To understand deep learning we need to understand kernel learning

ZIYU WANG

Reading Group Presentation
10 December 2018

Contents

GP-BNN Correspondence

The DGP corresponding to a single-layer BNN prior is

$$w_i^{(1)} \sim \mathcal{N}(0, C_1/N), \qquad i \in [N];$$
$$w_i^{(0)} \sim \mathcal{N}(0, C_0), \qquad i \in [N];$$
$$f(x) := \sum_{i \le N} w_i^{(1)} \varphi(w_i^{(0)} x).$$

Let $\varepsilon_i^{(1)} := \sqrt{N} w_i^{(1)}$. Now

$$\begin{bmatrix} f(x) \\ f(x') \end{bmatrix} = \frac{1}{\sqrt{N}} \sum_{i \le N} \begin{bmatrix} \varepsilon_i^{(1)} \varphi(w_i^{(0)} x) \\ \varepsilon_i^{(1)} \varphi(w_i^{(0)} x') \end{bmatrix} \tag{1}$$

is a sum of i.i.d. rvs. So when $\varphi(NormalRv)$ is well-behaved, by CLT $(f(x), f(x'))$ are jointly normal, and $p(f) \to \mathcal{GP}$.

When the final layer has multiple outputs, they are uncorrelated:

$$\mathbb{E}[f_0(x) f_1(x)]$$
$$= \mathbb{E}\left[\sum_{i,i'} w_{i \to 0}^{(1)} \varphi(w_i^{(0)} x) w_{i' \to 1}^{(1)} \varphi(w_{i'}^{(0)} x) \right] = 0.$$

Thus $[f_0(x); f_1(x')] \to \mathcal{N} \Rightarrow \{f_i\}$ form *independent* GPs.

The idea that for a multi-output NN, its top-layer outputs are *a priori* independent is disturbing to many (Neal; MacKay; etc).

They believe this behavior is undesirable for e.g. pattern recognition, and advocate priors that break this. The easiest fix is to choose a φ s.t. $\varphi(NormalSample)$ has unbounded moments.

It's not clear if the implication will be profound. An analogy is to impose a prior of $\mathcal{N}(0, I)$ when you should use a hierarchical model for covariance.

Bayesian DNN to GP.—The DGP becomes

$$\begin{aligned} f_i^{(\mu)}(x) &:= \sum_{k=1}^{N_\mu} w_{k\to i}^{(\mu)} g_k^{(\mu-1)}(x) + b_i^{(\mu)}, \\ g_i^{(\mu)}(x) &:= \varphi[f_i^{(\mu)}(x)], \end{aligned}$$

where $w_{(\cdot)}^{(\mu)} \sim \mathcal{N}(0, C_w^{(\mu)}), b_{(\cdot)}^{(\mu)} \sim \mathcal{N}(0, C_b^{(\mu)})$.

Let the first two moments of $[f_k^{(\mu-1)}(x); f_k^{(\mu-1)}(x')]$ be 0 and $\mathbf{K}_{x,x'}$ for all k. We can calculate the covariance $\mathbb{E}[f_i^{(\mu)}(x) f_j^{(\mu)}(x')]$:

$$\begin{aligned} &\mathbb{E}[f_i^{(\mu)}(x) f_j^{(\mu)}(x')] \\ =&\mathbb{E}\left\{\sum_{k,k'} w_{k\to i}^{(\mu)} g_k^{(\mu-1)}(x) w_{k'\to j}^{(\mu)} g_{k'}^{(\mu-1)}(x')\right\} + \delta_{ij} C_b^{(\mu)} \\ =&\delta_{ij}\mathbb{E}\left\{\sum_{k} w_{k\to i}^{(\mu)} g_k^{(\mu-1)}(x) w_{k\to i}^{(\mu)} g_k^{(\mu-1)}(x')\right\} + \delta_{ij} C_b^{(\mu)} \\ =&\delta_{ij}[C_w^{(\mu)} N_\mu \mathbb{E}[g_k^{(\mu-1)}(x) g_k^{(\mu-1)}(x')] + C_b^{(\mu)}] \\ =&\delta_{ij}[\tilde{C}_w^{(\mu)} \mathbb{E}_{(\varepsilon,\varepsilon')\sim\mathcal{N}(0,K)}[\varphi(\varepsilon)\varphi(\varepsilon')] + C_b^{(\mu)}]. \end{aligned} \tag{2}$$

Notice that up to now, we haven't used normality.

