Notes on Kernel and Rich Limits in Neural Network Training

Henry Smith

Yale University

January 15, 2022

1 Kernel and Rich Regimes in Overparametrized Models, Woodworth et al. 2020

1.1 Problem Setup

The paper considers models $f : \mathbb{R}^p \times \mathcal{X} \to \mathbb{R}$ which map parameters $\boldsymbol{w} \in \mathbb{R}^p$ and observations $\boldsymbol{x} \in \mathcal{X}$ to predictions $f(\boldsymbol{w}, \boldsymbol{x})$. Let $F(\boldsymbol{w}) \in \{f : \mathcal{X} \to \mathbb{R}\}$ be the predictor implemented by the parameters $\boldsymbol{w}$. Of particular interest to the authors are models which are linear in the observations $\boldsymbol{x}$ (but not necessarily in the parameters $\boldsymbol{w}$). Since, in this case, $F(\boldsymbol{w})$ is in the dual space of $\mathcal{X}$, then we have that by Riesz Representation, $F(\boldsymbol{w})(\boldsymbol{x}) = \langle \boldsymbol{\beta}_{\boldsymbol{w}}, \boldsymbol{x} \rangle$ for some $\boldsymbol{\beta}_{\boldsymbol{w}}$. The paper focuses on D-homogeneous models in the parameter space $\boldsymbol{w}$ as discussed in [3], which satisfy $F(c \cdot \boldsymbol{w}) = c^D F(\boldsymbol{w})$ for any $c \in \mathbb{R}_+$. For D-homogeneous networks, it is important to note that scaling the output by a factor of $c > 0$, which is the focus of [3], is equivalent to scaling the input by a factor of $c^{1/D}$.

Further, the paper considers the square loss function $L(\boldsymbol{w}) = \widetilde{L}(F(\boldsymbol{w})) = \sum_{i=1}^{N} (f(\boldsymbol{w}, \boldsymbol{x}_n) - y_n)^2$ for the model F over the training set $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)$. Woodworth and colleagues minimize the loss L using the gradient flow dynamics, which can be thought of as gradient descent with the stepsize η limiting to 0. More explicitly, the gradient flow dynamics are

$$\dot{\boldsymbol{w}}(t) = -\nabla L(\boldsymbol{w}(t)).$$

The principal result of [3] is that, under suitable conditions, the gradient flow dynamics on $\boldsymbol{w}$ approach those of a linearized model as the scale of initialization approaches infinity. To capture the scale of initialization, the authors consider the parameter $\alpha \in \mathbb{R}_+$. For fixed initialization scale α, let $\boldsymbol{w}_{\alpha, \boldsymbol{w}_0}(t)$ be the gradient flow path with the initial condition $\boldsymbol{w}_{\alpha, \boldsymbol{w}_0}(0) = \alpha \boldsymbol{w}(0)$.

Woodworth and colleagues are particularly interested in the case of $N \ll p$, where there are many global minimizers of $L(\boldsymbol{w})$ with $L(\boldsymbol{w}) = 0$. That is, the model is overparameterized/underdetermined and so we can fit the observations exactly. Of particular interest to the paper is which of the global minimizers $\boldsymbol{w}^{\infty}_{\alpha, \boldsymbol{w}_0} := \lim_{t \to \infty} \boldsymbol{w}_{\alpha, \boldsymbol{w}_0}(t)$ gradient flow converges to.

1.2 Kernel Regime

Locally, the gradient descent depends only on the first-order approximation in $\boldsymbol{w}$:

$$f(\boldsymbol{w}, \boldsymbol{x}) = f(\boldsymbol{w} - \boldsymbol{w}(t), \boldsymbol{x}) + \langle \boldsymbol{w} - \boldsymbol{w}(t), \phi_{\boldsymbol{w}(t)}(\boldsymbol{x}) \rangle + \mathcal{O}(\|\boldsymbol{w} - \boldsymbol{w}(t)\|^2),$$

where $\phi_{\boldsymbol{w}(t)}(\boldsymbol{x}) = \nabla_{\boldsymbol{w}} f(\boldsymbol{w}(t), \boldsymbol{x})$ is the *feature map* corresponding to the *tangent kernel* $K_{\boldsymbol{w}(t)}(\boldsymbol{x}, \boldsymbol{x}') = \langle \nabla_{\boldsymbol{w}} f(\boldsymbol{w}(t), \boldsymbol{x}), \nabla_{\boldsymbol{w}} f(\boldsymbol{w}(t), \boldsymbol{x}') \rangle$. Clearly this approximation is linear in $\boldsymbol{w}$ but need not be linear in $\boldsymbol{x}$ if the underlying model $F(\boldsymbol{w})$ is not linear in $\boldsymbol{x}$.

[3] provides an in-depth look at the *kernel regime* in which the gradient flow dynamics $\dot{w}(t)$ *depend only on a linear function in* $\boldsymbol{w}$. This means that $\phi_{\boldsymbol{w}(t)}(\boldsymbol{x})$, and thus the tangent kernel, is constant throughout training. Under certain conditions on the D-homogeneous model $F(\boldsymbol{w})$ and the loss function L, the kernel regime manifests as $\alpha \to \infty$ in the initialization $\boldsymbol{w}_\alpha(0) = \alpha \boldsymbol{w}(0)$. Since this is a theoretical limit, we use the kernel regime to refer to the case in which the gradient flow dynamics are "close to" the linearized dynamics and so the *tangent kernel does not change significantly* throughout training (see Theorem 2.4 in [3]).

In particular, under the kernel regime, training $f(\boldsymbol{w}, \boldsymbol{x})$ is equivalent to training the affine model $f_K(\boldsymbol{w}, \boldsymbol{x}) = \alpha^D f(\boldsymbol{w}(0), \boldsymbol{x}) + \langle \phi_{\boldsymbol{w}(0)}(\boldsymbol{x}), \boldsymbol{w} - \boldsymbol{w}(0) \rangle$ using kernelized gradient descent/flow with the kernel $K_{\boldsymbol{w}_0}$ and a bias term $f(\boldsymbol{w}_0, \boldsymbol{x})$. Observe that $\phi_{\boldsymbol{w}(t)}(\boldsymbol{x}) = \nabla_{\boldsymbol{w}(t)} f_K(\boldsymbol{w}, \boldsymbol{x}) = \phi_{\boldsymbol{w}(0)}(\boldsymbol{x})$, and so the tangent kernel is indeed constant throughout training. Minimizing the loss of this affine model with gradient flow reaches the solution nearest to initialization where distance is measured with respect to the RKHS norm determined by K_0. That is, $F(\boldsymbol{w}_\alpha^\infty) = \operatorname{argmin}_h \|h - F(\boldsymbol{w}_0)\|_{K_0}$ s.t. $h(X) = \boldsymbol{y}$. Here, our RKHS norm is $\|g\|_{K_0} = \inf\{\|\ell\|_2 : \ell \in \mathbb{R}^p, g(\boldsymbol{x}) = \langle \ell, \phi_{\boldsymbol{w}(0)}(\boldsymbol{x}) \rangle, \ \forall \boldsymbol{x} \in \mathcal{X}\}$. [3] advises choosing an unbiased initialization $\boldsymbol{w}_0$ such that $F(\boldsymbol{w}_0) = 0$. This means that the bias term $\alpha^D f(\boldsymbol{w}(0), \boldsymbol{x})$ depending on the initialization scale vanishes.

