^{-\frac{1}{2}}(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}} L^c \\ &\Lambda_{1:L}^{\frac{1}{2}}(U_a \otimes U_g)^T D^{-\frac{1}{2}}\Big)X^l.\end{aligned} \tag{12}$$

We further reduce this by evaluating $D^{-\frac{1}{2}}$ and defining $X_D^l = D^{-\frac{1}{2}} X^l \in \mathbb{R}^{mn}$ and $P^c = \Lambda_{1:L}^{\frac{1}{2}} L^c \Lambda_{1:L}^{\frac{1}{2}} \in \mathbb{R}^{L\times L}$. We note that this multiplication operation is memory-wise feasible.

$$F^c X^l = X_D^l - \Big(D^{-1}(U_a \otimes U_g)P^c(U_a \otimes U_g)^T X_D^l\Big). \tag{13}$$

Now, we map X_D^l to matrix normal distribution by an unvec$(\cdot)$ operation so that $X_D^s = \text{unvec}(X_D^l) \in \mathbb{R}^{m\times n}$ or equivalently $X_D^l = \text{vec}(X_D^s)$. Using a widely known relation for Kronecker product that is - $(U_a \otimes U_g)^T \text{ vec}(X_D^s) = \text{vec}(U_g^T X_D^s U_a)$, it follows:

$$F^c X^l = X_D^l - \Big(D^{-1}(U_a \otimes U_g)P^c \text{vec}(U_g^T X_D^s U_a)\Big). \tag{14}$$

Note that matrix multiplication is performed with small matrices. Repeating a similar procedure as above we obtain the equation below for $X_P^s = P^c\ (U_a \otimes U_g)^T X_D^l$,

$$\begin{aligned}F^c X^l &= X_D^l - \Big(D^{-1}(U_a \otimes U_g)X_P^s\Big) \\ &= X_D^l - \Big(D^{-1}\text{vec}(U_g X_P^s U_a^T)\Big).\end{aligned} \tag{15}$$

This completes the derivation. Lastly, we provide a remark below to summarize the main points.

Remark: We presented a new derivation to sample from (2), a low-rank and information formulation of MND. This analytical solution ensures (a) $O(N^3) >> O(L^3)$ for Cholesky decomposition, (b) $O(N^3) >> O(L^3)$ for a matrix inversion, (c) storage of small matrices U_g, U_a, a diagonal matrix D and identity matrices and finally (d) matrix multiplications that only involve these matrices. This is a direct benefit of our proposed LRA that preserves Kronecker structure in eigenvectors. Furthermore, this result shows the sparse information form as a new scalable Gaussian posterior family for approximate Bayesian inference.

3. Theoretical Analysis and Results

Some of the interesting theoretical properties are as follows with proofs provided in section 4.

3.1. More Accurate Information Matrix

Theoretical results of adding a diagonal correction term to Kronecker factored eigenbasis are captured below.

Lemma 1: *Let $\boldsymbol{I}$ be the real information matrix, and let $\boldsymbol{I}_{inf}$ and $\boldsymbol{I}_{efb}$ be the INF and EFB estimates of it respectively. It is guaranteed to have $\left\|\boldsymbol{I}-\boldsymbol{I}_{efb}\right\|_F \geq \left\|\boldsymbol{I}-\boldsymbol{I}_{inf}\right\|_F$.*

Corollary 1: *Let $\boldsymbol{I}_{kfac}$ and $\boldsymbol{I}_{inf}$ be KFAC and our estimates of real information matrix $\boldsymbol{I}$ respectively. Then, it is guaranteed to have $\left\|\boldsymbol{I}-\boldsymbol{I}_{kfac}\right\|_F \geq \left\|\boldsymbol{I}-\boldsymbol{I}_{inf}\right\|_F$.*

For interested readers, find the proof $\left\|\boldsymbol{I}-\boldsymbol{I}_{kfac}\right\|_F \geq \|\boldsymbol{I}-\boldsymbol{I}_{\text{efb}}\|_F$ in George et al. (2018). Note that $\left\|\boldsymbol{I}-\boldsymbol{I}_{kfac}\right\|_F \geq \|\boldsymbol{I}-\boldsymbol{I}_{\text{efb}}\|_F$ may not mean that $\left\|\boldsymbol{I}^{-1}-\boldsymbol{I}_{kfac}^{-1}\right\|_F \geq \left\|\boldsymbol{I}^{-1}-\boldsymbol{I}_{\text{efb}}^{-1}\right\|_F$ or vice versa. Yet, our proposed approximation yields better estimates than KFAC in the information form of MND.

3.2. Properties of Low-Rank Information Matrix

To our knowledge, the proposed sparse IM have not been studied before. Therefore, we theoretically motivate its design and validity for better insights.

Lemma 2: *Let $\boldsymbol{I}$ be the real Fisher information matrix, and let $\hat{\boldsymbol{I}}_{inf}$, $\boldsymbol{I}_{efb}$ and $\boldsymbol{I}_{kfac}$ be the low rank INF, EFB and KFAC estimates of it respectively. Then, it is guaranteed to have $\left\| diag(\boldsymbol{I}) - diag(\boldsymbol{I}_{efb})\right\|_F \geq \left\| diag(\boldsymbol{I}) - diag(\hat{\boldsymbol{I}}_{inf})\right\|_F = 0$ and $\left\| diag(\boldsymbol{I}) - diag(\boldsymbol{I}_{kfac})\right\|_F \geq \left\| diag(\boldsymbol{I}) - diag(\hat{\boldsymbol{I}}_{inf})\right\|_F = 0$. Furthermore, if the eigenvalues of $\hat{\boldsymbol{I}}_{inf}$ contains all non-zero eigenvalues of $\boldsymbol{I}_{inf}$, it follows: $\left\|\boldsymbol{I}-\boldsymbol{I}_{efb}\right\|_F \geq \left\|\boldsymbol{I}-\hat{\boldsymbol{I}}_{inf}\right\|_F$.*

Lemma 2 shows the optimally in capturing the diagonal variance while indicating that our approach also becomes effective in estimating off-diagonal entries if IM contains many close to zero eigenvalues. Validity of this assumption has been studied by Sagun et al. (2018) where it is shown that the Hessian of overparameterized DNNs tend to have many close-to-zero eigenvalues. Intuitively, from a graphical interpretation of IM, diagonal entries indicate information present in each nodes and off-diagonal entries are links of these nodes. Our sparsification scheme reduces the strength of the weak links while keeping the diagonal variance exact. This is a result of the diagonal correction after LRA which exploits spectrum sparsity of IM.

Lemma 3: *The low rank matrix $\hat{\Sigma} = \left((U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D\right)^{-1} \in \mathbb{R}^{N\times N}$ is a non-degenerate covariance matrix if the diagonal correction matrix D and LRA $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$ are both symmetric and positive definite. This condition is satisfied if $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T_{ii} < \mathbb{E}\left[\delta\theta_i^2\right]$ for all $i \in \{1, 2, \cdots, d\}$ and with $\Lambda_{1:L} \nsubseteq 0$.*

This comments on validity of the resulting posterior (a sufficient condition only) and proves that sparsifying the matrix can lead to a valid non-degenerate covariance if two conditions are met. As non-degenerate covariance can have a uniquely defined inverse, it is important to check these two conditions. We note that searching the rank can be automated with off-line computations that does not involve any data. Thus, it does not introduce significant overhead. In case D does not turn out to be, there are still several techniques that can deal with it. We recommend eigen-value clipping (Chen et al., 2018) or finding nearest positive semi-definite matrices (Higham, 1988). For a side note, above Lemma provides a sufficient condition and even if D is not positive definite, there is no indication that the given representation is an invalid form of covariance. These conditions have been a conservative guideline to make the likelihood term non-degenerate which we found to work well in practice. Lastly, D^{-1} is more numerically stable when we add a prior precision term and a scaling factor $(ND + \tau I)^{-1}$.

