Title
Equivalence of Kernel Methods and Linear Models in High Dimensions

Author
Sahraee Ardakan, Mojtaba

Publication Date
2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Equivalence of Kernel Methods and Linear Models in High Dimensions

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Statistics

by

Mojtaba Sahraee Ardakan

2022

ABSTRACT OF THE THESIS

Equivalence of Kernel Methods and Linear Models

in High Dimensions

by

Mojtaba Sahraee Ardakan

Master of Science in Statistics

University of California, Los Angeles, 2022

Alyson K. Fletcher, Chair

Empirical observation of high dimensional phenomena, such as the double descent behavior, has attracted a lot of interest in understanding classical techniques such as kernel methods, and their implications to explain generalization properties of neural networks that operate close to kernel regime. Many recent works analyze such models in a certain high-dimensional regime where the covariates are generated by independent sub-Gaussian random variables transformed through a covariance matrix and the number of samples and the number of covariates grow at a fixed ratio (i.e. proportional asymptotics). In this work we show that for a large class of kernels, including the neural tangent kernel of fully connected networks, kernel methods can only perform as well as linear models in this regime. More surprisingly, when the data is generated by a Gaussian process model where the relationship between input and the response could be very nonlinear, we show that linear models are in fact optimal, i.e. linear models achieve the minimum risk among all models, linear or nonlinear. These results suggest that more complex models for the data other than independent features are needed for high-dimensional analysis.

The thesis of Mojtaba Sahraee Ardakan is approved.

Arash A. Amini

Guido Montufar

Alyson K. Fletcher, Committee Chair

University of California, Los Angeles

2022

To Fateme,

my parents, and my brothers.

Contents

List of Figures

Acknowledgements

This dissertation would not have been possible without the help of my adviser, Prof. Alyson K. Fletcher and her constant support throughout my studies in the MS program in statistics and PhD program in electrical and computer engineering, as well the countless hours of discussion and helpful insights provided by Prof. Sundeep Rangan. I would like to thank Prof. Arash A. Amini who introduced me to the beautiful world of statistics and played an important role in my decision to pursue a master's degree in statistics concurrently with my PhD program. I would like to thank Prof. Guido Montufar for his helpful advice during my studies and my internships at Microsoft Research. I am thankful to Prof. P. Schniter who collaborated with us on many of our publications. My studies at University of California, Los Angeles were funded in part by NSF Grants 1738285 and 1738286, and ONR Grant N00014-15-1-2677.

Chapter 1

Introduction

Analysis of kernel methods have seen a resurgence after Jacot et al. showed in [33] an equivalence of wide fully connected neural networks[1], trained with gradient descent, with the so-called *neural tangent kernel* (NTK). Since then, such equivalence to kernel models has been established for many different architectures such as convolutional models and tensor programs which shows this equivalence in a systematic way for almost all the architectures that are used in practice [2, 3, 64, 65].

Informally, we can describe these equivalences by looking at the first order Taylor expansion of a neural network $f(x;\theta)$ where θ corresponds to all the parameters in the network and x is the input. Assume that the parameters are initialized at random (often independently and identically distributed with an appropriate distribution) to θ_0 and consider the first order Taylor expansion of $f(x;\theta)$ with respect to θ around θ_0

$$f(x;\theta) \approx f(x;\theta_0) + \langle \nabla_\theta f(x;\theta_0), \theta - \theta_0 \rangle.$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product in ℓ^2. It can be shown that in wide neural networks with certain random initialization of parameters, this linear approximation becomes

[1]Here wide corresponds to different notions depending on the architecture of the neural network, e.g. the number of hidden unit in the the fully connected layers, the number of convolutional channels, etc. going to infinity.

exact as the width of networks goes to infinity. This linear model in turn (modulo the initial function $f(x;\theta_0)$) is equivalent to a kernel model with feature map $x \mapsto \nabla_\theta f(x;\theta_0)$, and hence the reproducing kernel $K(x_i, x_j) = \langle \nabla_\theta f(x_i;\theta_0), \nabla_\theta f(x_j;\theta_0) \rangle$. This kernel is called the neural tangent kernel, and training neural networks with gradient descent is equivalent to training a kernel model using gradient descent in the feature space of the NTK described above. Equivalently, the neural network function would be the same (throughout the training) as a function in the reproducing kernel Hilbert space (RKHS) induced by the NTK kernel and learned by functional gradient descent. We briefly review the neural tangent kernel for fully connected networks in the next chapter in Section 2.3.

Contemporaneously, there has been growing interest in high-dimensional asymptotic analyses of machine learning methods in a regime where the number of input samples n and number of input features p grow proportionally as

$$p/n \to \beta,$$

for some $\beta > 0$ and the data follow some random distribution. This regime, which we call *proportional asymptotics*, often enables remarkably precise predictions on the behavior of complex algorithms (see, e.g. [37], and the references below). Classically, parametric methods were studied in the large sample limit where the number of samples tends to infinity, but the number of parameters in the model is fixed. Examples of results of this nature include consistency of estimators and asymptotic distribution of estimates such as asymptotics normality of maximum likelihood or more generally M-estimators. Analyzing algorithms and models in this asymptotic regime becomes a lot easier as many tools such as laws of large numbers and central limit theorem come to our help in this regime. Unfortunately, despite the relative ease of the analysis in this regime compared to the non-asymptotic regime, these asymptotic results might not be a good predictor of the behavior of algorithms in practice as in modern machine learning problems the number of parameters and the number of samples

are both large and often of the same order. For example, modern natural language processing models such as GPT-3 [12] and Megatron-Turing NLG [62] have hundreds of billions of parameters each and are trained on huge datasets such as The Pile [24] and data that is scraped from the web. Similarly, text to image models such as Imagen [58] and DALL.E 2 [54] also have billions of trainable parameters. In fact, this move towards the large models with billions of parameters is a common trend in many machine learning tasks and one of the major driving forces behind the improved performance of new models over the older and smaller models [13]. This trend of training ever larger models using very large datasets motivates us to study machine learning models and algorithms in the proportional asymptotics regime. By doing the analysis in this regime, one might hope that certain convergent behaviors might be seen that make the analysis much easier, and yet since the number of samples and parameters are of the same order, which is often the case in modern deep learning models, the results that we obtain in this regime are hopefully still predictive of the real world performance of such models.

Such high-dimensional analyses have also been instrumental in elucidating important behavior such as the *double descent* phenomenon formalized by [9]. A surprising empirical behaviour, the double descent phenomenon has been demonstrated to hold for a large class of models in high dimensions including kernel models and linear models by [31] and [6]. This has piqued the curiosity of the machine learning community regarding the asymptotic properties of Kernel methods and their explanatory power towards understanding the generalizability of neural networks.

In this work, we study kernel ridge estimators in proportional asymptotics. These estimators are learned via a regularized empirical risk minimization

$$\widehat{f}_{\text{ker}} = \underset{f \in \mathcal{H}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(y_i, f(x_i)\right) + \lambda \|f\|_{\mathcal{H}}^2,$$

where $\mathcal{H}$ is a reproducing kernel Hilbert space (RKHS) with reproducing kernel $K(\cdot, \cdot)$, $\mathcal{L}$ is

a loss function, $\{(x_i, y_i)\}_{i=1}^n$ are the training samples, $\|\cdot\|_{\mathcal{H}}$ denotes the Hilbert norm of the RKHS, and λ is the regularization parameter. We consider the training of such kernel models in an asymptotic random regime similar in form to several other high-dimensional analyses:

Proportional, quasi-uniform large scale limit: Consider a sequence of problems indexed by the number of data covariates p satisfying the following assumptions:

A1 (Quasi-uniform data) Training features are generated as $x_i = \Sigma_x^{1/2} z_i \in \mathbb{R}^p$ where $z_i \in \mathbb{R}^p$ has i.i.d. sub-Gaussian entries with $\mathbb{E}z_i = 0$, $\mathbb{E}|z_i|^2 = 1$. A test sample, $x_{\text{ts}} = \Sigma_x^{1/2} z_{\text{ts}}$, is generated similarly. The responses y_i have finite second moment, i.e. $\mathbb{E}[y_i^2] < \infty$ and the data (x_i, y_i) are i.i.d. Further, the covariance matrix Σ_x is positive definite with $\|\Sigma_x\|_2 = \mathcal{O}(1)$, and $\tau := \lim_{p\to\infty} \operatorname{tr}(\Sigma_x)/p < \infty$.

