Unveiling the Hessian's Connection to the Decision Boundary

Mahalakshmi Sabanayagam[1], Freya Behrens[2], Urte Adomaityte[3], Anna Dawid[4]

[1] Technical University of Munich, Germany
[2] Statistical Physics of Computation Lab, EPFL, Switzerland
[3] Department of Mathematics, King's College London, United Kingdom
[4] Center for Computational Quantum Physics, Flatiron Institute, New York

Summary

Motivation

Across deep learning setups, the Hessian of the training loss at the minimum exhibits some universal characteristics: its spectrum has few outliers, and the gradient information resides in the corresponding small subspace. **What is encoded by the top eigenvectors of the loss Hessian?**

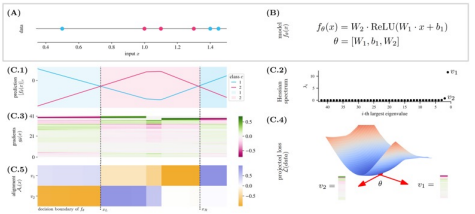
What did we do

We measure the cosine similarity (alignment) between the ***loss gradient of the individual input data and the eigenvectors of the loss Hessian*** at the minimum for various classification datasets and neural networks.

Contributions

- The top eigenvectors of the training loss Hessian encode the decision boundary learned by the neural network. Particularly, each eigenvector encodes a separate section of the decision boundary.
- The number of encoding eigenvectors usually equals the number of spectrum outliers (and the number of classes). However, more eigenvectors are needed to encode a complex, highly non-linear decision boundary.
- We propose a new, improved generalization measure that considers the simplicity of the decision boundary via the Hessian eigenvectors. In addition, we develop a technique to estimate the narrowest margin of the decision boundary in the input space.

An illustrative toy example



(A) The 1D toy dataset with 5 input points and 2 classes {pink, cyan}.

(B) A model f_θ with parameters θ takes an input x and returns logit probabilities for each class.

(C.1) Predictions of f_θ across the input space with $\hat{\theta}$ being a specific set of parameters that correctly classify the training data.

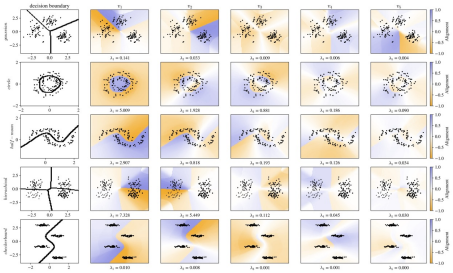
(C.2) There are two outliers in the Hessian eigenspectrum of the training loss calculated at the minimum $\hat{\theta}$. They correspond to the eigenvectors $v_1$and v_2 that are directions in the parameter space shown in **(C.4)**. When measuring their cosine similarity with gradients of the loss of individual points from the input **(C.3)**, we obtain the alignment in **(C.5)**.

Each outlier encodes one section of the decision boundary of the model f_θ relevant to the data that induced the loss landscape.

Take a look at the paper!

Top Hessian eigenvectors encode the decision boundary

Alignment of gradients of loss of input data with the top 5 Hessian eigenvectors at the minimum

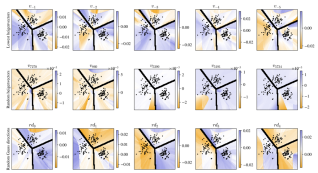


! Absolute alignment between the topmost eigenvectors and the points on the decision boundary is close-to-one

! each top eigenvector captures only a section of the boundary

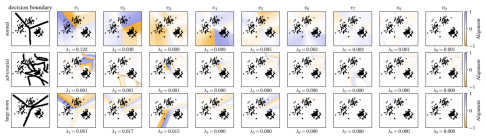
! the alignment does not necessarily switch between extreme values +1 and −1 across the decision boundary. The exact alignment values do not seem informative?