1.3 Example: Linear Regression

The primary contribution of the paper is the explicit characterization of the implicit bias when training with gradient descent as a function of α, the scale of initialization, in the following least-squares problem: For $\mathcal{X} = \mathbb{R}^d$, define the model

$$f(\boldsymbol{w}, \boldsymbol{x}) = \sum_{i=1}^{d} (\boldsymbol{w}_{+,i}^2 - \boldsymbol{w}_{-,i}^2) \boldsymbol{x}_i = \langle \boldsymbol{\beta}_{\boldsymbol{w}}, \boldsymbol{x} \rangle, \quad \boldsymbol{w} = \begin{bmatrix} \boldsymbol{w}_+ \\ \boldsymbol{w}_- \end{bmatrix} \in \mathbb{R}^{2d}, \quad \boldsymbol{\beta}_{\boldsymbol{w}} = \boldsymbol{w}_+^2 - \boldsymbol{w}_-^2$$

where $\boldsymbol{z}^2$ for $\boldsymbol{z} \in \mathbb{R}^d$ denotes element-wise squaring. This model represents a "diagonal" neural network where each input component $\boldsymbol{x}_i$ is connected to a positive input unit with weight $\boldsymbol{w}_{+,i}^2$ and a negative input unit with weight $\boldsymbol{w}_{-,i}^2$. This differs from the least-squares problem discussed in [4] where there is a single set of positive weights representing the squared least-squares coefficients (i.e. we have model $f(\boldsymbol{w}, \boldsymbol{x}) = \sum_{i=1}^d \boldsymbol{w}_i^2 \boldsymbol{x}_i$). The authors prefer the former formulation of the least-squares problem for two reasons: (1) by choosing $\boldsymbol{w}_0$ such that $\boldsymbol{w}_+ = \boldsymbol{w}_-$, then the model vanishes at initialization $F(\alpha \boldsymbol{w}) = 0$ without this being a saddle point of the objective and (2) the image of $F(\boldsymbol{w})$ is all linear functionals i.e. $\operatorname{img}(F(\boldsymbol{w})) = \mathcal{X}^*$.

The authors study the underdetermined case $N \ll d$ when there are many solution vectors $\boldsymbol{\beta} \in \mathbb{R}^d$ that satisfy $X\boldsymbol{\beta} = \boldsymbol{y}$. Let $\boldsymbol{\beta}_{\alpha, \boldsymbol{w}_0}^\infty$ be the solution reached by gradient flow when initialized at $\boldsymbol{w}_+(0) = \boldsymbol{w}_-(0) = \alpha \boldsymbol{w}_0$. Woodworth and colleagues first consider the case of $\boldsymbol{w}_0 = \vec{1}$ and then generalize their results.

For $\boldsymbol{w}_0 = \vec{1}$, it is easy to verify that the tangent kernel at initialization is $K_{\boldsymbol{w}(0)}(\boldsymbol{x}, \boldsymbol{x}') = 8\alpha^2 \langle \boldsymbol{x}, \boldsymbol{x}' \rangle$, which is simply a scaling of the standard inner product kernel (just write out $\nabla_{\boldsymbol{w}} f(\boldsymbol{w}, \boldsymbol{x})|_{\boldsymbol{w} = \boldsymbol{w}(0)}$). That is, we have $\|\boldsymbol{\beta}\|_{K_0} \propto \|\boldsymbol{\beta}\|_2$. And so in the kernel regime, $\boldsymbol{\beta}_{\alpha, \vec{1}}^\infty$ is equal to the minimum ℓ_2 solution $\boldsymbol{\beta}_{\ell_2}^* := \operatorname{argmin}_{X\boldsymbol{\beta} = y} \|\boldsymbol{\beta}\|_2$. Conversely, as $\alpha \to 0$, we approach what Woodworth and colleagues classify as the "rich limit." Under the rich limit, the tangent kernel changes significantly during optimization, and the model

$f(\boldsymbol{w}, \boldsymbol{x})$ is not linear in $\boldsymbol{w}$. For the linear regression problem under consideration with $\boldsymbol{w}_0 = \vec{1}$, [4] proves in Corollary 2 that $\lim_{\alpha \to 0} \boldsymbol{\beta}^{\infty}_{\alpha, \vec{1}} = \boldsymbol{\beta}^*_{\ell_1} = \operatorname{argmin}_{X\boldsymbol{\beta} = y} \|\boldsymbol{\beta}\|_1$.

And so we understand that for the current model of interest with initialization $\boldsymbol{w}_0 = \vec{1}$, the gradient flow solution in the kernel regime $\alpha \to \infty$ is the minimum ℓ_2 solution and the solution in the rich regime $\alpha \to 0$ is the minimum ℓ_1 solution.

1.4 Interpolation Between the Rich and Kernel Regimes

What the authors seek to understand, however, is how the gradient flow solution interpolates between the rich and kernel limits. That is, what can we say about the implicit bias when training the linear regression model for α small, for example? This is the subject of the following theorem:

Theorem 1. *For any $0 < \alpha < \infty$, if the gradient flow solution $\boldsymbol{\beta}^{\infty}_{\alpha, \vec{1}}$ for the squared parameterization model satisfies $X\boldsymbol{\beta}^{\infty}_{\alpha, \vec{1}} = \boldsymbol{y}$, then*

$$\boldsymbol{\beta}^{\infty}_{\alpha, \vec{1}} = \underset{\boldsymbol{\beta}}{argmin}\ Q_\alpha(\boldsymbol{\beta}) \text{ s.t. } X\boldsymbol{\beta} = \boldsymbol{y}$$

where $Q_\alpha(\boldsymbol{\beta}) = \alpha^2 \sum_{i=1}^d q\left(\frac{\boldsymbol{\beta}_i}{\alpha^2}\right)$ and $q(z) = \int_0^z arcsinh\left(\frac{u}{2}\right) du = 2 - \sqrt{4 + z^2} + z arcsinh\left(\frac{z}{2}\right)$.

See [5] for further analysis of how $Q_\alpha(\boldsymbol{\beta})$ reproduces the ℓ_2 and ℓ_1 minimization problems as $\alpha \to \infty$ and $\alpha \to 0$, respectively. In Theorem 2, Woodworth and colleagues quantify the scale of α necessary to guarantee approximation of the minimum ℓ_1 and ℓ_2 solutions:

Theorem 2. *For any $0 < \epsilon < d$, under the setting of Theorem 1 with $\boldsymbol{w}_0 = \vec{1}$,*

$$\alpha \leq \min\left\{ \left(2(1+\epsilon)\left\|\boldsymbol{\beta}^*_{\ell_1}\right\|_1\right)^{-\frac{2+\epsilon}{2\epsilon}}, \exp\left(-d/(\epsilon \left\|\boldsymbol{\beta}^*_{\ell_1}\right\|_1)\right) \right\} \implies \left\|\boldsymbol{\beta}^{\infty}_{\alpha, \vec{1}}\right\|_1 \leq (1+\epsilon)\left\|\boldsymbol{\beta}^*_{\ell_1}\right\|_1$$

$$\alpha \geq \sqrt{2(1+\epsilon)(1+2/\epsilon)\left\|\boldsymbol{\beta}^*_{\ell_2}\right\|_2} \implies \left\|\boldsymbol{\beta}^{\infty}_{\alpha, \vec{1}}\right\|_2^2 \leq (1+\epsilon)\left\|\boldsymbol{\beta}^*_{\ell_2}\right\|_2^2.$$

Theorem 2 illustrates a challenge with approximating the rich limit: while only polynomially large α suffices to approximate $\boldsymbol{\beta}^*_{\ell_2}$, one must have *exponentially small* α to approximate $\boldsymbol{\beta}^*_{\ell_1}$. As a result, experiments close to the rich limit may require very small initializations and, as a result, may be computationally intractable. The work from [1] suggests a remedy to this problem: increasing the depth (i.e. degree of homogeneity) of the model.

