Lecture 9: Optimization in feature-learning regime (cont'd) Learning lower bounds

Last lecture recap:

- express a model $f(x;\theta) = \frac{1}{m}\sum_{j=1}^{m} \phi(x;\theta_j) = \int \phi(x;\theta)\, \mu(d\theta)$
- gradient flow wrt parameters (θ_j) $\leftrightarrow$ wasserstein gradient flow wrt measure μ.

Today:

→ convergence properties

→ learning lower bounds

→ Recall the continuity equation (Liouville eq) that μ_t solves:

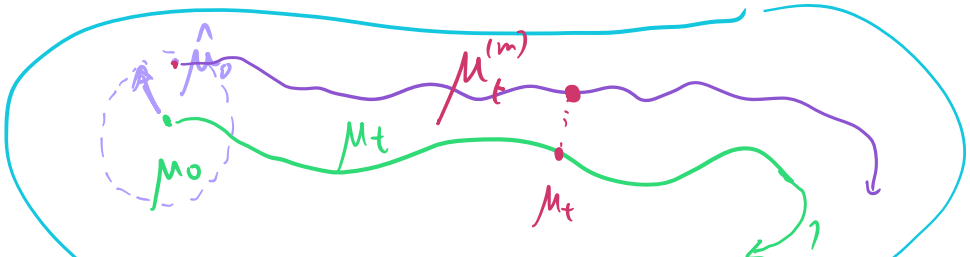
$$\partial_t \mu_t = \operatorname{div}\left(\nabla G(\cdot;\mu_t)\cdot \mu_t\right) \quad *$$

Natural Questions: $\partial_t \mu_t = \Phi(\mu_t)$

(i) under what conditions does this PDE converge in time towards the (global) minimizers of $\mathcal{L}$? (convergence in t).

(ii) How are the dynamics affected by over parametrization (convergence in width, m)

$\hat{\mu}_0^{(m)} = \frac{1}{m}\sum_{j=1}^{m} \delta_{\theta_j}$



$\theta_i \overset{iid}{\sim} \mu_0$

μ^*

$\mathcal{P}(D)$

$\mu^* = \underset{\mu}{\operatorname{argmin}}\ \mathcal{L}[\mu]$

↳ $\mu_t^{(m)}$ solves (*) with initial condition given by $\mu_0^{(m)}$.

Theorem [CB, R, ÉVE, MNM, SS '18] For any fixed $T > 0$, $\mu_T^{(m)}$ converges weakly to μ_T as $m \to \infty$, where μ_t solves (*) with initial condition μ_0.

↳ In other words, sampling and "learning" commute in the limit of $m, t \to \infty$.

$$\left(\lim_{t\to\infty} \lim_{m\to\infty} \mu_t^{(m)} = \lim_{m\to\infty} \lim_{t\to\infty} \mu_t^{(m)} \right).$$

The evolution $(\mu_t)_t$ is referred as the mean-field limit of the system. It provides a description of the system at any point in time that "neglects" interactions.

Global convergence in time

Recall that (*) $\partial_t \mu_t = \operatorname{div}(\nabla G(\cdot, \mu_t) \cdot \mu_t)$ is the WGF associated with the functional

$$\mathcal{L}[\mu] = -\int F \, d\mu + \frac{1}{2} \iint K(\theta, \theta') \, d\mu \, d\mu'.$$

↳ Stationary point of WGF satisfy

$$0 = \operatorname{div}(\nabla G(\cdot;\mu_t)\,\mu_t)$$ Euler–Lagrange equation.