A2 (Proportional asymptotics) Number of samples n and number of input features p scale as $\lim_{p\to\infty} p/n = \beta$ for some constant $0 < \beta < \infty$.

A3 (Kernel) The kernel function is of the form

$$K(x_i, x_j) = g\left(\frac{\|x_i\|_2^2}{p}, \frac{\langle x_i, x_j\rangle}{p}, \frac{\|x_j\|_2^2}{p}\right) \tag{1.1}$$

where g is C^3 around (τ, τ, τ) and $(\tau, 0, \tau)$.

Under these assumptions the main result of this work can be summarized as:

Kernel ridge regression offers no gain over linear models.

The class of kernels in (1.1) is quite large and includes many of the commonly used kernels in practice. These include inner product kernels such as polynomial kernels and kernels that are a function of the Euclidean norm such as radial basis functions and Laplace kernels. Furthermore, the NTK of fully connected networks as well as residual networks with fully connected blocks also have this form. Our result does not disregard kernel methods (or neural networks) as a whole, but serves as a caution regarding the proportional quasi-uniform large scale limit model while examining the asymptotic properties of kernels. A result of this nature

regarding the high-dimensional degeneracy of two layer neural networks has been studied in [32]. Note that we use the term *quasi-uniform* to describe a data model that satisfies A1. This is a non-standard terminology and it should not be interpreted as a data that spans the whole space uniformly. Rather, we use it to describe data that spans the whole directions in space transformed via a covariance matrix.

1.1 Summary of the Results

To be precise, we show three surprising results concerning kernel regression in the proportional, quasi-uniform large scale limit:

1. First, we show kernel models only learn linear relations between the covariates x and the response y in this regime. Consequently, kernel models (including neural networks in the kernel regime) have no benefit over linear models in this regime.

2. Our second result considers the training dynamics of the kernel and linear models. We show that under gradient descent, in the high dimensional setting, dynamics of the kernel model and a linear model are equivalent throughout training.

3. Finally, we consider the case where the true data is generated from a kernel model with some unknown parameters. In this case, the relation between x and y can be highly nonlinear. An example of such a model is that y is generated from x via a neural network with random, unknown parameters. In this case, we show that in the high-dimensional limit, the linear networks provide the minimum generalization error. That is, again, nonlinear kernel methods provide no benefit and training a wide neural network would result in a linear model.

The main take-away of this work is that under certain data distribution assumptions that are widely used in theoretical papers, a large class of kernel methods, including fully connected neural networks (and residual architectures with fully connected blocks) in kernel regime,

can only learn linear functions. Therefore, in order to theoretically understand the benefits that they provide over linear models, more complex data distributions should be considered. Informally, if $x \in \mathbb{R}^p$ covers this space in every direction (not necessarily isotropically), and the number of samples grows only linearly in the dimension of this space, many kernels can only see linear relationships between the covariates and the response. In other words, we argue that if we seek high-dimensional models for analyzing performance of neural networks, other distributional assumptions will be needed.

The proofs of our results rely on a generalization of Theorem 2.1 and 2.2 of [21] which is presented in the Appendix in Theorem 4. This generalization might be of independent interest for other works.

1.2 Organization of this Work

In Chapter 2 we review some background material that are used throughout this work. We first briefly introduce reproducing kernel Hilbert spaces and the kernel ridge regression problem. We state different formulations of this problem in function space, in feature space, as well as the dual parameterization of this problem. Next, we review the Gaussian regression problem which we use to show that linear models are Bayes optimal in the proportional, quasi-uniform large scale limit. Finally, we summarize the neural tangent kernel results for fully connected network as well as closed form recursive formulae to evaluate the NTK of suitably normalized fully connected ReLU networks. These results are extensively used in our experiments.

In Chapter 3 the main theoretical results of this work are presented. All of our results are obtained in the proportional, quasi-uniform large scale limit. We first briefly review related literature. Next, we show that in this asymptotic regime, kernel models learn linear models. Then, we show that when kernel models that are trained by gradient descent in the feature space, the models are linear throughout the training. Finally, we show that when the data

has a Gaussian process prior, linear models are in fact Bayes optimal, i.e. no learning method can beat suitably regularized linear models. We conclude this chapter by providing a sketch of the proof of the main results. The details of the proofs are deferred to the Appendix in Chapter 5 for clarity of the text.

Lastly, in Chapter 4 we validate each of our theoretical results with a series of experiments. We also include and example in which the Assumptions A1-A3 are violated and hence kernel methods outperform linear models. We conclude this work with conclusions and future directions.

Chapter 2

Background on Kernel Methods and Neural Networks in Kernel Regime

In this chapter, we present a short overview of some of the concepts that are used frequently in this work. We begin by briefly introducing reproducing kernel Hilbert spaces and kernel ridge regression. Next, we review Gaussian process regression. Finally, we end this chapter by reviewing the neural tangent kernel for fully connected networks.

2.1 Kernel Regression

In kernel regression, the estimator $\widehat{y}(x)$ is a function that belongs to a reproducing kernel Hilbert space (RKHS). A kernel $K : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ that is an inner product in a possibly infinite dimensional space $\mathcal{H}$ called the *feature space*, i.e. $K(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}$ where $\phi : \mathbb{R}^p \to \mathcal{H}$ is called the feature map. With this feature map, the functions in the RKHS are of the form $f(x) = \langle \phi(x), \theta \rangle_{L^2}$ which is a nonlinear function in x but linear in the parameters θ. In this work, we consider kernels of the form in equation (1.1), which includes inner product kernels as well as shift-invariant kernels. Many commonly used kernels such as RBF kernels, polynomial kernels, as well as the neural tangent kernel are of this form.

In kernel methods, the estimator is often learned via a regularized ERM

$$\widehat{f}_{\text{ker}} = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(y_i, f(x_i)\right) + \lambda \|f\|_{\mathcal{H}}^2, \tag{2.1}$$

where $\mathcal{L}$ is a loss function and $\|f\|_{\mathcal{H}} := \sqrt{\langle f, f \rangle_{\mathcal{H}}}$ is the RKHS norm. By writing $f(x) = \langle \phi(x), \theta \rangle$ as a parametric function with parameters $\theta \in \mathcal{H}$, this optimization over the function space can be written as an optimization over the parameter space as

$$\begin{aligned} \widehat{f}_{\text{ker}}(x) &= \langle \phi(x), \widehat{\theta} \rangle \\ \widehat{\theta} &= \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(y_i, \langle \phi(x_i), \theta \rangle\right) + \lambda \|\theta\|_{L^2}^2. \end{aligned} \tag{2.2}$$

Note that this optimization is often very high-dimensional as the dimension of feature space could be very high or even infinite. By the representer theorem [61], the solution to the optimization problem in (2.1) has the form

$$\widehat{f}_{\text{ker}}(x) = \sum_{i=1}^{n} K(x, x_i)\alpha_i,$$

which is sometimes referred to as the dual parameterization since this form can also be obtained from the dual formulation of the optimization problem in (2.2). By the reproducing property of the kernel, it is easy to show that $\|\hat{f}_{\text{ker}}\|_{\mathcal{H}}^2 = \alpha^{\mathsf{T}} K(X_{\text{tr}}, X_{\text{tr}})\alpha$ where $\alpha = [\alpha_1, \ldots, \alpha_n]^{\mathsf{T}}$. The optimization problem in (2.1) can then be written in terms of α as

$$\widehat{\alpha} = \arg\min_{\alpha} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\big(y_i, K(X_{\text{tr}}, x_i)\alpha\big) + \lambda \alpha^{\mathsf{T}} K(X_{\text{tr}}, X_{\text{tr}})\alpha, \tag{2.3}$$

where $K(X_{\text{tr}}, x_i) = [K(x_1, x), \ldots, K(x_n, x)]$. Throughout this work, for two matrices of data points $X_1 \in \mathbb{R}^{n_1 \times p}$ and $X_2 \in \mathbb{R}^{n_2 \times p}$ we use the notation $K(X_1, X_2)$ to represent the $n_1 \times n_2$ matrix with $[K(X_1, X_2)]_{ij} = K(X_{1,i}, X_{2,j})$. Observe that this optimization problem only depends on the kernel evaluated over the data points, and hence the optimization problem in

