On the Variance of the Fisher Information for Deep Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

The Fisher information matrix (FIM) has been applied to the realm of deep learning. It is closely related to the loss landscape, the variance of the parameters, second order optimization, and deep learning theory. The exact FIM is either unavailable in closed form or too expensive to compute. In practice, it is almost always estimated based on empirical samples. We investigate two such estimators based on two equivalent representations of the FIM. They are both unbiased and consistent with respect to the underlying "true" FIM. Their estimation quality is characterized by their variance given in closed form. We bound their variances and analyze how the parametric structure of a deep neural network can impact the variance. We discuss the meaning of this variance measure and our bounds in the context of deep learning.

1 Introduction

The Fisher information is one of the most fundamental concepts in statistical machine learning. Intuitively, it measures the amount of information carried by a single random observation when the underlying model varies along certain directions in the parameter space: if the variation does not change the underlying model, then an observation contains zero (Fisher) information and is non-informative regarding the parameter; Otherwise, if the variation significantly changes the model and has large information, then the estimation can be relatively more accurate, as described by the Cramér-Rao lower bound (CRLB). In machine learning, this concept is useful for defining intrinsic geometric structure of the parameter space, measuring model complexity, and performing gradient-based optimization [14].

Given a statistical model that is specified by a parametric form $p(\boldsymbol{z} \mid \boldsymbol{\theta})$ and a continuous domain $\boldsymbol{\theta} \in \mathcal{M}$, the Fisher information matrix (FIM) is a 2D tensor

$$\mathcal{I}(\boldsymbol{\theta}) = E_{p(\boldsymbol{z} \mid \boldsymbol{\theta})} \left(\frac{\partial \ell}{\partial \boldsymbol{\theta}} \frac{\partial \ell}{\partial \boldsymbol{\theta}^\top} \right), \tag{1}$$

where $E_{p(\boldsymbol{z} \mid \boldsymbol{\theta})}(\cdot)$, or simply $E_p(\cdot)$ if the model p is clear from the context, denotes the expectation w.r.t. $p(\boldsymbol{z} \mid \boldsymbol{\theta})$, and $\ell \stackrel{\text{def}}{=} \log p(\boldsymbol{z} \mid \boldsymbol{\theta})$ is the log-likelihood function. Under weak conditions (see Lemma 5.3 [8] for the univariate case), the FIM has the equivalent expression $\mathcal{I}(\boldsymbol{\theta}) = E_{p(\boldsymbol{z} \mid \boldsymbol{\theta})} \left(-\frac{\partial^2 \ell}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} \right)$. Given N observations $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N$, these two equivalent expressions of the FIM lead to two different estimators

$$\hat{\mathcal{I}}_1(\boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{\partial \ell_i}{\partial \boldsymbol{\theta}} \frac{\partial \ell_i}{\partial \boldsymbol{\theta}^\top} \right) \quad \text{and} \quad \hat{\mathcal{I}}_2(\boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \left(-\frac{\partial^2 \ell_i}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} \right), \tag{2}$$

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

where $\ell_i = \log p(\boldsymbol{z}_i \,|\, \boldsymbol{\theta})$ is the log-likelihood of the i'th observation $\boldsymbol{z}_i$. The notations $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ are abused for simplicity as they actually depend on the random observations $\boldsymbol{z}_i$.

These estimators are universal and independent to the parametric form $p(\boldsymbol{z} \,|\, \boldsymbol{\theta})$. They are expressed in terms of the 1st- or 2nd-order derivatives of the log-likelihood. Usually, we already have these derivatives to perform gradient-based learning, and therefore we can save computational cost through the reuse of these derivatives to estimate the Fisher information. This is especially useful for deep learning, where the computational overhead of the FIM can be significant.

It is straightforward from the law of large numbers and the central limit theorem that both estimators are unbiased and consistent.

Proposition 1.

$$E_{p(\boldsymbol{z} \,|\, \boldsymbol{\theta})}\left(\hat{\mathcal{I}}_1(\boldsymbol{\theta})\right) = E_{p(\boldsymbol{z} \,|\, \boldsymbol{\theta})}\left(\hat{\mathcal{I}}_2(\boldsymbol{\theta})\right) = \mathcal{I}(\boldsymbol{\theta}).$$
$$\forall \epsilon > 0, \lim_{N \to \infty} \mathrm{Prob}(|\hat{\mathcal{I}}_1(\boldsymbol{\theta}) - \mathcal{I}(\boldsymbol{\theta})| + |\hat{\mathcal{I}}_2(\boldsymbol{\theta}) - \mathcal{I}(\boldsymbol{\theta})| > \epsilon) = 0,$$

where $\mathrm{Prob}(\cdot)$ *denotes the probability of the parameter statement being true.*

How far can $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ derivate from the true FIM $\mathcal{I}(\boldsymbol{\theta})$, and how fast can they converge to $\mathcal{I}(\boldsymbol{\theta})$ as the number of observations increases? It is important to analyze the variance of $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$, so that we are in control of the estimation quality, and measurements or algorithms based on the FIM can be performed accurately.

We made the following contributions that will unfold in the following Sections 2 to 4. In Section 5, we give our concluding remarks.

- We review and rediscover two equivalent expression of the FIM in the context of deep feed-forward networks (Section 2);
- We give in closed form the variance (extending to meaningful upper bounds) and discuss the convergence rate of the two estimators $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ (Section 3);
- We analyze how the 1st- and 2nd-order derivatives of the neural network can affect the estimation of the FIM (Section 4).

2 Feed-forward Networks with Exponential Family Output

This section realizes the concept of Fisher information in a feed-forward network with exponentially family output and explains why its estimators will be useful in theory and practice.