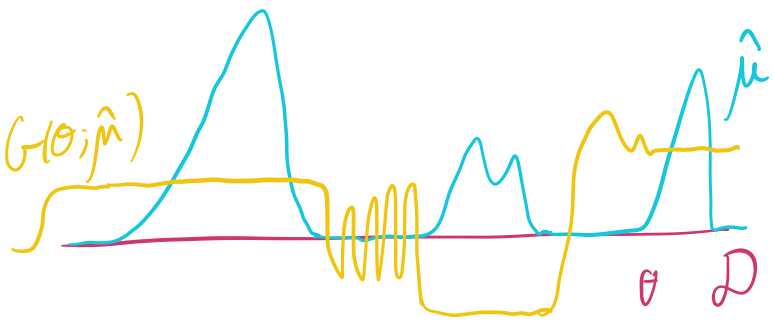
Claim: $\hat{\mu}$ is a stationary point if

$$\nabla G(\theta;\hat{\mu}) = 0 \quad \text{for } \theta \in \operatorname{supp}(\hat{\mu}) \tag{1}$$

↳ Global minimums μ^* of $\mathcal{L}$ satisfy

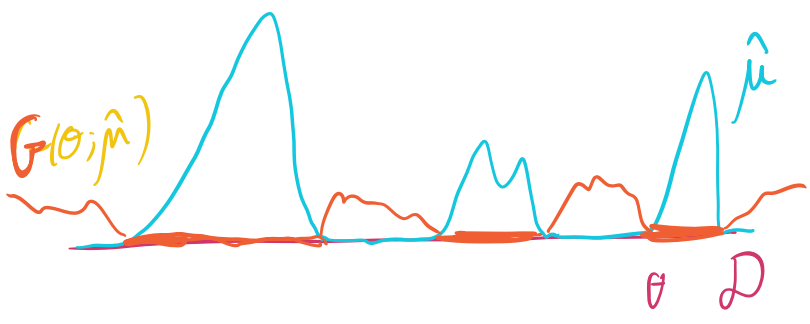
$$\begin{cases} G(\theta;\mu^*) = 0 & \text{for } \theta \in \operatorname{supp}(\mu^*) \\ G(\theta;\mu^*) \geq 0 & \text{for any } \theta. \end{cases} \tag{2}$$

Q: When can we guarantee that (1) $\Rightarrow$ (2) ?



← stationary point of WGF

//



← global minimum of $\mathcal{L}$.

Two challenges that we need to adress:

① → Not "concentrate" mass too soon along dynamics

② → From $\nabla G(\theta;\mu^*)=0$ to $G(\theta;\mu^*)=0$?

↳ For the first challenge, we can address it by using initial conditions with full support (CB); alternative: add noise into the gradient ⇝ adds a term $\Delta\mu_t$ that enforces full support (MNM).

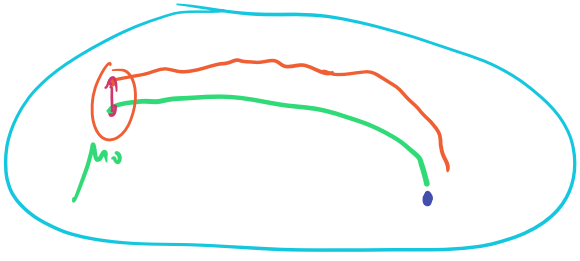
↳ The second challenge is addressed thanks to the homogeity of the neuron. $\phi(\lambda\theta;x)=\lambda^p\phi(\theta;x)$

deriv w.r.t. λ

$$\phi(\theta;x) = p^{-1}\left\langle \theta, \nabla\phi(\theta;x)\right\rangle$$

[Euler Rule for homogenous]

[CB]
[R, EVE]



Convergence in m

→ If we have an initial condition μ_0 that leads to global convergence, how many neurons are needed to approximate it uniformly in time ?

→ For $t=0$; $f(x;\theta_1\cdots\theta_m)$ with $\theta_i \overset{iid}{\sim} \mu_0$ approximates the mean function

$$\bar{f}(x;\mu_0) = \mathbb{E}_{\theta\sim\mu_0}[\phi(x;\theta)]$$

→ Approximation error satisfies

$$\mathbb{E}\,\|\, f(\cdot;\theta_1\cdots\theta_m) - \bar{f}\,\|^2 = \frac{T(\mu_0)}{m}, \text{ with}$$

↳ Monte-Carlo Rate.

$$T(\mu_0) = \mathbb{E}_{\mu}\,\|\,\phi(\cdot;\theta)\|_\nu^2 - \|\bar{f}\|_\nu^2$$

Q: How does this error evolve during training?