A hand-waving argument.—If we further require

$$\{f_k^{(\mu-1)} : k \in [N_{\mu-1}]\}$$

be jointly normal, they become independent, and by multivariate CLT, as $N_\mu \to \infty$, $\{f_k^{(\mu)} : k \in [N_\mu]\}$ are jointly normal $\Rightarrow$ independent GPs with kernel (2).

As $N_1 \to \infty$, $f_k^{(1)}(\cdot)$ become independent GPs following the single-layer argument.

Thus we claim the DNN prior converges to a GP if we "set $N_\mu \to \infty$ for $\mu = 1, 2, \ldots$ consecutively".

A rigorous proof for weak convergence.—can be found in Matthews et al (2018). The idea is that

(1) $f_n \xrightarrow{w} \mathcal{GP} \Leftrightarrow \texttt{FiniteLinearProj}[f_n] \xrightarrow{w} \mathcal{N}$;
(2) CLT holds for *exchangeable* and uncorrelated rvs.

Simulations.—

(1) 4-layer DNNs with width 50 well approximates the GP prior, in terms of 12-dimensional cross-sectional MMD.
(2) On a 2D regression task with tens of input, log marginal likelihood of BNN / GP differs by $\sim 10\%$.
(3) On the Yacht dataset, using GP / BNN to optimize kernel hyper-parameters results in significantly different results; under their settings BNN consistently outperforms GP.

A Spectral View to Kernel Ridge Regression

Now it is tempting[1] to study the MAP given a GP prior, i.e. the KRR estimator.

$$\begin{aligned} \hat{f} &:= \operatorname{argmin}_f -\|f(X_{\text{tr}}) - Y_{\text{tr}}\|_2^2 + \alpha\|f\|_{\mathcal{H}} \\ \Rightarrow \quad \hat{f}(x_{\text{te}}) &= K_{\text{er}}(K_{\text{rr}} + \alpha I)^{-1} Y_{\text{tr}}. \end{aligned}$$

As we will see, generalization performance of KRR relates to the *intrinsic dimensionality* of the regression problem.

A Functional / Asymptotic View.—Let $X_{\text{tr}} \sim P(x)$. As $N_{\text{train}} \to \infty$, eigenvectors of K_{rr} can be viewed as eigenfunctions of the following operator:

$$\begin{aligned} K_{\text{rr}}: \quad f(\mathbf{X}_{\text{tr}}) &\mapsto K_{\text{rr}} f(X_{\text{tr}}) \qquad \xrightarrow{N_{\text{train}} \to \infty} \\ \mathbf{K}: \quad f &\mapsto \int f(\cdot) k(\cdot, y) P(dy). \end{aligned}$$

As $N_{\text{train}}, N_{\text{test}} \to \infty$, the KRR estimator

$$Y_{\text{tr}} \mapsto K_{\text{er}}(K_{\text{rr}} + \lambda I)^{-1} Y_{\text{tr}}$$

can also be viewed as an operator on $L^2(P)$, namely

$$g \mapsto \mathbf{K}(\mathbf{K} + \lambda \cdot \text{id})^{-1} g.$$

Let the eigenfunctions and eigenvalues of K be

$$\{\lambda_i, \psi_i(\cdot)\}_{i=1}^{\infty},$$

now

$$\mathbf{K}(\mathbf{K} + \alpha \cdot \text{id})^{-1} g \quad = \quad \sum_i \frac{\lambda_i}{\lambda_i + \alpha} \langle \psi_i, g \rangle \psi_i$$

$\Longrightarrow$ Asymptotically, KRR performs low-pass filtering.