Consider the prediction task where $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y})$ with features $\boldsymbol{x}$ and labels $\boldsymbol{y}$; with corresponding prediction model specified by a joint distribution $p(\boldsymbol{x}, \boldsymbol{y} \,|\, \boldsymbol{\theta}) = p(\boldsymbol{x})p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta})$. The marginal distribution $p(\boldsymbol{x})$ is parameter-free, usually fixed as the empirical distribution $p(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\boldsymbol{x} - \boldsymbol{x}_i)$ w.r.t. a set of observations $\{\boldsymbol{x}_i\}_{i=1}^N$, where $\delta(\cdot)$ is the Dirac delta. The predictor $p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta})$ is a neural network with parameters $\boldsymbol{\theta} = \{\boldsymbol{W}_{l-1}\}_{l=1}^L$ and exponential family output units, given by

$$\begin{aligned}
p(\boldsymbol{y} \,|\, \boldsymbol{x}) &= \exp\left(\boldsymbol{t}^\top(\boldsymbol{y})\boldsymbol{h}_L - F(\boldsymbol{h}_L)\right), \\
\boldsymbol{h}_l &= \sigma(\boldsymbol{W}_{l-1}\bar{\boldsymbol{h}}_{l-1}), \quad (l = 1, \ldots, L) \\
\bar{\boldsymbol{h}}_l &= (\boldsymbol{h}_l^\top, 1)^\top, \\
\boldsymbol{h}_0 &= \boldsymbol{x},
\end{aligned} \tag{3}$$

where $\boldsymbol{t}(\boldsymbol{y})$ is the sufficient statistics of the prediction model, $F(\boldsymbol{h}) = \log \int \exp(\boldsymbol{t}^\top(\boldsymbol{y})\boldsymbol{h}) d\boldsymbol{y}$ is the log-partition function, and $\sigma : \Re \to \Re$ is a (typically non-linear) activation function applied element-wise. Further let $n_l = \dim(\boldsymbol{h}_l(\boldsymbol{x}))$ denote the size of layer l. In particular, the neural network $\boldsymbol{h}_L$ specifies the natural parameter of the exponential family. All intermediate variables $\boldsymbol{h}_l$ are extended to include a constant scalar 1 in $\bar{\boldsymbol{h}}_l$, so that a linear layer can simply be expressed as $\boldsymbol{W}_{l-1}\bar{\boldsymbol{h}}_{l-1}$. We need the following lemma which gives the FIM of the exponential family $p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta})$.

Lemma 2. *For the neural network model specified in Eq.* (3)*,*

$$\mathcal{I}(\boldsymbol{h}_L) = \mathrm{Var}(\boldsymbol{t}(\boldsymbol{y})) = \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{h}_L},$$

where $\mathrm{Var}(\cdot)$ *denotes the covariance matrix w.r.t.* $p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta})$*,* $\boldsymbol{\eta} \overset{def}{=} \boldsymbol{\eta}(\boldsymbol{h}_L) \overset{def}{=} F'(\boldsymbol{h}_L)$ *is the expectation parameters, and the vector-vector-derivative* $\partial \boldsymbol{\eta} / \partial \boldsymbol{h}_L$ *denotes the Jacobian matrix.*

The derivatives of the log-likelihood $\ell(\boldsymbol{\theta}) = \log p(\boldsymbol{x}, \boldsymbol{y} \,|\, \boldsymbol{\theta})$ characterizing its landscape, are essential to compute the FIM. The score function (gradient of ℓ) is

$$\frac{\partial \ell}{\partial \boldsymbol{\theta}} = \left(\frac{\partial \boldsymbol{h}_L}{\partial \boldsymbol{\theta}} \right)^\top (\boldsymbol{t}(\boldsymbol{y}) - \boldsymbol{\eta}(\boldsymbol{h}_L)) = \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta}} (\boldsymbol{t}_a - \boldsymbol{\eta}_a). \tag{4}$$

In this paper, we mix the usual Σ-notation of summation with the Einstein notation: In the same term, an index appearing in both upper- and lower-positions indicates a sum over this index. For example, $\boldsymbol{t}_a \boldsymbol{h}^a = \sum_a \boldsymbol{t}_a \boldsymbol{h}_a$. We found this mixed representation of sums helps to simplify our expressions without causing confusion. From Eq. (4), the Hessian of ℓ is given by

$$\frac{\partial^2 \ell}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} = (\boldsymbol{t}_a - \boldsymbol{\eta}_a) \frac{\partial^2 \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} - \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\eta}_a}{\partial \boldsymbol{\theta}^\top} = (\boldsymbol{t}_a - \boldsymbol{\eta}_a) \frac{\partial^2 \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} - \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta}} \mathcal{I}_{ab}(\boldsymbol{h}_L) \frac{\partial \boldsymbol{h}_L^b}{\partial \boldsymbol{\theta}^\top}. \tag{5}$$

The FIM has the equivalent expression (expectation of the Hessian of $-\ell$) as long as the activation function is smooth enough.

Theorem 3. *For any activation function* $\sigma \in C^2(\Re)$ *(both* $\sigma'(x)$ *and* $\sigma''(x)$ *exist and are continuous), we have* $\mathcal{I}(\boldsymbol{\theta}) = E_p \left(-\frac{\partial^2 \ell}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} \right)$.

ReLU networks do not have this equivalent expression as $\mathrm{ReLU}(x)$ is not differentiable at $x = 0$.

By the definition of the FIM, or its equivalent formula in Theorem 3, we arrive at the same expression

$$\mathcal{I}(\boldsymbol{\theta}) = \left(\frac{\partial \boldsymbol{h}_L}{\partial \boldsymbol{\theta}} \right)^\top \mathcal{I}(\boldsymbol{h}_L) \frac{\partial \boldsymbol{h}_L}{\partial \boldsymbol{\theta}^\top} = \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta}} \mathcal{I}_{ab}(\boldsymbol{h}_L) \frac{\partial \boldsymbol{h}_L^b}{\partial \boldsymbol{\theta}^\top}. \tag{6}$$

This specific definition of the FIM has been previously been called the 'conditional' FIM [5].

To compute $\mathcal{I}(\boldsymbol{\theta})$ in closed form, one need to first compute the Jacobian matrix of size $\dim(\boldsymbol{\theta}) \times n_L$, then perform the matrix multiplication in Eq. (6). Once the parameter $\boldsymbol{\theta}$ is updated, the FIM has to be recomputed. This is infeasible in practice for large networks.

