Scaling description of generalization with number of parameters in deep learning

Mario Geiger *, Arthur Jacot *, Stefano Spigler, Franck Gabriel, Levent Sagun, Stéphane d’Ascoli, Giulio Biroli, Clément Hongler, and Matthieu Wyart

arxiv 1901.01608
Deep learning

1/learning from example
2/ can predict!

• Revolution in Artificial Intelligence
• Principles to understand why It works lacking
Set-up

- binary classification task, \( P \) training data \( \{x_i, y_i = \pm 1\} \)

- Deep net \( f_W(x_i) \) with \( N \) parameters, width \( h \) (\( N \sim h^2 \))

\[
a_\beta = \rho \left( \sum_{\alpha \in \text{previous layer}} W_{\alpha, \beta} a_\beta - B_\beta \right)
\]

\( \rho \): non-linear function
Learning

- Learning: gradient descent in loss function \( \mathcal{L} = \frac{1}{P} \sum_{i} l_i(f_\mathbf{w}(x_i)) \)

- Quadratic Hinge Loss:
  \[
  l_i(f_\mathbf{w}(x_i)) = 0 \quad \text{if} \quad f_\mathbf{w}(x_i)y_i > 1 \\
  l_i(f_\mathbf{w}(x_i)) = (f_\mathbf{w}(x_i)y_i - 1)^2 \quad \text{if} \quad f_\mathbf{w}(x_i)y_i < 1
  \]

- \( \mathcal{L} = 0 \iff f_\mathbf{w}(x_i)y_i > 1 \forall i \) satisfability problem
Learning = descent in Loss Landscape

- High dimensional, not convex landscape.

**Question:** why not stuck in bad local minima? Landscape geometry?

Choromanska et al. 15, Soudry, Hoffer 17’ Cooper 18’ Baity-Jesy et al. 18

- sharp jamming transition in the landscape separating glassy landscape from an over-parametrized-phase with $\mathcal{L} = 0$.

Achievable if $N \sim P$  

Geiger et al. 18, Spigler et al. 18 (see Silvio’s talk)

- Why deep nets have predictive power while $N > P$, or even $N >> P$?
Empirical tests: MNIST (parity)

- $6 \times 10^4$ images of digits

- position of transition depends on dynamics (GD, adams, fire...)

Geiger et al., arxiv 180909349
Generalization

2 interesting asymptotic regimes:

• peak at the jamming transition

• performance improves with $N$ in the SAT phase???

works by Rakhlin, Srebro: increased regularization with $N$

Quantitative description? importance of $N=$
Quantifying fluctuations induced by initialization

- fixed data set, output function $f$ stochastic due to initialization
- This stochasticity is reduced as $N$ grows \cite{Neal et al. arxiv 1810591}

\[ \bar{f}_N : \text{ensemble average of } f_N \text{ on (20) initial conditions} \]

\[ \| f \|_\mu^2 = \int d\mu(x) f(x)^2 \]

\[ \| f_N - \bar{f}_N \|_\mu \sim N^{-1/4} \]

(to be explained later)
Test and practical consequences

- Reduce fluctuations by averaging

\[ \bar{\epsilon}_N : \text{test error of } \bar{f}_N \]

- Test error becomes nearly flat for \( N > N^* \), optimal near \( N^* \)

- **Best procedure: ensemble average near jamming transition!!!**
Scaling argument for generalization error

- seek to compute $\langle \epsilon_N \rangle \sim \bar{\epsilon}_N$
  using $\delta f_N = f_N - \bar{f}_N$ very small

- signed distances $\delta(x)$ becomes small. If smooth:

\[
\delta(x) = \frac{\delta f_N(x)}{||\nabla \bar{f}_N(x)||} + O(\delta f_N^2)
\]

\[
\left\{ \begin{array}{l}
\delta(x) \sim ||\delta f_N||_\mu \\
\langle \delta(x) \rangle \sim ||\delta f_N||^2_\mu
\end{array} \right.
\]
Scaling argument for generalization error

\[ \Delta \epsilon = \int_B dx^{d-1} \left[ \frac{\partial \epsilon}{\partial \delta(x)} \delta(x) + \frac{1}{2} \frac{\partial^2 \epsilon}{\partial^2 \delta(x)} \delta^2(x) + \mathcal{O}(\delta^3(x)) \right]. \]

\[ \langle \Delta \epsilon \rangle = c_0 \| \delta f \|^2 + \mathcal{O}(\| \delta f \|^3) \]

expect \( c_0 > 0 \) if \( \bar{\epsilon} \) small

\[ \langle \epsilon_N \rangle - \bar{\epsilon}_N \sim \| \bar{f}_N - f_N \|^2 \sim 1/\sqrt{N} \]
Propagation in infinitely wide nets at t=0