A: Under mild assumptions in the Mean-Field limit, this error remains bounded (at the M-C scale), uniformly in time:

Theorem [CRBE '20]

$$\lim_{t\to\infty}\lim_{m\to\infty} m\cdot\mathbb{E}\,\|\,\hat{f}_t^{(m)} - f_t^{n}\,\|^2 \le T(\mu_\infty)$$

$\int\phi\,d\mu_t^{(m)}$ $\int\phi\,d\mu_t$

Remark: CLT analog of the LLN (Law of large numbers) Mean-field convergence.

→ Extending the finite-time horizon propagation-of-chaos results; based on Gronwall's inequality.

if $\|\partial_t\hat{\mu}_t - \partial\mu_t\| \le \beta(t)\cdot\|\hat{\mu}_t - \mu_t\|$ then

$$\|\hat{\mu}_T - \mu_T\| \le \|\hat{\mu}_0 - \mu_0\|\exp\left(\int_0^T \beta(s)\,ds\right)$$

$\cdot \varphi(\hat{\mu}_t) - \varphi(\mu_t)$

→ We use a different technique, based on a Volterra-type kernel.

→ MF take-aways:

⊕ Rich qualitative description of empirically observed phenomena.

⊕ Consistent with the "mystery" of over-parametrization.
↳ as m grows, optim is "easier" with generalization under control.

⊕ Provably more expressive than NTK.

⊖ Qualitative results: no rates of convergence neither in time nor in m (width)!

Q: Can we hope to efficiently learn any target function f^* that can be well approximated in our shallow function class $\mathcal{F}$?

Learning Lower Bounds

↳ Recall from first lecture the example of learning 1-Lipschitz functions

→ Key source of hardness:

there are "too many" ($\exp(d)$) candidate functions in my class.

→ Learning Parities with gradient Descent [Shalev-Swartz, Shamir, Shammah 17]

- Let $v^* \in \{0,1\}^d$ a point in the d-dim hypercube chosen uniformly at random.
- Now consider the target function

$$f_{v^*} : \{0,1\}^d \longrightarrow \{-1,1\}$$

$$x \longmapsto (-1)^{\langle x, v^* \rangle}$$

In words, $f_{v^*}(x)$ is the parity of x within the subset v^*.

→ Dataset $\{x_i\}_{i=1..n} \sim \text{Unif}\{0,1\}^d$

$$y_i \,|\, x_i = f_{v^*}(x_i)$$

→ We wish to learn f_{v^*} from the dataset $\{(x_i, y_i)\}_i$ using a shallow Relu network.

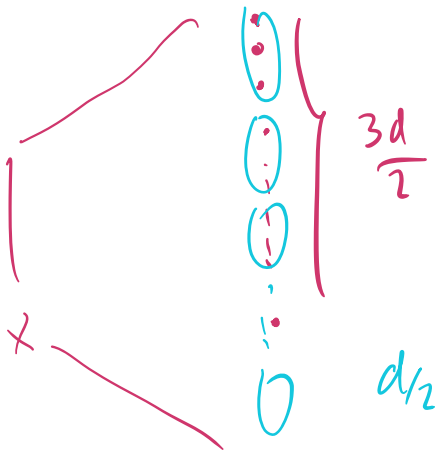
$$f(x; \theta_1 \dots \theta_M) = \sum_{j=1}^{M} a_j \sigma(\langle x, w_j \rangle + b_j)$$

$$\min_{\theta_1 \dots \theta_M} \frac{1}{n} \sum_{i=1}^{n} \left(f(x_i; \vec{\theta}) - y_i \right)^2$$

M: width of net
n: # of training point.