1.5 The Relation of Kernel and Rich Limits to Model Generalization

Interestingly, the kernel and rich limits are related to the generalizability of the model. Woodworth et al. illustrate this phenomenon by generating a training set $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim \mathcal{N}(0, I)$ and $y_n \sim \mathcal{N}(\langle \boldsymbol{\beta}^*, \boldsymbol{x}_n \rangle, 0.01)$, where $\boldsymbol{\beta}^*$ is an r^* sparse vector whose nonzero entries are equal to $1/\sqrt{r^*}$. The authors note that for $N \leq d$, gradient flow generally reaches a zero training error solution, but not all solutions generalize the same. In particular, they show that in the rich limit, $N = \Omega(r^* \log d)$ training points suffices for $\boldsymbol{\beta}^*_{\ell_1}$ to "generalize well"; in the kernel limit, though, good generalization requires $N = \Omega(d)$. This tells us that, in general, *good generalization of our model requires us to train close to the rich regime.* As previously noted, though, one cannot take α to be arbitrary small: as $w(0) \to \vec{0}$, we reach a saddle point of the objective function. Thus, [5] suggests working at the edge of the rich limit, ensuring generalizability of the solution while also making sure that the optimization problem is feasible. One can see this in section 7 of [5], where the authors perform experiements with neural networks on the MNIST and CIFAR10 datasets.

1.6 Reconsidering $\boldsymbol{w}_0$

So far, we have only considered the problem of $\boldsymbol{w}_0 = \vec{1}$. The authors give a more general statement of Theorem 1 as follows:

Theorem 1. *For any $0 < \alpha < \infty$ and $\boldsymbol{w}_0$ with no zero entries, if the gradient flow solution $\boldsymbol{\beta}^{\infty}_{\alpha,\boldsymbol{w}_0}$ for the squared parameterization model satisfies $X\boldsymbol{\beta}^{\infty}_{\alpha,\boldsymbol{w}_0} = \boldsymbol{y}$, then*

$$\boldsymbol{\beta}^{\infty}_{\alpha,\boldsymbol{w}_0} = \underset{\boldsymbol{\beta}}{argmin}\ Q_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta})\ s.t.\ X\boldsymbol{\beta} = \boldsymbol{y}$$

where $Q_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta}) = \sum_{i=1}^{d} \alpha^2 \boldsymbol{w}_{0,i}^2 q\left(\frac{\boldsymbol{\beta}_i}{\alpha^2 \boldsymbol{w}_{0,i}^2}\right)$ and $q(z) = \int_0^z arcsinh\left(\frac{u}{2}\right) du = 2 - \sqrt{4+z^2} + z arcsinh\left(\frac{z}{2}\right)$.

The authors then note that for small z, $q(z) = \frac{z^2}{4} + \mathcal{O}(z^4)$ and so for $\alpha \to \infty$

$$Q_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta}) = \sum_{i=1}^{d} \frac{\boldsymbol{\beta}_i^2}{4\alpha^2 \boldsymbol{w}_{0,i}^2} + \mathcal{O}(\alpha^{-6}).$$

That is, $Q_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta})$ is proportional to the ℓ_2 norm weighted by $\text{diag}(1/\boldsymbol{w}_0^2)$. Conversely, for large $|z|$, $q(z) = |z| \log |z| + \mathcal{O}(1/|z|)$, and so as $\alpha \to 0$,

$$\frac{1}{\log(1/\alpha^2)} Q_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta}) = \sum_{i=1}^{d} |\boldsymbol{\beta}_i| + \mathcal{O}(1/\log(1/\alpha^2)).$$

This tells us that in the rich limit, $Q_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta})$ is proportional to $\|\boldsymbol{\beta}\|_1$ regardless of the shape of the initialization $\boldsymbol{w}_0$. In summary, the specific initialization $\boldsymbol{w}_0$ *does* affect the implicit bias in optimization under the kernel regime, but *not* in the rich limit. For neural networks with i.i.d. initialized units, the particular value of $\boldsymbol{w}_0$ is analogous to the distribution used to initialize each unit. That is, changing the initialization distribution changes the tangent kernel at initialization and thus the kernel regime behavior; it does not, however, affect the rich limit.

1.7 Implicit Bias and Weights

The authors consider the possibility that the implicit bias is minimizing the ℓ_2 norm from initialization:

$$\boldsymbol{\beta}^{R}_{\alpha,\boldsymbol{w}_0} := F\left(\underset{\boldsymbol{w}}{\text{argmin}}\ \|\boldsymbol{w} - \alpha\boldsymbol{w}_0\|_2^2 \text{ s.t. } L(\boldsymbol{w}) = 0\right) = \underset{\boldsymbol{\beta}}{\text{argmin}}\ R_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta}) \text{ s.t. } X\beta = y$$

$$\text{where} \quad R_{\alpha,\boldsymbol{w}_0}(\boldsymbol{\beta}) = \min_{\boldsymbol{w}} \|\boldsymbol{w} - \alpha\boldsymbol{w}_0\|_2^2 \text{ s.t. } F(\boldsymbol{w}) = \boldsymbol{\beta}.$$

They remark that for the least-squares model $f(\boldsymbol{w}, \boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$, we indeed have $\boldsymbol{\beta}^{\infty}_{\alpha,\boldsymbol{w}_0} = \boldsymbol{\beta}^{R}_{\alpha,\boldsymbol{w}_0}$. That is, the implicit bias is captured by $R_{\alpha,\boldsymbol{w}_0}$. For the two-homogeneous model under consideration, however, this is not the case. Indeed for the case of $\boldsymbol{w}_0 = \vec{1}$, we have $Q_{\alpha,\vec{1}}(\boldsymbol{\beta}) = R_{\alpha,\vec{1}}(\boldsymbol{\beta})$ for $\alpha \to 0$ and $\alpha \to \infty$. From Figure 2 of [5], we observe that, in general, $Q_{\alpha,\vec{1}}(\boldsymbol{\beta}) \neq R_{\alpha,\vec{1}}(\boldsymbol{\beta})$, and they are not rescalings of each other. In particular, $R_{\alpha,\vec{1}}(\boldsymbol{\beta})$ approaches the rich limit $\|\boldsymbol{\beta}\|_1$ polynomially in α, while we know from previously that this is not the case for $Q_{\alpha,\vec{1}}(\boldsymbol{\beta})$.

