§2. Escaping Saddle Points.

1. <u>Def.</u> $f: \mathbb{R}^d \to \mathbb{R}$. x is said to be a <u>local minimizer</u> of f if $\exists$ open $U \subseteq \mathbb{R}^d$ s.t. $x \in U$ and $f(x') \geq f(x)$ $\forall x' \in U$.

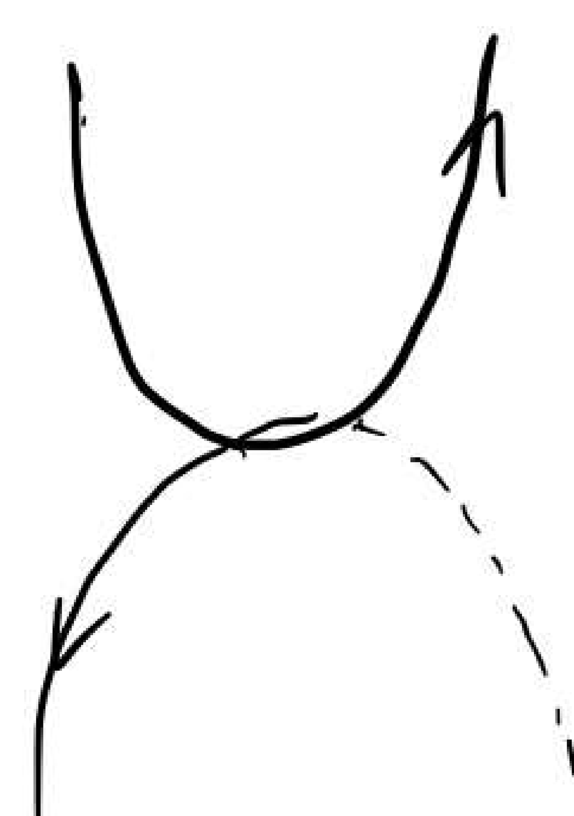
<u>2nd order condition</u>. $\nabla f(x) = 0$, $\nabla^2 f(x) \succ 0$

$\Rightarrow$ x is a strict local minimizer

* $\underbrace{\nabla f(x) = 0, \ \nabla^2 f(x) \succeq 0}_{\text{2nd order stationary point}} \not\Rightarrow$ local minimizer.

e.g. $f(x) = x^3$. $x = 0$

$f(x_1, x_2) = x_1^2 - x_2^4$.



2. <u>Def.</u> $f: \mathbb{R}^d \to \mathbb{R}$. ρ-Hessian Lipschitz. We say x is a <u>ε-second order stationary point</u> of f if.

$$\|\nabla f(x)\| \leq \varepsilon \quad \text{and} \quad \lambda_{\min}(\nabla^2 f(x)) \geq -\sqrt{\rho\varepsilon}.$$

- error terms: Consider $f(x) = \Theta(\rho x^3)$

3. Goal: Assume $f: \mathbb{R}^d \to \mathbb{R}$ is ℓ-smooth and ρ-Hessian Lipschitz. Noisy GD/GF can efficiently find a ε-second order stationary point. ($\Leftarrow$ they can escape saddle points with $\lambda_{\min}(\nabla^2 f(x)) < -\sqrt{\rho\varepsilon}$.)

4. The quadratic case

Consider $f(x) = x^T A x$ where $A = \mathrm{diag}(-1, 1, \dots, 1)$.

clear that $\hat{x} = 0$ is a saddle point

Claim. For any reasonable x_0, GF will not get stuck at $\hat{x}$.

pf. $\frac{d}{dt} x_t = -\nabla f(x_t) = -2Ax_t$

$\Rightarrow \frac{d}{dt} x_{k,t} = -2A_{kk} x_{k,t}$

$\Rightarrow x_{k,t} = x_{k,0} \exp(-2A_{kk} t)$

$\Rightarrow x_{k,t} \to 0 \ (k \neq 1)$, $|x_{1,t}| \to \infty$

As long as $|x_{1,0}| \geq 1/\mathrm{poly}(d)$, $|x_{1,t}|$ will become $\Omega(1)$ in $O(\log d)$ time.

Lemma. (Anti-concentration of ball volume).

$x \sim \mathrm{Unif}(B(r))$. $\forall \delta \in (0,1)$. with proba. $\geq 1-\delta$, we have

$|x_1| \geq r\delta / (2\sqrt{d})$

Thus, small ball perturbation

$\Rightarrow$ escaping the saddle point.