The two estimators $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ in Eq. (2) provide a computationally inexpensive way to estimate the FIM. Given $\boldsymbol{\theta}$ and $\boldsymbol{x}$, one can draw i.i.d. samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N \sim p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta})$. Both $\partial \ell_i / \partial \boldsymbol{\theta}$ and $\partial^2 \ell_i / \partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top$ can be evaluated directly through auto-differentiation that is highly optimized for modern GPUs. In most of the time, we only need the diagonal blocks of the FIM. Therefore $\partial^2 \ell_i / \partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top$ can be efficiently computed in real time. If $p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta})$ has the parametric form in Eq. (3), from Eqs. (4) and (5), the estimators in Eq. (2) becomes

$$\hat{\mathcal{I}}_1(\boldsymbol{\theta}) = \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta}} \cdot \frac{1}{N} \sum_{i=1}^N (\boldsymbol{t}_a(\boldsymbol{y}_i) - \boldsymbol{\eta}_a)(\boldsymbol{t}_b(\boldsymbol{y}_i) - \boldsymbol{\eta}_b) \cdot \frac{\partial \boldsymbol{h}_L^b}{\partial \boldsymbol{\theta}^\top}, \tag{7}$$

$$\hat{\mathcal{I}}_2(\boldsymbol{\theta}) = \left(\boldsymbol{\eta}_a - \frac{1}{N} \sum_{i=1}^N \boldsymbol{t}_a(\boldsymbol{y}_i) \right) \frac{\partial^2 \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} + \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta}} \mathcal{I}_{ab}(\boldsymbol{h}_L) \frac{\partial \boldsymbol{h}_L^b}{\partial \boldsymbol{\theta}^\top}. \tag{8}$$

Recall that the notation of $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ is abused as they depend on $\boldsymbol{x}$ and $\boldsymbol{y}_1 \cdots \boldsymbol{y}_N$. Notably, $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ depends on the Jacobian matrix of the mapping $\boldsymbol{\theta} \to \boldsymbol{h}_L$, and the empirical variance of the minimal sufficient statistic $\boldsymbol{t}(\boldsymbol{y}_i)$ of the output exponential family; $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ depends on both the Jacobian and Hessian of $\boldsymbol{\theta} \to \boldsymbol{h}_L$, and the empirical average of $\boldsymbol{t}(\boldsymbol{y}_i)$. The second term in Eq. (8) is exactly the FIM. If the output units belong to non-exponential families, *e.g.*, a statistical mixture model, one falls back to the general formulae in Eq. (2).

The CRLB states that any unbiased estimator $\hat{\boldsymbol{\theta}}_i$ of the parameter $\boldsymbol{\theta}_i$ satisfies $\mathrm{Var}(\hat{\boldsymbol{\theta}}_i) \geq [\mathcal{I}(\boldsymbol{\theta}_i)]^{-1}$. Its application relies on an accurate estimation of $\mathcal{I}(\boldsymbol{\theta}_i)$. In Lemma 2, the FIM is with respect to the output of the neural network. As such, $\mathcal{I}(\boldsymbol{h}_L)$ can be used to obtain a lower bound for any unbiased estimator of the optimal neural network, $\mathrm{Var}(\hat{\boldsymbol{h}}_i) \geq [\mathcal{I}_{ii}(\boldsymbol{h}_L)]^{-1}$. Similarly for Eq. (6), we can consider unbiased estimators for the weights of the neural network $\hat{\boldsymbol{\theta}}$. Here we have variance $\mathrm{Var}(\hat{\boldsymbol{\theta}}_i) = [\mathcal{I}_{ii}(\boldsymbol{\theta})]^{-1}$. If the scale of the FIM is relatively small as compared to is variance, the FIM estimation is more likely to be a small positive value (or even become zero or negative). This will lead to a large error when applying the CRLB.

Related Work A large body of efforts tries to approximate the FIM or define "FIM-like" tensors for performing natural gradient descent in deep learning [1, 9, 12, 17]. Efficient computational methods are developed to evaluate the FIM inverse [13]. Variants of natural gradient descent are considered [16, 10]. Ollivier [12] provided algorithm procedures to compute the unit-wise FIM and discussed Monte Carlo natural gradient.

Our estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ is not the "empirical Fisher" (see *e.g.* section 11 [9]), as $\boldsymbol{y}_i$ is randomly sampled from $p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta})$, making $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ an unbiased estimator. The difference between this empirical Fisher and the FIM is clarified [7]. Similarly, our estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ is not the Hessian of the loss, as $\boldsymbol{y}_i$ is randomly sampled rather than fixed to the given target. The two FIM estimators have been examined in latent variable models [4]. To the best of our knowledge, their variance has not been analyzed theoretically in machine learning.

Recently, the structure of the FIM (or its partial approximations) are examined in deep learning settings. In particular, the FIM of randomized networks are analyzed [2], often using spectral analysis [15, 5] based on random matrix theory. An insight is that most of the eigenvalues of the FIM are close to 0 while the high end of spectrum has large values [6].

In information geometry [1], higher order tensors are used to describe the intrinsic structure in the space of probability distributions. The affine connection is a 3D tensor; the Riemannian curvature is a 4D tensor. The third cumulants of the sufficient statistics gives an affine connection (belonging to the α-connections or the Amari-Čensov tensor) of the exponential family. The moments of the FIM are in a higher-order of the sufficient statistics. Similar concepts are examined in higher-order asymptotic theory. In statistics, higher order moments of the maximum likelihood estimator (MLE) were discussed (see *e.g.* [3]). These moments are associates with parameter estimators and differ from the concept of the FIM estimators.

3 The Variance of the FIM Estimators

In this section, we measure the quality of the two estimators $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ given by their variances. Given the same sample size N, a smaller variance is preferred as the estimator is more accurate and is likely to be close to the true FIM $\mathcal{I}(\boldsymbol{\theta})$. We focus on the deep learning architecture specified in Eq. (3). In particular, we consider how the structure of the exponential family impacts the variance.