1.8 Higher Order Models

As the authors note, deeper models correspond to higher order homogeneity (ex. a depth-D ReLU neural network is D-homogeneous). Accordingly, they generalize their model to a depth-D diagonal neural network

$$F_D(\boldsymbol{w}) = \boldsymbol{\beta}_{\boldsymbol{w},D} = \boldsymbol{w}_+^D - \boldsymbol{w}_-^D \quad \text{and} \quad f_D(\boldsymbol{w}, \boldsymbol{x}) = \langle \boldsymbol{w}_+^D - \boldsymbol{w}_-^D, \boldsymbol{x} \rangle.$$

As Woodworth et al. remark, this is simply a linear model with an unconventional parameterization, or a depth-D matrix factorization problem with commutative (diagonal) measurement matrices and diagonal factor matrices as studied by [1]. This formulation leads us to the following theorem:

Theorem 3. *For any* $0 < \alpha < \infty$ *and* $D \geq 3$*, if* $X\boldsymbol{\beta}^{\infty}_{\alpha,D} = y$*, then*

$$\boldsymbol{\beta}^{\infty}_{\alpha,D} = \mathit{argmin}_{\boldsymbol{\beta}} Q^D_\alpha(\boldsymbol{\beta}) \text{ s.t. } X\boldsymbol{\beta} = \boldsymbol{y}$$

where $Q^D_\alpha(\beta) = \alpha^D \sum_{i=1}^d q_D(\boldsymbol{\beta}_i/\alpha^D)$ *and* $q_D = \int h_D^{-1}$ *is the antiderivative of the unique inverse of* $h_D(z) = (1-z)^{-\frac{D}{D-2}} - (1+z)^{-\frac{D}{D-2}}$ *on* $[-1, 1]$*. Furthermore,* $\lim_{\alpha \to 0} \boldsymbol{\beta}^{\infty}_{\alpha,D} = \boldsymbol{\beta}^*_{\ell_1}$ *and* $\lim_{\alpha \to \infty} \boldsymbol{\beta}^{\infty}_{\alpha,D} = \boldsymbol{\beta}^*_{\ell_2}$*.*

The minimum ℓ_1 solution in the rich limit for depth-D diagonal neural networks has already been observed by Arora et al. While the rich and kernel limits do not change as the depth of the diagonal network increases, the transition between the rich and kernel regimes *does* change. In particular, *for higher orders of homogeneity, the transition into the extreme regimes is sharper.* Even at $D = 3$, the scale of α needed to approximate $\left\|\boldsymbol{\beta}^*_{\ell_1}\right\|_1$ is polynomial rather than exponential in the case of $D = 2$ (see Figure 3(b) in [5]).

2 On Lazy Training in Differential Programming, Chizat, Oyallon, and Bach 2018

Chizat's paper is particularly important as it provides much of the theoretical grounding upon which [5] is built.

2.1 Lazy Training

Let us consider parameter space $\mathbb{R}^p$, Hilbert space $\mathcal{F}$, a smooth model $h : \mathbb{R}^p \to \mathcal{F}$, and a smooth loss $R : \mathcal{F} \to \mathbb{R}_+$. Using gradient-based methods (i.e. gradient flow), the paper seeks to minimize the objective function $F : \mathbb{R}^p \mapsto \mathbb{R}_+$ defined

$$F(w) := R(h(w)).$$

With the initialization w_0, the authors also define the *linearized model* $\overline{h}(w) = h(w_0) + Dh(w_0)(w - w_0)$ around w_0. The corresponding objective function is

$$\overline{F}(w) := R(\overline{h}(w)).$$

At the beginning of training, it is known that the optimization paths of F and $\overline{F}$ are close to each other. The *lazy training* regime (what [5] refers to as the "kernel regime") occurs when the gradient flow dynamics of F and $\overline{F}$ are close for *all* $t \geq 0$, not just near $t = 0$.

A natural question to arise from this discussion is when the lazy training regime occurs. The authors provide a general criterion for the lazy regime, which relates the relative change of the objective $\Delta(F)$ to the relative change of the differential $\Delta(Dh)$.

In particular, assume that the initial set of parameters w_0 is not a minimizer so that $F(w_0) > 0$ and is not a critical point of the objective F so that $\nabla F(w_0) \neq 0$. Consider the gradient descent step $w_1 = w_0 - \eta \nabla F(w_0)$ with small stepsize $\eta > 0$. The relative change of the objective for the step is $\Delta(F) := \frac{|F(w_1)-F(w_0)|}{F(w_0)} \approx \eta \frac{\|\nabla F(w_0)\|^2}{F(w_0)}$, and the relative change in the differential of h measured in the operator norm is $\Delta(Dh) := \frac{\|Dh(w_1)-Dh(w_0)\|}{\|Dh(w_0)\|} \leq \eta \frac{\|\nabla F(w_0)\| \cdot \|D^2 h(w_0)\|}{\|Dh(w_0)\|}$. Lazy training occurs when the *differential of h does not change significantly whereas the loss does* i.e. $\Delta(F) \gg \Delta(Dh)$. Using the above estimates for $\Delta(F), \Delta(Dh)$, this is guaranteed whenever

$$\frac{\|\nabla F(w_0)\|}{F(w_0)} \gg \frac{\|D^2 h(w_0)\|}{\|Dh(w_0)\|}.$$

In the particular case where $R(y) = \frac{1}{2}\|y - y^\star\|^2$ for some $y^\star \in \mathcal{F}$ (i.e. the loss under consideration is the square loss), then we have the simpler criterion

$$\kappa_h(w_0) := \|h(w_0) - y^\star\| \frac{\|D^2 h(w_0)\|}{\|Dh(w_0)\|^2} \ll 1.$$

This comes from the approximation $\|\nabla F(w_0)\| = \|Dh(w_0)^T(h(w_0) - y^\star)\| \approx \|Dh(w_0)\| \cdot \|h(w_0) - y^\star\|$.

2.2 Scaling the Output

Of particular importance to this work is how scaling the output $h(w)$ affects lazy training. Chizat and colleagues note that for the square loss, scaling the model h by factor of $\alpha > 0$ results in

$$\kappa_{\alpha h}(w_0) = \frac{1}{\alpha}\|\alpha h(w_0) - y^\star\| \frac{\|D^2 h(w_0)\|}{\|Dh(w_0)\|^2}.$$

And so as long as $\|\alpha h(w_0) - y^\star\|$ is bounded, taking $\alpha \to \infty$ leads to the lazy training regime. The authors discuss numerous strategies by which to ensure $h(w_0) = 0$, and so $\|\alpha h(w_0) - y^\star\|$ is indeed bounded.

For D-homogeneous models, multiplying the initialization w_0 by a factor of λ is equivalent to scaling the output by λ^D. That is,

$$\kappa_h(\lambda w_0) = \frac{1}{\lambda^D}\|\lambda^D h(w_0) - y^\star\| \frac{\|D^2 h(w_0)\|}{\|Dh(w_0)\|^2}.$$

Importantly, neural networks with homogeneous activation functions and linear, *but not affine*, operators are important examples of homogeneous models.

2.3 Lazy Training and the Scaled Model

In the remainder of their work, Chizat et al. prove that for $\alpha > 0$ large, the training dynamics for the scaled objective

$$F_\alpha(w) := \frac{1}{\alpha^2} R(\alpha h(w))$$

are close to those for the linearized model

$$\overline{F}_\alpha(w) := \frac{1}{\alpha^2} R(\alpha \overline{h}(w)),$$

where $\overline{h}(w) := h(w_0) + Dh(w_0)(w - w_0)$ and $w_0 \in \mathbb{R}^p$ is a fixed initialization. That is, we prove the prior assertion that for scaling parameter α large, we indeed achieve lazy training as defined at the beginning of the paper.

The necessary assumptions on the model $h : \mathbb{R}^p \to \mathcal{F}$ and loss $R : \mathcal{F} \to \mathbb{R}_+$ are as follows: the parametric model h is differentiable with a locally Lipschitz differential Dh. Moreover, R is differentiable with a Lipschitz gradient.

