The Model centric Data manifold in Deep Learning.

Rita Fioresi

ALMA-AI Workshop on Foundations of AI,
Universitá di Bologna
rita.fioresi@unibo.it

April 27, 2021

The Geometric Structure of Data

Deep Learning and classification tasks:

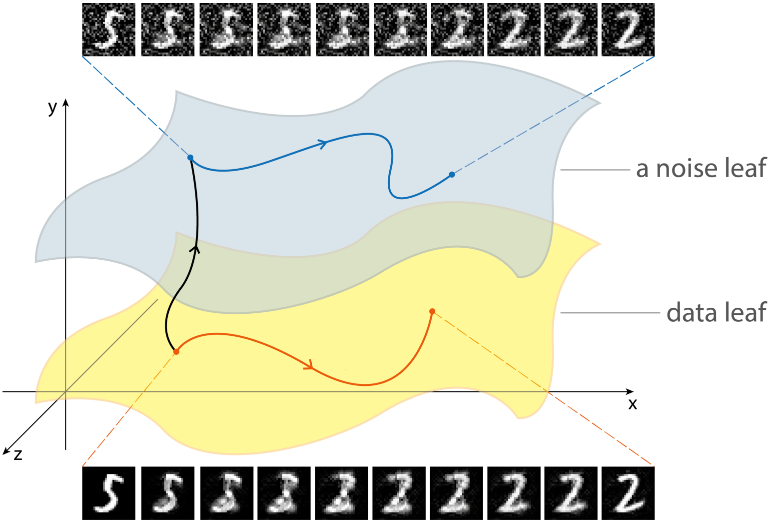
- Data occupies a domain in $\mathbb{R}^n$ (e.g. MNIST in $\mathbb{R}^{784}$, $n = 784 = 28 \times 28$ pixels)
- The data domain is mostly composed of meaningless noise: data occupy a thin region of it!

Main result:

1. A partially trained neural network decomposes the data domain in $\mathbb{R}^n$ as the disjoint union of submanifolds (the **leaves** of a foliation).
2. The dimension d of every submanifold (every leaf of the foliation) is bounded by the number of classes C of our classification model: $d << n$ (e.g. MNIST $d = 9 << 784$).

Data leaf versus Noise leaf

The data domain is the disjoint union of subdomains (foliation) and the **training data are all on one leaf**.



Information Geometry

Information Geometry: studies geometrical structures on manifolds in the parameter space and the data domain.

Amari, S.-I. *Natural gradient works efficiently in learning. Neural computation*, 10(2):251276, 1998.

Amari Loss: $l(x, w) = -\log(p(y|x, w))$

Loss function: $L(x, w) = \mathbb{E}_{y \sim q}[l(x, w)]$

$$L(x, w) = \mathbb{E}_{y \sim q}[-\log(p(y|x, w))] = \mathrm{KL}(q(y|x)||p(y|x, w)) + \mathrm{constant}$$

$p(y|x, w) = (p_i(y|x, w))_{i=1,\dots,C}$: discrete probability distribution of data x
C: classification labels y.
w: parameters

The Fisher matrix F and the Local Data Matrix G

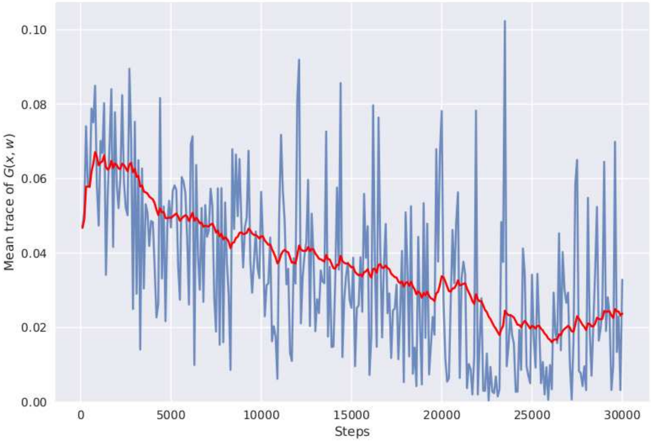
$$\begin{aligned} F(x,w) &= \mathbb{E}_{y\sim p}[\nabla_w \log p(y|x,w) \cdot (\nabla_w \log p(y|x,w))^T] \\ G(x,w) &= \mathbb{E}_{y\sim p}[\nabla_x \log p(y|x,w) \cdot (\nabla_x \log p(y|x,w))^T]. \end{aligned}$$