3.1 Variance in closed form

We first consider the variance of $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ in Eq. (2) as real matrices in $\Re^{\dim(\boldsymbol{\theta})\times\dim(\boldsymbol{\theta})}$. As the samples $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N$ are i.i.d., their variances are given by

$$\mathrm{Var}(\hat{\mathcal{I}}_1(\boldsymbol{\theta})) = \frac{1}{N}\mathrm{Var}\left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_i}\frac{\partial \ell}{\partial \boldsymbol{\theta}_j}\right), \quad \mathrm{Var}(\hat{\mathcal{I}}_2(\boldsymbol{\theta})) = \frac{1}{N}\mathrm{Var}\left(\frac{\partial^2 \ell}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j}\right). \tag{9}$$

For the neural network model in Eq. (3), we further have those variances on the RHS (right-hand-side) in closed form. The variance term of the first estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ is presented in the follow theorem.

Theorem 4.

$$\mathrm{Var}\left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_i}\frac{\partial \ell}{\partial \boldsymbol{\theta}_j}\right) = \partial_i \boldsymbol{h}_L^a(\boldsymbol{x})\partial_j \boldsymbol{h}_L^b(\boldsymbol{x})\partial_i \boldsymbol{h}_L^c(\boldsymbol{x})\partial_j \boldsymbol{h}_L^d(\boldsymbol{x}) \cdot \left(\mathcal{K}_{abcd}(\boldsymbol{t}) - \mathcal{I}_{ab}(\boldsymbol{h}_L) \cdot \mathcal{I}_{cd}(\boldsymbol{h}_L)\right),$$

where the 4D tensor

$$\mathcal{K}_{abcd}(\boldsymbol{t}) = \mathbb{E}[(\boldsymbol{t}_a - \boldsymbol{\eta}_a(\boldsymbol{h}_L(\boldsymbol{x})))(\boldsymbol{t}_b - \boldsymbol{\eta}_b(\boldsymbol{h}_L(\boldsymbol{x})))(\boldsymbol{t}_c - \boldsymbol{\eta}_c(\boldsymbol{h}_L(\boldsymbol{x})))(\boldsymbol{t}_d - \boldsymbol{\eta}_d(\boldsymbol{h}_L(\boldsymbol{x})))]$$

is the 4th (unscaled) central moment [1] *of* $\boldsymbol{t}(\boldsymbol{y})$ *and* $\partial_i \boldsymbol{h}_L(\boldsymbol{x}) \stackrel{\text{def}}{=} \partial \boldsymbol{h}_L(\boldsymbol{x}) / \partial \boldsymbol{\theta}_i$.

Notably, as $\boldsymbol{t}(\boldsymbol{y})$ is the sufficient statistics of an exponential family, the derivatives of the log-partition $F(\boldsymbol{h}_L(\boldsymbol{x}))$, w.r.t. the natural parameters $\boldsymbol{h}$, are equivalent to the *cumulants* of $\boldsymbol{t}(\boldsymbol{y})$. The cumulants correspond to the coefficients of the Taylor expansion of the logarithm of the moment generating function [11]. Importantly, the cumulants of order 3 and below are equivalent to the central moments (see *e.g.* Lemma 2). However, this in not the case for the 4th central moment which much be expressed as a quadratic combination of the 2nd and 4th cumulants.

Lemma 5.

$$\mathcal{K}_{abcd}(\boldsymbol{t}) = \kappa_{a,b,c,d} + \mathcal{I}_{ab}(\boldsymbol{h}_L) \cdot \mathcal{I}_{cd}(\boldsymbol{h}_L) + \mathcal{I}_{ac}(\boldsymbol{h}_L) \cdot \mathcal{I}_{bd}(\boldsymbol{h}_L) + \mathcal{I}_{ad}(\boldsymbol{h}_L) \cdot \mathcal{I}_{bc}(\boldsymbol{h}_L),$$

where

$$\kappa_{a,b,c,d} \stackrel{\text{def}}{=} \frac{\partial^4 F(\boldsymbol{h}_L(\boldsymbol{x}))}{\partial \boldsymbol{h}_a \partial \boldsymbol{h}_b \partial \boldsymbol{h}_c \partial \boldsymbol{h}_d}.$$

In the 1D case, the 4th central moment simplifies to $\mathcal{K}(\boldsymbol{t}) = \partial^4 F(\boldsymbol{h}_L) + 3(\partial^2 F(\boldsymbol{h}_L))^2$.

For the second estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$, the variance only depends on the 2nd central moment.

Theorem 6.

$$\operatorname{Var}\left(-\frac{\partial^2 \ell}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j}\right) = \partial^2_{ij} \boldsymbol{h}_L^{\alpha}(\boldsymbol{x}) \partial^2_{ij} \boldsymbol{h}_L^{\beta}(\boldsymbol{x}) \mathcal{I}_{\alpha\beta}(\boldsymbol{h}_L),$$

where $\partial^2_{ij} \boldsymbol{h}_L(\boldsymbol{x}) \stackrel{\text{def}}{=} \partial^2 \boldsymbol{h}_L(\boldsymbol{x}) / \partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j$ [2].

Remark 6.1. *In Theorems 4 and 6, the 4D tensor* $(\mathcal{K}_{abcd}(\boldsymbol{t}) - \mathcal{I}_{ab}(\boldsymbol{h}_L) \cdot \mathcal{I}_{cd}(\boldsymbol{h}_L))$ *and the matrix* $\mathcal{I}_{\alpha\beta}(\boldsymbol{h}_L)$ *are the variance of the exponentially family FIM estimators* $\hat{\mathcal{I}}_1(\boldsymbol{h}_L)$ *and* $\hat{\mathcal{I}}_2(\boldsymbol{h}_L)$. *Theorems 4 and 6 can be regarded as the transformation of the variance w.r.t. the coordinate transformation* $\boldsymbol{\theta} \to \boldsymbol{h}_L$ *of the exponential family.*

The variance in Theorems 4 and 6 has two different components: ① the derivatives of the deep learning network $\boldsymbol{x} \to \boldsymbol{h}_L$; and ② the central (unscaled) moments of $\boldsymbol{t}(\boldsymbol{y})$. The 4D tensor $\mathcal{K}_{abcd}(\boldsymbol{t})$ and the 2D FIM $\mathcal{I}_{ab}(\boldsymbol{h}_L)$ correspond to the 4th and 2nd central moments of $\boldsymbol{t}(\boldsymbol{y})$, respectively. Intuitively, the larger the scale of the Jacobian or the Hessian of the neural network mapping $\boldsymbol{\theta} \to \boldsymbol{h}_L$, and/or the larger the central moments of exponential family, the larger the corresponding variance when estimating the FIM.

