Implicit Bias of Wide Two-layer Neural Networks Trained with the Logistic Loss

Jihu Lee

May 20, 2021

Table of Contents

Introduction

- **Implicit bias of gradient methods**
  - Lyu and Li (2019): homogeneous neural networks the training trajectory converges in direction to a critical point of some nonconvex max-margin problem
  - Improve this result for the two-layer case: characterize the learnt classifier as the solution of a *convex* max-margin problem

Introduction

- **Dynamics of infinitely-wide neural networks**
  - Describe the training dynamics by a Wasserstein gradient flow
  - Chizat and Bach (2018): convex loss, diverse-enough initialization, convergent gradient flow $\rightarrow$ its limit is a global minimizer
  - This paper includes the cases when the gradient flow diverges

Notation

- $\mathcal{M}(\mathbb{R}^p)$: set of nonnegative finite Borel measures on $\mathbb{R}^p$
- $\mathcal{P}_2(\mathbb{R}^p)$: set of probability measures with finite second moment
- $\Delta^{m-1} = \left\{p \in \mathbb{R}_+^m; 1^\intercal p = 1\right\}$: simplex

Table of Contents

2-homogeneous neural networks

- Binary calssification with a training set $(x_i, y_i)_{i \in [n]}$ with $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$

$$h_m(w, x) = \frac{1}{m} \sum_{j=1}^{m} \phi(w_j, x) \tag{1}$$

- m: number of units, $w = (w_j)_{j \in [m]}$: trainable parameters
- This setting covers two-layer neural networks where m is the size of the hidden layer

2-homogeneous neural networks

- Focus on the case where ϕ is 2-*homogeneous* and *balanced*
- **(A1)** The function ϕ is 2-*homogeneous* in its first variable, i.e., $\phi(rw, x) = r^2\phi(w, x)$ and it is *balanced*, i.e. $\exists T$ s.t. $\phi(T(\theta), \cdot) = -\phi(\theta, \cdot)$
- Ex) **ReLU**, S-ReLU

Parameterizing with a measure

$$h(\mu, x) = \int_{\mathbb{R}^p} \phi(w, x) d\mu(w) \tag{2}$$

- Finite width networks as in Eq.(1) are recovered when μ is a discrete measure with m atoms

Parameterizing with a measure

- Eq.(2) can be reduced to a *convex neural network* parameterized by an unnormalized measure

$$\int_{\mathbb{S}^{p-1}} \phi(\theta) d\left[\Pi_2(\mu)\right](\theta) = \int_{\mathbb{R}^p} \|w\|^2 \phi(w/\|w\|) d\mu(w) \qquad (3)$$

Max-margins and functional norms

- Margin of a predictor f: $min_{i \in [n]} y_i f(x_i)$
- **Variation norm**
  - $\mathcal{F}_1$: space of functions that can be written as $f(x) = \int_{\mathcal{S}^{p-1}} \phi(\theta, x) d\nu(\theta)$
  - Infimum of $\nu(\mathcal{S}^{p-1})$ over all such decompositions defines a norm: *variation norm* on $\mathcal{F}_1$
- **RKHS norm**
  - $\mathcal{F}_2$: space of functions of the form $f(x) = \int_{\mathcal{S}^{p-1}} \sigma(b + c^\intercal x) g(b,c) d\tau(b,c)$ for some square-integrable function $g \in L^2(\tau)$
  - Infimum of $\|g\|_{L^2(\tau)} = (\int |g(b,c)|^2 d\tau(b,c))^{\frac{1}{2}}$ defines a norm

Max-margins and functional norms

- $\mathcal{F}_1$ max-margin classifier

$$\gamma_1 := \max_{\nu \in \mathcal{M}_+(\mathbb{S}^{p-1}), \nu(\mathbb{S}^{p-1}) \leq 1} \min_{i \in [n]} y_i \int_{\mathbb{S}^{p-1}} \phi(\theta, x) d\nu(\theta) \tag{4}$$

- $\mathcal{F}_2$ max-margin classifier

$$\gamma_2 := \max_{\|g\|_{L^2(\tau)} \leq 1} \min_{i \in [n]} y_i \int_{\mathbb{S}^{p-1}} \sigma(b + c^\top x_i) g(b, c) d\tau(b, c) \tag{5}$$

Training dynamics in the infinite width limit

- Given a loss l, define the empirical risk associated to a predictor h_m as $\frac{1}{n}\sum_{i=1}^{n} l(-y_i h_m(w, x_i))$
- **(A2)** The loss l is differentiable with a locally Lipschitz-continuous gradient. It has an *exponential tail*, it is strictly increasing and there exists $c > 0$ such that $l'(u) \geq c$ for $u \geq 0$
- Ex) logistic loss: $l(u) = log(1 + e^u)$, exponential loss: $l(u) = e^u$
- **(A3)** The family $(\phi(\cdot, x_i))_{i \in [n]}$ is linearly independent and for $i \in [n]$, the function $\phi(\cdot, x_i)$ is differentiable with a Lipschitz-continuous gradient and subanalytic