Why is this interesting?.—The filtering effect is significant, at least for problems with moderately dimension:

Ex. (Belkin, 2018) When P is the Lebesgue measure on $\mathbb{R}^d$, k is RBF with bandwidth σ^2, we will have

$$\lambda_n \sim \exp(-C\sigma^2 n^{1/d}).$$

ψ_i s.t. $\lambda_i \ll \alpha$ is basically filtered out with this regularizer. As λ_i decays exponentially, even a very small value of α effectively makes KRR a *truncated series estimator* (Belkin, 2018; Section 4). This is desirable as higher-frequency eigenfunctions are harder to estimate, and are undesirable, if you believe in the $\mathcal{GP}(0, k)$ prior.

> Your universal-approximating NN / GP / KRR regressors may encode stronger assumptions than you expect.

More example of eigenvalue decay.—

- A k-th order polynomial kernel has at most k non-zero eigenvalues;

[1] Meanwhile, recall it is not necessarily the weight-space MAP.

- When $\dim \operatorname{supp} P = d' < d$ and k is the RBF kernel, $\lambda_n \prec \exp(-C\sigma^2 n^{1/d'})$.

Eigenspectrum of $\mathbf{K}$ is a sensible measure of model complexity, and is determined by both the kernel and the geometry of $P(x)$.

Neural Tangent Kernel

(Full-batch) gradient descent in weight space:

$$\begin{aligned}\theta_{\ell+1} &\leftarrow \theta_\ell - \epsilon \frac{\partial L(\theta)}{\partial \theta} \\ &= \theta_\ell - \epsilon \left[\frac{\partial f(\mathbf{X})}{\partial \theta}\right]^\top \frac{\partial L[f(\mathbf{X})]}{\partial f(\mathbf{X})}\end{aligned}$$

in function space:

$$f_{\ell+1}(\mathbf{X}) \leftarrow$$

$$f_\ell(\mathbf{X}) - \epsilon \underbrace{\frac{\partial f(\mathbf{X})}{\partial \theta}\left[\frac{\partial f(\mathbf{X})}{\partial \theta}\right]^\top}_{\text{preconditioner}} \underbrace{\frac{\partial L[f(\mathbf{X})]}{\partial f_\ell(\mathbf{X})}}_{\text{intended F-S update } d_\ell}$$

What does this preconditioner do?.—Consider a linear model

$$f(\mathbf{X}) := \underbrace{\mathbf{X}}_{N_{train} \times N_X} \underbrace{\theta}_{N_X \times 1}.$$

If $\mathbf{X}$ are random features corresponding to a certain kernel k, the preconditioner

$$\lim_{N_X \to \infty} \mathbf{X}\mathbf{X}^\top = k(\mathbf{X}, \mathbf{X})$$

is the gram matrix.

If we decompose the intended function-space update

$$d_\ell =: \sum_i \langle d_\ell, \psi_i \rangle \, \psi_i$$

the update rule

$$f_{\ell+1} - f_\ell = \epsilon \sum_i \lambda_i \langle d_\ell, \psi_i \rangle \, \psi_i$$

decreases the learning rate for higher-frequency eigenfunctions.

Early stopping acts as a low-pass filter.

Again, the result is "exponentially strong": increasing training time to K times the original only add in $O(\log K)$ eigenfunctions.