3.2 Variance Bounds

We aim to derive meaningful and simpler upper bounds of the variances presented in Theorems 4 and 6. However, these upper bounds are not straightforward to understand or compute. Using the Cauchy-Schwarz inequality, we can "decouple" the derivatives of the neural network mapping and the central moments of the exponential family and separate them into different terms.

Corollary 7.

$$\operatorname{Var}\left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_i} \frac{\partial \ell}{\partial \boldsymbol{\theta}_j}\right) \leq \|\partial_i \boldsymbol{h}_L(\boldsymbol{x})\|_2^2 \cdot \|\partial_j \boldsymbol{h}_L(\boldsymbol{x})\|_2^2 \cdot \|\mathcal{K}(\boldsymbol{t}) - \mathcal{I}(\boldsymbol{h}_L) \otimes \mathcal{I}(\boldsymbol{h}_L)\|_F,$$

where $\|\cdot\|_F$ *is the Frobenius norm and* $\otimes$ *is the tensor-product:* $(\mathcal{I}(\boldsymbol{h}_L) \otimes \mathcal{I}(\boldsymbol{h}_L))_{abcd} \stackrel{\text{def}}{=} \mathcal{I}_{ab}(\boldsymbol{h}_L) \cdot \mathcal{I}_{cd}(\boldsymbol{h}_L)$.

[1] The kurtosis of a random variable is defined by its 4th standardized (both centered and normalized) moment. Here, $\mathcal{K}(\cdot)$ denotes the 4th central moment but not the kurtosis.

[2] In this paper, the derivatives are by default taken w.r.t. $\boldsymbol{\theta}$. Therefore, $\partial_i \stackrel{\text{def}}{=} \frac{\partial}{\partial \boldsymbol{\theta}_i}$ and $\partial^2_{ij} \stackrel{\text{def}}{=} \frac{\partial^2}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j}$.

Table 1: Cumulants of common exponential family distributions, given as derivatives of the log-partition function. The BASE column corresponds to the base measure of the distribution. † The normal distribution has unit standard deviation ($\sigma = 1$).

DIST.	$F(\boldsymbol{h})$	$\boldsymbol{h} := \boldsymbol{h}_L$	$\partial^2 F(\boldsymbol{h})$	$\partial^4 F(\boldsymbol{h})$	BASE
BERNOULLI	$\log(1+\exp(\boldsymbol{h}))$	$\log {}^{p}/_{1-p}$	$p(1-p)$	$p(1-p)(6p^2-6p+1)$	1
NORMAL†	${}^{\boldsymbol{h}^2}/_2$	μ	1	0	${}^{(e^{-x^2/2})}/\sqrt{2\pi}$
POISSON	$\exp(\boldsymbol{h})$	$\log \lambda$	λ	λ	${}^{1}/_{y!}$

Corollary 8.

$$\mathrm{Var}\left(-\frac{\partial^2 \ell}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j}\right) \leq \|\partial^2_{ij} \boldsymbol{h}_L(\boldsymbol{x})\|_2^2 \cdot \|\mathcal{I}(\boldsymbol{h}_L)\|_F.$$

We consider an additional upper-bound to split up the "1D central moment" terms from the cross terms. By the triangle inequality,

$$\|\mathcal{K}(\boldsymbol{t}) - \mathcal{I}(\boldsymbol{h}_L) \otimes \mathcal{I}(\boldsymbol{h}_L)\|_F \leq \sqrt{\sum_{a=1}^{n_L} \left(\partial^4_{aaaa} F(\boldsymbol{h}_L) + 2\partial^2_{aa} F(\boldsymbol{h}_L)\right)^2}$$
$$+ \sqrt{\sum_{\substack{i,j,k,l \\ |\{i,j,k,l\}|>1}} \left(\partial^4_{ijkl} F(\boldsymbol{h}_L) + \partial^2_{ik} F(\boldsymbol{h}_L) \cdot \partial^2_{jl} F(\boldsymbol{h}_L) + \partial^2_{il} F(\boldsymbol{h}_L) \cdot \partial^2_{jk} F(\boldsymbol{h}_L)\right)^2}, \tag{10}$$

where $|\{i,j,k,l\}|$ denotes the set cardinality. Similarly,

$$\|\mathcal{I}(\boldsymbol{h}_L)\|_F \leq \sqrt{\sum_{a=1}^{n_L} (\partial^2_{aa} F(\boldsymbol{h}_L))^2} + \sqrt{\sum_{i \neq j} (\partial^2_{ij} F(\boldsymbol{h}_L))^2}. \tag{11}$$

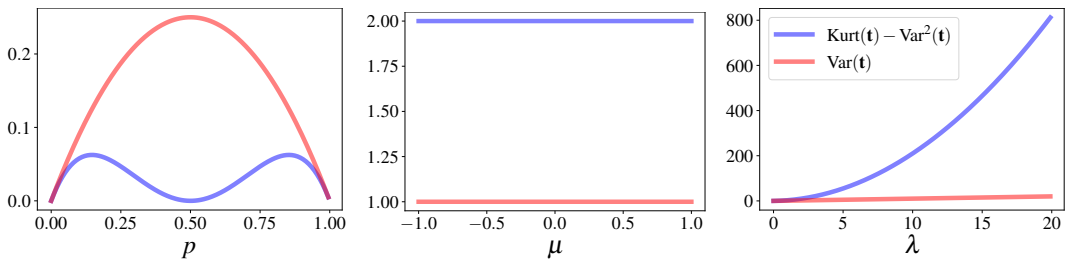
The size of the diagonal elements of the central moments is given by the 1st term on the RHS of Eqs. (10) and (11). It measures the element-wise central moments of $\boldsymbol{t}(\boldsymbol{y})$. The size of the non-diagonal elements, given by the 2nd term on the RHS of Eqs. (10) and (11), correspond to how different dimensions of $\boldsymbol{t}(\boldsymbol{y})$ interact with each other. If the last layer is 1D, *i.e.* $\boldsymbol{t}(\boldsymbol{y})$ is a scalar, then these cross-interaction terms disappear. In another simplified case, if the dimensions of $\boldsymbol{t}(\boldsymbol{y})$ is conditionally independent given $\boldsymbol{x}$, the cross-interaction terms also disappear. Understanding the scale of these 1D central moment terms helps us understand the scale of the variances.

