OptiML Group Meeting

Implicit Bias of Large Depth Networks: a Notion of Rank for Nonlinear Functions

Arthur Jacot (ICLR 2023 notable-top-25%)

Speaker: Hanseul Cho

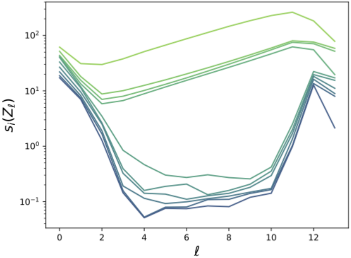
2023.09.08. (Fri)

OptiML
Optimization &
Machine Learning
Laboratory

Introduction

- "Implicit bias": What functions are favored by a neural network when fitting the training data?
  - Different architectures and training procedures can lead to widely different biases.

- Long story short; a fully trained (ℓ_2-regularized) deep neural network prefers to be low rank as getting deeper and deeper. [Jacot, 2023]



(a) Spectrum of the weights W_ℓ throughout the network.

Table of Contents

1. Preliminaries
    - Representation Cost (of a Function)
    - Rank of Non-linear Functions
2. Main results
    - Infinite Depth Regime
    - Finite Depth: Approximate Rank-1 Regime (Large Depth)
    - Finite Depth: Rank Recovery with Intermediate Depth
3. Practical Implications
    - Classification
    - Autoencoders (AE)

Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions

2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth

3. Practical Implications
   - Classification
   - Autoencoders (AE)

Representation cost

Definition

Representation cost of a function f:

$$R(f;\Omega,\sigma,L) = \min_{\boldsymbol{w}:F_{\boldsymbol{w}}|_\Omega = f|_\Omega} \|\boldsymbol{w}\|^2,$$

where $F_{\boldsymbol{w}}$ is a depth-L fully-connected neural network with homogeneous[a] non-linearity σ.

[a] $\sigma(c\cdot x) = c\cdot\sigma(x)$ $(c \geq 0)$; *e.g.*, ReLU, LeakyReLU, $\ldots$.

- It "describes the natural bias on the represented function $F_{\boldsymbol{w}}$ induced by adding ℓ_2-regularization on the weight $\boldsymbol{w}$"[1]:

$$\min_{\boldsymbol{w}} C(F_{\boldsymbol{w}}) + \lambda\|\boldsymbol{w}\|^2 = \min_f C(f) + \lambda R(f;\Omega,\sigma,L).$$

[1] We assume sufficiently large (but finite) width of the network. The paper says it is enough to use the width about $N(N+1)$ for the size of the training set N.

Representation cost

$$R(f;\Omega,\sigma,L) = \min_{\boldsymbol{w}:F_{\boldsymbol{w}}|_\Omega = f|_\Omega} \|\boldsymbol{w}\|^2$$

- Linear networks: sparsity/low-rank biases ($p = {}^2/_L$)
  - For diagonal linear networks of depth L, a linear function $f(\boldsymbol{v}) = \boldsymbol{v}^\intercal \boldsymbol{x}$ has $R(f)/L = \|\boldsymbol{v}\|_p^p$ [Gunasekar et al., 2018].
  - For fully-connected linear networks of depth L, a linear function $f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x}$ has $R(f)/L = \|\boldsymbol{A}\|_p^p := \sum_{i=1}^{\mathrm{rank}(\boldsymbol{A})} s_i(\boldsymbol{A})^p$ [Dai et al., 2021].
  - The biases become stronger with depth L; in the infinite depth limit $L \to \infty$, they converge to $\|\boldsymbol{v}\|_0$ and $\mathrm{rank}(\boldsymbol{A})$, respectively.
- Shallow ($L = 2$) nonlinear networks with a homogeneous activation: $R(f)$ takes the form of an ℓ_1 norm.
- How about deep nonlinear neural networks (with homogeneous non-linearity)?
  - The re-scaled representation cost $R(f)/L$ converges to some notion of rank in nonlinear networks as $L \to \infty$.

Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions

2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth

3. Practical Implications
   - Classification
   - Autoencoders (AE)

Rank of piece-wise linear functions

- We consider piece-wise linear (possibly non-linear in a whole) functions with a finite number of linear regions (called **finite piece-wise linear functions, FPLF**).
- A rank on FPLF must satisfy:
  1. The rank of a function is a nonnegative integer;
  2. $\mathrm{Rank}(f \circ g) \leq \min\{\mathrm{Rank}\, f, \mathrm{Rank}\, g\}$;
  3. $\mathrm{Rank}(f + g) \leq \mathrm{Rank}\, f + \mathrm{Rank}\, g$;
  4. If f is affine $(f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b})$ then $\mathrm{Rank}\, f = \mathrm{rank}(\boldsymbol{A})$.
- ...inspired by properties of rank for linear maps.