For the primary results of the paper, the authors consider the gradient flow of the objective function F_α. For initialization w_0, the gradient flow is the path $(w_\alpha(t))_{t\geq 0}$ in the space of parameters $\mathbb{R}^p$ that satisfies $w_\alpha(0) = w_0$ and solves the ordinary differential equation

$$w'_\alpha(t) = -\nabla F_\alpha(w_\alpha(t)) = -\frac{1}{\alpha} Dh(w_\alpha(t))^T \nabla R(\alpha h(w_\alpha(t))),$$

where Dh^T is the adjoint of the differential Dh. The authors compare this dynamic to the gradient flow $(\overline{w}_\alpha(t))_{t\geq 0}$ of $\overline{F}_\alpha$ that satisfies $(\overline{w}_\alpha(0) = w_0$ and solves

$$\overline{w}'_\alpha(t) = -\nabla \overline{F}_\alpha(\overline{w}_\alpha(t)) = -\frac{1}{\alpha} Dh(w_0)^T \nabla R(\alpha \overline{h}(\overline{w}_\alpha(t))).$$

The first major result confirms that when $h(w_0) = 0$, then large α leads to lazy training. It is important to point out that here, we do not assume anything about the convexity of the loss R.

Theorem 2.2. *Assume that $h(w_0) = 0$. Given a fixed time horizon $T > 0$, it holds that $\sup_{t\in[0,T]} \|w_\alpha(t) - w_0\| = \mathcal{O}(1/\alpha)$,*

$$\sup_{t\in[0,T]} \|w_\alpha(t) - \overline{w}_\alpha(t)\| = \mathcal{O}(1/\alpha^2) \quad \textit{and} \quad \sup_{t\in[0,T]} \left\|\alpha h(w_\alpha(t)) - \alpha\overline{h}(\overline{w}_\alpha(t))\right\| = \mathcal{O}(1/\alpha^2).$$

Each of these three statements is of key importance. The first tells us that as α increases, $w_\alpha(t)$ is closer to initialization $w_\alpha(0)$ [in the ℓ_2 norm]. The second tells us that the gradient flow of F_α approaches the gradient flow of $\overline{F}_\alpha$ as α grows. The last tells us that the scaled model αh approaches the scaled linearized model $\alpha\overline{h}$ for α large. The constants in Theorem 2.2 depend *exponentially on the time horizon T*.

In the next portion of their paper, the author gives *uniform* bounds in time and convergence results under the assumption that the loss R is strongly convex. By this assumption, $\overline{F}_\alpha$ is strictly convex on the affine hyperspace $w_0 + \ker Dh(w_0)^\perp$ which contains the linearized gradient flow $(\overline{w}_\alpha(t))_{t\geq 0}$. Therefore, $(\overline{w}_\alpha(t))_{t\geq 0}$ converges linearly to the unique global minimizer of $\overline{F}_\alpha$ (since we assumed strong convexity of R). The authors first treat the overparameterized case, when $Dh(w_0)$ is surjective. As they discuss, rank $Dh(w_0)$ gives the degrees of freedom of the model around initialization w_0.

Theorem 2.4. *Consider the M-smooth and m-strongly convex loss R with minimizer $y^\star$ and condition number $\kappa := M/m$. Assume that σ_{min}, the smallest singular value of $Dh(w_0)^T$, is positive and that the initialization satisfies $\|h(w_0)\| \leq C_0 := \sigma_{min}^3/(32\kappa^{3/2}\|Dh(w_0)\| \, Lip(Dh))$, where $Lip(Dh)$ is the Lipschitz constant of Dh. If $\alpha > \|y^\star\|/C_0$, then for $t \geq 0$, it holds*

$$\|\alpha h(w_\alpha(t)) - y^\star\| \leq \sqrt{\kappa}\, \|\alpha h(w_0) - y^\star\| \exp(-m\sigma_{min}^2 t/4).$$

If moreover $h(w_0) = 0$, it holds as $\alpha \to \infty$, $\sup_{t\geq 0} \|w_\alpha(t) - w_0\| = \mathcal{O}(1/\alpha)$,

$$\sup_{t\geq 0} \left\|\alpha h(w_\alpha(t)) - \alpha\overline{h}(\overline{w}_\alpha(t))\right\| = \mathcal{O}(1/\alpha) \quad \textit{and} \quad \sup_{t\geq 0} \|w_\alpha(t) - \overline{w}_\alpha(t)\| = \mathcal{O}(\log\alpha/\alpha^2).$$

Notice that there is no dependence here on the time horizon T, as there was in Theorem 2.2. Instead, we get a bound that is uniform in $t \geq 0$.

Removing the overparameterized assumption, the authors prove linear convergence of the gradient flow for large values of α:

Theorem 2.5. *Assume that $\mathcal{F}$ is separable, R is strongly convex, $h(w_0) = 0$, and rank $Dh(w)$ is constant on a neighborhood of w_0. Then there exists $\alpha_0 > 0$ such that for all $\alpha > \alpha_0$, the gradient flow converges at a geometric rate (asymptotically independent of α) to a local minimum of F_α.*

And so we have that, for sufficiently large α, $\lim_{t\to\infty} w_\alpha(t)$ is a strict local minimizer. It is not in general, though, a global minimizer of F_α *because the image of $Dh(w_0)$ need not contain the global minimizer of R.*

2.4 Applications to Neural Networks

Of importance to our present research, Chizat and colleagues discuss how these theoretical results may be applied to deep leaning problems. In these problems, we typically have a function f that maps from $\mathbb{R}^p \times \mathbb{R}^d$ to $\mathbb{R}^k$, where $\mathbb{R}^d$ is the input space and $\mathbb{R}^k$ is the output space. To relate this f to the problem under consideration, let $\mathcal{F}$ be a Hilbert space of functions from $\mathbb{R}^d$ to $\mathbb{R}^k$. For example, we can have $\mathcal{F} = L^2(\rho_x, \mathbb{R}^k)$ where ρ_x is the distribution of input samples. So then $h : \mathbb{R}^p \to L^2(\rho_x, \mathbb{R}^k)$ maps a vector of parameters to a predictor. That is, $h : w \mapsto f(w, \cdot)$.

One particular application is the two-layer neural network

$$f_m(w, x) = \alpha(m) \sum_{j=1}^{m} b_j \cdot \sigma(a_j \cdot x),$$

where m is the size of the hidden layer, $\sigma : \mathbb{R} \to \mathbb{R}$ is an activation function, and $(\theta_j)_{j=1}^m$, $\theta_j = (a_j, b_j) \in \mathbb{R}^{d+1}$ are the parameters. Here $\alpha(m)$ is the scaling factor as previously discussed. Note that in order for h to be differentiable, we must have that σ is smooth.

3 Neural Tangent Kernel: Convergence and Generalization in Neural Networks, Jacot, Gabriel, and Hongler 2018

The paper by Jacot and colleagues studies the training of artificial neural networks in the infinite width limit. They consider fully-connected ANNs with layers numbered from 0 (input) to L (output), each containing $n_0, \ldots, n_L$ neurons, and with a Lipschitz, twice-differentiable nonlinearity function $\sigma : \mathbb{R} \to \mathbb{R}$. The authors remark that the smoothness assumption on σ, while useful for proving the desired results, do not seem to be necessary. Popular activation functions like the ReLU are clearly not twice-differentiable, and so it is reassuring that the result holds true outside of the differentiability assumptions. The paper focuses on what the authors call the ANN realization function $F^{(L)} : \mathbb{R}^p \to \mathcal{F}$, which maps parameters θ to functions f_θ in the space $\mathcal{F}$. As is true in a general fully-connected neural network, the parameters consist of the connection matrices $W^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell+1}}$ and bias vectors $b^{(\ell)} \in \mathbb{R}^{n_{\ell+1}}$ for $\ell = 0, \ldots, L-1$.