In Table 1 we present some common simple 1D exponential families and their central moments. Furthermore, we plot $\mathcal{K}(\boldsymbol{t}) - \mathcal{I}(\boldsymbol{h}_L) \cdot \mathcal{I}(\boldsymbol{h}_L)$ and $\mathcal{I}(\boldsymbol{h}_L)$ in Fig. 1. As such we can compare each of the central moment quantities for the two estimators' variance for these examples.

In particular, we find when considering the Bernoulli distribution the central moment quantity in Theorem 4 is less than or equal to the variance found in Theorem 6 regardless of the value of p given by $\boldsymbol{h}_L$. Notably, when $p = 0.5$, we find that the variance of the first estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ will be 0 – regardless of the structure of $\boldsymbol{h}_L$. For the normal distribution (with unit standard deviation), both central moment quantities are constant. Thus comparing the scale of variances needs to take into account the structure of $\boldsymbol{h}_L$. For the Poisson distribution, the variance $\mathcal{I}(\boldsymbol{h}_L)$ increases linearly with the average number of events λ. On the other hand, $\mathcal{K}(\boldsymbol{t}) - \mathcal{I}(\boldsymbol{h}_L) \cdot \mathcal{I}(\boldsymbol{h}_L)$ increases quadratically. Thus, if $\boldsymbol{h}_L$ is large the variance found in Theorem 4 will increase faster than Theorem 6. This suggest that we either want to use estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ with a Poisson distribution or control the scale of neural network output $\boldsymbol{h}_L$.

3.3 Positive Definiteness

By definition, the FIM of any statistical model is positive semidefinite (p.s.d.). The first estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ is naturally on the p.s.d. manifold (space of p.s.d. matrices). On the other hand, $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ can



(a) Bernoulli. (b) Normal ($\sigma = 1$). (c) Poisson.

Figure 1: Moment quantities in variances of estimators given by Theorems 4 and 6.

"fall off" the p.s.d. manifold. It is important to examine the likelihood for $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ having a negative spectrum and the corresponding scale, so any algorithm (*e.g.* natural gradient) relying of the FIM being p.s.d. can be adapted.

Eq. (8) can be re-expressed as the sum of a p.s.d. matrix and a linear combination of n_L symmetric matrices. We provide a probabilistic bound that $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ stays on the manifold given conditions on the variance of $\boldsymbol{t}(\boldsymbol{x})$ and eigenvalues on the sum of matrices.

Theorem 9. *Let $\lambda^{\mathcal{I}}_{\min}$ denote the smallest eigenvalue of $\mathcal{I}(\boldsymbol{\theta})$ and $\lambda^{a}_{\max}$ denote the largest eigenvalue of $\partial^2 \boldsymbol{h}^a_L(\boldsymbol{x})$. Let $\lambda^{\star}_{\max} \stackrel{def}{=} \max\{1, \lambda^1_{\max}, \dots \lambda^{n_L}_{\max}\}$. Then with probability at least*

$$\prod_{a=1}^{n_L} (1 - \iota_a)\,, \quad \textit{where } \iota_a = \frac{n_L^2 \cdot (\lambda^{\star}_{\max})^2 \cdot \mathrm{Var}(\boldsymbol{t}_a)}{n_L^2 \cdot (\lambda^{\star}_{\max})^2 \cdot \mathrm{Var}(\boldsymbol{t}_a) + N^2 \cdot (\lambda^{\mathcal{I}}_{\min})^2},$$

the estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ with N samples is a p.s.d. matrix.

The probability found in Theorem 9 can be further weakened for interpretability. Using Bernoulli's inequality, with probability at least $(1 - n_L \max_a \iota_a)$, $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ is p.s.d.

It follows the bound becomes uninformative as the output layer size n_L increases and as the maximal eigenvalues of the Hessian of $\boldsymbol{h}_L$ increases. This also occurs as the variance of $\boldsymbol{t}_a$ increases. In contrast, as the true FIM's minimal eigenvalue increases the lower bound increases. In particular, with sample rate $\mathcal{O}(N^{-2})$, estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ will be a p.s.d. matrix. In practice, for over-parametrized networks, we have $\lambda^{\mathcal{I}}_{\min} = 0$ which results in Theorem 9 becoming trivial. In any case, we need to consider the scale of the negative spectrum.

Theorem 10. *Let $\lambda^a > 0$ denote the spectral radius (largest absolute value of the spectrum) of $\partial^2 \boldsymbol{h}^a_L(\boldsymbol{x})$ and $\lambda_{\min}(\hat{\mathcal{I}}_2(\boldsymbol{\theta}))$ denotes the smallest eigenvalue of $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$. We have*

$$\lambda_{\min}\left(\hat{\mathcal{I}}_2(\boldsymbol{\theta})\right) \geq -\lambda^a \left| \boldsymbol{\eta}_a - \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{t}_a(\boldsymbol{y}_i) \right|.$$

The above Theorem 10 guarantees that in the worst case, the scale of the negative spectrum of $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ is controlled. By Lemma 2, $\mathrm{Var}(\boldsymbol{\eta}_a - \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{t}_a(\boldsymbol{y}_i)) = \frac{1}{N}\mathcal{I}_{aa}(\boldsymbol{\theta})$. Therefore, as N increases or $\mathcal{I}_{aa}(\boldsymbol{\theta})$ decreases, the negative spectrum of $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ will shrink.

3.4 Convergence Rate

The rate of convergence for each of the estimators are of particular interest when considering the practical viability of using the estimators. Through the Chebyshev inequality, we can get a simple element-wise convergence rate for each estimator.

