Approximate Fisher Information Matrix to Characterise the Training of Deep Neural Networks

Nikita Kiselev

MIPT, 2024

September 24, 2024

Motivation

Main idea

Training of deep neural networks relies heavily on the careful selection of mini-batch sizes and learning rates. This selection process is often subjective and lacks clear guidelines. The proposed methodology use efficiently computed measures derived from the Fisher information matrix. These measures can help practitioners monitor and control the training process, leading to improved training convergence and generalization.

SGD objective

Image Classification task

Given the dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{|\mathcal{D}|}$, where the i^{th} image $\mathbf{x}_i : \Omega \to \mathbb{R}$ is annotated with the label $y_i \in \{1, ..., C\}$. This dataset is divided into training $\mathcal{T} \in \mathcal{D}$ and testing $\mathcal{S} \in \mathcal{D}$. The full model is defined by $f(\mathbf{x}, \theta)$, where $\theta \in \mathbb{R}^P$ denotes all model parameters. The objective is to minimise the multi-class cross entropy loss $\ell(\cdot)$ on the training set $\mathcal{T}$, as follows:

$$\theta^* = \arg\min_{\theta} \frac{1}{|\mathcal{T}|} \sum_{i \in \mathcal{T}} \ell\left(y_i, f(\mathbf{x}_i, \theta)\right). \tag{1}$$

SGD

$$\theta_{k+1} = \theta_k - \frac{\alpha_k}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} \nabla \ell(y_i, f(\mathbf{x}_i, \theta_k)), \tag{2}$$

where $\mathcal{B}_k$ is the mini-batch for the k^{th} iteration of the minimisation process.

Hessian matrix approximation

Loss landscape shape

The shape of the loss function can be characterised by the spectrum of the $\nabla^2 \ell(y_i, f(\mathbf{x}_i, \theta_k))$, where sharpness is defined by the magnitude of the eigenvalues.

Fisher information matrix

We define the Fisher information matrix as follows:

$$\mathbf{F}_k = \nabla \ell(y_{i \in \mathcal{B}_k}, f(\mathbf{x}_{i \in \mathcal{B}_k}, \theta_k)) \nabla \ell(y_{i \in \mathcal{B}_k}, f(\mathbf{x}_{i \in \mathcal{B}_k}, \theta_k))^\top, \tag{3}$$

where $\mathbf{F}_k \in \mathbb{R}^{P \times P}$.

Idea

Fisher matrix can be a reasonable approximation of the Hessian matrix at the end of the training (assuming sufficient training has been done).

Fisher matrix approximation

Fisher matrix calculation

The calculation of $\mathbf{F}_k$ depends on the Jacobian $\mathbf{J}_k = \nabla \ell(y_{i \in \mathcal{B}_k}, f(\mathbf{x}_{i \in \mathcal{B}_k}, \theta_k))$, with $\mathbf{J}_k \in \mathbb{R}^{P \times |\mathcal{B}_k|}$.

Computational and memory complexity

Given that $\mathbf{F}_k = \mathbf{J}_k \mathbf{J}_k^\top \in \mathbb{R}^{P \times P}$ scales with $P \in [O(10^6), O(10^7)]$ and that we are only interested in the spectrum of $\mathbf{F}_k$, we can compute instead $\widetilde{\mathbf{F}}_k = \mathbf{J}_k^\top \mathbf{J}_k \in \mathbb{R}^{|\mathcal{B}_k| \times |\mathcal{B}_k|}$ that scales with the mini-batch size $|\mathcal{B}_k| \in [O(10^1), O(10^2)]$.

Spectrum consistency

Note that the rank of $\widetilde{\mathbf{F}}_k$ and $\mathbf{F}_k$ is at most $|\mathcal{B}_k|$, which means that the spectra of $\widetilde{\mathbf{F}}_k$ and $\mathbf{F}_k$ are the same given that both will have at most $|\mathcal{B}_k|$ non-zero eigenvalues.

Methodology

1. Running average of the truncated condition number of $\widetilde{\mathbf{F}}_k$

$$\bar{C}_K = \frac{1}{K}\sum_{k=1}^{K} c_k, \quad c_k = \frac{\sigma_{\max}(\widetilde{\mathbf{F}}_k)}{\sigma^{+}_{\min}(\widetilde{\mathbf{F}}_k)}. \tag{4}$$

2. Weighted cumulative sum of the energy of $\widetilde{\mathbf{F}}_k$

$$L_K = \sum_{k=1}^{K} l_k, \quad l_k = \frac{\alpha_k}{|\mathcal{B}_k|}\left(\mathrm{Tr}\left(\widetilde{\mathbf{F}}_k\right)\right)^{\frac{1}{2}}. \tag{5}$$

3. Dynamic sampling

Change the mini-batch size during the training in order to achieve good convergence and generalisation.