Before introducing the next theoretical property we define,

$$\hat{\boldsymbol{I}}^{\text{top}}_{1:K} = (U_A \otimes U_G)_{1:K}\Lambda_{1:K}(U_A \otimes U_G)^T_{1:K} \tag{16}$$

as a low rank EFB estimates of true Fisher that preserves top K eigenvalues. Similarly, $\hat{\boldsymbol{I}}^{\text{top}}_{1:L}$ can be defined which preserves top L eigenvalues. In contract, our proposal to preserve Kronecker structure in eigenvectors $\hat{\boldsymbol{I}}_{1:L}$ is denoted as shown below. Now, we start our analysis with Lemma 2.

$$\hat{\boldsymbol{I}}_{1:L} = (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T. \tag{17}$$

Lemma 4: *Let $\boldsymbol{I} \in \mathbb{R}^{N\times N}$ be the real Fisher information matrix, and let $\hat{\boldsymbol{I}}^{top}_{1:K} \in \mathbb{R}^{N\times N}$, $\hat{\boldsymbol{I}}^{top}_{1:L} \in \mathbb{R}^{N\times N}$ and $\hat{\boldsymbol{I}}_{1:L} \in \mathbb{R}^{N\times N}$ be the low rank estimates of $\boldsymbol{I}$ of EFB obtained by preserving top K, L and top K plus additional J resulting in L eigenvalues. Here, we define $K < L$. Then, the approximation error of $\hat{\boldsymbol{I}}_{1:L}$ is as follows: $\left\|\boldsymbol{I}-\hat{\boldsymbol{I}}^{top}_{1:L}\right\|_F \geq \left\|\boldsymbol{I}-\hat{\boldsymbol{I}}_{1:L}\right\|_F \geq \left\|\boldsymbol{I}-\hat{\boldsymbol{I}}^{top}_{1:K}\right\|_F$.*

This bound provides an insight that if preserving top L eigenvalues result in prohibitively too large covariance matrix, our LRA provides an alternative to preserving top K eigenvalues given that K < L. In practice, note that $\hat{\boldsymbol{I}}_{1:L}$ is a memory-wise feasible option as we formulate $\hat{\boldsymbol{I}}_{1:L} = (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$ which preserves the Kronecker structure in eigenvectors. This can be a case where evaluating $(U_a \otimes U_g)$ or $(U_a \otimes U_g)_{1:K}$ is not feasible to store.

4. Proofs

4.1. More Accurate of Information Matrix

Proposition 1: *Let $\boldsymbol{I} \in R^{N\times N}$ be the real information matrix, and let $\boldsymbol{I}_{inf} \in R^{N\times N}$ and $\hat{\boldsymbol{I}}_{inf} \in R^{N\times N}$ be our estimates of it with rank d and k such that $k < d$. Their diagonal entries are equal that is $\boldsymbol{I}_{ii} = \boldsymbol{I}_{inf_{ii}} = \hat{\boldsymbol{I}}_{inf_{ii}}$ for all i = 1, …, N.*

proof: The proof trivially follows from the definitions of $\boldsymbol{I} \in R^{N\times N}$, $\boldsymbol{I}_{\text{inf}} \in R^{N\times N}$ and $\hat{\boldsymbol{I}}_{\text{inf}} \in R^{N\times N}$. As the exact Fisher is an expectation on outer products of back-propagated gradients, its diagonal entries equal $\boldsymbol{I}_{ii} = \mathbb{E}\left[\delta\theta_i^2\right]$ for all i = 1, 2, …, N.

In the case of full ranked $\boldsymbol{I}_{inf}$, substituting $D_{ii} = \mathbb{E}\left[\delta\theta_i^2\right] - \sum_{j=1}^{nm}(v_{i,j}\sqrt{\Lambda_j})^2$ with $\sum_{j=1}^{nm}(v_{i,j}\sqrt{\Lambda_j})^2 = (U_A \otimes U_G)\Lambda(U_A \otimes U_G)_{ii}^T$ results in equation 18 for all i = 1, 2, …, N.

$$\begin{aligned}\boldsymbol{I}_{inf_{ii}} &= (U_A \otimes U_G)\Lambda(U_A \otimes U_G)_{ii}^T + D_{ii} \\ &= (U_A \otimes U_G)\Lambda(U_A \otimes U_G)_{ii}^T + \mathbb{E}\left[\delta\theta_i^2\right] \\ &\quad - (U_A \otimes U_G)\Lambda(U_A \otimes U_G)_{ii}^T \\ &= \mathbb{E}\left[\delta\theta_i^2\right]\end{aligned} \tag{18}$$

Similarly, we substitute $\hat{D}_{ii} = \mathbb{E}\left[\delta\theta_i^2\right] - \sum_{j=1}^{L}(\hat{v}_{i,j}\sqrt{\Lambda_{1:L}})^2$ with $\sum_{j=1}^{L}(\hat{v}_{i,j}\sqrt{\Lambda_{1:L}})^2 = (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)_{ii}^T$ which results in equation 19 for all i = 1, 2, …, N.

$$\begin{aligned}\hat{\boldsymbol{I}}_{inf_{ii}} &= (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)_{ii}^T + D_{ii} \\ &= (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)_{ii}^T + \mathbb{E}\left[\delta\theta_i^2\right] \\ &\quad - (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)_{ii}^T \\ &= \mathbb{E}\left[\delta\theta_i^2\right]\end{aligned} \tag{19}$$

Therefore, we have $\boldsymbol{I}_{ii} = \boldsymbol{I}_{\text{inf}_{ii}} = \hat{\boldsymbol{I}}_{\text{inf}_{ii}}$ for all i = 1, 2, …, N.

Lemma 1: *Let $\boldsymbol{I}$ be the real information matrix, and let $\boldsymbol{I}_{inf}$ and $\boldsymbol{I}_{efb}$ be the INF and EFB estimates of it respectively. It is guaranteed to have $\left\|\boldsymbol{I} - \boldsymbol{I}_{efb}\right\|_F \geq \left\|\boldsymbol{I} - \boldsymbol{I}_{inf}\right\|_F$.*

proof: Let $e^2 = \|A - B\|_F^2$ define a squared Frobenius norm of error between the two matrices $A \in \mathbb{R}^{N\times N}$ and $B \in \mathbb{R}^{N\times N}$. Now, e^2 can be formulated as,

$$\begin{aligned}e_b^2 &= \|A - B\|_F^2 \\ &= \sum_i (A-B)_{ii}^2 + \sum_i \sum_{j\neq i} (A-B)_{ij}^2\end{aligned} \tag{20}$$

The first term of equation 20 belongs to errors of diagonal entries in B w.r.t A whilst the second term is due to the off-diagonal entries.

Now, it follows that,

$$\begin{gathered}\|\boldsymbol{I} - \boldsymbol{I}_{\text{efb}}\|_F \geq \|\boldsymbol{I} - \boldsymbol{I}_{\text{inf}}\|_F \\ e_{efb}^2 \geq e_{inf}^2 \\ \textstyle\sum_i (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ij}^2 \geq \\ \textstyle\sum_i (\boldsymbol{I} - \boldsymbol{I}_{\text{inf}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{inf}})_{ij}^2 \\ \textstyle\sum_i (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ij}^2 \geq \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{inf}})_{ij}^2 \\ \textstyle\sum_i (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ij}^2 \geq \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ij}^2\end{gathered}$$

Note that $\sum_i (\boldsymbol{I} - \boldsymbol{I}_{\text{inf}})_{ii}^2 = 0$ using proposition 1. Furthermore, $\sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{inf}})_{ij}^2 = \sum_i \sum_{j\neq i} (\boldsymbol{I} - \boldsymbol{I}_{\text{efb}})_{ij}^2$ since by definition, $\boldsymbol{I}_{\text{efb}}$ and $\boldsymbol{I}_{\text{inf}}$ have the same off-diagonal terms.

Corollary 1: *Let $\boldsymbol{I}_{kfac}$ and $\boldsymbol{I}_{inf}$ be KFAC and our estimates of real information matrix $\boldsymbol{I}$ respectively. Then, it is guaranteed to have $\left\|\boldsymbol{I} - \boldsymbol{I}_{kfac}\right\|_F \geq \left\|\boldsymbol{I} - \boldsymbol{I}_{inf}\right\|_F$.*

For interested readers, find the proof $\left\|\boldsymbol{I} - \boldsymbol{I}_{kfac}\right\|_F \geq \|\boldsymbol{I} - \boldsymbol{I}_{\text{efb}}\|_F$ in George et al. (2018).