5. General loss function.

- Algorithm: Run GD/GF and occasionally add a small perturbation.

- High-level idea:

Suppose x_0 is near a saddle point with at least one descent direction. Assume $\nabla f(x_t)$ is small $\forall t \in [0, T] \Rightarrow x_t \approx x_0$

$\Rightarrow \nabla^2 f(x_t) \approx \nabla^2 f(x_0)$.

$$\frac{d}{dt} \|\nabla f(x_t)\|^2 = 2\langle \nabla f(x_t), \frac{d}{dt} \nabla f(x_t)\rangle$$

$$= 2\langle \nabla f(x_t), \nabla^2 f(x_t)\, \dot{x}_t\rangle$$

$$= -2\langle \nabla f(x_t), \nabla^2 f(x_t) \nabla f(x_t)\rangle$$

$$\approx -2\langle \nabla f(x_t), \nabla^2 f(x_0) \nabla f(x_t)\rangle$$

$\Rightarrow$ $\nabla f(x_t)$ will blow up along the descent dir. Contradiction.

Thm. (Thm 2 of Jin et al. (2017))

$f: \mathbb{R}^d \to \mathbb{R}$ ℓ-smooth, ρ-Hessian Lipschitz. $\forall \varepsilon < \ell^2/\rho$, $\delta \in (0,1)$, with proba. $\geq 1-\delta$. noisy GD can output a ε-second order stationary pt. with

$$O\left(\frac{\ell\,(f(x_0) - f_*)}{\varepsilon^2} \log^4\left(\frac{d\ell\,(f(x_0) - f_*)}{\varepsilon^2 \delta}\right)\right)$$

iterations.

* $\mathrm{poly}\log(d)$

§3. Neural Tangent Kernel (NTK)

1. Themes of DL Theory.

- "Traditional" supervised learning:
  - Why GD + NN can (over)fit the training set?
    (Global convergence)
  - What solutions can GD find? Why they can generalize?
    (algorithmic regularization.
  - .....
- Not so supervised. (Pretraining + finetuning).
  - Different pre-training tasks. (contrastive learning, reconstruction ....)
  - What representations do they learn?
  - Why they can be used in downstream tasks.
  - .....
- In-context learning (Prompting)
  - We don't know what questions to ask yet ...
- Generic template.
  - Why X works?
  - Why X is better than Y?

2. Background of NTK

- (Zhang et al., 2016). (experimental)

  GD + NN can often globally minimize the loss, even when the labels are random.

- Over-parameterized networks.

  - Original meaning: # params > n ← # training samples

  - what people mean in DL theory:

    # neurons (per layer) = poly(r) ← latent dim.

    or poly(d) ← input dim.

    or poly(d, n)

    or exp(d) or ∞

3. Setup

- Training set $\{(x_i, y_i)\}_{i=1}^{n} \subseteq \mathbb{R}^d \times \mathbb{R}$.

- Learner network.

$$f(x; W, a) = \frac{1}{\sqrt{m}} a^T \sigma(Wx) = \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k \sigma(w_k \cdot x)$$

where $x \in \mathbb{R}^d$ input.

$m \in \mathbb{N}$. # neurons.

$W \in \mathbb{R}^{m \times d}$ first layer weights

$\sigma : \mathbb{R} \to \mathbb{R}$. ReLU.

$a \in \mathbb{R}^m$ output weights

- Initialization. $a_k \sim \mathrm{Unif}(\{\pm 1\})$

  $w_k \sim N(0, I_d)$.

- Training algorithm: Fix a. Train W with GD + MSE. $L(W) = \frac{1}{2} \sum_{i=1}^{n} (y_i - f(x_i; W))^2$

4. Derivatives.

$$\nabla_{w_k} L(W) = \sum_{i=1}^{n} (y_i - f(x_i;W)) \nabla_{w_k} f(x_i;W)$$

$$= -\sum_{i=1}^{n} (y_i - f(x_i;W)) \frac{a_k}{\sqrt{m}} \nabla_{w_k} \sigma(w_k \cdot x)$$

$$= -\frac{a_k}{\sqrt{m}} \sum_{i=1}^{n} (y_i - f(x_i;W))\, \sigma'(w_k \cdot x_i)\, x_i$$