Two different notions of rank

Definition

The **Jacobian rank** of an FPLF f is $\mathrm{Rank}_J(f;\Omega) = \max_{\boldsymbol{x}\in\Omega} \mathrm{Rank}\, Jf(\boldsymbol{x})$, taking the max over the set of differentiable points $\boldsymbol{x}$.

Definition

The **bottleneck rank** of an FPLF f is $\mathrm{Rank}_{BN}(f;\Omega)$ is the smallest integer k such that there is a factorization as the composition of two FPLFs $f|_\Omega = (h \circ g)|_\Omega$ with inner dimension k.

- In general, $\mathrm{Rank}_J(f;\Omega) \leq \mathrm{Rank}_{BN}(f;\Omega)$ but "$=$" does not always hold (Proposition 1).
- In particular, if $f = \psi \circ g \circ \phi$ for an affine function $g(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$ and two bijections ψ and ϕ, $\mathrm{Rank}_J(f;\Omega) = \mathrm{Rank}_{BN}(f;\Omega) = \mathrm{rank}\boldsymbol{A}$.

Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions
2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth
3. Practical Implications
   - Classification
   - Autoencoders (AE)

Table of Contents

1 Preliminaries
- Representation Cost (of a Function)
- Rank of Non-linear Functions

2 Main results
- Infinite Depth Regime
- Finite Depth: Approximate Rank-1 Regime (Large Depth)
- Finite Depth: Rank Recovery with Intermediate Depth

3 Practical Implications
- Classification
- Autoencoders (AE)

Infinite depth regime

The rescaled representation cost $R_\infty(f;\Omega,\sigma) := \lim_{L\to\infty} \frac{R(f;\Omega,\sigma,L)}{L}$ converges to a value 'sandwiched' between the two notions of rank.[2]

Theorem 1

For any bounded domain Ω and any FPLF f,

$$\mathrm{Rank}_J(f;\Omega) \le R_\infty(f;\Omega,\sigma) \le \mathrm{Rank}_{BN}(f;\Omega)$$

Furthermore, $R_\infty(f;\Omega,\sigma)$ satisfies properties 2 to 4 of the rank.

- Conjecture: $R_\infty(f;\Omega,\sigma) = \mathrm{Rank}_{BN}(f;\Omega)$.
- Infinitely deep neural networks are biased towards functions with a low Jacobian (and possibly bottleneck) rank.

[2] Note: I found there's no proof of the convergence & well-definedness of $R_\infty(f)$ (!).

Infinite depth regime: Proof sketch

- The first inequality follows from taking $L \to \infty$ in Proposition 3.

Proposition 3

Let f be an FPLF, then at any differentiable point $\boldsymbol{x}$, we have

$$\|Jf(\boldsymbol{x})\|_{2/L}^{2/L} := \sum_{i=1}^{\operatorname{rank}(Jf(\boldsymbol{x}))} s_i(Jf(\boldsymbol{x}))^{2/L} \le \frac{R(f;\Omega,\sigma,L)}{L}.$$

- The proof of the second inequality is constructive. An FPLF $f = h \circ g$ can be represented as a network in 3 consecutive parts:
  1. the first part (of depth L_g) representing g,
  2. in the middle $L - L_g - L_h$ identity layers on $\mathbb{R}^k$,
  3. the final part (of depth L_h) representing h.

  The overall parameter norm is

$$\|\boldsymbol{w}\|_2^2 = \|\boldsymbol{w}_g\|_2^2 + k(L - L_g - L_h) + \|\boldsymbol{w}_h\|_2^2.$$

  The middle part dominates as $L \to \infty$.

Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions

2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth

3. Practical Implications
   - Classification
   - Autoencoders (AE)

Problem setting

- From now on, we focus on global minima functions $f_{\widehat{\boldsymbol{w}}}$ for MSE loss

$$\mathcal{L}_\lambda(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N} \left\| f_{\boldsymbol{w}}(\boldsymbol{x}_i) - \boldsymbol{y}_i \right\|^2 + \frac{\lambda}{L}\left\| \boldsymbol{w} \right\|^2$$

  and consider how $R(f_{\widehat{\boldsymbol{w}}})/L$ is affected by the depth L.
- The training dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$ is sampled from a distribution with bounded support Ω.

