Foundations of Deep Learning

Lecture 05

Optimization Landscape of Neural Networks

Aurelien Lucchi

Fall 2024

Today

Discuss properties of the loss surface of deep neural networks.

How do they depend on the type of architecture, including parameters such as width, depth, activation functions, etc?

Definitions

Definition 1 (Global minimum)

Given a function $f(\mathbf{w}) : \mathbb{R}^d \to \mathbb{R}$, a point $\mathbf{w}^*$ is called a global minimum of f if for every $\mathbf{w}$, we have $f(\mathbf{w}^*) \ \leq f(\mathbf{w})$.

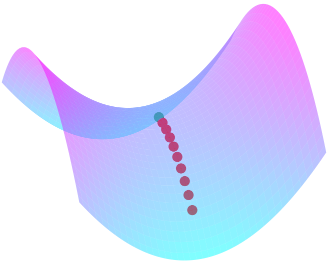
Definition 2 (Local minimum)

Given a function $f(\mathbf{w}) : \mathbb{R}^d \to \mathbb{R}$, a point $\mathbf{w}^*$ is called a local minimum of f if there exists a neighborhood of size ϵ, i.e. $\{\mathbf{w} \in \mathbb{R}^d \mid \|\mathbf{w} - \mathbf{w}^*\| \leq \epsilon\}$ such that $f(\mathbf{w}^*) \ \leq f(\mathbf{w})$.

Definitions

Definition 3 (Saddle point)

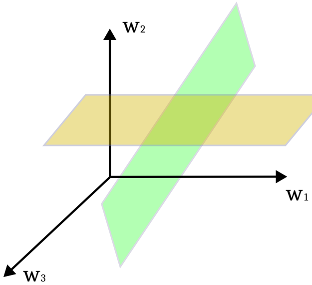
Given a function $f(\mathbf{w}) : \mathbb{R}^d \to \mathbb{R}$, a point $\mathbf{w}_s$ is called a saddle point of f if $\nabla f(\mathbf{w}_s) = 0$ and the Hessian $\nabla^2 f(\mathbf{w}_s)$ is indefinite, i.e. it has both positive and negative eigenvalues.



Over-parametrization

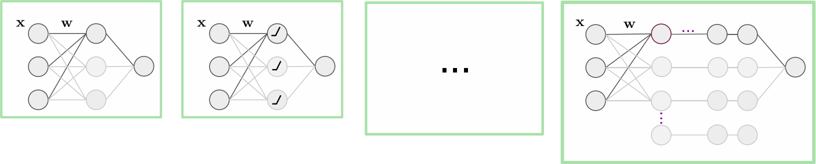
Consider a linear system of the form $F(\mathbf{w}) = \mathbf{y}$ where $\mathbf{w} \in \mathbb{R}^d, \mathbf{y} \in \mathbb{R}^n$ and $F : \mathbb{R}^d \to \mathbb{R}^n$.

As an example, take $n = 2, d = 3$.



This is an over-parametrized model ($n < d$). There is a manifold of solutions corresponding to the intersection of the planes.

Landscapes of Neural Networks: Overview



A) Linear network

B) Two-layer ReLU network

C) Midly-overparametrized network

D) Over-parametrized network/NTK regime

- A) Linear networks (*Kawaguchi (2016)*): Every local minima is global.
- B) Small-size two-layer ReLU networks (*Safran and Shamir (2018); Yun et al. (2018)*): there exist spurious local minima and there is a high probability of reaching them
- D) Over-parametrized networks (*Du et al. (2018); Allen-Zhu et al. (2018)*): every local minima is global

Section 1

Deep Linear Networks

Two-layer network [Baldi and Hornik(1989)]

- Consider a linear network with two layers whose weights are defined by matrices $\mathbf{A} \in \mathbb{R}^{d\times p}$ and $\mathbf{B} \in \mathbb{R}^{p\times d}$.
- Assume $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{R}^d$.
- Network has one hidden layer with p $(p \leq d)$ units, and one output layer with d units.