Recall GF: $\frac{d}{dt} w_k = -\nabla_{w_k} L(W)$.

$$\frac{d}{dt} f(x_j) = \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k \frac{d}{dt} \sigma(w_k \cdot x_j)$$

$$= \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k \left\langle \sigma'(w_k \cdot x_j)\, x_j, \frac{d}{dt} w_k \right\rangle$$

$$= \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k \left\langle \sigma'(w_k \cdot x_j)\, x_j, \frac{a_k}{\sqrt{m}} \sum_{i=1}^{n} (y_i - f(x_i;W))\, \sigma'(w_k \cdot x_i)\, x_i \right\rangle$$

$$= \frac{1}{m} \sum_{i=1}^{n} (y_i - f(x_i;W)) \sum_{k=1}^{m} \cancel{a_k^2}^{\,1}\, \sigma'(w_k \cdot x_i)\, \sigma'(w_k \cdot x_j) \langle x_i, x_j \rangle$$

Define the NTK

$$H(x,x') = \frac{1}{m} \sum_{k=1}^{m} \sigma'(w_k \cdot x)\, \sigma'(w_k \cdot x_j) \langle x_i, x_j \rangle$$

and $H_{ij} = H(x_i, x_j)$ $\quad i,j \in [n]$.

Define $y = (y_i)_{i=1}^{n}$, $f_t = (f(x_i;W_t))_{i=1}^{n}$

Then $$\frac{d}{dt} f_t = H_t (y - f_t).$$

Remarks.

a) H_t depends on t.

b). NTK and these formula themselves are quite generic, while the NTK technique is not.

c) If $H_t \succeq \lambda_0 I_d$ for some $\lambda_0 > 0$, $\forall t$ then $f_t \to y$ linearly.

5. The NTK technique.

Choose m and the initialization scale (im)properly to ensure $H_t \approx H_0$ throughout training and reduce the problem to a convex one.

6. Define $H^\infty \in \mathbb{R}^{n\times n}$ by

$$
\begin{aligned}
H^\infty_{ij} &= \lim_{m\to\infty} H_{ij}(0) \\
&= \lim_{m\to\infty} \frac{1}{m}\sum_{k=1}^{m} \sigma'(w_k(0)\cdot x_i)\,\sigma'(w_k(0)\cdot x_j)\,\langle x_i, x_j\rangle \\
&= \mathbb{E}_{w\sim N(0,I_d)}\left[\sigma'(w\cdot x_i)\,\sigma'(w\cdot x_j)\,\langle x_i, x_j\rangle\right].
\end{aligned}
$$

Define $\lambda_0 := \lambda_{min}(H^\infty)$.

Fact. If $x_i \not\parallel x_j$, $\forall i\neq j$, then $\lambda_0 > 0$.

Assume $\lambda_0 > 0$.

Du et al., 2019. Gradient descent provably optimizes over-parametrized neural networks

7. **Lemma.** Choose $m = \Omega\left(\frac{n^2}{\lambda_0^2} \log\left(\frac{n}{\delta}\right)\right)$.

With proba. $\geq 1-\delta$, we have

$\|H_0 - H^\infty\|_2 \leq \lambda_0/4$, whence $\lambda_{min}(H_0) \geq \frac{3}{4}\lambda_0$.

Lemma. Initialization $w_1, \dots, w_m$.

Any $\hat{w}_1, \dots, \hat{w}_m \in \mathbb{R}^d$ with $\|w_k - \hat{w}_k\| \leq O\left(\frac{\delta \lambda_0}{n^2}\right) =: R$.

Let H and $\hat{H}$ be the corresponding NTKs. With proba. $\geq 1-\delta$ over the random init., we have

$\|\hat{H} - H\|_2 \leq \lambda_0/4$, whence $\lambda_{min}(\hat{H}) \geq \lambda_0/2$.

Deterministic.
The infinite-width limit.
Depends only on the initialization scheme

H^∞

$\approx$

Depends only on the actual random initialization. $\to$ H_0

H_t $\leftarrow$ Also depends on the training procedure.

8. <u>Thm</u>. Assume $\|x_i\|, |y_i| \le C$. Choose $m = \Omega\left(\frac{n^6}{\delta^3 \lambda_0^4}\right)$.

With proba. $\geq 1 - O(\delta)$, GF converges to a point with

$\|y - f_t\| \le \varepsilon$ within $O\left(\frac{1}{\lambda_0} \log\left(\frac{\|y - f_0\|}{\varepsilon}\right)\right)$ amount of time.