4.2. Properties of Low-Rank Information Matrix

Lemma 2: *Let $\boldsymbol{I}$ be the real Fisher information matrix, and let $\hat{\boldsymbol{I}}_{inf}$, $\boldsymbol{I}_{efb}$ and $\boldsymbol{I}_{kfac}$ be the low rank INF, EFB and KFAC estimates of it respectively. Then, it is guaranteed to have $\left\|\ diag(\boldsymbol{I}) - \ diag(\boldsymbol{I}_{efb})\right\|_F \geq \left\|\ diag(\boldsymbol{I}) - \ diag(\hat{\boldsymbol{I}}_{inf})\right\|_F = 0$ and $\left\|\ diag(\boldsymbol{I}) - diag(\boldsymbol{I}_{kfac})\right\|_F \geq \left\|\ diag(\boldsymbol{I}) - \ diag(\hat{\boldsymbol{I}}_{inf})\right\|_F = 0$. Furthermore, if the eigenvalues of $\hat{\boldsymbol{I}}_{inf}$ contains all non-zero eigenvalues of $\boldsymbol{I}_{inf}$, it follows: $\left\|\boldsymbol{I} - \boldsymbol{I}_{efb}\right\|_F \geq \left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{inf}\right\|_F$.*

proof: The first part follows from proposition 1 which states that for all the elements i, $\boldsymbol{I}_{ii} = \hat{\boldsymbol{I}}_{\text{inf}}$, $\left\|\ \text{diag}(\boldsymbol{I}) - \text{diag}(\boldsymbol{I}_{\text{efb}})\right\|_F \geq \left\|\ \text{diag}(\boldsymbol{I}) - \text{diag}(\hat{\boldsymbol{I}}_{\text{inf}})\right\|_F = 0$ and $\left\|\ \text{diag}(\boldsymbol{I}) - \text{diag}(\boldsymbol{I}_{\text{kfac}})\right\|_F \geq \left\|\ \text{diag}(\boldsymbol{I}) - \text{diag}(\hat{\boldsymbol{I}}_{\text{inf}})\right\|_F = 0$. This results by the design of the method, in which, we correct the diagonals in parameter space after LRA.

For the second part of the proof, lets recap that Lemma 2 (Wely's idea on eigenvalue perturbation) that removing zero eigenvalues does not affect the approximation error in terms of Frobenius norm. This then implies that off-diagonal elements of $\hat{\boldsymbol{I}}_{\text{inf}}$ and $\boldsymbol{I}_{\text{efb}}$ are equivalent. Then,:

$$\begin{gathered}\|\boldsymbol{I} - \boldsymbol{I}_{\text{efb}}\|_F \geq \left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{\text{inf}}\right\|_F \\ e_{efb}^2 \geq e_{inf}^2\end{gathered}$$

$$\sum_i (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ij}^2 \geq \sum_i (\boldsymbol{I}-\hat{\boldsymbol{I}}_{\text{inf}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I}-\hat{\boldsymbol{I}}_{\text{inf}})_{ij}^2$$
$$\sum_i (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ij}^2 \geq \sum_i \sum_{j\neq i} (\boldsymbol{I}-\hat{\boldsymbol{I}}_{\text{inf}})_{ij}^2$$
$$\sum_i (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ii}^2 + \sum_i \sum_{j\neq i} (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ij}^2 \geq \sum_i \sum_{j\neq i} (\boldsymbol{I}-\boldsymbol{I}_{\text{efb}})_{ij}^2$$

Again, $\sum_i (\boldsymbol{I}-\hat{\boldsymbol{I}}_{\text{inf}})_{ii}^2 = 0$ according to proposition 1 for all the elements i, which completes the proof.

Lemma 3: *The low rank matrix* $\hat{\Sigma} = \left((U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D\right)^{-1} \in \mathbb{R}^{N\times N}$ *is a non-degenerate covariance matrix if the diagonal correction matrix* D *and LRA* $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$ *are both symmetric and positive definite. This condition is satisfied if* $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)_{ii}^T < \mathbb{E}\left[\delta\theta_i^2\right]$ *for all* $i \in \{1, 2, \cdots, N\}$ *and with* $\Lambda_{1:L} \not\subseteq 0$.

proof: Let us first rewrite $\hat{\boldsymbol{I}}_{\text{inf}} = (U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D$ in the following form.

$$\begin{aligned}
&(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T + D = \\
&(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}}\Lambda_{1:L}^{\frac{1}{2}}(U_a \otimes U_g)^T + D \\
&= \left[(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}}\right]\left[(U_a \otimes U_g)\Lambda_{1:L}^{\frac{1}{2}}\right]^T + D \\
&= UU^T + D
\end{aligned} \tag{21}$$

Now, if D and $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$ is both symmetric and positive definite, it follows that for an arbitrary vector $x \in \mathbb{R}^d$, $x^T UU^T x > 0$ as eigenvalues $R_i > 0$ by construction. Furthermore, $x^T Dx > 0$ also holds by the definition of positive definiteness. Therefore, we have $x^T(UU^T + D)x = x^T UU^T x + x^T Dx > 0$ which leads to the proof that $\boldsymbol{I}_{\text{inf}}$ is positive definite if D and $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$ is both symmetric and positive definite. As this results in non-degenerate IM, the covariance Σ is non-degenerate as well.

Trivially following the definition of $D_{ii} = \mathbb{E}\left[\delta\theta_i^2\right] - (U_A \otimes U_G)\Lambda(U_A \otimes U_G)_{ii}^T$, $D_{ii} > 0$ for all i when $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)_{ii}^T < \mathbb{E}\left[\delta\theta_i^2\right]$. Again, by the definition of $\Lambda_{ii} = \mathbb{E}\left[(V^T\delta\theta)_i^2\right] \geq 0$, $\Lambda_{1:L}$ containing no zero eigenvalues result in the positive definite matrix $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T$, which completes the proof.

Lemma 4: *Let* $\boldsymbol{I} \in \mathbb{R}^{N\times N}$ *be the real Fisher information matrix, and let* $\hat{\boldsymbol{I}}_{1:K}^{top} \in \mathbb{R}^{N\times N}$, $\hat{\boldsymbol{I}}_{1:L}^{top} \in \mathbb{R}^{N\times N}$ *and* $\hat{\boldsymbol{I}}_{1:L} \in \mathbb{R}^{N\times N}$ *be the low rank estimates of* $\boldsymbol{I}$ *of EFB obtained by preserving top K, L and top K plus additional J resulting in L eigenvalues. Here, we define* $K < L$. *Then, the approximation error of* $\hat{\boldsymbol{I}}_{1:L}$ *is as follows:* $\left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{1:L}^{top}\right\|_F \geq \left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{1:L}\right\|_F \geq \left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{1:K}^{top}\right\|_F$.

proof: From the definition, $(U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T = V\Lambda V^T$ is PSD as $\Lambda_{ii} = \mathbb{E}\left[(V^T\delta\theta)_i^2\right] \geq 0$ for all elements i and $VV^T = I$ with I as an identity matrix (orthogonality). Naturally, low rank approximations $(U_A \otimes U_G)_{1:L^{\text{top}}}\Lambda_{1:L^{\text{top}}}(U_A \otimes U_G)_{1:L^{\text{top}}}^T$, $(U_A \otimes U_G)_{1:K^{\text{top}}}\Lambda_{1:K^{\text{top}}}(U_A \otimes U_G)_{1:K^{\text{top}}}^T$ and $(U_a \otimes U_g)\Lambda_{1:L}(U_a \otimes U_g)^T = (U_A \otimes U_G)_{1:L}\Lambda_{1:L}(U_A \otimes U_G)_{1:L}^T$ are again PSD by the fact that low rank approximation does not introduce negative eigenvalues.