Fact: When $M > \frac{3d}{2}$, a shallow ReLu net can express f_v for <u>any</u> v. (no error).

proof · set $\begin{cases} w_j = v & \text{for } j = 1 \dots 3d/2 \\ w_j = 0 & \text{for } j > 3d/2 \end{cases}$



For $i = 1 \dots d/2$ set biases and output weights as

$j = 3i + \{1,2,3\}$

$$b_j = \begin{cases} -(2i - \frac{1}{2}) \\ -2i \\ -(2i + \frac{1}{2}) \end{cases} \qquad a_j = \begin{cases} 1 \\ -2 \\ 1 \end{cases}$$

$$\sum_{j = 3i + \{1,2,3\}} a_j \sigma(\langle x, w_j \rangle + b_j) = \begin{cases} 0 & \text{if } \langle x, w_j \rangle \neq 2i \\ 1/2 & \text{if } \langle x, w_j \rangle = 2i \end{cases}$$

$$\rightarrow \sum_i \left(\sum_{j = 3i + \{1,2,3\}} a_j \sigma(\langle x, w_j \rangle + b_j) \right) = \begin{cases} 1/2 & \text{if } \langle x, v \rangle \text{ even} \\ 0 & \text{otherwise.} \end{cases}$$

So we are expressing f_v (up to linear rescaling).

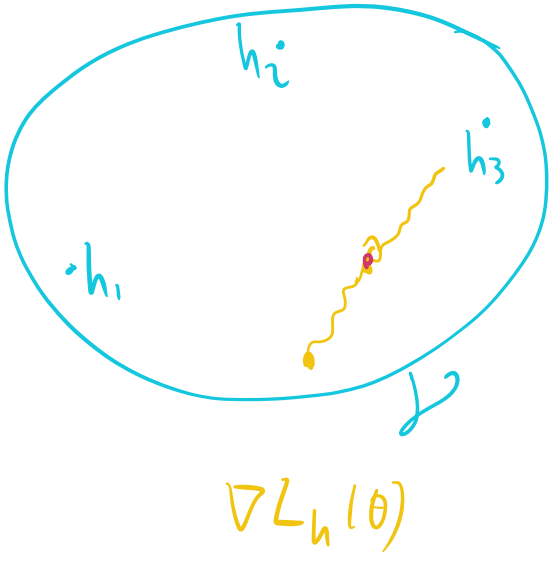
→ Can we learn such function with gradient descent?

→ Consider the optimisation given by

$$\min_{\theta} L_h(\theta) = \mathbb{E}_x\left[\ell\left(\tilde{f}_\theta(x), h(x)\right)\right]$$

$\tilde{f}_\theta$: is a differentiable class (eg NNs)

h : is a target hypothesis picked from a set V.



→ We consider how sensitive the gradient of L_h wrt θ is to h (target function). We measure this through the variance

$$Var(V; L; \theta) = \mathbb{E}_h \left\| \nabla_\theta L_h(\theta) - \mathbb{E}_h \nabla L_h(\theta) \right\|^2$$

L_2 measures how much "signal" from target ($h \in V$) is contained in the gradient at point θ.

Theorem [SSS '17]. Suppose V is a discrete set of target functions satisfying $\langle h, h' \rangle_\nu = 0$ if $h \neq h'$ and $\|h\|_\nu^2 \geq \varepsilon$

- Assume f_θ is differentiable wrt θ, with

$$\mathbb{E}_x\left[\|\nabla_\theta \tilde{f}_\theta(x)\|^2\right] \leq G(\theta)^2$$

- Pick ℓ: squared loss.

Then $$\mathrm{Var}(V; L; \theta) \leq \frac{G(\theta)^2}{|V|}$$

Proof: [SSS'17] → very easy.

→ ## When D = Random parities f_v

↳ Fact: $\begin{cases} \langle f_v, f_{v'} \rangle = 0 & \text{if } v \neq v' \\ \|f_v\|^2 = 1 \end{cases}$

↳ We conclude that

$$\mathrm{Var}(V, L; \theta) \leq \frac{G(\theta)^2}{2^d}$$

↳ The gradient at any point will be concentrated at a value independent of target. (bad!)

① → learning parities is GD-hard. Is it hard in general?

$$f_v(x) = (-1)^{\langle x, v \rangle}$$

$$\forall i, \quad \langle x_i, v \rangle = f_v(x_i) \pmod 2$$

↳ Gaussian elimination solves parities in linear time!