Experiments Setup

- **Data:** CIFAR-10, CIFAR-100, SVHN, and MNIST.
- **Models:** ResNet and DenseNet.
- **Optimizer:** SGD with momentum of 0.9 and learning rate schedule as follows: $\times 0.1$ at the 50% training epochs, and $\times 0.1$ at the 75%.
- **Metric:** Accuracy on training and testing samples.

Mini-batch Size and Learning Rate

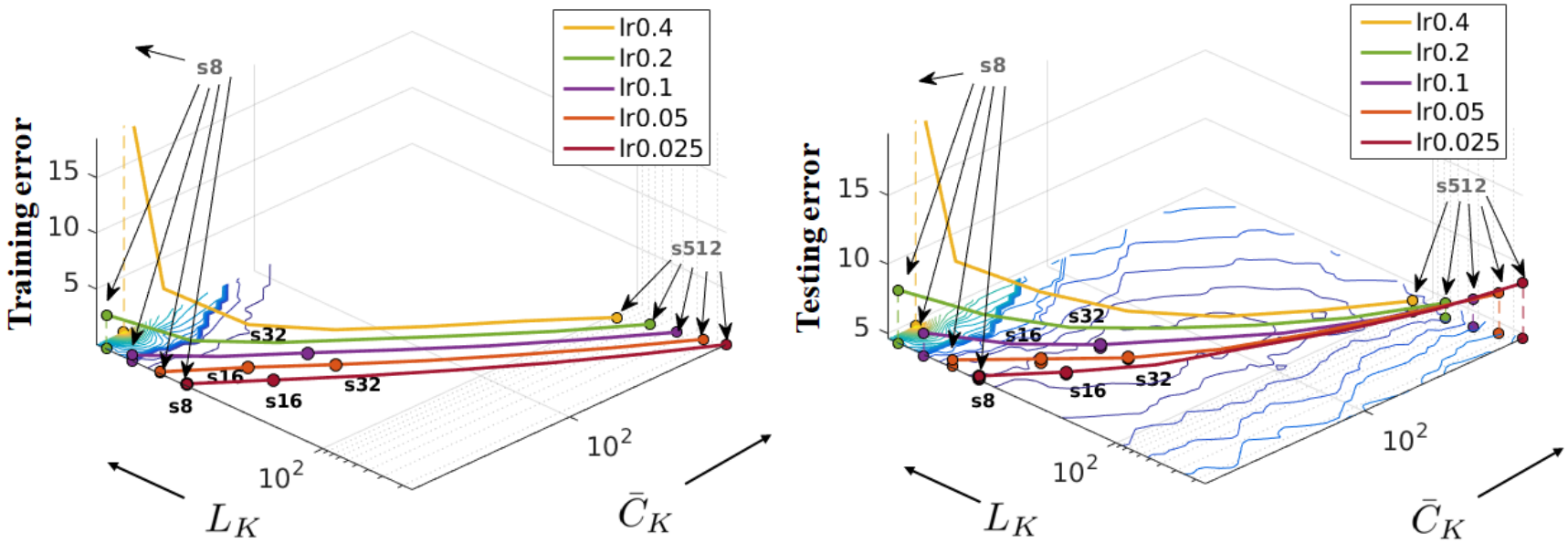


Figure: ResNet on CIFAR-10. 1) Each configuration has a unique $\bar{C}_K$ and L_K signature, where no configuration overlays over each other in the space; 2) $|\mathcal{B}_k|$ is directly proportional to $\bar{C}_K$ and inversely proportional to L_K; 3) α_k is directly proportional to $\bar{C}_K$ and L_K; and 4) small $\bar{C}_K$ and large L_K indicate poor training convergence, and large $\bar{C}_K$ and small L_K show poor generalisation, so the best convergence and generalisation requires a small value for both measures.