Gradient flow of the smooth-margin objective

- Consider maximizing minus the log of the empirical risk

$$S(u) = -\log\left(\frac{1}{n}\sum_{i=1}^{n} l(-u_i)\right) \tag{6}$$

- Objective function: $F_m(w) = S(\hat{h}_m(w))$
- $\hat{h}_m(w) = (y_i h_m(w, x_i))_{i\in[n]}$

$$\frac{d}{dt}w(t) = m\nabla F_m(w(t)) \tag{7}$$

Wasserstein gradient flow

- Training dynamics: $\mu_{t,m} = \frac{1}{m}\sum_{j=1}^{m} \delta_{w_j(t)}$ in $\mathcal{P}_2(\mathbb{R}^p)$
- $F(\mu) = S(\hat{h}(\mu))$: functional on $\mathcal{P}_2(\mathbb{R}^p)$
- $\hat{h}(\mu) = (y_i h(\mu, x_i))_{i\in[n]}$
- $F'_\mu(w) = \sum_{i=1}^{n} y_i \phi(w, x_i) \nabla_i S(\hat{h}(\mu))$

Definition (Wasserstein gradient flow)
A Wasserstein gradient flow for the functional F is a path $(\mu_t)_{t\geq 0}$ such that there exists a flow $X : \mathbb{R}_+ \times \mathbb{R}^p \to \mathbb{R}^p$ satisfying $\mu_t = (X_t)_{\#\mu_0}$ (where $X_t(\cdot) = X(t, \cdot)$), $X(0, \cdot) = X_0 = id_{\mathbb{R}^p}$ and for all $(t, w) \in \mathbb{R}_+ \times \mathbb{R}^p$,

$$\frac{d}{dt} X(t, w) = \nabla F_{\mu_t}(X(t, w))$$

Infinite width limit of training

Theorem (Infinite width limit of training)
Under $(A1-3)$*, if the sequence* $(w_j(0))_{j\in\mathbb{N}_*}$ *is such that* $\mu_{0,m}$ *converges in* $\mathcal{P}_2(\mathbb{R}^p)$ *to* μ_0*, then* $\mu_{t,m}$ *converges in* $\mathcal{P}_2(\mathbb{R}^p)$ *to the unique Wasserstein gradient flow of* F *starting from* μ_0*. The convergence is uniform on bounded time intervals.*

Table of Contents

Implicit bias of gradient flow

Theorem (Implicit bias)
Under $(A1-3)$, *assume that* $\Pi_2(\mu_0)$ *has full support on* $\mathbb{S}^{p-1}$. *If* $\nabla S(\hat{h}(\mu_t))$ *converges and* $\bar{\nu}_t = \Pi_2(\mu_t)/([\Pi_2(\mu_t)](\mathbb{S}^{p-1}))$ *converges weakly to some* $\bar{\nu}_\infty$, *then this limit* $\bar{\nu}_\infty$ *is a maximizer for the* $\mathcal{F}_1$*-max margin problem in Eq. (4)*

Observations

- Limit $\bar{\nu}_\infty$ of a *non-convex* dynamics is a *global* minimizer of Eq. (4)
- Convergence of $\nabla S(\hat{h}(\mu_t))$ and $\bar{\nu}_t$ is an open question
- Unlike in the convex case, the dynamics does not completely forget where it started from

Training finite width neural networks

Corollary
Under the assumptions of Theorem 3, assume that the sequence $(w_j(0))_{j \in \mathbb{N}_*}$ is such that $\mu_{0,m}$ converges in $\mathcal{P}_2(\mathbb{R}^p)$ to μ_0. Then, denoting $\bar{\nu}_{m,t} = \Pi_2(\mu_{m,t}) / \left[\Pi_2(\mu_{m,t})\right](\mathbb{S}^{p-1})$, it holds

$$\lim_{m,t \to \infty} (\min_{i \in [n]} y_i \int \phi(\theta, x_i) d\bar{\nu}_{m,t}) = \gamma_1$$

Table of Contents

Insights on the convergence rate and choice of step-size

- Simplified dynamics: $w_j(t) = r_j(t)\theta_j$, where $r_j(t)$ is trained and θ_j is fixed at init

$$F_m(r) = -log(\frac{1}{n}\sum_{i=1}^{n} exp(-\frac{1}{m}\sum_{j=1}^{m} z_{i,j} r_j^2))$$

- $z_{i,j} = y_i \phi(\theta_j, x_i)$: signed fixed features
- $l = exp$: exponential loss