Let $\mathbf{W} = \mathbf{AB}$, and define the following loss function,

$$\mathcal{L}(\mathbf{W}) = \sum_{i=1}^{n} \|\mathbf{y}_i - \mathbf{ABx}_i\|^2. \tag{1}$$

Two-layer network [Baldi and Hornik(1989)]

Define the following covariance matrices, $\Sigma_{\mathbf{xx}} = \sum_i \mathbf{x}_i \mathbf{x}_i^\top$, $\Sigma_{\mathbf{xy}} = \sum_i \mathbf{x}_i \mathbf{y}_i^\top$ and $\Sigma_{\mathbf{yy}} = \sum_i \mathbf{y}_i \mathbf{y}_i^\top$.

Matrix form Let $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_n]$ and the output data matrix $\mathbf{Y} = [\mathbf{y}_1 \dots \mathbf{y}_n]$. Then

$$\begin{aligned}\mathcal{L}(\mathbf{W}) &= \|\mathbf{Y} - \mathbf{ABX}\|_F^2 \\ &= \mathrm{tr}[(\mathbf{Y} - \mathbf{ABX})(\mathbf{Y} - \mathbf{ABX})^\top].\end{aligned}$$

First-order condition

Theorem 4

For any fixed $d \times p$ *matrix* $\mathbf{A}$ *the function* $\mathcal{L}(\mathbf{A}, \mathbf{B})$ *is convex in the coefficients of* $\mathbf{B}$ *and attains its minimum for any* $\mathbf{B}$ *satisfying the equation*

$$\mathbf{A}^\top \mathbf{A} \mathbf{B} \Sigma_{\mathbf{xx}} = \mathbf{A}^\top \Sigma_{\mathbf{yx}}. \tag{2}$$

Proof

We will use the following expressions:

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{B}} \operatorname{tr}[\mathbf{ABC}] &= \mathbf{A}^\top \mathbf{C}^\top \\
\frac{\partial}{\partial \mathbf{B}} \operatorname{tr}[\mathbf{CB}^\top \mathbf{A}^\top] &= \mathbf{A}^\top \mathbf{C} \\
\frac{\partial}{\partial \mathbf{B}} \operatorname{tr}[\mathbf{ABCB}^\top \mathbf{A}^\top] &= \mathbf{A}^\top \mathbf{ABC}^\top + \mathbf{A}^\top \mathbf{ABC}.
\end{aligned} \tag{3}$$

Proof

First-order condition

Theorem 5

For any fixed $p \times d$ *matrix* $\mathbf{B}$ *the function* $\mathcal{L}(\mathbf{A}, \mathbf{B})$ *is convex in the coefficients of* $\mathbf{A}$ *and attains its minimum for any* $\mathbf{A}$ *satisfying the equation*

$$\mathbf{A}\mathbf{B}\Sigma_{\mathbf{xx}}\mathbf{B}^\top = \Sigma_{\mathbf{yx}}\mathbf{B}^\top. \tag{4}$$

Proof.

Left as an exercise. □

Optimal weights

Theorem 6

Assume that $\Sigma_{\mathbf{xx}}$ *is invertible. If two matrices* $\mathbf{A}$ *and* $\mathbf{B}$ *define a critical point of* $\mathcal{L}$ *(i.e., a point where* $\frac{\partial \mathcal{L}}{\partial a_{ij}} = \frac{\partial \mathcal{L}}{\partial b_{ij}} = 0$*) then the global map* $\mathbf{W} = \mathbf{AB}$ *is of the form*

$$\mathbf{W} = P_{\mathbf{A}} \Sigma_{\mathbf{xy}} \Sigma_{\mathbf{xx}}^{-1}, \tag{5}$$

where $P_{\mathbf{A}}$ *the matrix of the orthogonal projection onto the subspace spanned by the columns of* $\mathbf{A}$*.*

Critical points

Theorem 7

If $\Sigma_{\mathbf{xx}}$ and $\Sigma_{\mathbf{xy}}$ are full rank and $\Sigma = \Sigma_{\mathbf{yx}}\Sigma_{\mathbf{xx}^{-1}}\Sigma_{\mathbf{xy}}$ is full rank, then any local minimum is global and other critical points are saddle points.

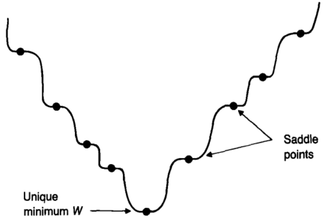


Figure: The landscape of the objective function (figure taken from [Baldi and Hornik(1989)].