In words, GF will fit the training set before W_k moves too far away from the initialization.

Pf. Define $T_* := \min\{T_1, T_2\}$ where

$$T_1 := \inf\{t \geq 0 : \|f_t - y\| \le \varepsilon\}$$

$$T_2 := \inf\{t \geq 0 : \exists k, \|W_k(t) - W_k(0)\| \geq R\}$$

By the previous lemmas, $\forall t \le T_* \le T_2$, we have

$$\frac{d}{dt}\|y - f_t\|^2 = -2\langle y - f_t, H_t(y - f_t)\rangle$$

$$\le -\lambda_0 \|y - f_t\|^2.$$

$$\Rightarrow \|y - f_t\|^2 \le \|y - f_0\|^2 \exp(-\lambda_0 t).$$

$$(\Leftrightarrow \|y - f_t\| \le \|y - f_0\| \exp(-\lambda_0 t/2))$$

Meanwhile, we have

$$\|\nabla_{W_k} L(W)\| = \left\|\frac{a_k}{\sqrt{m}} \sum_{i=1}^n (y_i - f(x_i))\sigma'(W_k \cdot x)x\right\|$$

$$\le \frac{C}{\sqrt{m}} \sum_{i=1}^n |y_i - f(x_i)|$$

$$\le C\sqrt{\frac{n}{m}} \|y - f_t\|$$

$$\Rightarrow \|W_k(t) - W_k(0)\| \le C\sqrt{\frac{n}{m}} \int_0^t \|y - f_s\| \, ds$$

$$\le C\sqrt{\frac{n}{m}} \|y - f_0\| \int_0^t \exp(-\lambda_0 s/2) \, ds$$

$$\le \underbrace{2C\sqrt{\frac{n}{m}} \|y - f_0\|}_{\le R}.$$

$\Rightarrow T_*$ cannot be attained by T_2.

$\Rightarrow T_* = T_2 \le O\left(\frac{1}{\lambda_0} \log\left(\frac{\|y - f_0\|}{\varepsilon}\right)\right).$

Question. Where did we use $a_k \sim \mathrm{Unif}\{\pm 1\}$?

Answer.
$$\mathop{\mathbb{E}}_{a,W} \|f_0\|^2 = \mathop{\mathbb{E}}_{a,W} \left\| \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k \sigma(w_k \cdot x) \right\|^2$$
$$= \frac{1}{m} \sum_{i,j=1}^{m} \mathop{\mathbb{E}}_{a,W} \left\{ a_i a_j \, \sigma(w_i \cdot x) \, \sigma(w_j \cdot x) \right\}.$$

If. $a_k \sim \mathrm{Unif}\{\pm 1\}$. then
$$= \frac{1}{m} \sum_{k=1}^{m} \mathop{\mathbb{E}}_{W} \sigma^2(w_k \cdot x) = \mathop{\mathbb{E}}_{W} \sigma^2(w_k \cdot x)$$

If $a_k = 1$, then
$$= \frac{1}{m} \sum_{i,j}^{m} \mathop{\mathbb{E}}_{W} \sigma(w_i \cdot x) \, \sigma(w_j \cdot x)$$
$$\geq \Omega\left(m \left(\mathop{\mathbb{E}}_{w_k} \sigma(w_k \cdot x) \right)^2 \right).$$

As a result.
$$\|w_k(t) - w_k(0)\| \leq 2 (\frac{\eta}{\sqrt{m}} \|y - f_0\|$$
$$\not\to 0 \text{ as } m \to \infty.$$

9. NTK $\Leftrightarrow$ random feature.

If the movement is small.

$$f(x;W(t)) = \frac{1}{\sqrt{m}}\sum_{k=1}^{m} a_k\,\sigma(W_k\cdot x)$$

$$\approx \underbrace{f(x;W(0))} + \frac{1}{\sqrt{m}}\sum_{k=1}^{m} a_k\langle W_k(t)-W_k(0),\, x\rangle\,\sigma'(W_k\cdot x)$$

$\hookrightarrow \approx 0$ when m is large.