For general non-linear models, the preconditioner, as a function of θ, is time-varying. So the following may seem mind-blowing:

Thm. *For infinitely wide DNN with $|\sigma'|$ and $|\sigma''|$ bounded,*

(1) The preconditioner at initialization converges in probability to a fixed transform;
*(2) and it **remains constant** during training.*

Notations.—for $l \in [L], i \in [N_l], j \in [N_{l-1}]$

$$W_{ij}^{(l)} \sim \mathcal{N}(0, I),$$

$$\tilde{\alpha}^{(l)}(x) := \frac{1}{\sqrt{N_{L-1}}} W^{(l)} \alpha^{(l-1)}(x) + b^{(l)}, \quad \text{(pre-activation)}$$

$$\alpha^{(l)} := \sigma(\tilde{\alpha}^{(l)}) \quad \text{(activation)}$$

Initial transform.—Let $\tilde{\alpha}_i^{(l)}, \alpha_i^{(l)} \in \mathbb{R}^{N_{train} \times 1}$ denote the evaluation on $\mathbf{X}$.

$$\frac{\partial \tilde{\alpha}_i^{(l)}}{\partial \Theta} \left(\frac{\partial \tilde{\alpha}_{i'}^{(l)}}{\partial \Theta} \right)^\top = \sum_{\theta \in \text{layer } l} \frac{\partial \tilde{\alpha}_i^{(l)}}{\partial \theta} \left(\frac{\partial \tilde{\alpha}_{i'}^{(l)}}{\partial \theta} \right)^\top + (\theta \notin \text{layer } l) \tag{3}$$

We will show by induction that when we take sequentially the limit $n_i \to \infty$ for $i < l$, the above

(1) Converges in probability to some deterministic matrix $K^{(l)}$, if $i = i'$;
(2) Converges to 0 otherwise.

(a) For $\theta \in$ layer l, say $\theta = W_{ji}^{(l)}$ for some j,

$$\sum_j \frac{\partial \tilde{\alpha}_i^{(l)}}{\partial W_{ji}^{(l)}} \left(\frac{\partial \tilde{\alpha}_{i'}^{(l)}}{\partial W_{ji}^{(l)}} \right)^\top = \delta_{ii'} \sum_{j=1}^{N_{L-1}} \frac{1}{N_{L-1}} \alpha_j^{(l-1)} \left(\alpha_j^{(l-1)} \right)^\top \xrightarrow{\text{CLT}} \delta_{ii'} \Sigma^{(l-1)}.$$

(in the last line, we first let $N_{1\ldots l-2} \to \infty$, now $\{\alpha_j^{(l-1)} : j \in [N_{l-1}]\}$ converge to i.d. GPs with deterministic kernel $\Sigma^{(l-1)}$ at initialization.)

$$\frac{\partial \tilde{\alpha}_i^{(l)}}{\partial b_i^{(l)}} \left(\frac{\partial \tilde{\alpha}_{i'}^{(l)}}{\partial b_i^{(l)}}\right)^\top = \delta_{ii'} \mathbf{1}\mathbf{1}^\top.$$

(b) For $\theta \notin$ layer l, $\frac{\partial \tilde{\alpha}_i^{(l)}}{\partial \theta} \left(\frac{\partial \tilde{\alpha}_{i'}^{(l)}}{\partial \theta}\right)^\top$ equals

$$\left[\sum_j \frac{\partial \tilde{\alpha}_i^{(l)}}{\partial \alpha_j^{(l-1)}} \frac{\partial \alpha_j^{(l-1)}}{\partial \theta}\right] (\text{Transpose}, i \mapsto i')$$

$$= \frac{1}{N_{l-1}} \left[\sum_j W_{ij}^{(l)} \frac{\partial \tilde{\alpha}_j^{(l-1)}}{\partial \theta} \sigma'(\tilde{\alpha}_j^{(l-1)})\right] (\text{T}, i \mapsto i')$$

$$\xrightarrow{p} \frac{1}{N_{l-1}} \sum_j W_{ij}^{(l)} W_{i'j}^{(l)} (\sigma'(\tilde{\alpha}_j^{(l-1)}))^2 K^{(l-1)}.$$

The last line holds when N_l is fixed and $N_{1\ldots l-2} \to \infty$. Set $N_{l-1} \to \infty$, and proof completes by LLN (?).