Deep linear network [Kawaguchi(2016)]

Consider the model $\hat{\mathbf{Y}} = \mathbf{W}_{H+1}\mathbf{W}_H \dots \mathbf{W}_1\mathbf{X}$ and the loss $\mathcal{L}(\mathbf{W}) = \frac{1}{2}\|\hat{y}(\mathbf{W}, \mathbf{X}) - \mathbf{Y}\|^2$ and denote by p the smallest width in the hidden layers.

Theorem 8

For any depth $H \geq 1$ and for any layer widths and any input-output dimensions, the loss surface has the following properties:

1. *It is non-convex and non-concave*
2. *Every local minimum is a global minimum*
3. *Every critical point that is not a global minimum is a saddle point*
4. *If rank$(\mathbf{W}_H\mathbf{W}_{H-1} \dots \mathbf{W}_2) \geq p$, the Hessian at any saddle point has at least one negative eigenvalue.*

Section 2

Vanishing and Exploding Gradients

Gradient of a Vector-Valued Function

Consider the function $f(\mathbf{x}) = \mathbf{A}\mathbf{x} \in \mathbb{R}^p$, where $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{A} \in \mathbb{R}^{p \times d}$. Our goal is to compute the gradient $\frac{df}{d\mathbf{x}}$.
First, note that the dimension of the gradient $\frac{df}{d\mathbf{x}}$ is $\mathbb{R}^{p \times d}$.
Let's compute the partial derivative of f w.r.t. a single x_j. We have

$$f_i(\mathbf{x}) = \sum_{j=1}^{d} A_{ij} x_j \implies \frac{\partial f_i}{\partial x_j} = A_{ij}. \tag{6}$$

Collecting all the partial derivatives in the Jacobian, we obtain the following expression for the gradient:

$$\frac{df}{d\mathbf{x}} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_d} \\ & \dots & \\ \frac{\partial f_p}{\partial x_1} & \dots & \frac{\partial f_p}{\partial x_d} \end{pmatrix} = \begin{pmatrix} A_{11} & \dots & A_{1d} \\ & \dots & \\ A_{p1} & \dots & A_{pd} \end{pmatrix} = \mathbf{A} \in \mathbb{R}^{p \times d}. \tag{7}$$

Gradient of a Matrix-Valued Function (1/2)

Consider the function $f(\mathbf{x}) = \mathbf{A}\mathbf{x} \in \mathbb{R}^p$, where $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{A} \in \mathbb{R}^{p\times d}$. Our goal is to compute the gradient $\frac{df}{d\mathbf{A}}$.
First, note that the dimension of the gradient $\nabla f := \frac{df}{d\mathbf{A}}$ is $\mathbb{R}^{p\times(p\times d)}$. Let's compute the partial derivative of f w.r.t. a single x_j. We have

$$f_i(\mathbf{x}) = \sum_{j=1}^{d} A_{ij} x_j \implies \frac{\partial f_i}{\partial A_{iq}} = x_q. \tag{8}$$

Collecting all the partial derivatives, we can compute partial derivative of f_i w.r.t. the i-th row of $\mathbf{A}$:

$$\begin{aligned} \frac{\partial f_i}{\partial A_{i,:}} &= \mathbf{x}^\top \in \mathbb{R}^{1\times 1\times d} \\ \frac{\partial f_i}{\partial A_{k\neq i,:}} &= \mathbf{0}^\top \in \mathbb{R}^{1\times 1\times d}. \end{aligned} \tag{9}$$

Gradient of a Matrix-Valued Function (2/2)

Stacking the partial derivatives, we obtain the gradient of f_i w.r.t. $\mathbf{A}$:

$$\frac{\partial f_i}{\partial \mathbf{A}} = \begin{pmatrix} \mathbf{0}^\top \\ \cdots \\ \mathbf{0}^\top \\ \mathbf{x}^\top \\ \mathbf{0}^\top \\ \cdots \\ \mathbf{0}^\top \end{pmatrix} \in \mathbb{R}^{1\times(p\times d)}. \tag{10}$$

One can use the Kronecker product notation $\otimes$ to write the total derivative of f w.r.t. $\mathbf{A}$ as

$$\frac{\partial f}{\partial \mathbf{A}} = \mathbf{x}^\top \otimes \mathbf{I}. \tag{11}$$

Setting

Consider a regression problem with a single datapoint $\mathbf{x} \in \mathbb{R}^d$ and a corresponding target $\mathbf{y} \in \mathbb{R}^d$.