3.1 The Kernel Gradient

The training of an ANN involves optimizing f_θ in the function space $\mathcal{F}$ with respect to a functional cost $C : \mathcal{F} \to \mathbb{R}$. And even for a convex functional cost C, the composite cost $C \circ F^{(L)} : \mathbb{R}^p \to \mathbb{R}$ is in general highly non-convex.

The authors show that during training, the network function f_θ follows descent along the kernel gradient with respect to the Neural Tangent Kernel (NTK). That is, during training the network function f_θ evolves along the negative kernel gradient

$$\partial_t f_{\theta(t)} = -\nabla_{\Theta^{(L)}} C|_{f_{\theta(t)}}$$

with respect to the *neural tangent kernel* (NTK)

$$\Theta^{(L)}(\theta) = \sum_{p=1}^{P} \partial_{\theta_p} F^{(L)}(\theta) \otimes \partial_{\theta_p} F^{(L)}(\theta).$$

The realization function $F^{(L)}$ is not linear (in $\theta \in \mathbb{R}^p$). As a consequence, $\partial_{\theta_p} F^{(L)}(\theta)$, and thus the neural tangent kernel, depend on the parameters θ. The NTK is random at initialization (since the authors initialize the parameters as i.i.d. $\mathcal{N}(0,1)$) and varies during training.

For a more rigorous treatment of the kernel gradient, see section 3 of [5].

3.2 Key Results

Below, we summarize the principal results of the paper without focusing too much on the particular math.

The first key result tells us that as the width of the ANN tends to infinity, the NTK converges in probability to an explicit deterministic limit.

Theorem 1. *For a network of depth L at initialization, with a Lipschitz nonlinearity σ, and in the limit as the layers width $n_1, \ldots, n_{L-1} \to \infty$, the NTK $^{(L)}$ converges in probability to a deterministic limiting kernel*

$$\Theta^{(L)} \to \Theta_{\infty}^{(L)} \otimes Id_{n_L}.$$

A specific definition for this limiting kernel is given in the paper and is in terms of the Gaussian processes to which the output functions $f_{\theta,k}$ tend in the infinite width limit $n_1, \ldots, n_{L-1} \to \infty$.

The second main theorem discusses the behavior of the NTK during training. Particularly, in the infinite-width limit, the NTK remains asymptotically constant during training. Recall that this is exactly the “kernel regime” discussed in both [5] and [3] where the model's gradient flow dynamic approaches that of a linearized model.

Theorem 2. *Assume that σ is a Lipschitz, twice differentiable nonlinearity function, with bounded second derivative. For any T such that the integral $\int_0^T \|d_t\|_{p^{in}}\, dt$ stays stochastically bounded, as $n_1, \ldots, n_{L-1} \to \infty$ we have, uniformly for $t \in [0,T]$*

$$\Theta^{(L)}(t) \to \Theta_{\infty}^{(L)} \otimes Id_{n_L}.$$

As is discussed in section 3 of the paper, the convergence of the neural tangent kernel to a critical point of the cost function C *is guaranteed for positive definite kernels.* The authors note that the limiting NTK Θ_∞ is positive definite if the span of the derivatives $\partial_{\theta_p} F^{(L)},\ p = 1, \ldots, P$ becomes dense in $\mathcal{F}$ with respect to the p^{in} norm as the width grows to infinity.

TODO: ask Professor Zhou about convergence along the principal components of the input data with respect to the NTK (see section 5 of the paper)

4 Implicit Regularization in Matrix Factorization, Gunasekar et al. 2017

4.1 Matrix Factorization Problem

Consider the matrix factorization problem

$$\min_{X\succeq 0} F(X) = \|\mathcal{A}(X) - y\|_2^2 ,$$

where $\mathcal{A} : \mathbb{R}^{n\times n} \to \mathbb{R}^m$ is a linear operator specified by $\mathcal{A}(X)_i = \langle A_i, X\rangle$, $A_i \in \mathbb{R}^{n\times n}$ and $y \in \mathbb{R}^m$. Here $\langle A_i, X\rangle = \sum_{k=1}^{n}\sum_{j=1}^{n} \overline{(A_i)}_{kj} X_{kj}$ is the Frobenius inner product. Moreover, the authors restrict their attention to symmetric, positive semidefinite X and symmetric, linearly independent A_i.

Instead of working with $X \in \mathbb{R}^{n\times n}$, however, the authors factor $X = UU^T$, where $U \in \mathbb{R}^{n\times d}$:

$$\min_{X\succeq 0} f(X) = \left\|\mathcal{A}(UU^T) - y\right\|_2^2 .$$

Notice that by setting $d < n$, we are enforcing that matrix X must be low-rank; for $d = n$, we reproduce the original problem.

4.2 Underdetermined Problems and Gradient Descent

Gunasekar and colleagues are particularly interested in the case of $m \ll n^2$ (the number of measurement matrices A_i is much smaller than the number of entries in X). Under this regime, the problem is undetermined and there are many global minima satisfying $\mathcal{A}(X) = y$. Indeed, for sufficiently large d, the authors achieve zero training error in the underdetermined problem. What they are surprised by is that for $d > m/n$ with initialization U_0 close to zero and small stepsize, they still achieve small relative error. For $d < m/n$, the low-rank structure of U can gaurantee generalization and reconstruction (see references in paper).

By observing the nuclear norms of the solutions for the simulations, the authors hypothesize that the gradient descent solution is also the minimum nuclear norm solution when U is full dimensional, the stepsize is sufficiently small, and the initialization approaches zero. That is, they suggest that under suitable conditions, the gradient descent solution satisfies

$$\operatorname*{argmin}_{X\succeq 0} \|X\|_* \quad \text{s.t. } \mathcal{A}(X) = y.$$

[1] subsequently shows that this is not the case.

4.3 Key Findings and Results

Instead of proving the lofty previous result, the authors settle on the following conjecture:

Conjecture. *For any full rank X_{init}, if $\widehat{X} = \lim_{a\to 0} X_\infty(\alpha X_{init})$ exists and is a global optima for the matrix factorization problem with $\mathcal{A}(\widehat{X}) = y$, then $\widehat{X} \in argmin_{X\succeq 0} \|X\|_*$ s.t. $\mathcal{A}(X) = y$.*

Here, $\lim_{\alpha\to 0} X_\infty(X_{\text{init}})$ is the limit point of the gradient flow dynamics

$$\dot{X}_t = \mathcal{A}^*(r_t)X_t - X_t\mathcal{A}^*(r_t)$$

initialized at $X_0 = X_{\text{init}}$. These dynamics are equivalent to the gradient flow on the matrix $U \in \mathbb{R}^{n\times d}$ in the matrix factorization problem:

$$\dot{U}_t = \mathcal{A}^*(\mathcal{A}(U_t U_t^T) - y)U_t.$$

Biting off a smaller portion of this conjecture, the authors prove the result for the case of $A_1, \ldots, A_m$ commutative matrices:

Theorem 1. *In the case where the matrices $\{A_i\}_{i=1}^m$ commute, if $\widehat{X} = \lim_{\alpha\to 0} X_\infty(\alpha X_{init})$ exists and is a global optima for the matrix factorization problem with $\mathcal{A}(\widehat{X}) = y$, then $\widehat{X} \in argmin_{X\succeq 0} \|X\|_*$ s.t. $\mathcal{A}(X) = y$.*

One should notice that linear regression falls under this case: when the initial matrix X_{init} and "measurements" A_i are diagonal matrices, then we have an equivalent parameterization of the vector least-squares problem in terms of *squares of the least-squares coefficients.* [1] Notice that we cannot work with the least-squares coefficients themselves since we require that X is positive semidefinite. Recall that the least-squares problem is

$$\operatorname*{argmin}_{\beta\in\mathbb{R}^n} \|y - W\beta\|_2^2 ,$$

where $W \in \mathbb{R}^{m\times n}$ is the matrix whose rows are the observations, $\beta \in \mathbb{R}^n$ is the least-squares coefficient vector, and $y \in \mathbb{R}^m$ is the response vector.