Now, a well known fact from dimensional reduction literature is that low rank approximation preserving the top eigenvalues result in best approximation errors in terms of Frobenius norm for the given rank. Informally stating Wely's ideas on eigenvalue perturbation:

Let $B \in \mathbb{R}^{m\times n}$ with rank smaller or equal to p (one can also use complex space $\mathbb{C}$ instead of $\mathbb{R}$) and let $E = A - B$ with $A \in \mathbb{R}^{m\times n}$. Then, it follows that,

$$\begin{aligned}
\|A-B\|_F^2 &= \sigma_1(A-B)^2 + \cdots + \sigma_\mu(A-B)^2 \\
&\geq \sigma_{p+1}(A-B)^2 + \cdots + \sigma_\mu(A-B)^2 \\
&= \left\|A - B_{1:p}\right\|_F^2,
\end{aligned} \tag{22}$$

where $\sigma_1, \cdots \sigma_\mu$ are the singular values of A with $\mu = \min(n, m)$. The convention here is that $\sigma_i(A)$ is the ith largest singular value and $\sigma_i(A) = 0$ for $i >$ rank(A). Using this insight, and the fact that in the given settings, squared singular values are variances in new space lead to:

$$\left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{1:K}^{\text{top}}\right\|_F \geq \left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{1:L}\right\|_F \geq \left\|\boldsymbol{I} - \hat{\boldsymbol{I}}_{1:L}^{\text{top}}\right\|_F$$

This completes the proof of Lemma 4.

5. Implementation Details

The following experiments are implemented using Tensorflow (Abadi et al., 2016): (i) toy regression, (ii) UCI benchmark, (iii) active learning and (iv) classification on MNIST and CIFAR10. The KFAC library from Tensorflow [2] was used to implement the Fisher estimator (Martens & Grosse, 2015) for our methods and the works of Ritter et al. (2018a). On the other hand, Pytorch (Paszke et al., 2019) has been used for ImageNet and adversarial defense experiments [3]. The plug-and-play code is made available for Pytorch.

Note that empirical Fisher usually is not a good estimate of the Hessian, as it is typically biased (Martens & Grosse, 2015; Kunstner et al., 2019). Instead, KFAC library offers several estimation modes. We have used the gradients mode for KFAC whereas the exact mode was used for Diag. NVIDIA Tesla and 1080Ti are used for all the experiments.

[2] Available at https://github.com/tensorflow/kfac

[3] Available at TBD

Table 2. **UCI benchmark.** Root mean squared error (RMSE) is reported for the used set-up of UCI benchmark experiments. Note that as test log-likelihood depends on accuracy (in addition to uncertainty estimation), we have used linearized LA on the output space so that the test accuracy or RMSE is kept the same amongst the compared LA-based approaches. Our model overfits in some datasets so that effectiveness of having Bayesian Neural Network can be seen clearly. This also does not affect our results as all LA-based methods are built on top of the same model.

Datasets	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine	Yacht
RMSE	3.361±0.929	6.181±0.727	0.573±0.070	0.164±0.005	0.010±0.0001	4.322±0.153	4.516±0.123	0.637±0.034	9.568±1.132

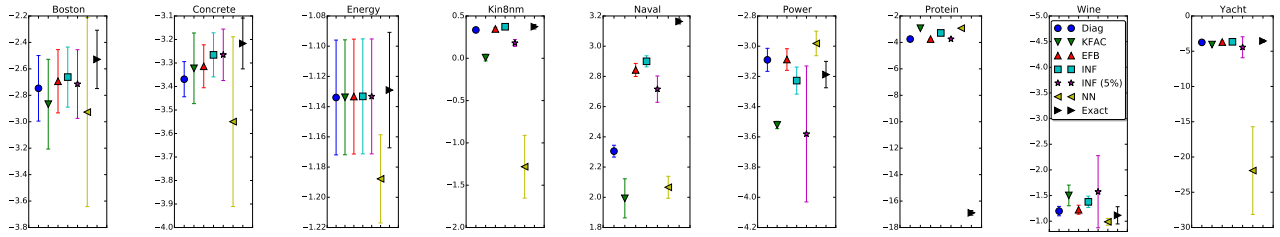

Figure 1. **Evaluating predictive uncertainty on UCI datasets.** We report test log likelihood on the y-axis and compare Diag, KFAC, EFB, INF variants, NN and Exact. Here, exact refers Laplace Approximation using the block-wise exact information matrix while NN denotes a deterministic neural network. Using linear approximation to the predictive uncertainty (MacKay, 1992a), accuracy between each methods are kept the same, which ensures fair comparisons using test log likelihood. Higher the better.

5.1. Small scale experiments

The training details are as follows: Adam has been used with a learning rate of 0.001 with zero prior precision or L2 regularization coefficient ($\tau = 0.2$ for KFAC, $\tau = 0.45$ for Diag, $N = 1$ and $\tau = 0$ for both FB and INF have been used). Mean squared error (MSE) loss is used. The exact block-wise Hessian and their approximations for the given setup contained zero values on its diagonals. This can be interpreted as zero variance in the IM, meaning no information, resulting in an IM being degenerate for the likelihood term. In such cases, the covariance may not be uniquely defined (Thrun et al., 2004). Therefore, we treated these variances as deterministic, making the IM non-degenerate (similar findings reported by MacKay (1992b)). We have used Numpyro (Phan et al., 2019) for the implementations of HMC, with 50000 samples. For BBB, we have used an open-source implementation [4] where the Gaussian noise is sampled in a batch initially, and a symmetric sampling is deployed. Lastly, $K_{mc} = 100$ samples were used.

Experiments on UCI benchmark [5] have been conducted to evaluate various IM estimates and their effects on predictive uncertainty estimation. We evaluate LA-based approaches only which has an advantage that the only differences between each approaches are approximations of IM. To explain, inference principle, network architectures, and training convergence can be kept the same. We note that, this is difficult for approaches based on variational inference. Due to this, meaningful comparisons can be often difficult as specific details such as number of epochs can have significant effects on the results (Mukhoti et al., 2018). In this line of argument, we have further used so-called linearized LA (MacKay, 1992a; Foong et al., 2019) instead of sampling based evaluation (Gal, 2016; Ritter et al., 2018a):

$$p(y^*|x^*, x, y) \approx \mathcal{N}(f_{\theta_{\text{MAP}}}(x^*), \Sigma_{\text{alea}} + \delta\theta(x^*)^T \Sigma_{\text{epis}} \delta\theta(x^*)).$$

As shown, the mean of prediction $f_{\theta_{\text{MAP}}}(x^*)$ depends only on the new test data x^*. As test log-likelihood depends on the accuracy of the predictor, we can keep the accuracy of the predictors the same among-st various LA-based approaches (reported in table 2). Furthermore, as the covariance matrix for predictive uncertainty only depends on *aleatoric* uncertainty Σ_{alea}, gradients on the new test input $\delta\theta(x^*)$ and episdemic uncertainty Σ_{epis}, comparisons between LA-based approaches are simpler as the experiments can be implemented in a way that the difference lies only in various approximations of Σ_{epis}. Closely following Foong et al. (2019), layer-wise exact IM has been established and the same hyperparameter settings are applied across other LA-based approaches [6]. Figure 1 reports the results of UCI experiments where we compare the reliability of uncertainty estimates using the test log-likelihood as a measure. As shown, we find that our approach compares well to the others. Note that in Power and Protein, LA approaches were

[4] Available at https://github.com/ThirstyScholar/bayes-by-backprop

[5] Dataset and splits have been taken from https://github.com/yaringal/DropoutUncertaintyExps.

[6] We also present the results of sampling based evaluations in section 6.4 where we extensively search hyperparameters for each LA methods separately.

under-performing even when compared to a deterministic DNN. This is a known limitation of LA: the approximated posterior may cover areas of low probability mass, the approaches perform similar to a deterministic DNN or become unstable. Here, our experiments also indicate that improvements in terms of Frobenius norm of error may not directly translate to performance in uncertainty estimation, atleast for LA, which requires in-depth treatment for future works.