Approximate rank-1 regime: huge depth

Proposition 2 (Regarding Jacobian rank)

There exists a constant C_N (which solely depends on the training dataset) such that for any large L, at any global minimum $\widehat{\boldsymbol{w}}$ of the loss $\mathcal{L}_\lambda$,

$$\|Jf_{\widehat{\boldsymbol{w}}}(\boldsymbol{x})\|_{2/L}^{2/L} \overset{\text{Prop. 3}}{\leq} \frac{R(f_{\widehat{\boldsymbol{w}}};\Omega,\sigma,L)}{L} \leq 1 + \frac{C_N}{L}.$$

- Proof sketch: One can build a function with bottleneck rank 1 that fits any given training dataset!
- Together with Prop. 3, the second singular value of Jacobian is exponentially small.

Proposition 4 (Regarding bottleneck rank)

Take any set of $\tilde{N}$ data points with non-constant outputs, there is a layer $\ell_0 \leq L$ such that the first two singular values $s_1,\ s_2$ of the hidden representation (*i.e.*, activation) $Z_{\ell_0} \in \mathbb{R}^{n_{\ell_0}\times\tilde{N}}$ satisfies $\frac{s_2}{s_1} = O(L^{-1/4})$.

Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions

2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth

3. Practical Implications
   - Classification
   - Autoencoders (AE)

Rank recovery

- If $\mathrm{Rank}_{BN}(f^*) = 1$ for true function f^*: deep network can recover the rank 1.
- What if $\mathrm{Rank}_{BN}(f^*) > 1$?
    - Too deep networks might underestimate the rank of true function.
    - Too shallow models (like a linear ridge regression model) often overestimate the rank (since the jacobian is a.s. full rank).
- When does rank recovery occur?
    - A large amount of training data hinders rank underestimation.
    - There is a (intermediate) range of depths that may be possible to recover the true rank.

C_N explodes with N

Theorem 2

Given a Jacobian-rank k true function f^* on a bounded domain Ω, for all $\epsilon > 0$ there is a constant c_ϵ such that for any BN-rank 1 function $\hat{f}$ that fits $\hat{f}(\boldsymbol{x}_i) = f^*(\boldsymbol{x}_i)$ a dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$ sampled i.i.d. from a distribution with a support Ω, we have

$$\frac{R(\hat{f}; \Omega, \sigma, L)}{L} > c_\epsilon N^{\frac{2}{L}(1-\frac{1}{k})}$$

with probability $\geq 1 - \epsilon$.

- Idea: for large N, fitting the dataset with a rank 1 function requires large derivatives (which scale as the $\mathrm{TSP}(\boldsymbol{y}_1, ..., \boldsymbol{y}_N) \gtrsim N^{(1-\frac{1}{k})}$), which in turn implies a large parameter norm.
- Recall from Prop. 2 that $\frac{R(f_{\widehat{\boldsymbol{w}}}; \Omega, \sigma, L)}{L} \leq 1 + \frac{C_N}{L}$ for any $\widehat{\boldsymbol{w}}$.
- As N increases, larger and larger depths are required for the bound in Prop. 2 to be meaningful.
- Better upper bound independent of N?

Another upper bound of $\frac{1}{L}R(f_{\widehat{\boldsymbol{w}}})$

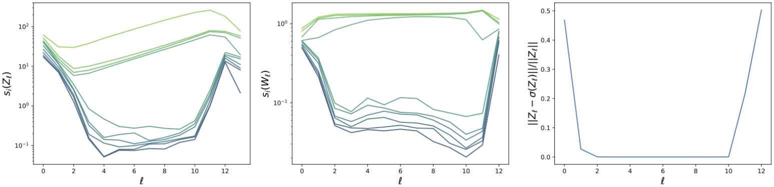
Proposition 5

Let f^* be FPLF with $\mathrm{Rank}_{BN}(f^\star) = k$. Then there is a constant C which solely depends on f^* such that any minimizer function $f_{\hat{\boldsymbol{w}}}$ satisfies

$$\frac{R(f_{\hat{\boldsymbol{w}}}; \Omega, \sigma, L)}{L} \leq k + \frac{C}{L}.$$

- If N is large enough, there are parameters $\boldsymbol{w}^*$ with a smaller norm than any choice of parameters fitting the data with a rank 1 function. $\Rightarrow$ rank underestimation never happen.
- Since C is independent of N, there is a range of depths where the upper bound of Prop. 5 is below that of Prop. 2.

Experiment: rank recovery



(a) Spectrum of the weights W_ℓ throughout the network.

(b) Spectrum of the activations Z_ℓ throughout the network.

(c) Impact of the nonlinearity on the representations Z_ℓ.