As a result, we have the following Corollary to Theorem 1:

Corollary 2. *Let $x_\infty(x_{init})$ be the limit point of gradient flow on $\min_{u\in\mathbb{R}^n} \|Ax(u) - y\|_2^2$ with initialization x_{init}, where $x(u)_i = u_i^2$, $A \in \mathbb{R}^{m\times n}$, and $y \in \mathbb{R}^n$. If $\lim_{\alpha\to 0} x_\infty(\alpha\vec{1})$ exists and $A\widehat{x} = y$, then $\widehat{x} \in argmin_{x\in\mathbb{R}_+^m} \|x\|_1$ s.t. $Ax = y$.*

Clearly, this result is crucial to the least-squares problem (as a "diagonal" neural network) considered in [5].

5 Implicit Regularization in Deep Matrix Factorization, Arora et al. 2019

5.1 Matrix Completion Problem

Given a matrix $W^* \in \mathbb{R}^{m\times n}$ and a randomly chosen subset of observed entries, recover the unseen entries of W^*. View each entry of W^* as a data point: observed entries are the training set, unobserved entries are the test set. This is an undetermined system with multiple solutions. Previous work has shown that if we assume that W^* is low rank, certain technical assumptions are met, and sufficiently many entries are observed, then we can achieve approximate or exact recovery of W^*. [4] conjectured that under suitable conditions, the gradient descent solution for the matrix factorization problem is (always) the minimum nuclear norm solution.

5.2 Matrix Completion Problems as Neural Networks

One can use shallow neural networks to solve matrix completion problems. Consider parameterizing the solution W as the product of two matrices $W = W_2W_1$ and optimizing the non-convex objective for fitting

[1] Note that, in this case, $A_1, \ldots, A_m$ are the observations and the diagonal entries of X in terms of the *squares of* least-squares coefficients.

the observed entries. This optimization problem can be viewed as a depth-2 linear neural network (that is, the activation functions are linear and there are no added bias terms).

This two-layer problem can be naturally extended to the **deep matrix factorization problem** considered in the paper: for $W \in \mathbb{R}^{d \times d'}$, $d_1, \ldots, d_{N-1} \in \mathbb{N}$, we parameterize

$$W = W_N W_{N-1} \cdots W_1,$$

where $W_j \in \mathbb{R}^{d_j \times d_{j-1}}$ with $d_n = d, d_0 = d'$. N is the *depth* of the matrix factorization, matrices $W_1, \ldots, W_N$ are the *factors*.

5.3 Key Findings and Results

The present paper considers whether the implicit regularization induced by a deep matrix factorization is stronger than the shallow factorization. Recall that [4] only considers the shallow factorization $X = UU^T$. The authors then study the scenario when the number of observed entries m is too small for exact recovery. They show that, in this regime, matrix factorizations outperform minimum nuclear norm solutions. Further, there is a tendency towards low rank solutions W, which intensifies with the depth N of the network.

The first original result of the paper extends Theorem 1 from [4] for depth-N matrix completion problems:

Theorem 2. *Suppose $N \geq 3$ and the matrices $A_1, \ldots, A_m$ commute. Then if $\bar{W}_{deep} := \lim_{\alpha \to 0} W_{deep,\infty}(\alpha)$, where $W_{deep,\infty}(\alpha) = W_N(t) W_{N-1}(t) \cdots W_1(t)$ with $W_j(0) = \alpha I$, $\dot{W}_j(t) = -\frac{\partial \phi}{\partial W_j}(W_1(t), \ldots, W_N(t))$ $t \geq 0$, exists and is a global optimum for*

$$\min_{W \in \mathcal{S}_+^d} l(W) := \frac{1}{2} \sum_{i=1}^{m} (y_i - \langle A_i, W \rangle)^2,$$

where $A_1 \ldots, A_m \in \mathbb{R}^{d \times d}$ are symmetric and linearly independent (and $\mathcal{S}_+^d$ is the set of symmetric, positive semidefinite matrices) with $l(\bar{W}_{deep}) = 0$, then it holds that $\bar{W}_{deep} \in argmin_{W \in \mathcal{S}_+^d, l(W)=0} \ \|W\|_*$. Here, $\|\cdot\|_*$ denotes the nuclear norm.

The authors then consider whether the gradient descent solution to the matrix factorization problem always tends to the minimum nuclear norm solution. They show that when the number of observed entries is sufficiently large relative to the rank of the matrix, factorizations of all depth admit solutions that tend to the minimum nuclear norm. However, when fewer entries are observed, neither shallow nor deep factorizations minimize the nuclear norm. This contradicts the previously mentioned conjecture made by [4].

Arora and colleagues then prove the following result:

Theorem 3. *The signed singular values of the product matrix W(t) evolve by*

$$\dot{\sigma}_r(t) = -N(\sigma_r^2(t))^{1-1/N} \cdot \langle \nabla l(W(t)), u_r(t), v_r^T(t) \rangle, \qquad r = 1, \ldots, \min\{d, d'\}.$$

If the matrix factorization is non-degenerate i.e. has depth $N \geq 2$, the singular values need not be signed (we can assume $\sigma_r(t) \geq 0$ for all t).

Theorem 3 establishes a tendency of gradient descent for matrix factorization problems to low rank solutions, which intensifies with the depth N of the network. This is because the dynamics promote solutions that have a few large singular values and many small ones; the gap grows more extreme the deeper the matrix factorization (see Figure 3 on p. 9).

6 On the Global Convergence of Gradient Descent for Over-parameterized Models Using Optimal Transport, Chizat and Bach 2018

6.1 Problem Statement

Chizat and Bach discuss the problem of finding an element in the Hilbert space $\mathcal{F}$ that minimizes a smooth, convex loss function $R : \mathcal{F} \to \mathbb{R}_+$ and that is a linear combination of a few elements from a large given parameterized set $\{\phi(\theta)\}_{\theta\in\Theta} \subset \mathcal{F}$. They state that a general formulation of the problem is to describe the linear combination through an *unknown signed measure* μ on the parameter space and solve for

$$J^* = \min_{\mu\in\mathcal{M}(\Theta)} J(\mu), \qquad J(\mu) := R\left(\int \phi d\mu\right) + G(\mu)$$

where $\mathcal{M}(\Theta)$ is the set of signed measures on the parameter space Θ and $G : \mathcal{M}(\Theta) \to \mathbb{R}$ is an optional convex regularizer. In the present paper, the authors focus on the infinite-dimensional case where the parameter space is a domain of $\mathbb{R}^d$ and $\theta \to \phi(\theta)$ is differentiable.