5.2. Active Learning Experiments

Details of experiment settings are as follows: we split each one into training, validation, and test set with 20, 100 (which is reasonable when compared with the size of test set) and 100 data points randomly for 20 times, respectively. The remaining points serve as pool set, in which we are not allowed to obtain their labels. So we have 20 splits in this experiment. Once the model chooses the data point in the pool set and move it into the training set, its label becomes available, which simulates the scenario where humans annotate it. The experiments progress as follows: firstly, we trained the model with the initial training set and select one point from pool set which will be put into the training set with its label. Then we train the model again and proceed into the next iteration. In each iteration, we evaluate our model on the test set and report the root mean square error (RMSE). We select the model during training and the corresponding hyperparameter (τ) based on its performance on the validation set. While the range of N is $[0.5 * N, 1.0 * N]$, where N is the size of the training set in current iteration. The range of τ is $[1, 200, 400]$. For other hyperparameters, we use learning rate of 0.01 for boston housing and energy, 0.001 for wine and L2 regularization of 0.0 for boston housing and wine, $1e-5$ for energy. The mini-batch size is set to the initial size of training set, 20. Only 1 point is selected in each iteration and the number of iteration is set to 20. Regarding model selection, we use early stopping based on the RMSE on the validation set and the maximum epoch is set to 40.

5.3. MNIST and CIFAR10 Experiments

For MNIST, the dropout layer is used to the FC layers with a rate of 0.6. An important information is the size of each layers. The first layer constitutes 32 filters with 5 by 5 kernel, followed by the second layer with 64 filters and 5 by 5 kernel. The first fully connected layer then constitutes 1024 units and the last one ends with 10 units. We note that, this validates our method on memory efficiency as the third layer has a large number of parameters, and its covariance, being quadratic in its size, cannot be stored in our utilized GPUs. For CIFAR10 experiments, the most relevant settings are: the first layer constitutes 5 by 5 kernel with 64 filters. This is then again followed by the same. Units of 384, 192, and 10 have been used for the fully connected layers. Lastly, random cropping, flipping, brightness changes and contrast are the applied data augmentations.

Table 3. **Necessity of low rank approximation and reduction in complexity.** Reduced dimensions from N to the chosen rank L per layer are reported for both MNIST and CIFAR10 experiments. CNN stand for convolution while FC is for fully connected layers. The complexity of sampling $O(N^3)$ are reduced to $O(L^3)$. Employed strategy here was to keep the maximum for the rank K, which results in a seemingly arbitrary rank L.

MNIST	Dim N [-]	Dim L [-]	Percent [%]
CNN-1	800	450	**56.25**
CNN-2	51200	5185	**10.12**
FC-1	3211264	5625	**0.18**
FC-2	10240	4775	**46.63**
CIFAR	Dim N [-]	Dim L [-]	Percent [%]
CNN-1	4800	4800	**100**
CNN-2	102400	2112	**2.06**
FC-1	884736	3980	**0.45**
FC-2	73728	5499	**7.45**
FC-3	1920	1920	**100**

Implementation of deep ensemble (Lakshminarayanan et al., 2017) is kept rather simple by not using the adversarial training, but we combined 15 networks that were trained with different initialization. The same architecture and training procedure were used for all. For dropout, we have tried a grid search of dropout probabilities of 0.5 and 0.8, and have reported the best results. For the methods based on LA, we have performed grid search on hyperparameters N of (1, 50000, 100000) and 100 values of τ were tried using known class validation set. Note that for every method, and different datasets, each method required different values of τI to give a reasonable accuracy. Figure 3 depicts examples on MNIST where minimum ECE were selected. The LRA is imposed as a way to tackle the challenges of computational intractability. To empirically access the reduction in complexity, we depict the parameter and low rank dimensions N and L respectively in table 3. As shown, LRA based sampling computations reduce the computational complexity significantly. Furthermore, this explains the necessity of LRA - certain layers (e.g. FC-1 of both MNIST and CIFAR experiments) are computationally intractable to store, infer and sample.

5.4. ImageNet Experiments

To demonstrate that our method does not require changes in the training procedure, we used pre-trained weights from Pytorch [7]. Results are discussed in section 6.5. SWAG and SWA are the available baselines which have been evaluated on the ImageNet dataset and implementations are officially

[7] Available at https://pytorch.org/docs/stable/torchvision/models.html

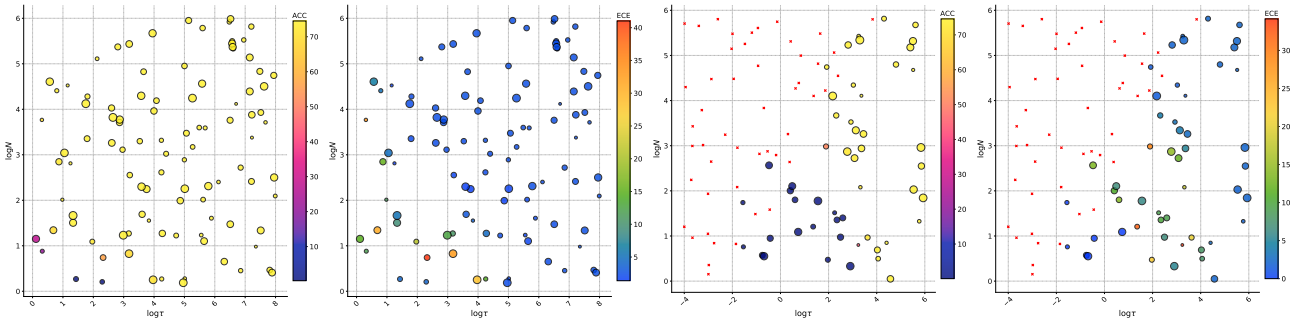

Figure 2. **Hyperparameter search** Results of random hyperparameter search for DenseNet121. From left to right: Diag Acc., Diag ECE, KFAC Acc. and KFAC ECE. Red crosses indicate configurations that where not invertible due to degeneracy or numerical instability. For accuracy, higher is better while for ECE lower is better.

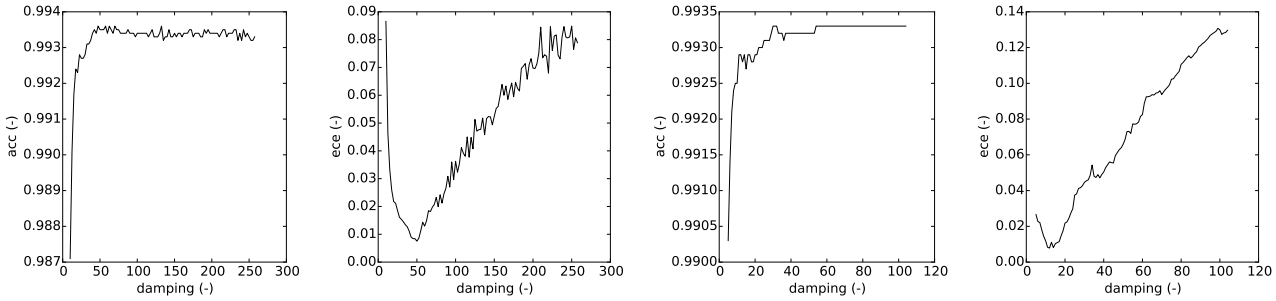

Figure 3. **Grid search results.** For Diag (left two figures) and KFAC (right two) an extensive grid search has been conducted to ensure fair comparison. Here, we report the results with pseudo observation term of 50000 on MNIST. This ensures that a main difference to DEF Laplace is the expression for model uncertainty as the inference and network architectures are kept the same.

open-sourced [8]. We closely followed the described experimental procedure. For the out-of-domain data we have used artistic impressions and paintings of landscapes and objects [9]. Lastly, performing multiple forward passes for the entire validation set is still a computationally expensive task. We therefore chose $K_{mc} = 30$ during inference, which we empirically found sufficient for the convergence, similar to Maddox et al. (2019).

We performed an extensive hyperparamter search for all LA methods using 100 randomly sampled pairs of N and τ selected from a log-scale between 0 and 10. We resorted to random search instead of grid search, as it tends to yield stronger results with a smaller number of samples (Bergstra & Bengio, 2012). The results for the accuracy (Acc.) and expected calibration error (ECE) are shown in figure 2. The ECE can be extremely low in insufficiently regularized areas, because the accuracy is also very low there, which is why we show the results for both metrics.