Functional Relations for the Proposed Measures

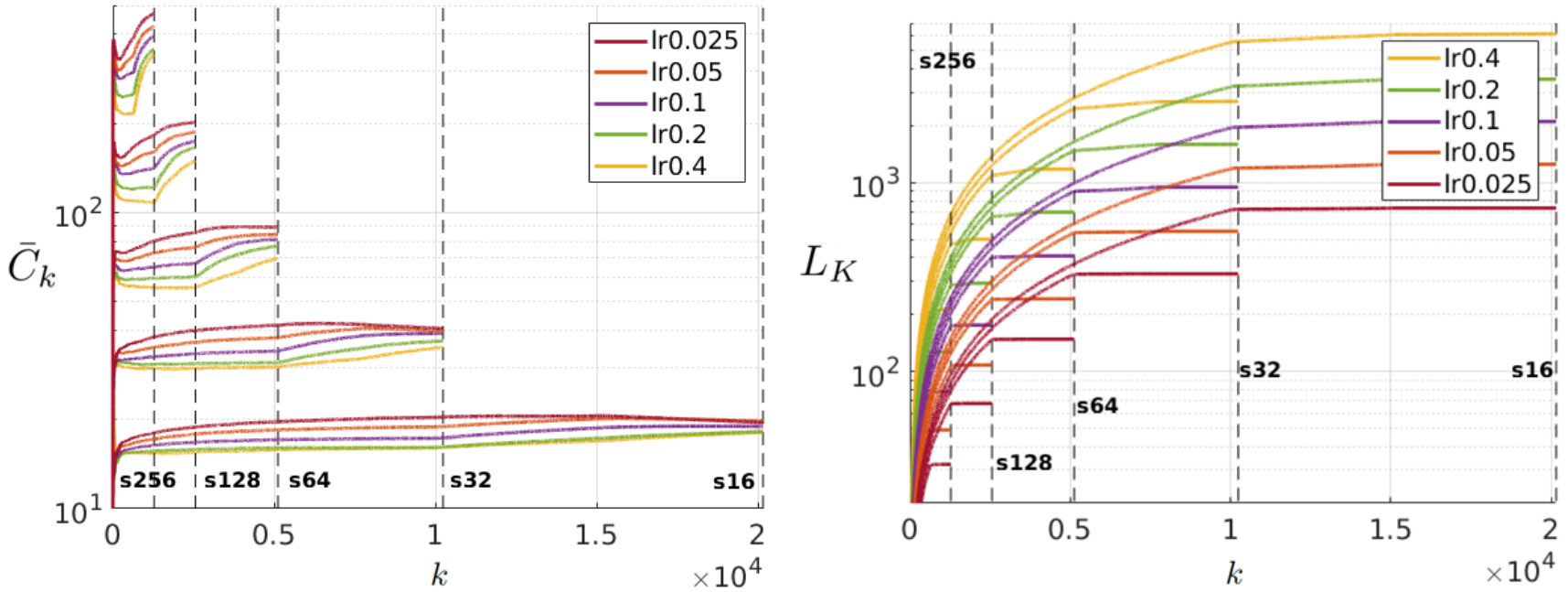


Figure: ResNet on CIFAR-10. 1) $\bar{C}_K$ tends to cluster at similar values for training processes performed with the same mini-batch sizes, independently of the learning rate; 2) L_K is more likely to cluster at similar values for training processes performed with the same learning rate, independently of the mini-batch size, particularly at the first half of the training.