While this framework may appear highly theoretical, it includes training neural works with a single hidden layer. Particularly, suppose we wish to select a function belonging to a certain class that maps features in $\mathbb{R}^{d-1}$ to labels in $\mathbb{R}$. In this case, we would have Hilbert space $\mathcal{F} = L^2(\mathbb{R}^{d-1}, \rho_x)$, quadratic or logistic loss function R, and $\phi(\theta) : x \mapsto \sigma(\sum_{i=1}^{d-1} \theta_i x_i + \theta_d)$ with activation function $\theta : \mathbb{R} \to \mathbb{R}$

6.2 Particle Gradient Descent

The authors discuss particle gradient descent, a technique for finding approximate minimizers of the above problem. Particle gradient descent exploits the differentiability of ϕ and consists of discretizing the unknown measure μ as a mixture of m particles parameterized by their positions and weights. This finite-dimensional problem is

$$\min_{\boldsymbol{w}\in\mathbb{R}^m, \boldsymbol{\theta}\in\Theta^m} J_m(\boldsymbol{w}, \theta) \quad \text{where} \quad J_m(\boldsymbol{w}, \boldsymbol{\theta}) := J\left(\frac{1}{m}\sum_{i=1}^{m} w_i \delta_{\theta_i}\right)$$

and can be solved with classical gradient-based algorithms.

The contributions of the paper are as follows: Chizat and Bach first introduce a more general class of problems and study the particle gradient flow for m large (i.e. when there are many particles). They characterize the many particle limit as a *Wasserstein gradient flow*. Then, under certain assumptions on ϕ and the initialization, they prove that if the Wasserstein gradient flow converges, then the limit is a global minimizer of J. Under these same assumptions, if $(\boldsymbol{w}^{(m)}(t), \theta^{(m)}(t))_{t\geq 0}$ are gradient flows for J_m suitably initialized, then

$$\lim_{m,t\to\infty} J(\mu_{m,t}) = J^* \quad \text{where} \quad \mu_{m,t} = \frac{1}{m}\sum_{i=1}^{m} w_i^{(m)}(t)\delta_{\theta_i^{(m)}(t)}.$$

That is, in the infinite particle limit and as t in the gradient flow tends to infinity, then the solution to the particle problem is also a solution (i.e. global minimizer) to the infinite-dimensional problem we originally stated.

7 Universal Approximation Bounds for Superpositions of a Sigmoidal Function, Barron 1993

7.1 Main Results

Theorem 1. *For every function in $\Gamma_{B,C}$, every sigmoidal function ϕ, every probability measure μ, and every $n \geq 1$, there exists a linear combination of sigmoidal functions $f_n(x)$ of the form*

$$f_n(x) = \sum_{k=1}^{n} c_k \phi(a_k \cdot x + b_k) + c_0$$

such that

$$\int_B (f(x) - f_n(x))^2 \mu(dx) \leq \frac{(2C)^2}{n}.$$

The coefficients of the linear combination in f_n may be restricted to satisfy $\sum_{k=1}^{n} |c_k| \leq 2C$ and $c_0 = f(0)$.

Theorem 4. *Let $f(x), x \in H$ be a function on a Hilbert space H with $C_f = \int_H \omega| F(d\omega) < \infty$; then for every $r > 0$, every sigmoidal function ϕ on $\mathbb{R}$, every probability measure μ on H, and every $n \geq 1$, there is a linear combination of sigmoidal functions $f_n(x) = \sum_{k=1}^{n} c_k \phi(a_k \cdot x + b_k) + c_0$ such that*

$$\int_{B_r} (f(x) - f_n(x))^2 \mu(dx) \leq (2rC_f)^2/n,$$

where $B_r = \{x \in H : |x| \leq r\}$ is the Hilbert space ball of radius r.

8 Approximation and Estimation Bounds for Artificial Neural Networks, Barron 1993

8.1 Main Results

The paper begins with a summary of the problem presented in [2], but also makes an insightful remark: "In the case that $\hat{f}_n$ is a neural network function estimated from data, the norm $\|f - \hat{f}_n\|_{L_2(\mu,B)}$ measures the ability of the neural network to generalize to new data drawn with distribution μ. In contrast, the empirical risk $(1/N)\sum_{i=1}^{N}(f(X_i) - f_n(X_i))^2$ only measures the accuracy of the observed data points $X_i,\ i = 1, 2, \ldots, N$." Barron then remarks that the first step to obtain a bound on the statistical risk $\|f - \hat{f}_n\|$ is to bound the approximation error $\|f - f_n\|$ of the best neural network of size n. To do so, he uses a special case of Theorem 1 from [2]:

Theorem 1. *Given an arbitrary sigmoidal function ϕ, an arbitrary target function f with C_f finite, and a probability measure μ on a domain $[-1, 1]^d$, then for every $n \geq 1$, there exists an artificial neural network of the form*

$$f_n(x) = f_n(x, \theta) = \sum_{k=1}^{n} c_k \phi(a_k \cdot x + b_k) + c_0$$

such that

$$\|f - f_n\| \leq \frac{C_f}{\sqrt{n}}.$$

For functions f with $C_f \leq C$, the parameters may be restricted to satisfy $\sum_{k=1}^{n} |c_k| \leq C$, $|c_0 - f(0)| \leq C$, and $|b_k| \leq |a_k|_1$.

Theorem 2. *Let a neural network be estimated by least squares with a complexity penalty (see pp. 121-122), where the range of Y and each candidate function is restricted to a known interval of length b, then for any $\lambda > 5b^2/3$, for all $n \geq 1$, and all $N \geq 1$,*

$$\mathbb{E}\|f - \hat{f}_{n,N}\| \leq \gamma R_{n,N}(f) + \frac{2\gamma\lambda}{N}$$

and

$$\mathbb{E}\|f - \hat{f}_{N}\| \leq \gamma R_{N}(f) + \frac{2\gamma\lambda}{N}$$

where $\gamma = (3\lambda + b^2)/(3\lambda - 5b^2)$. Thus,

$$\mathbb{E}\|f - \hat{f}_{N}\| \leq \mathcal{O}(R_N(f)).$$

Here, $\hat{f}_{n,N}$ is the least-squares estimator with a complexity penalty, and $\hat{f}_N = \hat{f}_{\hat{n},N}$ is the minimum complexity estimator (we estimate both n and θ).

References

[1] S. Arora, N. Cohen, W. Hu, and Y. Luo, *Implicit regularization in deep matrix factorization*, Advances in Neural Information Processing Systems, 32 (2019), pp. 7413–7424.

[2] A. R. Barron, *Universal approximation bounds for superpositions of a sigmoidal function*, IEEE Transactions on Information theory, 39 (1993), pp. 930–945.

[3] L. Chizat, E. Oyallon, and F. Bach, *On lazy training in differentiable programming*, arXiv preprint arXiv:1812.07956, (2018).

[4] S. Gunasekar, B. Woodworth, S. Bhojanapalli, B. Neyshabur, and N. Srebro, *Implicit regularization in matrix factorization*, in 2018 Information Theory and Applications Workshop (ITA), IEEE, 2018, pp. 1–10.

[5] B. Woodworth, S. Gunasekar, J. D. Lee, E. Moroshko, P. Savarese, I. Golan, D. Soudry, and N. Srebro, *Kernel and rich regimes in overparametrized models*, in Conference on Learning Theory, PMLR, 2020, pp. 3635–3673.