Invariance during training.—Now α, W varies with t. The authors showed that as $N_{1\ldots l-2} \to \infty$, with probability 1

$$\|\alpha_j^{(l-1)}(t) - \alpha_j^{(l-1)}(0)\|_2 = O\left(\frac{1}{\sqrt{N_{l-1}}}\right), \tag{4}$$

$$\|W_{j:}^{(l)}(t) - W_{j:}^{(l)}(0)\|_2 = O\left(\frac{1}{\sqrt{N_{l-1}}}\right) \tag{5}$$

hold uniformly for all t. Now (each matrix element in) both sums in (3) changes by $O\left(\frac{1}{\sqrt{N_{l-1}}}\right)$, and taking $N_{l-1} \to \infty$ completes the proof.

Rem. (1) You should not be bothered by (4), as collectively $\|\alpha^{(l-1)}\|(t)$ can still change. Intuitively, (4) should hold because at any time, the gradient $\alpha_j^{(l-1)}$ receives is scaled by $N_{l-1}^{-1/2}$.

(2) Notice K is independent of training data. So a more general statement is that in the complete, infinite-dimensional function space,

$$\frac{df(\mathcal{X})}{dt} = K_{ntk}(\mathcal{X}, \mathbf{X}_{train}) d_t(\mathbf{X}). \tag{6}$$

(3) Before we proceed, here is the definition of neural tangent kernel.

$$K^{(0)}_{\text{ntk}} = K^{(0)}_{\text{PriorGP}},$$
$$K^{(L)}_{\text{ntk}} = K^{(L)}_{\text{GP}} + K^{(L-1)}_{\text{ntk}} \odot \mathbb{E}_{f\sim K^{(L-1)}_{\text{GP}}}[\sigma'(f)\,(\sigma'(f))^\top].$$

Notice it is different from the prior GP kernel; however, for common nonlinearities the expectation < 1, so maybe NTK and prior GP kernel aren't too different.

Early stopping as LP filter.—this argument still applies, but with K_{ntk} instead of K_{gp}.

More magic in least square regression.—For least-square regression, $d_t(\mathbf{X}) = f_t(\mathbf{X}) - \mathbf{Y}$. If we assume K_{ntk} is indeed fixed throughout training, (6) is a linear ODE and we can obtain $f(\mathcal{X})_{t\to\infty}$ in closed form. The result is

$$f_\infty(x) = K_{er}K_{rr}^{-1}\mathbf{Y} + (f_0(x) - K_{er}K_{rr}^{-1}f_0(\mathbf{X})),$$

where $K_{er} = K_{ntk}(x, \mathbf{X}), K_{rr} = K_{ntk}(\mathbf{X}, \mathbf{X})$.

The second term is 0 for $x \in \mathbf{X}$, and the first term is the *maximum a posteriori estimation* for $\mathcal{GP}(0, K_{ntk})$ with $\sigma^2 = 0$.

Furthermore, as f_0 corresponds to a randomly initialized DNN, it converges to a GP. So we can derive the variance for out-of-sample x:

$$\begin{aligned}\text{Cov}[f_\infty(\mathcal{X})] =& K^{(0)}_{ee} + K_{er}K_{rr}^{-1}K^{(0)}_{rr}K_{rr}^{-1}K_{re} - \\ & K_{er}K_{rr}^{-1}K^{(0)}_{re} - K^{(0)}_{er}K_{rr}^{-1}K_{re}.\end{aligned}$$

where $K^{(0)}$ are Gram matrices corresponding to the GP kernel, and K corresponds to the NT kernel. If we assume $K \approx K^{(0)}$, this is the GP posterior covariance!

Gradient descent is* Bayesian inference.

*: only holds for RF expansion.

Connection to existing empirical findings.—

(1) It is known for long that GD ensemble produces sensible uncertainty estimate. On UCI regression datasets, a NIPS17 paper shows the performance of GD ensemble matches that of MFVI and MC dropout, but is still worse than SoTA BNN; for adversarial defense ensembled GD is notably better than a single MAP.
(2) In general, $K \neq K^{(0)}$. And it is hard to relate their eigendecompositions.
(3) There are other ensemble-like heuristics that produces better uncertainty estimates. E.g. our f-POVI; randomized prior (NIPS18). Intuitively, QMC is better than MC.