Define $\alpha_k = (W_k(t)-W_k(0))/\sqrt{m}$. Then

$$f(x;W(t)) \approx \sum_{k=1}^{m}\langle \alpha_k,\, a_k\,\sigma'(W_k(0)\cdot x)\,x\rangle.$$

$\hookrightarrow$ a linear model over the <u>fixed</u> random feature mapping

$$x \mapsto \left(a_k\,\sigma'(W_k(0)\cdot x)\,x\right)_{k=1}^{m}$$

10. Random linear features cannot learn a single linear model

- Input distr. $x \sim N(0, I_d)$
- Target function. $f_*(x) = \langle w_*, x \rangle$. for some fixed unit vector w_*.
- Learner: $f(x; u) = \sum_{k=1}^{m} u_k \langle w_k, x \rangle$

$$= \langle \underbrace{\sum_{k=1}^{m} u_k w_k}_{=: w_u}, x \rangle$$

where $w_k \sim N(0, I_d/d)$

Claim. If $m \le d/2$, then w.h.p.

$$\min_u \text{MSE} = \min_u \mathbb{E}_x \{ (f(x;u) - f_*(x))^2 \} \ge \frac{1}{4}$$

Pf. $\text{MSE} = \mathbb{E}_x \langle w_u - w_*, x \rangle^2 = \| w_u - w_* \|^2$

$\Rightarrow \min_u \text{MSE} =$ dist. from w_* to $\text{span}(w_1, \dots, w_m)$

By symmetry, we may assume w.l.o.g. that $(w_1, \dots, w_m)$ are fixed and w_* is a random unit vector. Moreover assume $w_k = e_k$.

$$\Rightarrow L(w_*) := \min_u \text{MSE} = 1 - \sum_{k=1}^{m} [w_*]_k^2$$

$$\ge 1 - \underbrace{\sum_{k=1}^{d/2} [w_*]_k^2}$$

$\hookrightarrow$ $\Theta(1)$-Lipschitz with median $= 1/2$

$\xrightarrow{\text{Lévy}}$ with proba. $\ge 1 - \exp(-\Theta(d))$,

$L(w_*) \ge 1/4$.

Lemma. (Lévy's inequality). Let $v \sim \mathrm{Unif}(\mathbb{S}^{d-1})$.

$g: \mathbb{R}^d \to \mathbb{R}$. L-Lipschitz. $\exists C, c > 0$ s.t. $\forall \varepsilon$.

$$\mathbb{P}\left[\, |g(v) - \mathrm{median}(g)| \geq \varepsilon \,\right] \leq C \exp\left(-\frac{c\varepsilon^2 d}{L^2}\right).$$

- Informal version of Thm. 1.2 of Thm 4.2 of Yehudai and Shamir (2019).

$\mathcal{F} = \left\{ x \mapsto f(Wx) : W = \begin{bmatrix} W_1 \\ \vdots \\ W_p \end{bmatrix}, W_k \sim \mathrm{Unif}(\mathbb{S}^{d-1}) \right\}$. $\quad N(x) = \sum_{k=1}^{m} u_i f(W_i x)$.

$\forall W^* \in \mathbb{R}^d$ with $\|W^*\| = d^2$, $\exists$ bias $b_* \in \mathbb{R}$ with $|b| = O(d^3)$ s.t. w.h.p.

$$\mathbb{E}\left\{ \left(N(x) - \sigma(W_* \cdot x + b_*)\right)^2 \right\} \leq 1/50.$$

$$\Rightarrow \quad m \cdot \max_k |u_k| \geq \exp(\Omega(d)).$$

11. Continuity argument aka. induction/minimal counterexample in continuous time

- Example $f \in C^0(\mathbb{R})$. $\dot{x}_t = f(x_t)$. $f(0) > 0$. $x_0 > 0 \Rightarrow x_t > 0 \ \forall t$.

- Example. Comparative Gronwall.

"Progress" X_t. "error" Y_t. Suppose when $Y_t \le \delta = \frac{1}{\mathrm{poly}(d)}$, we have

$$\begin{cases} \dot{X}_t \le -A_t X_t \\ \dot{Y}_t \le K A_t Y_t \end{cases} \quad \text{or} \quad \begin{cases} \dot{X}_t \le -A_t X_t \\ \dot{Y}_t \le K A_t Y_t + B \end{cases}$$

- Example. Suppose when $Y_t \le \delta \le 1/\mathrm{poly}(d)$, we have $\dot{X}_t \le -A X_t$. $\dot{Y}_t \le X_t Y_t$.

*. Present this before the NTK results.