(2.1) can be solved without ever working in the feature space $\mathcal{H}$. If we let X_{tr} to represent the data matrix with x_i as its ith row, and y_{tr} the vector of observations, then for the special case of square loss the optimization problem in (2.3) has the closed form solution $\widehat{\alpha} = (1/nK(X_{\text{tr}}, X_{\text{tr}}) + \lambda I)^{-1} y_{\text{tr}}/n$ which corresponds to the estimator

$$\widehat{f}_{\mathsf{krr}}(x) = \frac{K(x, X_{\text{tr}})}{\sqrt{n}} \left(\frac{1}{n} K(X_{\text{tr}}, X_{\text{tr}}) + \lambda I \right)^{-1} \frac{y_{\text{tr}}}{\sqrt{n}}. \tag{2.4}$$

Similarly, when $\mathcal{L}$ is the square loss, the optimization problem in (2.2) is a quadratic problem in the parameters θ and has the optimal solution

$$\widehat{\theta} = \left(\frac{1}{n} \phi(X_{\text{tr}})^{\mathsf{T}} \phi(X_{\text{tr}}) + \lambda I \right)^{-1} \phi(X_{\text{tr}})^{\mathsf{T}} \frac{y}{n},$$

where X_{tr} is the data matrix and $\phi(X_t r)$ is a matrix in which the ith row is $\phi(x_i)$. Therefore, this model has the form

$$\begin{aligned} \widehat{f}_{\mathsf{krr}}(x) &= \phi(x)\widehat{\theta} \\ &= \frac{\phi(x)\phi(X_{\text{tr}})^{\mathsf{T}}}{\sqrt{n}} \left(\frac{1}{n} \phi(X_{\text{tr}})\phi(X_{\text{tr}})^{\mathsf{T}} + \lambda I \right)^{-1} \frac{y_{\text{tr}}}{\sqrt{n}}, \end{aligned}$$

where we have used the so-called push-through identity

$$(\lambda I + UU^{\mathsf{T}})^{-1} U = U(\lambda I + U^{\mathsf{T}} U)^{-1}$$

which is commonly used in kernel methods. Note that (with some abuse of notation) the identity matrices on the left-hand side and the right-hand side have different dimensions. Also note that since $K(x, y) = \langle \phi(x), \phi(y) \rangle$ this is exactly the same model as the one in (2.4), which shows the equivalence of (2.1) and (2.2).

2.2 Gaussian Process Regression

A Gaussian process f is a stochastic process in which for every fixed set of points $\{x_i\}_{i=1}^n$, the joint distribution of $(f(x_1), \ldots, f(x_n))$ has multivariate Gaussian distribution. As in multivariate Gaussian distribution, the distribution of a Gaussian process is completely determined by its first and second order statistics, known as the mean function and covariance kernel respectively. If we denote the mean function by $\mu(\cdot)$ and the covariance kernel by $K(\cdot, \cdot)$, then for any finite set of points

$$\Big(f(x_1), f(x_2), \ldots, f(x_n)\Big) \sim \mathcal{N}(\mu, K),$$

where μ the vector of mean values $\mu_i = \mu(x_i)$ and K is the covariance matrix with $K_{ij} = K(x_i, x_j)$. Next, assume that a priori we set the mean function to be zero everywhere. Then, the problem of Gaussian process regression can be stated as follows: we are given training samples $\{(x_i, y_i)\}_{i=1}^n$

$$y_i = f(x_i) + \xi_i, \quad \xi_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2),$$

where f is a zero mean Gaussian process with covariance kernel K. Given a test point x_{ts}, we are interested in the posterior distribution of $y_{\text{ts}} := f(x_{\text{ts}}) + \xi_{\text{ts}}$ given the training samples. Defining X_{tr} and y_{tr} as in previous section we have

$$\begin{bmatrix} y_{\text{tr}} \\ y_{\text{ts}} \end{bmatrix} | X_{\text{tr}}, x_{\text{ts}} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K(X_{\text{tr}}, X_{\text{tr}}) + \sigma^2 I & K(X_{\text{tr}}, x_{\text{ts}}) \\ K(x_{\text{ts}}, X_{\text{tr}}) & K(x_{\text{ts}}, x_{\text{ts}}) + \sigma^2 \end{bmatrix} \right),$$

where $K(X_{\text{tr}}, X_{\text{tr}})$ is the kernel matrix evaluated at training points. Therefore, we have $y_{\text{ts}} | y_{\text{tr}}, X_{\text{tr}}, x_{\text{ts}} \sim \mathcal{N}(\widehat{y}_{\text{ts}}, \sigma^2_{\text{ts}})$ where

$$\begin{aligned} \widehat{y}_{\text{ts}} &= K(x_{\text{ts}}, X_{\text{tr}})(K(X_{\text{tr}}, X_{\text{tr}}) + \sigma^2 I)^{-1} y_{\text{tr}}, \\ \sigma^2_{\text{ts}} &= \sigma^2 + K(x_{\text{ts}}, x_{\text{ts}}) - K(x_{\text{ts}}, X_{\text{tr}})(K(X_{\text{tr}}, X_{\text{tr}}) + \sigma^2 I)^{-1} K(X_{\text{tr}}, x_{\text{ts}}). \end{aligned} \tag{2.5}$$

The minimum mean squared error (MMSE) estimator is defined as the estimator that minimizes the square risk

$$\widehat{f}_{\text{MMSE}} = \arg\min_{f \in \mathcal{F}} \mathbb{E}[(Y_{\text{ts}} - f(X_{\text{ts}}))^2 | X_{\text{tr}}, y_{\text{tr}}],$$

where $\mathcal{F}$ is the class of all measureable functions of X. For a given x_{ts}, we have $\widehat{f}_{\text{MMSE}}(x_{\text{ts}}) = \widehat{y}_{\text{ts}}$ where $\widehat{y}_{\text{ts}}$ minimizes the posterior risk

$$\mathcal{E}(x_{\text{ts}}) := \mathbb{E}[(\widehat{y}_{\text{ts}} - y_{\text{ts}})^2 | x_{\text{ts}}, X_{\text{tr}}, y_{\text{tr}}]$$

and the expectation is with respect to the randomness in f as well as $\{\xi_i\}$. The estimator that minimizes this risk is the mean of the posterior, i.e. $\widehat{y}_{\text{ts}}$ in (2.5) is the Bayes optimal estimator with respect to mean squared error and its mean squared error is $\mathcal{E}(x_{\text{ts}}) = \sigma_{\text{ts}}^2$. Note that while this estimator is linear in the training outputs, it is nonlinear in the input data.

In this work, the problem of Gaussian process regression arises for systems that are in the Gaussian kernel regime. More specifically, assume that we have training and test data $\{(x_i, y_i)\}_{i=1}^n$ and $(x_{\text{ts}}, y_{\text{ts}})$ that are generated by a parametric model $y = f(x, \theta) + \xi$ where $\xi \sim \mathcal{N}(0, \sigma^2)$. Furthermore, assume that conditioned on X_{tr} and x_{ts}

$$[f(x_{\text{ts}}, \theta), f(x_1, \theta), \ldots, f(x_n, \theta)]^{\mathsf{T}},$$

which is $n+1$-dimensional vector of the function values on the training and test inputs is jointly Gaussian and zero mean. Also, for x and x', in the training and test inputs define the kernel function by

$$K(x, x') := \mathbb{E}_\theta [f(x, \theta) f(x', \theta)].$$

Then the problem of estimating $\widehat{y}_{\text{ts}}$ can be considered as a Gaussian regression problem. An important instance of this kernel model is when $f(x, \theta)$ a wide neural network with parameters

θ drawn from random Gaussian distributions and a linear last layer. In this case, one can show that conditioned on the input, all the preactivation signals in the neural network, i.e. all the signals right before going through the nonlinearities, as well as the gradients with respect to the parameters are Gaussian processes as discussed below.

2.3 Neural Tangent Kernel

Consider a neural network function $f(x, \theta) = \tilde{\alpha}^{(L)}(x, \theta)$ defined recursively as

$$\begin{aligned} \alpha^{(0)}(x, \theta) &= x, \\ \tilde{\alpha}^{(\ell+1)}(x, \theta) &= \frac{1}{\sqrt{n_\ell}} W^{(\ell)} \alpha^{(\ell)}(x, \theta) + \vartheta b^{(\ell)}, \\ \alpha^{(\ell)}(x, \theta) &= \sigma(\tilde{\alpha}^{(\ell)}(x, \theta)), \end{aligned}$$

where σ is a elementwise nonlinearity, $W^{(\ell)} \in \mathbb{R}^{n_{\ell+1} \times n_\ell}$, and θ is the collection of all weights $W^{(\ell)}$ and biases $b^{(\ell)}$ which are all initialized with i.i.d. draws from the standard normal distribution. As noted in many works [17, 38, 46, 50], conditioned on the input signals, with a Lipschitz nonlinearity $\sigma(\cdot)$, the entries of the preactivations $\tilde{\alpha}^{(\ell)}$ converge in distribution to an i.i.d. Gaussian processes in the limit of $n_1, \ldots, n_{L-1}, n_L \to \infty$ with covariance $\Sigma^{(\ell)}$ defined recursively as

$$\begin{aligned} \Sigma^{(1)}(x, x') &= \frac{1}{n_0} x^\mathsf{T} x' + \vartheta^2 \\ \Sigma^{(\ell+1)}(x, x') &= \mathbb{E}_{(u,v) \sim \mathcal{N}(0, \Sigma^{(\ell)})} \sigma(u)\sigma(v) + \vartheta^2. \end{aligned} \tag{2.6}$$

Therefore, if the ground truth model that generates the data is a random deep network plus noise, the optimal estimator would be as in (2.5) with the covariance in (2.6) used as the kernel.