Key Facts:

$$\begin{aligned} \mathrm{KL}(p(y|x,w+\delta w)||p(y|x,w)) &= (\delta w)^T F(x,w)(\delta w) + \mathcal{O}(||\delta w||^3) \\ \mathrm{KL}(p(y|x+\delta x,w)||p(y|x,w)) &= (\delta x)^T G(x,w)(\delta x) + \mathcal{O}(||\delta x||^3) \end{aligned}$$

The Fisher matrix F provides a natural metric on the **parameter space** during dynamics of the stochastic gradient descent.
The Local Data matrix G provides a **natural metric on the data domain**.

The local data matrix G during optimization



This is why we do not want a fully trained model: the information is lost at equilibrium!

Properties of the local data matrix

1. $G(x, w)$ is a positive semidefinite symmetric matrix.
2. $\mathrm{rank}\ G(x, w) < C$.

Dataset	$G(x, w)$ size	$\mathrm{rank}\ G(x, w)$ bound
MNIST	784	10
CIFAR-10	3072	10
CIFAR-100	3072	100
ImageNet	150528	1000

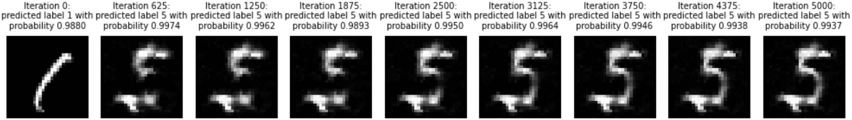
Main result.

1. At each point in the data domain in $\mathbb{R}^n$, $\ker G(x, w)^\perp$ is tangent to a submanifold (**data leaf**) of dimension $\mathrm{rank}\ G(x, w) < C$
2. G defines a foliation on $\mathbb{R}^n$ of rank at most C (**Frobenius Thm**).

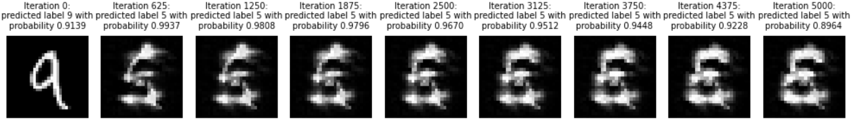
Moving on the data leaf: MNIST

Moving around in **on the data leaf**:

- We can connect any two data=images.
- Any path starting from one image and going to another goes through data with the same level of noise.

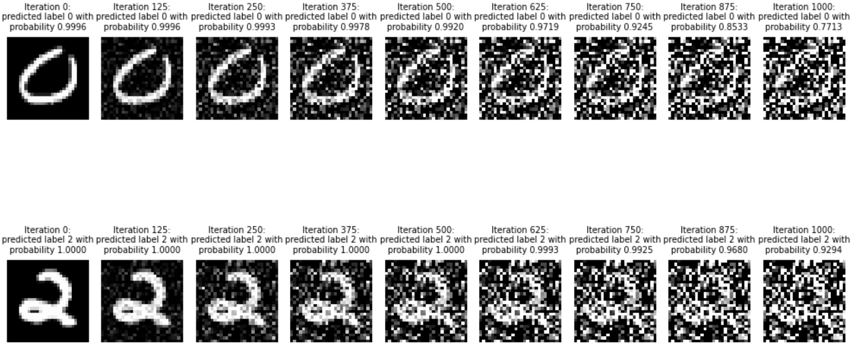


We can connect a digit from MNIST to a symbol **not** in MNIST moving on the **data leaf**:



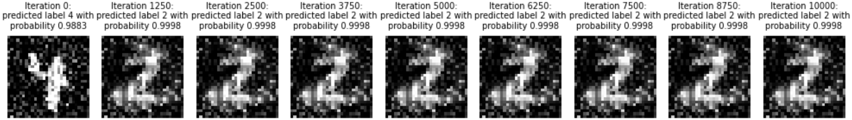
Moving away from the data leaf: MNIST

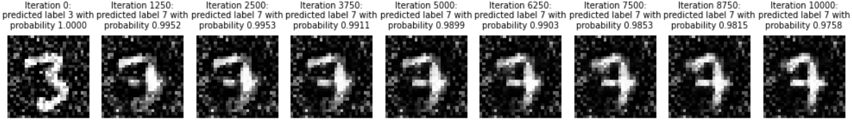
When moving **away** from a given data leaf, noise is added, but the accuracy is high.



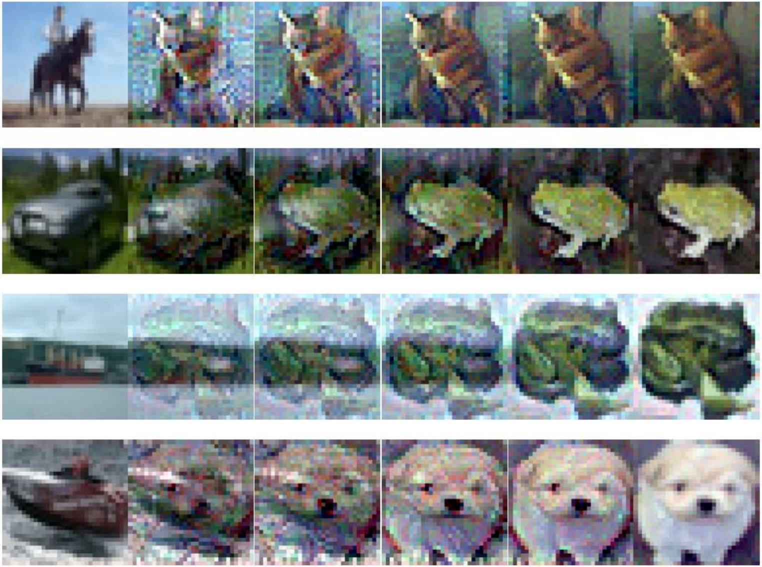
Moving on a noisy leaf: MNIST

We can connect a noisy datum with any other datum with the **same** level of noise:





Moving on the data manifold: CIFAR10



Conclusions

- Using a partially trained model we can construct a low dimensional submanifold the **data leaf** of $\mathbb{R}^n$ containing the data the model was trained with.
- We can navigate the data leaf and obtain either data or points with similarities to our data.
- Moving orthogonally to the data leaf will add noise to data, but the model will not change its accuracy.

1. Possible Applications:
   - Denoising of images: project a noisy data point on the data leaf to perform denoising.
   - Use the distance from the data leaf to recognize out-of- distribution examples
   - GAN: generate new images with the same label, by moving on the data leaf.

Bibliography

- Amari, S.-I. *Natural gradient works efficiently in learning. Neural computation*, 10(2):251276, 1998.
- Grementieri, L., Fioresi, R. *Model-centric Data Manifold: the Data Through the Eyes of the Model*, preprint, 2021.
- Bergomi, M. G., Frosini, P., Giorgi, D., and Quercioli, N. *Towards a topologicalgeometrical theory of group equivariant non-expansive operators for data analysis and machine learning*. Nature Machine Intelligence, 1 (9):423433, 2019.
- Sommer, S. and Bronstein, A. M. *Horizontal flows and manifold stochastics in geometric deep learning*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020.