Deep Linear Network

$$\hat{\mathbf{y}} := F(\mathbf{x}) = \mathbf{W}^{L:1}\mathbf{x}, \quad \mathbf{W}^{L:1} = \mathbf{W}^{L} \cdots \mathbf{W}^{1}, \quad \mathbf{W}^{k} \in \mathbb{R}^{d \times d}$$

Interested in the case of random weight matrices, e.g. at initialization.

Squared Loss Given a single target $\mathbf{y}$,

$$\ell_{\mathbf{x}.\mathbf{y}}(\hat{\mathbf{y}}) = \frac{1}{2}\|\mathbf{y} - \hat{\mathbf{y}}\|^2, \quad \frac{\partial \ell_{\mathbf{x},\mathbf{y}}}{\partial \hat{\mathbf{y}}} = \hat{\mathbf{y}} - \mathbf{y} = \mathbf{W}^{L:1}\mathbf{x} - \mathbf{y} =: \boldsymbol{\delta}$$

Gradient norm

$$\frac{\partial \ell}{\partial \mathbf{W}^k} = \overbrace{\left[\mathbf{W}^{k+1:L}\boldsymbol{\delta}\right]}^{\text{backward}} \cdot \overbrace{[\mathbf{W}^{k-1:1}\mathbf{x}]^\top}^{\text{forward}}$$
$$= \mathbf{W}^{k+1:L}[\mathbf{W}^{L:1}\mathbf{x}\mathbf{x}^\top - \mathbf{y}\mathbf{x}^\top]\mathbf{W}^{1:k-1},$$

with $\mathbf{W}^{k+1:L} := \left(\mathbf{W}^{k+1}\right)^\top \cdots \left(\mathbf{W}^{L}\right)^\top$.

Theorem 9

Let $\mathbf{W}^k$ be Gaussian matrices with iid entries such that $\mathbb{E}[w_{ij}] = 0$ and $\mathbb{E}[w_{ij}^2] = \sigma^2$, and let $\rho = \|\mathbf{x}\|^4$, $\gamma = \|\mathbf{x}\|^2\|\mathbf{y}\|^2$. Then:

$$\mathbb{E}\left\|\frac{\partial \ell}{\partial \mathbf{W}^k}\right\|_F^2 \leq 2\frac{\rho}{3\sigma^2}[\sigma^4 d(d+2)]^L + 2\gamma(\sigma^2 d)^{L-1}.$$

Remarks

Note that the exponents of the two contributions differ.

- Small σ: gradient is dominated by the term involving $\mathbf{y}$
- Large $\sigma^4 > d(d+2)$: term involving $\hat{\mathbf{y}}$ dominates (assuming $\rho \approx \gamma$)

Xavier initialization [Glorot and Bengio(2010)]
When d is very large, then the dominating term in the theorem is of the form $(\sigma^4 d^2)^L \implies$ Stabilization requires $\sigma = \frac{1}{\sqrt{d}}$.

Proof of Theorem 9

Starting point We bound the norm of the gradient as follows,

$$
\begin{aligned}
\left\| \frac{\partial \ell}{\partial \mathbf{W}^k} \right\|_F^2 &= \| \mathbf{W}^{k+1:L} \mathbf{W}^{L:1} \mathbf{x}\mathbf{x}^\top \mathbf{W}^{1:k-1} - \mathbf{W}^{k+1:L} \mathbf{y}\mathbf{x}^\top \mathbf{W}^{1:k-1} \|_F^2 \\
&\leq 2 \| \mathbf{W}^{k+1:L} \mathbf{W}^{L:1} \mathbf{x}\mathbf{x}^\top \mathbf{W}^{1:k-1} \|_F^2 \\
&\quad + 2 \| \mathbf{W}^{k+1:L} \mathbf{y}\mathbf{x}^\top \mathbf{W}^{1:k-1} \|_F^2 .
\end{aligned}
$$

Next, we bound each term independently.

Proof of Theorem 9

We start with a simple lemma that we will apply recursively...