**set-up:** initialization iid weights = $\frac{\omega}{h^{1/2}}$ where $\omega \sim \mathcal{N}(0, 1)$

- Non-trivial limit for propagation, pre-activation $\alpha \sim 1$ and $f \sim 1$
- pre-activation and output are iid gaussian processes as $h \to \infty$

Neal 96, williams 98, Lee et al 18, Ganguli et al.
Learning: Neural Tangent Kernel

Jacot, Gabriel, Hongler NIPS 18

manifold $f_w$

small $h$: $\frac{\partial f}{\partial w}$ evolves

large $h$: $\frac{\partial f}{\partial w}$ fixed

“lazy learning”:
- weights change a little bit $\omega^t - \omega^0 \sim 1/h$
- sufficient to change $f$ (positive interference)
- does not change $\frac{\partial f}{\partial w}$
Results

\[ \mathcal{L} = \frac{1}{P} \sum_{i}^{P} l_i(f_{\mathbf{W}}(x_i)) \]

gradient descent

\[ \frac{df(x)}{dt} = -\frac{1}{P} \sum_{i=1}^{P} \Theta^t_N(x_i, x) l'_i(f(x_i)) \]

\[ \Theta^t_N(x_i, x) = \sum_{\omega} \frac{\partial f^t(x_i)}{\partial \omega} \frac{\partial f^t(x)}{\partial \omega} \]

useless in general...

Theorem: kernel does not depend on initialization at large N, nor on time

\[ \lim_{N \to \infty} \Theta^t_N(x_i, x) = \Theta_\infty(x_i, x) \]

deep learning equivalent to kernel learning as \( N \to \infty \)
Finite N  

- Fluctuations of $\Theta_{N}^{t=0}$ go as $1/\sqrt{h} \sim N^{-1/4}$

- Evolution in time much smaller $\theta_{N}^{t} - \theta_{N}^{t=0} \sim 1/\sqrt{N}$

\[
\frac{df(x)}{dt} = -\frac{1}{P} \sum_{i=1}^{P} \Theta_{N}^{t}(x_i, x)l_i'(f(x_i))
\]

- Leads to fluctuations of similar magnitude for output function (proof mean square loss)

\[
||f_{N} - \bar{f}_{N}||_{\mu} \sim N^{-1/4}
\]
Is learning features useful?

• neurons pattern of activity barely changes as $N \to \infty$

$$\alpha^t(x) \sim \alpha^{t=0}(x) \sim 1/\sqrt{h}$$

• success of deep learning believes to be associated with the emergence of good features....

• Small effect FCC on MNIST.
Is learning features useful? CNN data
Conclusion

• Deep nets fit all data if N > N*, jamming transition

• Performance keep increasing passed N* because fluctuations induced by initialization diminish

• Fluctuations are induced by the fluctuations of the kernel, fixed at infinite N

• In practice: best procedure= ensemble averaging just above N*

• Question future: scaling performance swith P?
Results

• **Theorem 3:** Dynamics find global minimum of the loss if loss $l_i$ convex and activation function non-polynomial.

Gram matrix $\Theta_\infty(x_i, x_j)$ positive definite

$$\frac{df(x)}{dt} = -\frac{1}{P} \sum_{i=1}^{P} \Theta_\infty(x_i, x)l'_i(f(x_i))$$

• **Result 4:** Smoothness of $f^t(x)$ can be deduced

$$f^t(x) = f^{t=0}(x) + \sum_{i=1}^{P} c_i(t) \Theta_\infty(x, x_i)$$
Why does deep learning work?

• when can one fit the data (not stuck bad minimum)?

\textit{crank up the number of parameters}

• Why does it generalize well, even when the number of parameters is large?

\textit{Generalization keeps improving with number of parameters}...

**MENU:**

1/ Quantification of evolution of generalization with number of parameters

2/ Neural Tangeant Kernel (NTK)

3/ NTK and generalization as number of parameters becomes asymptotically large