The main result of [33] considers the problem of fitting a neural network to a training

data using gradient descent. It is shown that in the limit of wide networks (i.e. $n_\ell \to \infty$ for all ℓ), training a neural network with gradient descent is equivalent to fitting a kernel regression with respect to a specific kernel called the neural tangent kernel (NTK).

When $f(x, \theta)$ is a neural network with a scalar output, the neural tangent kernel (NTK) is defined as

$$K(x, x'; \theta) = \langle \nabla_\theta f(x; \theta), \nabla_\theta f(x'; \theta) \rangle.$$

In the case of networks with multiple outputs, a multi-dimensional kernel is defined in a similar way. In the limit of wide fully connected neural networks, [33] show that this kernel converges in probability to a kernel that is fixed throughout the training

$$K(x, x'; \theta) \xrightarrow{\mathsf{p}} K(x, x'; \theta_0).$$

Therefore, the main result of [33] can be summarized as follows: training wide neural networks is equivalent to learning kernel models in the RKHS induced by the neural tangent kernel above.

Similar to (2.6), neural tangent kernel can be evaluated via a set of recursive equations the details of which can be found in [33]. Similar results for architectures other than fully connected networks, such as convolutional models, recurrent networks, as well as general framework to show that most networks used in practice go to a kernel regime in a cetain high-dimensional limit have since been proven [2, 3, 64, 65].

For a fully connected network with ReLU nonlinearities, the NTK has a closed recursive form given by [11]. Let $f(x; \theta) = \sqrt{\frac{2}{n_{L-1}}} \langle w_L, a^{(L-1)} \rangle$ with $a^{(1)} = \sigma(W_1 x)$ and

$$a^{(\ell)} = \sigma\left(\sqrt{\frac{2}{n_{\ell-1}}} W_\ell a^{(\ell-1)} \right), \quad \ell = 2, \ldots, L-1,$$

where $\sigma(\cdot)$ is the ReLU function, $W_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$, $w_L \in \mathbb{R}^{L-1}$ and all the parameters w_L and $W_\ell, \ell = 1, 2, \ldots, L-1$, are initialized with i.i.d. entries drawn from $\mathcal{N}(0, 1)$. Then the

Gaussian process covariance (i.e. the covariance of preactivations in the network)as well as the NTK, $K(u,v) := K_L(u,v)$ can be obtained recursively by

$$\Sigma_\ell(u,v) = \|u\|\|v\|\kappa_1\left(\frac{\Sigma_{\ell-1}(u,v)}{\|u\|\|v\|}\right) \tag{2.7}$$

$$K_\ell(u,v) = \Sigma_\ell(u,v) + K_{\ell-1}(u,v)\kappa_0\left(\frac{\Sigma_{\ell-1}(u,v)}{\|u\|\|v\|}\right) \tag{2.8}$$

for $\ell = 1,\ldots,L$ and $K_0(u,v) = \Sigma_0(u,v) = u^\mathsf{T}v$ where

$$\kappa_0(t) = 1/\pi(\pi - \arccos(t))$$
$$\kappa_1(t) = 1/\pi\left(t\left(\pi - \arccos(t)\right) + \sqrt{1-t^2}\right).$$

In most of our experiments in the next chapters, we validate our results by considering neural networks that operate in (or close to) the NTK regime. There, we use these recursive formulae to evaluate the kernel for deep ReLU networks as well as its derivatives.

Chapter 3

Equivalence of Kernel Methods and Linear Models in High Dimensions

In this chapter we review some prior work and present the main results of this work. All of these results hold in a certain high-dimensional regime which we call the proportional, quasi-uniform large scale limit. Please refer to Chapter 1 for the details of this regime. First, we show that in this high-dimensional regime, kernel models are equivalent to linear model, i.e. the output of the kernel model is equal in probability to output of a linear model learned from the data with specific regularization parameters. Second, we show that if gradient descent is used to train the kernel model in feature space as well as the equivalent linear model formulation, both of these models are the same throughout training. In other words, their training dynamics also matches. Finally, for the case where the data has a Gaussian process prior, we show that the linear models are in fact optimal. We validate all of these results in our experiments. We also show empirically that if the assumptions of the proportional, quasi-uniform large scale limit are violated, these results would no longer hold.

3.1 Prior work

High-dimensional analyses in the proportional asymptotics regime similar to assumptions A1 to A3 have been widely-used in statistical physics and random matrix-based analyses of inference algorithms [66]. The high-dimensional framework has yielded powerful results in a wide range of applications such as estimation error in linear inverse problems [7, 20, 31, 37, 55], convolutional inverse problems [59], dynamics of deep linear networks [60], matrix factorization [34], binary classification [36, 63], inverse problems with deep priors [23, 52, 53], generalization error in linear and generalized linear models [22, 26, 27, 44], random features [18], and for choosing the optimal objective function for regression [1, 8] to name a few. Our result that, under a similar set of assumptions, kernel regression degenerates to linear models is thus somewhat surprising.

That being said, the result is not entirely new. Several authors have suggested that high-dimensional data modeled with i.i.d. covariates are inadequate [30, 48]. The results in this work can thus be seen as attempting to describe the limitations precisely. Several other works have also shown linearity of certain non-linear models in high dimensions under either more restrictive data distribution assumptions, for very specific models, or more restrictive kernel classes [14, 28, 47].

In this regard, the work is closest to [32]. The work [32] proves that for a two-layer fully-connected neural network, the training dynamics are equivalent to a linear model in inputs. They provide asymptotic rates for convergence in the early stages of training ($t < O(p \log p)$). Our result, however, considers a much larger class of kernels and is not limited to the NTK. In addition, we consider the dynamics throughout the training including the limit.

The generalization of kernel ridgeless regression is also discussed in this setting in [40]. The connections to double descent with explicit regularization has been analyzed in [43]. The authors in [19], characterize the limiting predictive risk for ridge regression and regularized discriminant analysis. [16] provides the error rates for KRR in the noisy case, and the generalization error in learning with random features with kernel approximation has been

discussed in [42]. A comparison between neural networks and kernel methods for Gaussian mixture classification is also is provided in [56].

The kernel approximation of the over-parameterized neural networks does not limit their performance in practical applications. In fact, these networks have surprisingly shown to generalize well [10, 51, 67]. Of course, in the non-asymptotic regime, these models also have very large capacity [5]. While this high capacity leads to learning complex functions, it is not always the case for the trained networks, and large models might still advocate for learning simpler functions. Works such as [32, 35] show that this simplicity can come from the implicit regularization induced by the training algorithms such as gradient descent for early-time dynamics. In this work, however, we show that in the high dimensional limit, this simplicity can be a result of the uniformity of input distribution over the space. In fact, we show that in this regime, kernel methods are no better than linear models.

3.2 Kernel Methods Learn Linear Models

In this section we show the first result of this work: in the proportional, quasi-uniform high-dimensional regime, fitting kernel models is equivalent to fitting a regularized least squares model with appropriate regularization parameters. A short review of reproducing kernel Hilbert spaces (RKHS) and kernel regression was presented in Section 2.1.

Suppose we have n data points (x_i, y_i), $i = 1, \dots, n$ with $x_i \in \mathbb{R}^p$, and an RKHS $\mathcal{H}$ corresponding to the kernel $K(\cdot, \cdot)$. Let $\tau = \lim_{p\to\infty} \operatorname{tr}(\Sigma_x)/p$, $\psi \in \mathbb{R}^n$ be a vector with $\psi_i = \|x_i\|_2^2/p - \tau$, and $\overline{\psi} = 1/n \sum_i \psi_i$.