Lemma 10

Let $\mathbf{W}$ *be a random matrix with iid entries such that* $\mathbb{E}[w_{ij}] = 0$ *and* $\mathbb{E}[w_{ij}^2] = \sigma^2$, $\mathbf{A}, \mathbf{B}$ *arbitrary matrices, then*

$$\mathbb{E}\|\mathbf{AWB}\|_F^2 = \sigma^2 \|\mathbf{A}\|_F^2 \|\mathbf{B}\|_F^2.$$

Moreover, if $\mathbf{A}, \mathbf{B}$ *are stochastic but* $\{\mathbf{A}, \mathbf{B}, \mathbf{W}\}$ *uncorrelated, then*

$$\mathbb{E}\|\mathbf{AWB}\|_F^2 = \sigma^2 \cdot \mathbb{E}\|\mathbf{A}\|_F^2 \cdot \mathbb{E}\|\mathbf{B}\|_F^2.$$

Proof

Corollary

By applying Lemma 10 recursively, we obtain the following:

Corollary 11

$$\mathbb{E}\|\mathbf{W}^{t:1}\mathbf{x}\|^2 = (d\sigma^2)^t\|\mathbf{x}\|^2$$

Lemma

Lemma 12 (Statistics after multiplication with a random matrix)

Let $\mathbf{W}$ *be a random matrix with iid entries such that* $\mathbb{E}[w_{ij}] = 0$ *and* $\mathbb{E}[w_{ij}^2] = \sigma^2$*, and kurtosis* κ*. Let* $\boldsymbol{\xi} \in \mathbb{R}^d$ *be an arbitrary random vector. Then*

$$\mathbb{E}\|\mathbf{W}\boldsymbol{\xi}\|_2^4 = d(d+2)\sigma^4\mathbb{E}\|\boldsymbol{\xi}\|_2^4 + (\kappa - 3)d\sigma^4\mathbb{E}\|\boldsymbol{\xi}\|_4^4.$$

Proof

$$\|\mathbf{W}\boldsymbol{\xi}\|_2^4 = \Big(\sum_i \Big(\sum_r w_{ir}\xi_r\Big)^2\Big)^2$$
$$= \sum_{i,j} \sum_r w_{ir}\xi_r \sum_s w_{is}\xi_s \sum_u w_{ju}\xi_u \sum_v w_{jv}\xi_v.$$

Then take an expectation...

Proof of Theorem 9 continued

Lemma 13

Let $\mathbf{W}^k$ *be random matrices with iid entries such that* $\mathbb{E}[w_{ij}] = 0$ *and* $\mathbb{E}[w_{ij}^2] = \sigma^2$. *For a fixed input/output pair* $(\mathbf{x}, \mathbf{y})$ *one has*

$$\mathbb{E}\left\|\frac{\partial \ell}{\partial \mathbf{W}^k}\right\|_F^2 \leq 2\sigma^2 \underbrace{\mathbb{E}\|\mathbf{W}^{k-1:1}\mathbf{x}\|_2^4}_{Lemma\ 12} \underbrace{\mathbb{E}\|\mathbf{W}^{k+1:L}\mathbf{W}^{L:k+1}\|_F^2}_{Lemma\ 10} + 2(d\sigma^2)^{L-1}\|\mathbf{x}\|^2\|\mathbf{y}\|^2.$$

Proof

Back to main theorem

Let $\mathbf{W}^k$ be Gaussian matrices with iid entries such that $\mathbb{E}[w_{ij}] = 0$ and $\mathbb{E}[w_{ij}^2] = \sigma^2$, and let $\rho = \|\mathbf{x}\|^4$, $\gamma = \|\mathbf{x}\|^2\|\mathbf{y}\|^2$. Then:

$$\mathbb{E}\left\|\frac{\partial \ell}{\partial \mathbf{W}^k}\right\|_F^2 \leq 2\frac{\rho}{3\sigma^2}[\sigma^4 d(d+2)]^L + 2\gamma(\sigma^2 d)^{L-1}.$$

Pierre Baldi and Kurt Hornik.
Neural networks and principal component analysis: Learning from examples without local minima.
Neural networks, 2(1):53–58, 1989.

Xavier Glorot and Yoshua Bengio.
Understanding the difficulty of training deep feedforward neural networks.
In **Proceedings of the thirteenth international conference on artificial intelligence and statistics**, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

Kenji Kawaguchi.
Deep learning without poor local minima.
In **Advances in Neural Information Processing Systems**, pages 586–594, 2016.