It is NOT accidental! Compare to the alignment with…

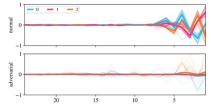


A complex boundary is characterized by many eigenvectors
☞ new generalization measure

Alignment of gradients of loss of input data with the top 9 Hessian eigenvectors at the minimum

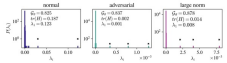


! more eigenvectors are needed to describe complex decision boundary



Alignment of gradients of loss of all training data with the top 25 Hessian eigenvectors for *gaussian*

! number of spectrum outliers depends on the decision boundary complexity (not only the number of classes)



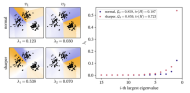
We propose a generalization measure detecting the complexity of the decision boundary

$$m_i = \frac{1}{n}\sum_{s=1}^{n} |\mathcal{A}_i(x_s)| \quad ; \quad \mathcal{G}_\theta = \frac{1}{p}\sum_{i=1}^{p} \mathbb{1}[m_i > \epsilon]$$

Ratio of the Hessian eigenvectors having on average non-zero* alignment with the training data.

Smaller the ratio, the simpler boundary, the better generalizing model.

* a larger alignment than a random direction



- our measure correctly identified well-generalizing models across all datasets and networks (including Iris, MNIST, CIFAR-10)
- It is also invariant to reparametrization!

Setting

classification problem

data $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$ where $x_i \in \mathbb{R}^d$ and $y_i \in \{1, \ldots, C\}$

overparametrized neural network

$f_\theta : \mathbb{R}^d \to \mathbb{R}^C \quad \theta \in \mathbb{R}^p \quad p \gg nd$

$\hat{y}_i = \arg\max f_\theta(x_i)$

Hessian of the training loss function computed at the minimum with eigenvalues and eigenvectors:

$H \in \mathbb{R}^{p \times p}$

$H_{i,j} = \frac{\partial^2}{\partial\theta_i \partial\theta_j}\mathcal{L}(\mathcal{D})$

λ_i and $v_i \in \mathbb{R}^p$

logit gradient $g_\theta : \mathbb{R}^d \to \mathbb{R}^p$

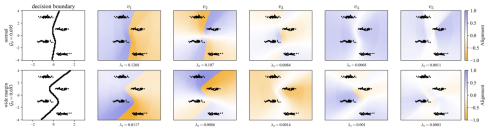
$g_\theta(x) = \frac{\partial}{\partial\theta}\mathcal{L}(\theta; \{x, \hat{y}\})$

Results here are for SGD + cross-entropy loss training, but they are invariant to the optimizer (e.g., Adam) and loss function (e.g., NLL loss).

Alignment between the *i*-th Hessian eigenvector and the logit gradient:

$$\mathcal{A}_i(x) = \frac{\langle g_\theta(x), v_i \rangle}{\|g_\theta(x)\| \|v_i\|}$$

Simplicity bias and estimation of the margin width



! the generalization measure cannot detect a poorer generalization caused by the simplicity bias!

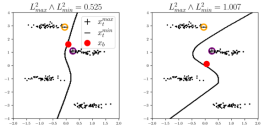
! however, the order of top eigenvectors follows the increasing margin of the sections of the boundary that they encode

☞ Margin estimation of various sections of the decision boundary using the corresponding Hessian eigenvector!

We need two input points for estimating the narrowest margin:

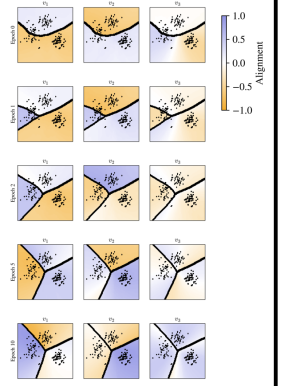
○ a training sample x_t that is closest to the boundary: it is chosen to have the largest alignment with the top Hessian eigenvector v_1

● a sample on the boundary x_b, close to x_t: we optimize the features of an input sample such that its gradient has a maximum alignment with the top Hessian eigenvector v_1



Better generalizing model → smaller generalization measure (simpler decision boundary) + larger margin

During training…



! The top Hessian eigenvectors encode the decision boundary also during training!

Scan for Code!