Consider two models fitted to this data:

1. Kernel ridge regression model $\widehat{f}_{\mathsf{krr}}$ which solves

$$\widehat{f}_{\mathsf{krr}} = \operatorname*{arg\,min}_{f\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda\|f\|_{\mathcal{H}}^2, \tag{3.1}$$

where $\|f\|_{\mathcal{H}} = \sqrt{\langle f, f\rangle_{\mathcal{H}}}$ is the Hilbert norm of the function.

2. Linear model $\widehat{f}_{\text{lin}}(x) = \gamma_1\langle\widehat{w}, x\rangle + \gamma_2\widehat{\theta}_2 + \frac{\gamma_3}{\sqrt{n}}\widehat{\theta}_3$ fitted by solving the ridge regression problem

$$\begin{aligned}(\widehat{w}, \widehat{\theta}_2, \widehat{\theta}_3, \widehat{\theta}_4) =& \arg\min_{w,\theta_2,\theta_3,\theta_4} J(w, \theta_2, \theta_3, \theta_4), \\ J(w, \theta_2, \theta_3, \theta_4) :=& \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \gamma_1\langle w, x_i\rangle - \gamma_2\theta_2 - \frac{\gamma_3}{\sqrt{n}}\theta_3 - \gamma_3\theta_4\phi_4(x_i)\right)^2 \\ &+ \lambda\|w\|_2^2 + \lambda(\theta_2^2 + \theta_3^2 + \theta_4^2),\end{aligned} \tag{3.2}$$

where $\phi_4(x) = \frac{\|x\|_2^2/p - \tau - \overline{\psi}}{\|\psi - \overline{\psi}1\|_2}$ and γ_is are constants that are defined in Theorem 1.

Our goal in this section is to show that in the proportional, quasi uniform large scale limit presented in Chapter 1 , the kernel model and the linear model are equivalent for specific values of the scaling parameters γ_1 and γ_2, and γ_3.

As we mentioned in Chapter 2.1, using the representer theorem [61], the optimal function in (3.1) also has the form

$$\widehat{f}_{\text{ker}}(x) = \sum_{i=1}^{n} K(x, x_i)\alpha_i,$$

where

$$\alpha = \left(\frac{1}{n}K(X_{\text{tr}}, X_{\text{tr}}) + \lambda I\right)^{-1}\frac{y}{n}$$

which gives us the following kernel model on test data x

$$\widehat{f}_{\mathsf{krr}}(x) = \frac{K(x, X_{\text{tr}})}{\sqrt{n}}\left(\frac{K(X_{\text{tr}}, X_{\text{tr}})}{n} + \lambda I\right)^{-1}\frac{y_{\text{tr}}}{\sqrt{n}}.$$

Similarly, the optimization in (3.2) is also a quadratic problem in w and θ_is which also has a closed form solution. Note that even though $\phi_4(x)$ is a nonlinear feature and is used in the learning problem, this feature is not used at the time of inference on the test data and hence the model that is learned is linear. Using this feature at the learning phase would affect the learned coefficients and hence if we completely ignore these features the models

that is learned would be different. Therefore, even though the coefficient of ϕ_4 is not used at inference time, we cannot simply ignore this feature.

To state the result we need to define the following constants related to the kernel and its associated function g from assumption A3

$$c_1 = g(\tau, 0, \tau) + \frac{\partial^2 g}{\partial z_2^2}(\tau, 0, \tau)\frac{\mathrm{tr}\Sigma_p^2}{2p^2}, \tag{3.3a}$$

$$c_2 = \frac{\partial g}{\partial z_2}(\tau, 0, \tau), \tag{3.3b}$$

$$c_3 = \frac{\partial g}{\partial z_1}(\tau, 0, \tau), \tag{3.3c}$$

where z_1 and z_2 denote the first and second argument of the kernel function g respectively.

Our first result shows that with an appropriate choice of γ_1, γ_2, and γ_3 the two models $\widehat{f}_{\mathsf{krr}}$ and $\widehat{f}_{\mathrm{lin}}$ are in fact equivalent.

Theorem 1. *Under Assumptions (A1-A3), if we use the same data to train* $\widehat{f}_{\mathsf{krr}}$ *in* (3.1) *and* $\widehat{f}_{\mathrm{lin}}$ *in* (3.2) *with*

$$\gamma_1 = \sqrt{c_1 + 2\overline{\psi}c_3}, \quad \gamma_2 = \sqrt{c_2}, \quad \gamma_3 = \left(2\sqrt{n}\|\psi - \overline{\psi}1\|_2 c_3\right)^{1/2},$$

where the constants c_1, c_2, *and* c_3 *are defined in equations* (3.3)*, then at a test sample,* x_{ts} *drawn from the same distribution as the training samples,*

$$\lim_{n,p\to\infty} |\widehat{f}_{\mathrm{lin}}(x_{\mathrm{ts}}) - \widehat{f}_{\mathsf{krr}}(x_{\mathrm{ts}})| \overset{\mathsf{p}}{=} 0.$$

Proof. See Appendix A. □

When the kernel considered is an inner product kernel, i.e. norms of the data points are not present in the kernel function, the results are simplified significantly. For these kernels, $c_3 = 0$ and hence both at the time of training and inference, the same linear model is used as the coefficients of ϕ_4 would be zero. This is stated in the next corollary.

Corollary 1. *Under the same assumptions as Theorem 1, if we further assume that the kernel is an inner product kernel, i.e.*

$$K(x_i, x_j) = g\left(\frac{\langle x_i, x_j\rangle}{p}\right)$$

then the kernel model in (3.1) *is equivalent to a linear model* $f_{\text{lin}}(x) = \gamma_1\langle\widehat{w}, x\rangle + \gamma_2\widehat{b}$ *where*

$$(\widehat{w}, \widehat{b}) = \arg\min_{w,b} \frac{1}{n}\sum_{i=1}^{n}(y_i - \gamma_1\langle w, x_i\rangle - \gamma_2 b)^2 + \lambda\|w\|_2^2 + \lambda b^2.$$

Therefore, in this case the kernel model is equivalent to a standard ridge regression problem except that usually the bias term b is not regularized whereas here we are also regularizing the bias term.

Remark 1. Note that the result in Theorem 1 is not uniform, i.e. it does not imply that the linear model and the kernel model are equal in probability for all the points in the domain of these functions in the proportional quasi-uniform regime, but rather over a random test point as given by assumption A1. However this suffices for understanding the generalization properties of these functions.

Remark 2. Since convergence in probability implies convergence in distribution, we also have that the generalization error of $\widehat{f}_{\mathsf{krr}}$ is the same as that of $\widehat{f}_{\text{lin}}$ for any bounded continuous metric.

Remark 3. Theorem 1 states a convergence in probability for a single test point. This holds for n_{ts} test samples so long as n_{ts} grows sublinearly in the number of training samples, i.e. $n_{\text{ts}} = n_{\text{tr}}^{\gamma}$, where $\gamma < 1$ and the outputs of kernel model and the linear model would be equal in probability over all these test samples. Refer to the proof of the theorem for more details.

Remark 4. The result in Theorem 1 is similar to Theorem 3.5 in [32] where the authors consider training the parameters of a two-layer wide neural network under similar data assumptions. As in our result, a similar feature that contains the norm of the inputs is also

also present in their work. However, our result holds for a much larger class of kernel models that includes the NTK of fully connected networks and not just two-layer neural networks.

3.3 Linear Training Dynamics of Kernel Models

Our next result shows that if a kernel ridge regression is solved using gradient descent, every intermediate estimator during training has an equivalent linear model.