The stability of the proposed measures

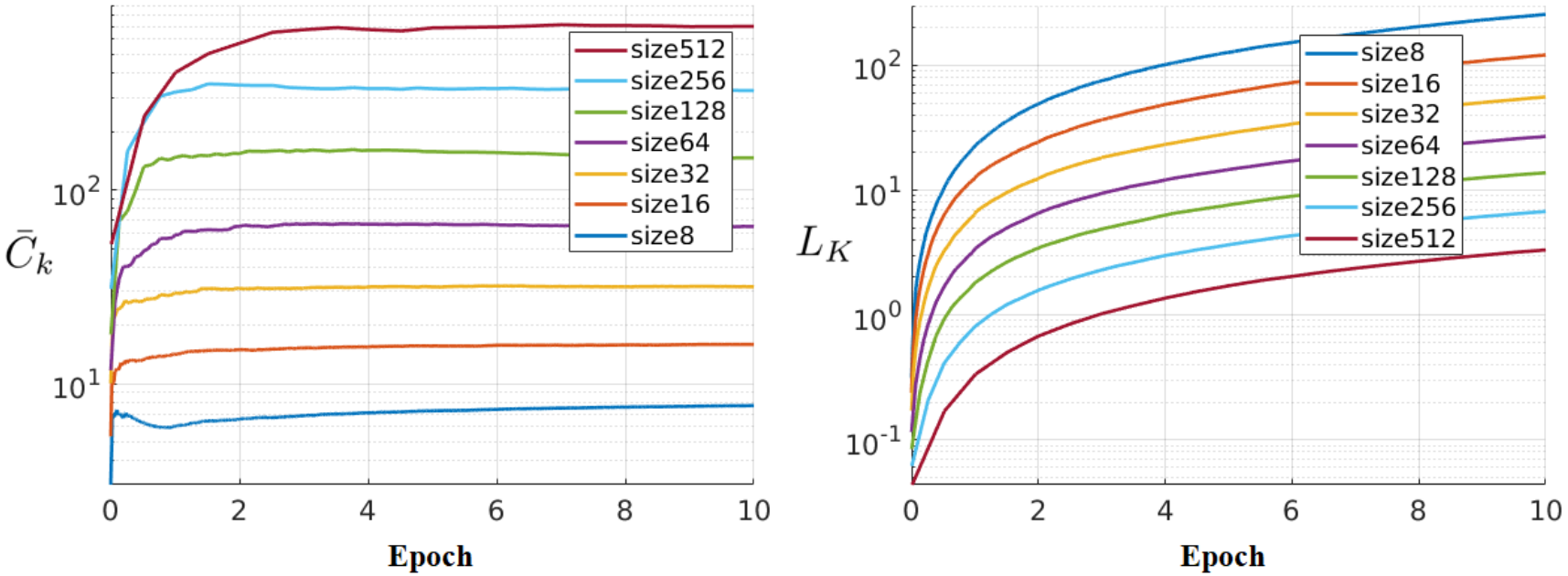


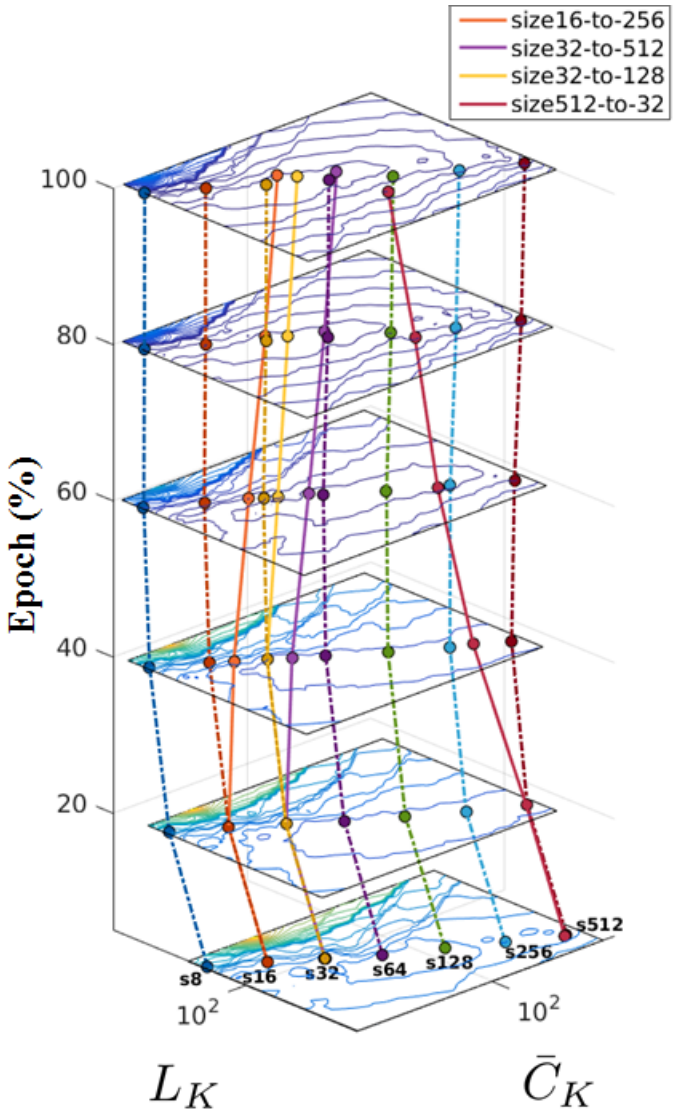
Figure: ResNet on CIFAR-10. Proposed measures are stable in terms of the relative positions of $\bar{C}_K$ and L_K values even during early training epochs.

Idea

Proposed $\bar{C}_K$ and L_K values can be used to guide dynamic sampling – a method that dynamically increases the mini-batch size during the training, by navigating the training procedure in the landscape of $\bar{C}_K$ and L_K.

Dynamic Sampling

- Divide the training process into five stages, each with equal number of training epochs and using a particular mini-batch size.
- $\bar{C}_K$ and L_K are pushed away from the initial mini-batch size region towards the final mini-batch size region.



Comparison of different sampling strategies

	Model (best of each)	CIFAR-10		
		s#	-$\emptyset$	**-MS**
ResNet	Name	**s32**	**s32-to-128**	**s32-to-128-MS**
	Test Error	$4.78\% \pm 0.05\%$	$4.90\% \pm 0.05\%$	$4.76\% \pm 0.13\%$
	p-value vs **s#**	–	0.0048	0.72
	p-value -$\emptyset$ vs **-MS**	–	0.029	
	Training Time (h)	7.7	7.0	7.1

Table: The comparison between the best "beacon" model **s#** (at **lr0.1**), and the best dynamic sampling models -$\emptyset$, and **-MS**. The latter (**-MS**) corresponds to sampling of mini-batch sizes over each different learning rate value.

Overview

In general, the results show that dynamic sampling allows a faster training and a similar classification accuracy, compared with the fixed sampling training of the beacons.

Literature

1. **Main article:** Liao Z. et al. Approximate fisher information matrix to characterize the training of deep neural networks // IEEE transactions on pattern analysis and machine intelligence.