[8]Available at https://github.com/wjmaddox/swa_gaussian

[9]Available at https://www.kaggle.com/c/painter-by-numbers/data

6. Further Results and Critical Analysis

6.1. Spectral Sparsity of Information Matrix

One of the key insight behind our work is that information matrix of overparameterized DNNs tends to have close to zero eigenvalues (equivalently sparse in its spectrum). Is this true for the considered experiments? To answer this question, we plot the eigenvalue histograms in figure 13. Figure 13 shows that the empirical findings of Sagun et al. (2018) hold well in our experiment set up. Two concrete observations are found: (i) with varying depth (figures 4, 5, 6, 7, 8), IM showed tendency to get more sparse (especially on the maximum eigenvalue), and (ii) with varying number of parameters in each layers (figures 9, 10, 11, 12) IM showed to be more sparse with the number of parameters. One possible insight is that the information of individual parameters tends to be smaller if there are more parameters for explaining the same amount of data.

6.2. Effects of Low Rank Approximation

We additionally study the effects of LRA on the approximation quality of IM when compared to the exact, block diagonal IM. We include exact diagonal approximation to

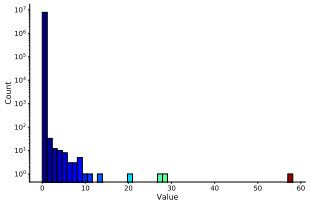

Figure 4. Densenet121

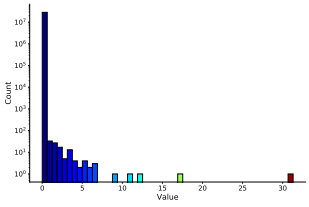

Figure 5. Densenet161

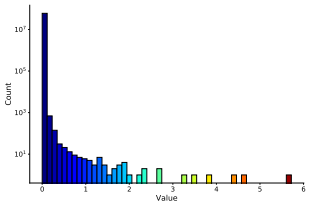

Figure 6. Resnet152

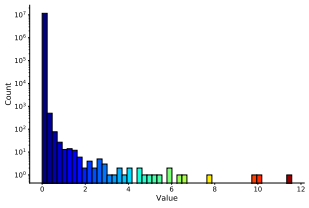

Figure 7. Resnet18

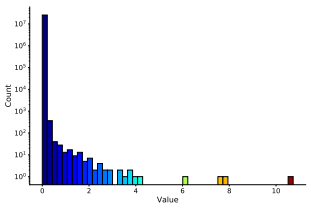

Figure 8. Resnet50

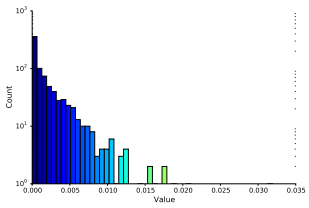

Figure 9. MNIST (layer 1)

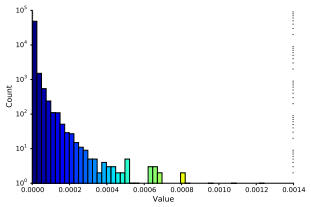

Figure 10. MNIST (layer 2)

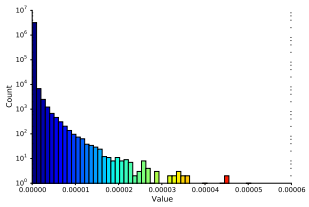

Figure 11. MNIST (layer 3)

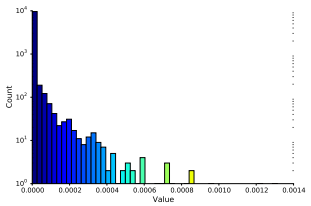

Figure 12. MNIST (layer 4)

Figure 13. **Eigenvalue histogram.** For figures 4 to 8, the eigenvalues of ImageNET architectures are shown. Here, x-axis plots the values whereas y-axis shows the counts in a log scale. From figure 9 to 12, we show the eigenvalues of MNIST (layer-wise differentiated. These figures empirically shows that the spectrum of information matrix is sparse (tend to have many values close to zeros) for all the considered architectures. Furthermore, in the considered set-up for MNIST dataset, more overparameterized layer tends to have more close-to-zero eigenvalues even within the same architecture.

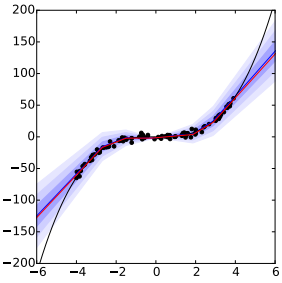

Figure 14. Diag

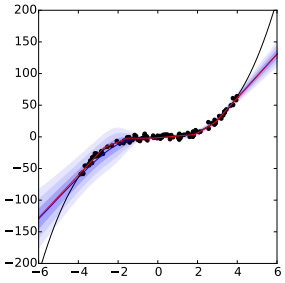

Figure 15. EFB

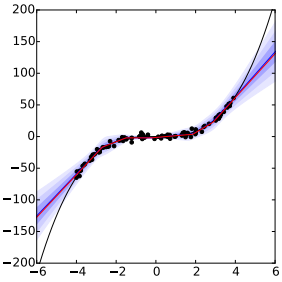

Figure 16. FB

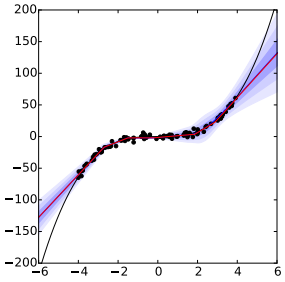

Figure 17. INF with rank 1

Figure 18. **Uncertainty on toy regression.** The black dots and the black lines are data points (x, y). The red and blue lines show predictions of the deterministic Neural Network and the mean output respectively. Upto three standard deviations are shown with blue shades.

the true IM while decrease the ranks of INF in steps of 25%. The results are depicted in table 4 which shows that due to the sparsity of IM, the error (with a measure on normalized frobenius norm) does not drastically increase with lower ranks. Diagonal approximation also results in the most severe approximation error on off-diagonal elements.

6.3. Additional Results on Toy Regression Experiments

Table 4. **UCI benchmark:** The normalized Frobenius norm of errors for the off-diagonal approximations w.r.t the true Fisher are depicted.

Dataset	Off-diagonals **Diag**	**INF (75%)**	**INF (50%)**	**INF (25%)**
Boston	1.000 ± 0.000	0.524±0.006	0.524±0.006	0.520±0.006
Concrete	1.000 ± 0.000	0.506±0.008	0.506±0.008	0.508±0.008
Energy	1.000 ± 0.000	0.504±0.006	0.504±0.006	0.514±0.006
Kin8nm	1.000 ± 0.000	0.526±0.005	0.526±0.005	0.546±0.005
Naval	1.000 ± 0.000	0.465±0.003	0.465±0.003	0.465±0.003
Power	1.000 ± 0.000	0.492±0.008	0.492±0.008	0.502±0.008
Protein	1.000 ± 0.000	0.541±0.021	0.541±0.021	0.541±0.021
Wine	1.000 ± 0.000	0.535±0.009	0.535±0.009	0.546±0.009
Yacht	1.000 ± 0.000	0.516±0.007	0.516±0.007	0.526±0.007

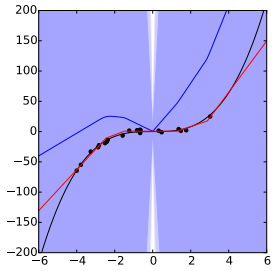

Figure 19. OKF

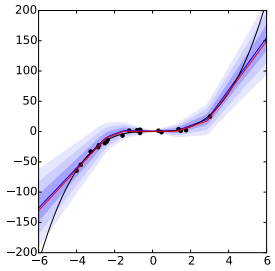

Figure 20. KFAC

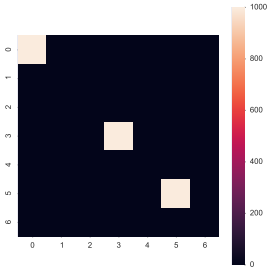

Figure 21. OKF Σ

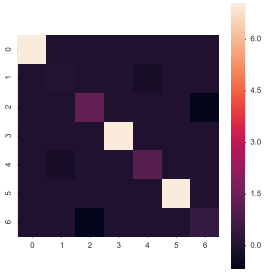

Figure 22. KFAC Σ

Figure 23. **Toy regression uncertainty and covariance visualization** (only the first layer is shown here). OKF Laplace means using equation 23 without further approximation in equation 24.