Consider a kernel model that is parameterized as $\widehat{f}_{\mathsf{krr}}(x) = \langle \eta(x), \widehat{\theta}_{\mathsf{krr}} \rangle$ (where $\eta(x)$ is a feature map, e.g. $\eta(x) = K(x, \cdot)$) that is trained by regularized empirical risk minimization:

$$\widehat{\theta}_{\mathsf{krr}} = \arg\min_{\theta_{\mathsf{krr}}} \frac{1}{n} \sum_{i=1}^{n} (y_i - \langle \eta(x_i), \theta_{\mathsf{krr}} \rangle)^2 + \lambda \|\theta_{\mathsf{krr}}\|_{L^2}^2.$$

The gradient descent iterates for this problem are with $\theta_{\mathsf{krr}}^0 = 0$ are

$$\theta_{\mathsf{krr}}^{t+1} = \Big(I - \rho\big((\eta(X_{\mathrm{tr}})^{\mathsf{T}} \eta(X_{\mathrm{tr}})/n + \lambda I\big) \Big) \theta_{\mathsf{krr}}^t + \rho \eta(X_{\mathrm{tr}})^{\mathsf{T}} \frac{y_{\mathrm{tr}}}{n}.$$

Here, $\eta(X_{\mathrm{tr}})$ is a matrix with $\eta(x_i)$ as its ith row and η is the learning rate. Therefore, the kernel model at the tth iteration of the gradient descent has the form $\widehat{f}_{\mathsf{krr}}^t(x) = \langle \eta(x), \widehat{\theta}_{\mathsf{krr}}^t \rangle$. Similarly, consider a linear model of the form $\widehat{f}_{\mathrm{lin}}(x) = \gamma_1 \langle \widehat{w}, x \rangle + \gamma_2 \widehat{\theta}_2 + \frac{\gamma_3}{\sqrt{n}} \widehat{\theta}_3$ fitted by solving the ridge regression problem in (3.2) via gradient descent. If we define the featur matrix over the data as $\Phi(X_{\mathrm{tr}})$ where

$$[\Phi(X_{\mathrm{tr}})]_{i*} = \left[\gamma_1 x_i, \quad \gamma_2, \quad \gamma_3 \sqrt{\frac{1}{n}}, \quad \gamma_3 \frac{\|x_i\|_2^2/p - \tau - \overline{\psi}}{\|\psi - \overline{\psi} 1\|_2} \right],$$

then the gradient descent updates for this model with the same learning rate ρ is

$$\theta^{t+1} = \Big(I - \rho\big((\Phi(X_{\mathrm{tr}})^{\mathsf{T}} \Phi(X_{\mathrm{tr}})/n + \lambda I\big) \Big) \theta_{\mathsf{krr}}^t + \rho \Phi(X_{\mathrm{tr}})^{\mathsf{T}} \frac{y_{\mathrm{tr}}}{n},$$

where $\theta = [w^{\mathsf{T}}, \theta_2, \theta_3, \theta_4]^{\mathsf{T}}$ and the gradient descent is initialized at zero. Therefore, the linear

model learned at the tth iteration of gradient descent has the form $\widehat{f}^t_{\text{lin}}(x) = \gamma_1\langle\widehat{w}^t, x\rangle + \gamma_2\widehat{\theta}^t_2 + \frac{\gamma_3}{\sqrt{n}}\widehat{\theta}^t_3$. We have the following result.

Theorem 2. *Under Assumptions A1-A3 and with*

$$\gamma_1 = \sqrt{c_1 + 2\overline{\psi}c_3}, \quad \gamma_2 = \sqrt{c_2}, \quad \gamma_3 = \left(2\sqrt{n}\|\psi - \overline{\psi}1\|_2 c_3\right)^{1/2},$$

for any step $t \geq 0$ of gradient descent (initialized at zero) and any test sample drawn from the same distribution as the training data we have

$$\lim_{p\to\infty} |\widehat{f}^t_{\text{ker}}(x_{\text{ts}}) - \widehat{f}^t_{\text{lin}}(x_{\text{ts}})| \overset{\text{p}}{=} 0.$$

Proof. The proof can be found in Appendix D. □

As in Corollary 1, for the case of inner product kernels this result can be simplified.

Corollary 2. *Let $\widehat{f}^t_{\text{lin}}(x) = \gamma_1\langle\widehat{w}^t, x\rangle + \gamma_2\widehat{b}^t$ where $(\widehat{w}, \widehat{b})$ are the parameters at the tth iteration of gradient descent on*

$$\min_{w,b} \frac{1}{n}\sum_{i=1}^{n} (y_i - \gamma_1\langle w, x_i\rangle - \gamma_2 b)^2 + \lambda\|w\|_2^2 + \lambda b^2.$$

with learning rate ρ. Then under assumptions A1-A3 with the further assumption that the kernel is an inner product kernel, for any step $t \geq 0$ of gradient descent and any test sample drawn from the same distribution as the training data the kernel model we have

$$\lim_{p\to\infty} |\widehat{f}^t_{\text{ker}}(x_{\text{ts}}) - \widehat{f}^t_{\text{lin}}(x_{\text{ts}})| \overset{\text{p}}{=} 0.$$

Remark 5. Theorem 2 provides an insight into the training dynamics of kernel models in the proportional uniform regime. This could potentially have implications regarding the Kernel-SVM solution in this regime, following the work of [49].

3.4 Optimality of Linear Models

Our last result shows that in the proportional uniform large scale limit, if the true model has a Gaussian process prior with a kernel that satisfies assumption A3, then linear models are in fact optimal, even though the true underlying relationship between the covariates and the responses could be highly nonlinear. See Appendix 2.2 for a review of Gaussian process regression.

Assume that we are given n training samples (x_i, y_i)

$$y_i = f^*(x_i) + \xi_i, \quad \xi_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2), \tag{3.4}$$

and the function f^* is a zero mean Gaussian process with covariance kernel $K(\cdot, \cdot)$. An example occurs in the so-called student-teacher set-up of [4, 25] where the unknown function is of the form

$$f(x) = g(x, \theta), \tag{3.5}$$

and $g(x, \theta)$ is a neural network with unknown parameters θ. If the network has infinitely wide hidden layers and the unknown parameters θ are generated with randomly with i.i.d. Gaussian coefficients with the appropriate scaling, the unknown function $f(x)$ in (3.5) becomes asymptotically a Gaussian process [17, 38, 46, 50].

Now assume that we are given a test sample from the same model $(x_{\text{ts}}, y_{\text{ts}})$ and we are interested in estimating y_{ts}. It is well known (see Appendix 2.2) that the Bayes optimal estimator with respect to squared error in this case is

$$\widehat{f}_{\text{opt}}(x_{\text{ts}}) = K(x_{\text{ts}}, X_{\text{tr}})(K(X_{\text{tr}}, X_{\text{tr}}) + \sigma^2 I)^{-1} y_{\text{tr}}, \tag{3.6}$$

and its Bayes risk is

$$\mathcal{E}_{\mathsf{opt}}(x_{\text{ts}}) = \sigma^2 + K(x_{\text{ts}}, x_{\text{ts}}) - K(x_{\text{ts}}, X_{\text{tr}})(K(X_{\text{tr}}, X_{\text{tr}}) + \sigma^2 I)^{-1} K(X_{\text{tr}}, x_{\text{ts}}).$$

Next consider a linear model $\widehat{f}_{\text{lin}}(x) = \gamma_1 \langle \widehat{w}, x \rangle + \gamma_2 \widehat{\theta}_2 + \frac{\gamma_3}{\sqrt{n}} \widehat{\theta}_3$ fitted by solving the regularized least squares problem in (3.2). Define the square error risk of this model as

$$\mathcal{E}_{\text{lin}}(x_{\text{ts}}) = \mathbb{E}[(y_{\text{ts}} - \widehat{f}_{\text{lin}}(x_{\text{ts}}))^2 | x_{\text{ts}}, X_{\text{tr}}, y_{\text{tr}}],$$

where the expectation is with respect to the randomness in f as well as the noise ξ_{ts}.

Theorem 3. *Under assumptions A1-A3 (where K is now interpreted as the covariance kernel) and the Gaussian data model* (3.4) *if the linear model $\widehat{f}_{\text{lin}}$ in equation* (3.2) *is trained with regularization parameter $\lambda = \sigma^2/n$ and constants*

$$\gamma_1 = \sqrt{c_1 + 2\overline{\psi} c_3}, \quad \gamma_2 = \sqrt{c_2}, \quad \gamma_3 = \left(2\sqrt{n} \|\psi - \overline{\psi} 1\|_2 c_3\right)^{1/2}, \tag{3.7}$$

where c_1, c_2 and c_3 are defined in (3.3)*, then $\widehat{f}_{\text{lin}}$ achieves the Bayes optimal risk for any test sample drawn from the same distribution as training data, i.e.*

$$\lim_{n \to \infty} |\mathcal{E}_{\text{lin}}(x_{\text{ts}}) - \mathcal{E}_{\mathsf{opt}}(x_{\text{ts}})| \overset{\mathsf{p}}{=} 0.$$

Proof. The result of Theorem 1 shows that with the specified choice of regularization parameter and γ_is in (3.7), the linear model and the kernel model in (3.6) are equivalent in the asymptotic regime

$$\lim_{n \to \infty} \widehat{f}_{\text{lin}}(x_{\text{ts}}) \overset{\mathsf{p}}{=} \widehat{f}_{\mathsf{opt}}(x_{\text{ts}}).$$

The result then immediately follows as the kernel model is Bayes optimal for squared error. □

This result is rather surprising as it claims even though the relationship between the covariates and the response could be quite nonlinear, in the proportional, quasi-uniform large scale limit the no learning algorithm can beat suitably tuned linear models as the Bayes optimal model is itself linear. In other words, in this regime, all we can learn from the data

are linear relationships.