Figure 2: DNN (depth $L = 13$ and width $w = 100$) trained on a MSE task with rank 4 true function $f^* : \mathbb{R}^{50} \to \mathbb{R}^{50}$, with $N = 500$ and $\lambda = 0.05/L$. At the end of training, we obtain $\|\theta\|^2/L \approx 6$. **(a)** First 10 singular values of the weight matrices W_ℓ for all ℓ. **(b)** First 10 singular values of the matrix of activations Z_ℓ for all ℓ. The representations are approx. rank 4 in the middle layers. **(c)** The impact of the nonlinearity at each layer ℓ, measured by the ratio $\|Z_\ell - \sigma(Z_\ell)\|_F / \|Z_\ell\|_F$ where $\tilde{Z}_\ell$ is the matrix of preactivations. This impact vanishes in the middle layers, supporting the intuition that the middle layers represent approximate identities.

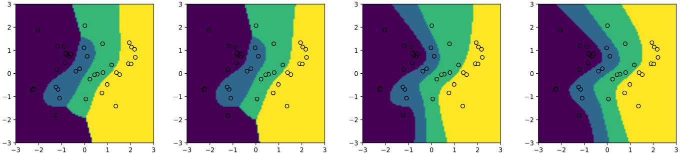
Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions
2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth
3. Practical Implications
   - Classification
   - Autoencoders (AE)

Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions
2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth
3. Practical Implications
   - Classification
   - Autoencoders (AE)

Deep networks learn low-dimensional topology



(a) $L = 2, \lambda = 0$ (b) $L = 2, \lambda = 10^{-3}$ (c) $L = 9, \lambda = 0$ (d) $L = 9, \lambda = 10^{-3}$

Figure 2: Classification on 4 classes (whose sampling distribution are 4 identical inverted 'S' shapes translated along the x-axis) for two depths and with or without L_2-regularization. The class boundaries in shallow networks **(A,B)** feature tripoints, which are not observed in deeper networks **(C,D)**.

- The bottleneck rank has an impact on the topology of the partitioning Ω into classes
- For instance, when $k = 1$, there would be no tripoints (*i.e.*, points at the boundary of 3 or more classes)
- The presence of explicit ℓ_2-regularization has little impact ($c.f.$, the cross-entropy loss leads to an implicit ℓ_2-regularization [Chizat and Bach, 2020, Gunasekar et al., 2018, Soudry et al., 2018])

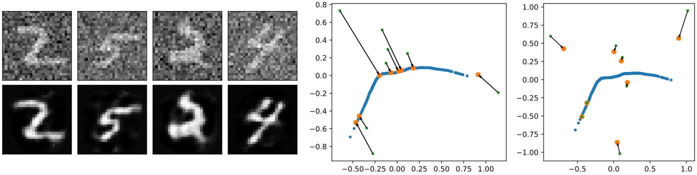
Table of Contents

1. Preliminaries
   - Representation Cost (of a Function)
   - Rank of Non-linear Functions

2. Main results
   - Infinite Depth Regime
   - Finite Depth: Approximate Rank-1 Regime (Large Depth)
   - Finite Depth: Rank Recovery with Intermediate Depth

3. Practical Implications
   - Classification
   - Autoencoders (AE)

Rank-recovering AEs naturally de-noise



(a) $L = 5$ (b) $L = 6$ (c) $L = 2$

Figure 3: Autoencoders trained on MNIST **(A)** and a 1D dataset on the plane **(B, C)** with a ridge $\lambda = 10^{-4}$. Plot **(A)** shows noisy inputs in the first line with corresponding outputs below. In plots **(B)** and **(C)** the blue dots are the training data, and the green dots are random inputs that are mapped to the orange dots pointed by the arrows. We see that for large depths **(A, B)** the learned autoencoder is naturally denoising, projecting points to the data distribution, which is not the case for shallow networks **(C)**.

- Consider learning an AE (i.e. learning $\hat{f}(x) = x$) on data of form $x = g(z)$ where $g : \mathbb{R}^k \to \mathbb{R}^d$ is an injective FPLF.
- Assuming $\mathrm{Rank}_{BN}\, \hat{f} = k$, $\hat{f}$ is locally an affine projection to a k-dimensional affine subspace (manifold of data points), which is equivalent to denoising.

Conclusion

- *TL;DR:* The representation cost of DNNs converges to a notion of nonlinear rank as the depth grows to infinity. This bias towards low-rank functions extends to large but finite widths.

References I

L. Chizat and F. Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on Learning Theory*, pages 1305–1338. PMLR, 2020.

Z. Dai, M. Karzand, and N. Srebro. Representation costs of linear neural networks: Analysis and design. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 26884–26896. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/e22cb9d6bbb4c290a94e4fff4d68a831-Paper.pdf.

S. Gunasekar, J. Lee, D. Soudry, and N. Srebro. Characterizing implicit bias in terms of optimization geometry. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1832–1841. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/gunasekar18a.html.

A. Jacot. Implicit bias of large depth networks: a notion of rank for nonlinear functions. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6iDHce-0B-a.

D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.