6.3.1. Effects of diagonal correction.

What is the relation between keeping diagonal elements of IM exact and predictive uncertainty? As effects of regularizing hyperparameters are removed to certain extent in the toy experiment, we study above mentioned question within this limited but controllable set-up.

For this purpose, we depict Diag, EFB, FB (layer-wise true IM) and INF with rank 1. The most comparable fit to HMC is given by FB while INF with rank 1 deteriorates when compared to its full rank counterpart. EFB for this set-up, produces considerable misfit to HMC. Importantly, since the only difference between EFB and DEF Laplace is a diagonal correction term, these results suggest that keeping diagonals of IM exact can result in accurate predictive uncertainty.

6.3.2. KFAC - a critical analysis.

Ritter et al. (2018a;b) reports that KFAC requires smaller sets of hyperparameters than Diag, which may suggest that KFAC produces better fits to the true posterior. Instead, we find that KFAC's approximation step for the prior incorporation may result in this phenomena. Concretely, lets define two variants as:

$$\mathrm{N}\boldsymbol{I}_{\mathrm{kfac}} + \tau I = \mathrm{N}(A \otimes G) + \tau I \quad \text{or} \tag{23}$$

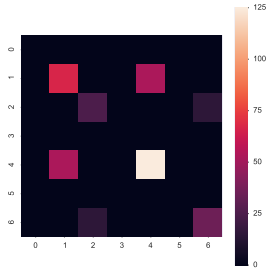

Figure 24. the Hessian [20].

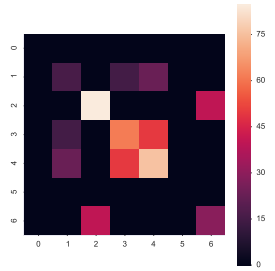

Figure 25. the Hessian [100]

Figure 26. **Visualization of the approximate information matrix with different data points.** Only the first layer chosen for the analysis. With increasing data points, the resulting information matrix becomes less degenerate.

$$\mathrm{N}\boldsymbol{I}_{\mathrm{kfac}} + \tau I \approx \left(\sqrt{\mathrm{N}}A_{\mathrm{i}-1} + \sqrt{\tau I}\right) \otimes \left(\sqrt{\mathrm{N}}G_{\mathrm{i}} + \sqrt{\tau I}\right). \tag{24}$$

Here, equation 24 has been the approximation step of Ritter et al. (2018a;b) while equation 23 is an exact variant. We denote the later as OKF. By reproducing the results of Ritter et al. (2018a), we depict the results in figure 23, in which we plot the predictive uncertainty obtained from OKF and KFAC under the same hyperparameter settings. Furthermore, a direct plot of covariance matrix can be found as well for the same hyperparameters. These results show that without the approximation step in equation 24, KFAC requires higher regularization hyperparameters, as similar as Diag. Looking into the covariance matrix directly, we also find that the magnitude of KFAC is smaller with this approximation. Therefore, our findings are that KFAC, due to the given approximation in incorporation of prior, requires smaller sets of regularization hyperparameters. Furthermore, as OKF does not seem to result in a similar phenomena, it might be difficult to conclude that KFAC, when compared to Diag, produces better fit to the true posterior.

6.3.3. Effects of data points size.

We now study the effects of dataset size to number of parameters. For this, we compare the dataset size 100 and 20. Results are depicted in figure 26. Notably, at using 20 data points resulted in more number of zero diagonal entries and corresponding rows and columns. This might be due to overparameterization of the model which results in under determined Hessian.

6.4. Effects of hyperparameters - UCI benchmark

Instead of linearized LA, we investigate the performance of LA-based methods with a full Bayesian analysis on UCI benchmarks. Rather than reporting the best performance of each methods with a single selected hyperparameter choice,

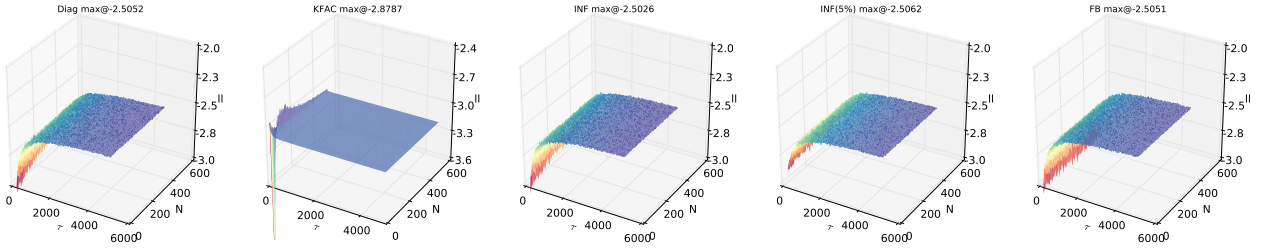

Figure 27. **Boston Hyperparameter Landscape.** Test-log likelihood on the z-axis. Ranging hyperparameters are display on XY plane. Maximum values are also displayed for Diag, KFAC, INF with two different ranks, and FB (true block-diagonal information matrix). Except KFAC, all LA-based approaches show similar behavior. Concrete, energy and protein showed similar tendency.

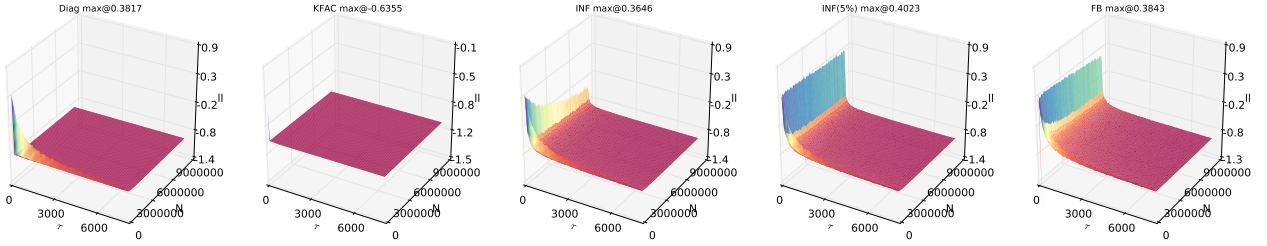

Figure 28. **Kin8nm Hyperparameter Landscape.** Test-log likelihood on the z-axis. Ranging hyperparameters are display on XY plane. Maximum values are also displayed for Diag, KFAC, INF with two different ranks, and FB (true block-diagonal information matrix). All LA-based approaches show different behavior. Naval, power, wine and yacht datasets have similar tendency.

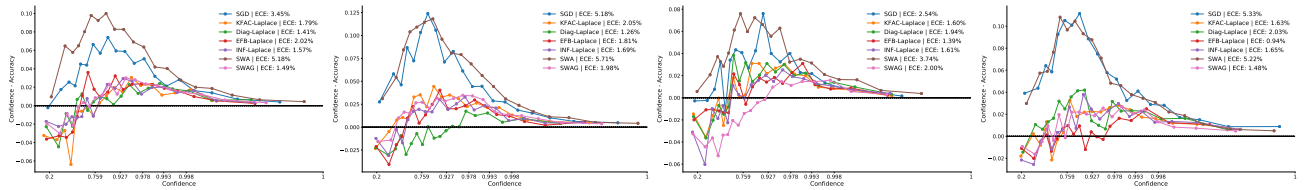

Figure 29. **Calibration results on large scale experiments:** From left to right: ResNet50, ResNet152, DenseNet121 and DenseNet161. Our method tends to outperform SWA and SWAG while being competitive to other fine tuned LA-based approaches.