Lemma 11. *Let $0 < \varepsilon < 1$. Then*

$$\left| [\hat{\mathcal{I}}_1(\boldsymbol{\theta})]_{ij} - [\mathcal{I}(\boldsymbol{\theta})]_{ij} \right| \leq \frac{1}{\sqrt{\varepsilon N}} \cdot \mathrm{Var}\left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_i} \frac{\partial \ell}{\partial \boldsymbol{\theta}_j} \right)$$

and

$$\left|[\hat{\mathcal{I}}_2(\boldsymbol{\theta})]_{ij} - [\mathcal{I}(\boldsymbol{\theta})]_{ij}\right| \leq \frac{1}{\sqrt{\varepsilon N}} \cdot \mathrm{Var}\left(-\frac{\partial^2 \ell}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j}\right)$$

hold with probability at least $1 - \varepsilon$.

Each of the estimators have a convergence rate of $\mathcal{O}(N^{-1/2})$. The rate's constants are determined by the variances of the estimators given by Theorems 4 and 6, and are influenced by the derivatives of the neural network and the moments of the output exponential family.

4 Effect of Neural Network Derivatives

In the previous section, we considered two estimators for the FIM. In particular, we only considered aspects of the exponential family in our model's setup. However, each of the results are dependent on the variances provided in Theorems 4 and 6. As such, the estimators also need to take into account the structure of the deep learning network. For the first estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$, the variance is dependent on the 1st-order gradient information of the network $\boldsymbol{h}_L(\boldsymbol{x})$. For the second estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$, additionally, the 2nd-order gradient information is also needed.

We give in closed form the layer-wise gradient of the log-likelihood ℓ and the last layer's output $\boldsymbol{h}_L$.

Lemma 12.

$$\frac{\partial \ell}{\partial \boldsymbol{W}_{l-1}} = \frac{\partial \ell}{\partial \boldsymbol{h}_l} \boldsymbol{h}_{l-1}^\top, \quad \frac{\partial \ell}{\partial \boldsymbol{h}_l} = \boldsymbol{B}_{l-1}\left(\boldsymbol{t}(\boldsymbol{y}) - \boldsymbol{\eta}(\boldsymbol{h}_L)\right), \quad \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{W}_l} = \boldsymbol{B}_l \boldsymbol{e}_a \boldsymbol{h}_l^\top,$$

where $\boldsymbol{e}_a$ *is the* a^{th} *standard basis vector and* $\boldsymbol{B}_l$ *is recursively defined by*

$$\begin{aligned} \boldsymbol{B}_{L-1} &= \boldsymbol{D}_{L-1}, \\ \boldsymbol{B}_{l-1} &= \boldsymbol{D}_{l-1} \boldsymbol{W}_{l-1}^\top \boldsymbol{B}_l, \\ \boldsymbol{D}_l &= \mathrm{diag}(\sigma'(\boldsymbol{W}_l \boldsymbol{h}_l)). \end{aligned}$$

Notice that

$$\hat{\mathcal{I}}_1(\boldsymbol{h}_l) = \frac{1}{N} \sum_{i=1}^{N} \frac{\partial \ell_i}{\partial \boldsymbol{h}_l} \frac{\partial \ell_i}{\partial \boldsymbol{h}_l^\top} = \boldsymbol{B}_l \left(\frac{1}{N} \sum_{i=1}^{N} \left(\boldsymbol{t}(\boldsymbol{y}) - \boldsymbol{\eta}(\boldsymbol{h}_L)\right) \left(\boldsymbol{t}(\boldsymbol{y}) - \boldsymbol{\eta}(\boldsymbol{h}_L)\right)^\top \right) \boldsymbol{B}_l^\top. \tag{12}$$

By noticing that $\boldsymbol{B}_l$ is recursively evaluated from the last layer (shallowest) to previous layers, the FIM of the hidden representation $\boldsymbol{h}_l$ can be recursively estimated based on $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$. It is similar to back-propagation, except that FIMs are back-propagated instead of gradients of the network.

To investigate the first estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and the loss landscape, we consider the following to bound the square of the Frobenius norm as present in Corollary 7.

Lemma 13. *Suppose that the activation function has bounded gradient* $|\sigma'(z)| \leq \sqrt{S}$. *Then*

$$\left\| \frac{\partial \boldsymbol{h}_L^a}{\partial \boldsymbol{W}_{l-1}} \right\|_F^2 \leq \left(\prod_{i=l}^{L} n_i \right) \cdot \left(\prod_{i=l}^{L-1} \|\boldsymbol{W}_{i-1}\|_F^2 \right) \cdot S^{L-l} \cdot \|\boldsymbol{h}_{l-1}\|_2^2. \tag{13}$$

Notably, given Lemma 13 we can see that the gradient constrained to the deep learning layers will be small if the weights and activation function gradients are bounded to be small. Common activation function have both bounded output and first-order derivative, or at least are locally Lipschitz, *i.e.*, sigmoid and ReLU activation functions. Thus, the interesting characteristics of the neural network are the size of the layers and the scale of the weights. If the size of the layers is immutable, then small weights help to control the bound. Regularizing the neural network in training is potentially a sufficient condition for bounding the variance of estimator $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$, while also potentially providing small variance values for the parameters.

The Hessian of the mapping $\boldsymbol{\theta} \to \boldsymbol{h}_L$, given by the 3D tensor $\partial^2 \boldsymbol{h}^\alpha(\boldsymbol{x})$, is important characterize the variance of the FIM estimator $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$. Corollary 8 leads to

$$\mathrm{Var}\left(\hat{\mathcal{I}}_2(\boldsymbol{\theta})\right) \leq \frac{\|\mathcal{I}(\boldsymbol{h}_L)\|_F}{N} \sqrt{\sum_{a=1}^{n_L} \partial^2 \boldsymbol{h}_L^a(\boldsymbol{x}) \circ \partial^2 \boldsymbol{h}_L^a(\boldsymbol{x})}, \tag{14}$$

where $\circ$ is the Hadamard product and the square root $\sqrt{\cdot}$ is applied element-wise. By Eq. (8), $\hat{\mathcal{I}}_2(\boldsymbol{\theta}) - \mathcal{I}(\boldsymbol{\theta})$ is a linear combination of the Hessian matrices $\partial^2 \boldsymbol{h}_L^a(\boldsymbol{x})$ with the coefficients given by $\boldsymbol{\eta}_a - \frac{1}{N}\sum_{i=1}^N \boldsymbol{t}_a(\boldsymbol{y}_i)$. For the given target $\tilde{\boldsymbol{y}}$, the log-likelihood is denoted as $\tilde{l} \stackrel{\text{def}}{=} \log p(\tilde{\boldsymbol{y}} \mid \boldsymbol{x}, \boldsymbol{\theta})$. By Eq. (5),

$$-\frac{\partial^2 \tilde{l}}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}} - \mathcal{I}(\boldsymbol{\theta}) = (\boldsymbol{\eta}_a - \boldsymbol{t}_a(\tilde{\boldsymbol{y}})) \frac{\partial^2 \boldsymbol{h}_L^a}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top}. \tag{15}$$