Insights on the convergence rate and choice of step-size

- Gradient ascent dynamics with initialization $r(0)$ and sequence of step-sizes $(\eta(t))_{t\in\mathbb{N}}$

$$r(t+1) = r(t) + \eta(t) m \nabla F_m(r(t))$$

- It is shown to converge to a max l_1-margin classifier without a rate

Insights on the convergence rate and choice of step-size

Proposition

Let $a_j(t) = r_j(t)^2/m$ *for* $j \in [m]$, $\beta(t) = \|a(t)\|_1$ *and* $\bar{a}(t) = a(t)/\beta(t)$. *For the step-sizes* $\eta(t) = 1/(16\|z\|_\infty\sqrt{t+1})$ *and a uniform initialization* $r(0) \propto 1$, *it holds*

$$\max_{0 \le s \le t-1} \min_{i \in [n]} z_i^\top \bar{a}(s) \ge \gamma_1^{(m)} - \frac{\|z\|_\infty}{\sqrt{t}}(8log(m) + log(t) + 1) - \frac{4Blog(n)}{\sqrt{(}t)}$$

where $\gamma_1^{(m)} := \max_{a \in \Delta^{m-1}} \min_{i \in [n]} z_i^\top a$ *and some* $B \le \infty$ *when* $\gamma_1^{(m)} > 0$

- Convergence of the best iterate to maximizers at an asymptotic rate $log(t)/\sqrt{(}t)$

Table of Contents

Training only the ouput layer

- Input layer being initialized randomly and fixed

$$F(r) = -log(\frac{1}{n}\sum_{i=1}^{n})exp(-\frac{1}{m}\sum_{j=1}^{m} z_{i,j} r_j)$$

- $z_{i,j} = y_i \sigma(b_j + x_i^\intercal c_j)$: signed output of neuron j for the training point i
- σ: non-linearity such as ReLU

Training only the ouput layer

- Gradient ascent dynamics with initialization $r(0)$ and sequence of step-sizes $(\eta(t))_{t\in\mathbb{N}}$

$$r(t+1) = r(t) + \eta(t) m \nabla F_m(r(t))$$

- It is shown to converge in $O(log(t)/sqrtt)$ to a l_2 max-margin classifier, for a step-size of order $1/\sqrt{t}$

Training only the ouput layer

Proposition
Let $a(t) = r(t)/m, \beta(t) = max\left\{1, \max_{0\le s\le t}\sqrt{m}\|a(t)\|_2\right\}$ *and* $\bar{a}(t) = a(t)/\beta(t)$*. Assume* $\gamma_2^{(m)} := \max_{\sqrt{m}\|a\|_2\le 1}\min_{i\in[n]} z_i^\intercal a > 0$*. For the step-sizes* $\eta(t) = \beta(t)\sqrt{2}/(\|z\|_\infty\sqrt{t+1})$ *and initialization* $r(0) = 0$*, it holds*

$$\max_{0\le s\le t-1}\min_{i\in[n]} z_i^\intercal \bar{a}(s) \ge \gamma_s^{(m)} - \frac{\|z\|_\infty}{\sqrt{t}}(2\sqrt{2} + \frac{\sqrt{3}log(n)}{\gamma_2^{(m)}})$$

Table of Contents

Dimension independent generalization bounds

- *projected inter class distance*:

$$\Delta_r(S_n) := \sup_P \left\{ \inf_{y_i \neq y_{i'}} \|P(x_i) - P(x_{i'})\|_2 ; \mathit{rank} - r \right\}$$

Dimesion independent generalization bounds

Theorem (Generalization bound)
For any $\epsilon \in (0,1)$ and $r \in [d]$, there exist $C(r), C_\epsilon(r > 0)$ such that the following holds. If $(x, y) \sim \mathbb{P}$ is such that for some $R > 0$ and $\Delta_r(\mathbb{P}) \leq C(r)$, it holds $\Delta_r(S_n) \leq \Delta_r(\mathbb{P})$ and $\|x\|_2 \leq R$ almost surely, then it holds with probability at least $1 - \delta$ over the choice of i.i.d. samples S_n, for f the $\mathcal{F}_1$-max margin classifier on S_n,

$$\mathbb{P}[yf(x) < 0] \leq \frac{C_\epsilon(r)}{\sqrt{n}}\left(\frac{R}{\Delta_r(\mathbb{P})}\right)^{\frac{r+3}{2-\epsilon}} + \sqrt{\frac{log(B)}{n}} + \sqrt{\frac{log(1/\delta)}{2n}}$$

where B some constant. The same bound applies to the $\mathcal{F}_2$-max margin classifier for $r = d$

Conclusion

- For wide two-layer ReLU neural networks, training both layers or only the ouput layer leads to very different implicit biases
- When training both, the classifier converges to a max-margin classifier for a non-Hilbertian norm
- This problem does not seem to be directly solvable with know convex methods in high dimension