Simulations.—

- For 4-layer FNN, NTK in $N = 500$ is notably different from the asymptotic value; $N = 10000$ gets closer.

- In both cases, the NTK stays closer to the initial value during the early phase of training.

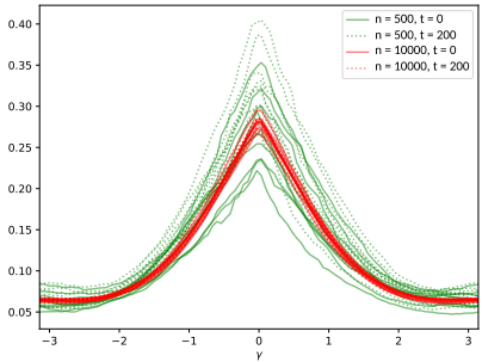


Figure 1: Convergence of the NTK to a fixed limit for two widths n and two times t.

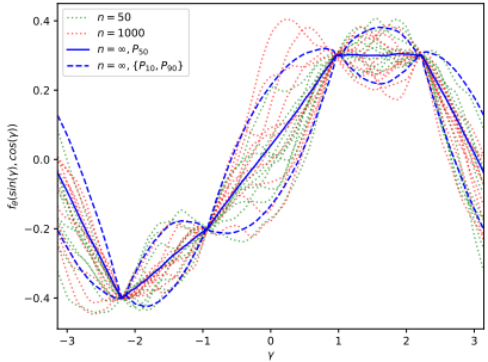
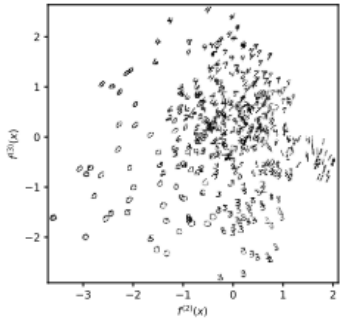


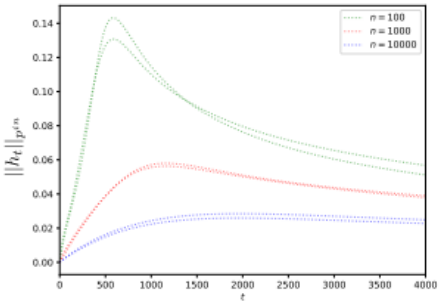
Figure 2: Networks function f_θ near convergence for two widths n and 10th, 50th and 90th percentiles of the asymptotic Gaussian distribution.

6.1 Convergence of the NTK

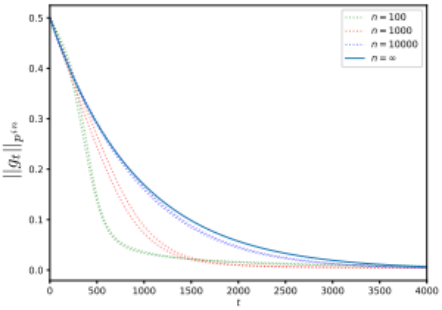
The first experiment illustrates the convergence of the NTK $\Theta^{(L)}$ of a network of depth $L = 4$ for two different widths $n = 500, 10000$. The function $\Theta^{(4)}(x_0, x)$ is plotted for a fixed $x_0 = (1, 0)$ and $x = (cos(\gamma), sin(\gamma))$ on the unit circle in Figure 1. To observe the distribution of the NTK, 10 independent initializations are performed for both widths. The kernels are plotted at initialization $t = 0$ and then after 200 steps of gradient descent with learning rate 1.0 (i.e. at $t = 200$). We approximate the function $f^*(x) = x_1 x_2$ with a least-squares cost on random $\mathcal{N}(0, 1)$ inputs.



(a) The 2nd and 3rd principal components of MNIST.



(b) Deviation of the network function f_θ from the straight line.



(c) Convergence of f_θ along the 2nd principal component.