Therefore the Hessian of the loss $-\tilde{l}$ differs from the FIM with another linear combination $\partial^2 \boldsymbol{h}_L^a$. During learning, the sampled $\boldsymbol{y}_i \sim p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta})$ will be close to the given $\tilde{\boldsymbol{y}}$ and our $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ can be regarded as the Hessian of the loss. The presented formal statements describe how the Hessian (which corresponds to the loss landscape) differs from the underlying FIM (which corresponds to the geometric structure of the parameter space). Interestingly, the information geometry only depends on the Jacobian of the map $\boldsymbol{\theta} \to \boldsymbol{h}_L$. The smaller the scale of the Hessian $\partial^2 \boldsymbol{h}_L^a(\boldsymbol{x})$, the closer the curvature of the loss surface is to that of the FIM.

5 Conclusion

The FIM reveals the intrinsic geometric structure of the parameter manifold. It is invariant to reparametrization and gives a theoretical limit of the minimum variance during parameter estimation. These perspectives lead to practical methods such as the natural gradient. The true FIM $\mathcal{I}(\boldsymbol{\theta})$ is usually expensive to obtain if at all possible. Estimators of the FIM based on empirical samples is used in the deep learning practice. We analyzed two different estimators $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ of the FIM of deep neural network with exponential family output units. These estimators are convenient to compute using auto-differentiation frameworks but randomly deviates from $\mathcal{I}(\boldsymbol{\theta})$. Our central results, Theorems 4 and 6 present the variance of $\hat{\mathcal{I}}_1(\boldsymbol{\theta})$ and $\hat{\mathcal{I}}_2(\boldsymbol{\theta})$ in closed form, which further extends to upper bounds in a simpler form. These variances are affected by the derivatives of neural network output $\boldsymbol{h}_L$ with respect to the weight parameters $\boldsymbol{\theta}$, as well as the property of $\boldsymbol{h}_L$ as an exponential family distribution. A large scale of 1st- and/or 2nd-order derivatives will lead to large variance when estimating the FIM. Our analytical results can be useful to measure the quality of the estimated FIM and could lead to variance reduction techniques.

References

[1] S. Amari. *Information Geometry and Its Applications*, volume 194 of *Applied Mathematical Sciences*. Springer-Verlag, Berlin, 2016.

[2] Shun-ichi Amari, Ryo Karakida, and Masafumi Oizumi. Fisher information and natural gradient learning in random deep networks. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 694–702. PMLR, 2019.

[3] K. O. Bowman and L. R. Shenton. *Properties of Estimators for the Gamma Distribution*. Marcel Dekker, New York, 1988.

[4] Maud Delattre and Estelle Kuhn. Estimating Fisher information matrix in latent variable models based on the score function. In *European Meeting of Statisticians (EMS)*, 2019.

[5] Tomohiro Hayase and Ryo Karakida. The spectrum of fisher information of deep networks achieving dynamical isometry. In *International Conference on Artificial Intelligence and Statistics*, pages 334–342. PMLR, 2021.

[6] Ryo Karakida, Shotaro Akaho, and Shun-ichi Amari. Universal statistics of Fisher information in deep neural networks: Mean field approach. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1032–1041. PMLR, 2019.

[7] F Kunstner, L Balles, and P Hennig. Limitations of the empirical Fisher approximation for natural gradient descent. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, pages 4133–4144. Curran Associates, Inc., 2020.

[8] E.L. Lehmann and George Casella. *Theory of Point Estimation*. Springer, second edition, 1998.

[9] James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

[10] James Martens and Roger Grosse. Optimizing neural networks with Kronecker-factored approximate curvature. In *International conference on machine learning*, pages 2408–2417. PMLR, 2015.

[11] Peter McCullagh. *Tensor methods in statistics*. Courier Dover Publications, 2018.

[12] Yann Ollivier. Riemannian metrics for neural networks i: feedforward networks. *Information and Inference: A Journal of the IMA*, 4(2):108–153, 2015.

[13] H. Park, S.-I Amari, and K. Fukumizu. Adaptive natural gradient learning algorithms for various stochastic models. *Neural Networks*, 13(7):755 – 764, 2000.

[14] R. Pascanu and Y. Bengio. Revisiting natural gradient for deep networks. In *ICLR*, 2014.

[15] Jeffrey Pennington and Pratik Worah. The spectrum of the fisher information matrix of a single-hidden-layer neural network. In *NeurIPS*, pages 5415–5424, 2018.

[16] Nicol N Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural computation*, 14(7):1723–1738, 2002.

[17] Ke Sun and Frank Nielsen. Relative Fisher information and natural gradient for learning large modular models. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3289–3298. PMLR, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Abstract summarizes our findings and the end of the introduction bullet poitns our contributions.
   (b) Did you describe the limitations of your work? [Yes] Limitations are discussed within the discussion paragraphs after theorems/lemmas/results are presented.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] We believe that our work sits within the theoretic space where there are no societal impacts.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] No breaches with respect to the online guidelines.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Full assumptions are listed in the theorems/lemmas/results. In particular we explicitly point out the assumption of the particular model we are considering (Eq. (3)) and the smoothness conditions we required (Theorem 3).
   (b) Did you include complete proofs of all theoretical results? [Yes] All complete proofs are presented in the supplemental material.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A] Only simple plots were made of well known central moments. No experiments.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A] No experiments.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] No experiments.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A] No experiments.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A] No other assets.
   (b) Did you mention the license of the assets? [N/A] No other assets.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A] No new assets.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] No data.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] No data.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]