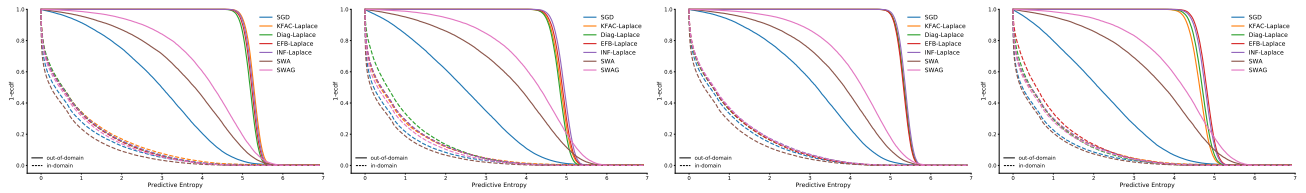

Figure 30. **Out-of-domain:** From left to right: ResNet50, ResNet152, DenseNet121 and DenseNet161. Our method tends to significantly outperform SWA and SWAG while being competitive to other fine tuned LA-based approaches.

we perform extensive grid searches and show the performance landscape. Such performance landscape can be informative for studying how more accurate approximation of information matrix translates to uncertainty estimation under the effects of hyperparameters. Note that this is possible on UCI datasets due to the small scale of the set up.

To this end, we search 10000 hyperparameters sets for each methods except KFAC, where we increase the size to 20000 hyperparameters [10]. The range of hyperparameters sets have been chosen differently for each datasets so that all the methods produce reliable predictions. We draw 100 K_{mc} samples for each predictions in order to have acceptable range of convergence for monte-carlo integration. Results are shown in figures 27 and 28 where we report following observations which falls into two categories.

- **Type 1 landscape:** Experiments on datasets namely boston housing, concrete, energy and protein showed similar behaviors. As seen in figure 27, the performance landscape (test log-likelihood) show similar curves for all the methods except KFAC. The maximum achievable performance have been found also similar with a marginal difference.
- **Type 2 landscape:** Experiments on datasets namely kin8nm, naval, power, wine and yacht showed a different tendency than type 1. While no methods significantly outperformed the other uniformly across all the datasets, the curve showed different behavior than type 1 as reported in figure 28. Each methods also showed different performance landscape.

These experiments suggest that for type 1, the benefits from having more accurate Fisher information is marginal. This suggests that improvements on accuracy of IM w.r.t Frobenius norm of error may not directly translate to more accurate uncertainty estimation within this context of LA.

For type 2 however, interesting differences can be found in the sense that INF variants and FB showed significantly more regimes of hyperparameter sets that outputs higher log-likelihood which can be benefits of having more accurate Fisher information matrix - when only smaller number of hyperparameter searches are possible, more accurate IM can result in better quality of predictive uncertainty. Understanding the causes of these behaviors to full generality seem a challenging research question as LA is tightly coupled with loss landscape of DNNs and further, how optimization affects generality and the shape of true posterior. One possible explanation for type 1 is that maintaining a single τ and N for all the layers may force all the methods to be regularized for fitting a few sharply peaked local mode of true posterior, hindering the benefits of having more accurate estimates of true Fisher information.

[10] We have doubled the search space for KFAC as it requires smaller sets of hyperparameters due to equation 24 instead of equation 23. Following this observation, we further decreased the minimum τ.

6.5. Additional ImageNet Results

The calibration and OOD detection experiments presented in the main text on ResNet18 were performed identically for the four additional architectures. We show the results in figures 29 and 30. The observations from the main text hold for the additional networks. All LA-based approaches can reduce the calibration error significantly compared to the deterministic network and SWA and are as good or better than SWAG. In out-of-domain separation, we find that the LA-based approaches perform comparably strong and are far superior to the other methods across all considered networks.

Table 5. **Wall clock time analysis on sampling.** Mean and standard deviation over 1000 draws are reported with a single thread.

Architecture	Diag [ms]	KFAC [ms]	EFB [ms]	INF [ms]
ResNet18	1.24 ± 0.06	8.23 ± 0.04	9.28 ± 0.06	4.74 ± 0.08
ResNet50	1.94 ± 0.11	15.47 ± 0.18	16.89 ± 0.09	12 ± 0.25
ResNet152	5.4 ± 0.07	32.08 ± 0.5	35.55 ± 0.07	32.62 ± 0.15
DenseNet121	4.41 ± 0.14	8.92 ± 0.19	10.22 ± 0.13	25.03 ± 0.65
DenseNet161	5.86 ± 0.11	16.48± 0.03	18.84 ± 0.54	35.95 ± 0.35

Table 5 also the wall clock analysis for sampling. Interestingly, for ResNet variants, INF is more efficient than KFAC and EFB due to the effects of low rank approximation. On the other hand, DenseNet variants have many small layers and therefore, rank reduction is less noticeable and cannot outweigh the disadvantage of having a more number of smaller operations in a sampling procedure. While KFAC and EFB maintain similar size matrices, EFB sampling is slower than KFAC, also due to more number of operations. Diag is as expected, the most efficient method. We note however, that Bayesian Neural Networks in general, has a disadvantage that prediction time is atleast 30 times slower (assuming 30 samples are taken) and thus, there may not be any practical advantages.

Table 6. **Wall clock time analysis on information matrix computation.** Values are rounded to the nearest.

Architecture	Diag [min]	KFAC [min]	EFB [min]	INF [min]
ResNet18	30	120	165	165
ResNet50	82	210	300	300
ResNet152	180	510	720	720
DenseNet121	100	360	465	465
DenseNet161	180	870	1060	1060

Next, table 6 reports the wall clock analysis for IM computations. In our implementation, EFB is computed after having KFAC and therefore, it takes more time than KFAC. Original implementation of EFB contains amortize eigen-decomposition and can be made more efficient than KFAC. INF is an offline procedure, and provides negligible over-

head to EFB. The total computation time for all the methods are less than a day on ImageNet, and thus, this analysis shows practicality and scalability of LA-based approaches.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., and Zheng, X. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pp. 265–283, 2016.

Ambikasaran, S. and O'Neil, M. Fast symmetric factorization of hierarchical matrices with applications. *CoRR*, abs/1405.0223, 2014.

Bergstra, J. and Bengio, Y. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(Feb):281–305, 2012.

Chen, S.-W., Chou, C.-N., and Chang, E. Y. Bda-pch: Block-diagonal approximation of positive-curvature hessian for training neural networks. *CoRR*, abs/1802.06502, 2018.

Foong, A. Y., Li, Y., Hernández-Lobato, J. M., and Turner, R. E. 'in-between'uncertainty in bayesian neural networks. *arXiv preprint arXiv:1906.11537*, 2019.

Gal, Y. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

George, T., Laurent, C., Bouthillier, X., Ballas, N., and Vincent, P. Fast approximate natural gradient descent in a kronecker factored eigenbasis. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pp. 9573–9583, 2018.

Higham, N. J. Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and its Applications*, 103:103–118, 1988. ISSN 0024-3795. doi: 10.1016/0024-3795(88)90223-6.

Kunstner, F., Hennig, P., and Balles, L. Limitations of the empirical fisher approximation for natural gradient descent. In *Advances in Neural Information Processing Systems*, pp. 4156–4167, 2019.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pp. 6402–6413, 2017.

MacKay, D. J. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992a.

MacKay, D. J. C. A practical bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472, 1992b.

Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P., and Wilson, A. G. A simple baseline for bayesian uncertainty in deep learning. In *Advances in Neural Information Processing Systems*, pp. 13132–13143, 2019.

Martens, J. and Grosse, R. B. Optimizing neural networks with kronecker-factored approximate curvature. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pp. 2408–2417, 2015.

Mukhoti, J., Stenetorp, P., and Gal, Y. On the importance of strong baselines in bayesian deep learning. *CoRR*, abs/1811.09385, 2018.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

Phan, D., Pradhan, N., and Jankowiak, M. Composable effects for flexible and accelerated probabilistic programming in numpyro. In *NeurIPS Workshop on Program Transformations 2019 (to appear)*, 2019.

Ritter, H., Botev, A., and Barber, D. A scalable laplace approximation for neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018a.

Ritter, H., Botev, A., and Barber, D. Online structured laplace approximations for overcoming catastrophic forgetting. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pp. 3742–3752, 2018b.

Sagun, L., Evci, U., Güney, V. U., Dauphin, Y., and Bottou, L. Empirical analysis of the hessian of over-parametrized neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*, 2018.

Thrun, S., Liu, Y., Koller, D., Ng, A. Y., Ghahramani, Z., and Durrant-Whyte, H. Simultaneous localization and mapping with sparse extended information filters. *The international journal of robotics research*, 23(7-8):693–716, 2004.