As was the case in Corollaries 1 and 2, when the covariance kernel is an inner product kernel, $c_3 = 0$ and the result of this theorem can be simplified similarly to equivalence between the optimal model and a simple linear model.

It is important to contrast this result with [29] and [4]. The works [4, 29] consider exactly the case where the true function is of the form (3.5) where $g(x, \theta)$ is a neural network with Gaussian i.i.d. parameters. However, in their analyses, the number of hidden units in both the true and trained network are *fixed* while the dimension of x and number of samples grow with proportional scaling. With a fixed number of hidden units, the true function is *not* a Gaussian process, and the model class is not a simple kernel estimator – hence, our results do not apply. Interestingly, in this case, the results of [4, 29] show that nonlinear models can significantly out-perform linear models. Hence, very wide neural networks can underperform networks with smaller numbers of hidden units. It is an open question as to which scaling of the number of hidden units, number of samples, and dimension yield degenerate results.

3.5 Sketch of Proofs

Here we provide the main ideas behind the proofs of our main theorems. The details of the proof of Theorem 1 can be found in Appendix C. Proof of Theorem 2 can be found in Appendix D.

3.5.1 Degeneracy of empirical kernel matrices

Our first result modifies Theorems 2.1 and 2.2 of [21] to kernels that are both functions of the inner product as well as the Euclidean norm of the inputs. This result is presented in Theorem 1 and may be of independent interest to the reader. Using this theorem, we can prove the next Proposition.

Proposition 1. *If K is a $n \times n$ kernel matrix with entries $K_{ij} = K(x_i, x_j)$ such that assumptions (A1-A3) hold, then*

$$\lim_{p\to\infty} \frac{1}{\sqrt{n}} \|K - M\|_2 \overset{\mathsf{p}}{=} 0,$$

where $M = c_1 1 1^\mathsf{T} + c_2 X X^\mathsf{T} + c_3(1\psi^\mathsf{T} + \psi 1^\mathsf{T})$ where c_1, c_2, c_3 are defined in equation (3.3), $X \in \mathbb{R}^{n\times p}$ is the design matrix with samples x_i as rows and $\psi \in \mathbb{R}^n$ with $\psi_i = \|x_i\|_2^2/p - \tau$.

Proof. See appendix A. □

In [21], a similar result is presented for kernels of the form $g(\langle x_i, x_j\rangle)$ or $g(\|x_i - x_j\|_2^2)$. Importantly, the NTK has a form that is neither $g(\langle x_i, x_j\rangle)$ or $g(\|x_i - x_j\|_2^2)$, but in fact of the form in equation (1.1), whereby Proposition 1 provides new insights into the behavior of empirical kernel matrices of the NTK for a large class of architectures.

3.5.2 Equivalence of Kernel and Linear Models

Proposition 1 is the main tool we use to show that kernel methods and linear methods are equivalent in the proportional, uniform large scale limit.

The model learned by the kernel ridge regression in equation (3.1) can be written as

$$\widehat{f}_{\mathsf{krr}}(x) = \frac{K(x, X_{\mathrm{tr}})}{\sqrt{n}} \left(\frac{K(X_{\mathrm{tr}}, X_{\mathrm{tr}})}{n} + \lambda I \right)^{-1} \frac{y_{\mathrm{tr}}}{\sqrt{n}}. \tag{3.8}$$

Using Proposition 1

We use Proposition 1 to show that there exists a feature map ϕ such that for the data $\overline{X} = [x_{\mathrm{ts}}^\mathsf{T}, X_{\mathrm{tr}}^\mathsf{T}]^\mathsf{T}$, we have

$$\lim_{p\to\infty} \frac{1}{\sqrt{n}} \|K(\overline{X}, \overline{X}) - \phi(\overline{X})\phi(\overline{X})^\mathsf{T})\|_2 \overset{\mathsf{p}}{=} 0.$$

Furthermore, the feature map ϕ is almost linear

$$\phi(x) = \left[\gamma_1 x, \quad \gamma_2, \quad \gamma_3\sqrt{\frac{1}{n}}, \quad \gamma_3\frac{\|x\|_2^2/p - \tau - \overline{\psi}}{\|\psi - \overline{\psi}1\|_2}\right].$$

Next we show that each of the kernel terms in (3.8) converge in probability to the same term with the kernel K replaced by a kernel that is induced by the feature map $\phi(\cdot)$. We should emphasize that ϕ is not a proper feature map in the strict sense as it has parameters in it that depend on all the data, but for the purpose of the proof, we can regard them as constants. See the proof of Theorem 1 for the details. Next, we use Lemma 3 to show that the models leanrned by the the kernel K and the kernel with feature map ϕ are equivalent. Finally, we prove that the model that is learned by this feature map is linear. This proves Theorem 1.

3.5.3 Equivalence Throughout Training

The proof of equivalence of the kernel model and linear model after t steps of gradient descent is very similar. The updates for parameters of the kernel model have linear dynamics. By unrolling the gradient update through time, we can write the parameters after t step as a summation over the past time steps. Using this, we can simplify the sums to write the output of the kernel model at time t over a test sample as

$$\widehat{f}^t_{\text{ker}}(x_{\text{ts}}) = \frac{K(x_{\text{ts}}, X_{\text{tr}})}{\sqrt{n}}\left((\frac{K(X_{\text{tr}}, X_{\text{tr}})}{n} + \lambda I\right)^{-1}\left(I - \left(I - \rho((\frac{K(X_{\text{tr}}, X_{\text{tr}})}{n} + \lambda I)\right)^t\right)\frac{y_{\text{tr}}}{\sqrt{n}},$$

where ρ is the step size of the gradient descent. Here, we could use the same argument as the proof of Theorem 1. We show that the kernel K can be evaluated with the feature map ϕ in the asymptotic limit. Hence, we can replace each term by the same term with kernel K replaced by the kernel induced by the feature map ϕ. Next, we use Lemma 3 to show that the two models evaluated at a text sample x_{ts} are equal in probability in the limit. We further show that the model learned by the feature map ϕ is linear throughout training. This completes the proof of Theorem 2. Refer to Appendix D for the details.

Chapter 4

Numerical Experiments and Conclusion

4.1 Linearity of Kernel Models for NTK

We demonstrate via numerical simulations the predictions made by our results in Theorems 1, 2, 3.

As shown in [39] and [41], wide fully connected neural networks can be approximated by their first order Taylor expansion throughout the training

$$f_{\text{lin}}(x) = f(x;\theta_0) + \langle \nabla_\theta f(x,\theta_0), \theta - \theta_0 \rangle,$$

and this approximation becomes exact in the limit that all the hidden dimensions of the neural network go to infinity. Therefore, training a network $f(x;\theta)$ by minimizing

$$\widehat{\theta} = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \big(y_i - \big(f(x;\theta) - f(x;\theta_0)\big)\big)^2 + \lambda \|\theta - \theta_0\|_2^2 \tag{4.1}$$

is equivalent (in the limit of wide network) to performing kernel ridge regression in an RKHS with feature map $x \mapsto \nabla_\theta f(x;\theta_0)$ and neural tangent kernel $K_L(x,x')$ as its kernel. See Section 2.3 for a brief review of the neural tangent kernel. Instead of removing the initial network, one can use a symmetric initialization scheme which makes the output of neural

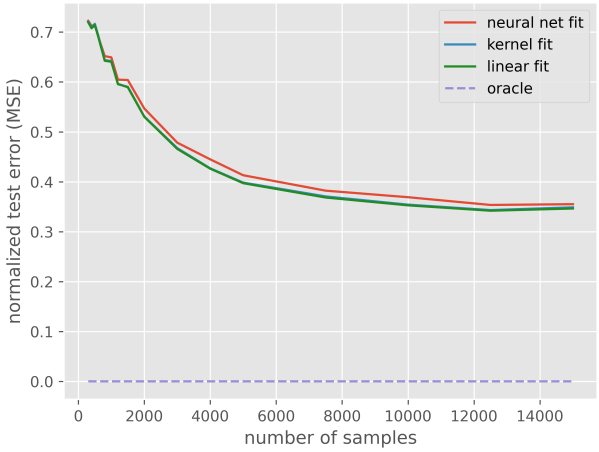

Figure 4.1: Comparison of test error with respect to the number otraining samples for three different models: (i) a neural network with a single hidden layer, (ii) NTK of a two layer fully-connected network, and (iii) the linear equivalent model prescribed by Theorem 1. The errors of the kernel model and the equivalent linear model match perfectly and neural network follows them very closely. The oracle model is the true model and represents the noise floor. We use $\lambda = 0.005$.

network zero at initialization without changing its NTK [15, 32, 68].

A key property of the NTK of fully connected neural networks is that it satisfies assumption A3 since it has the form in equation (1.1). Hence, if the input data x satisfies the requirements of this theorem, in the proportional asymptotics regime the NTK should behave like a linear kernel. The first and second order derivatives of the kernel function can be obtained by backpropagation through the recursive equations in (2.7) and (2.8).

Figure 4.1 illustrates a setting where kernel models and neural networks in the kernel regime perform no better than appropriately trained linear models. This verifies the main result of this work – Theorem 1.

We generate training data for $i = 1, 2, \ldots, n$ as

$$y_i = f^*(x_i) + \xi_i, \quad x_i \sim \mathcal{N}(0, I_{p\times p}), \quad \xi_i \sim \mathcal{N}(0, \sigma^2),$$

where $p = 1500$ and $\sigma^2 = 0.1$ and f^* is a fully-connected ReLU network with two hidden

layers with 100 hidden units each.

We train 3 models:

(i) A fully connected ReLU neural network with a single layer of 20,000 hidden units to fit this data using stochastic gradient descent (SGD) with momentum parameter 0.9. The initial network is remove from the output as in (4.1).

(ii) A kernel model as in equation (3.1) corresponding to the NTK of the model in (i) above. The kernel is evaluated using the recursive formulae given in (2.7) and (2.8).

(iii) A linear model as in equation (3.2) trained using the parameters prescribed by Theorem 1.

We compare the test error for these models, measured as $1 - R^2$ over $n_{\text{ts}} = 200$ test samples:

$$\mathcal{E}_{\text{ts}} = \frac{\sum_{i=1}^{n_{\text{ts}}} (y_{\text{ts},i} - \widehat{y}_{\text{ts},i})^2}{\sum_{i=1}^{n_{\text{ts}}} (y_{\text{ts},i})^2}, \tag{4.2}$$

We compare the test error for different number of training samples n averaged over 3 runs.

We can see that the test error of the NTK model and the equivalent linear model almost match perfectly over the whole range of number of training sample so muc h as the two curves are almost indistinguishable. The test error of the neural network model follows them very closely, matching them very well for smaller number of training samples.

There are two main sources of mismatch between the neural network model and the NTK model: first the width of the network while large (20,000) it is still finite, and secondly the training of the neural network model is stopped after 150 epochs, i.e. the neural network trained differs from the optimal neural network. Finally, the oracle model's performance is the noise floor.

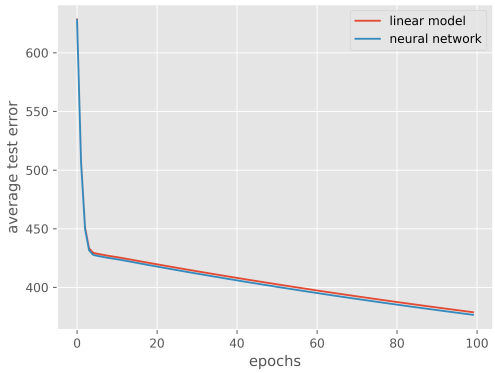

Figure 4.2: Equivalence of test error of linear model and the neural network vs. epochs of gradient descent.

4.2 Equivalence of Kernel and Linear Models Throughout Training

Next, we verify Theorem 2 by showing that the test error of the linear model and neural network match for all the steps of gradient descent. The setting is the same as in Section 4.1. We generate data using a random neural network with two hidden layers of 100 units each and train a neural network with a single hidden layer of 10,000 units as well as the linear model using gradient descent. We plot the the error of each of the models over the test data throughout the training. We train each model for 100 epochs. Figure 4.2 shows that the two models have approximately the same test error over the course of training.

4.3 Optimality of Linear Models

A polynomial kernels of degree d has the following form

$$K(x, x') = (\langle x, x' \rangle / p + c)^d,$$

where $x, x' \in \mathbb{R}^p$ and $c \geq 0$ is a constant that adjusts the influence of higher degree terms and lower degree terms. In this examples, we samples test and train samples from the following

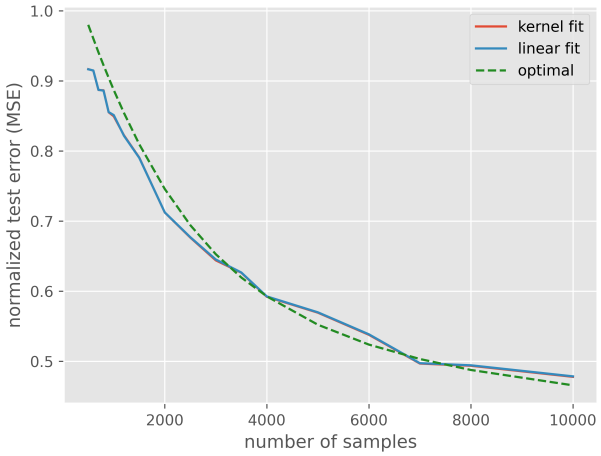

Figure 4.3: Normalized errors vs. number of training samples for a kernel model and the equivalent linear model for a data generated from a Gaussian process. The curves for the kernel and linear fit match almost perfectly. The dashed line corresponds to the theoretical optimal error given in equation (10).

model

$$x \sim \mathcal{N}(0, I_{p\times p}), \quad y = f(x) + \xi, \quad \xi \sim \mathcal{N}(0, \sigma^2),$$

where f is a Gaussian process with covariance kernel being a polynomial kernel. We use $c = 0.1, d = 2$ for the polynomial kernel and set $\sigma^2 = 0.1$, $p = 2,000$. We generate n_{tr} samples and train the kernel model and the equivalent linear model and estimate the normalized mean squared error of the estimator by averaging the normalized error over $n_{\text{ts}} = 500$ test samples. We use $\lambda = \sigma^2 = 0.1$ as the regularization parameter which makes the kernel estimator Bayes optimal (with respect to squared error). The results are averaged over 5 runs. The results are shown in Figure 4.3 where normalized errors (defined in equation (4.2)) are plotted against the number of training samples. The dashed line corresponds to optimal error curve obtained from Equation (3.6). The generalization errors for the linear model and the kernel model match almost perfectly which confirms Theorem 1 and as Theorem 3 proves both of the curves are very close to the optimal error curve. This figure verifies that the optimal estimator is indeed linear.

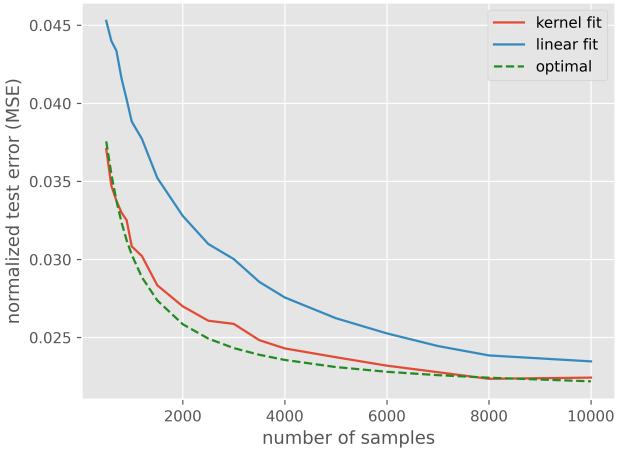

Figure 4.4: normalized error of kernel model, linear model and the Bayes optimal error with respect to number of training samples. When assumptions A1-A3 are not satisfied, the kernel model and linear model are not equivalent.

4.4 Counterexample: Beyond the Proportional Uniform Regime

Our results should not be misconstrued as ineffectiveness of kernel methods or neural networks. The equivalence of kernel models and linear models holds in the proportional, quasi-uniform data regime. However kernel models and neural networks outperform linear models when we deviate from this regime, as demonstrated in Figure 4.4.

This observation is closer to real-world experiences of the machine learning community, which perhaps suggests that the assumptions A1-A3 are unrealistic for understanding high dimensional phenomena relating large datasets and high dimensional models.

We consider a Gaussian process regression problem as in Section 4.3, but the input variables x are generated from a mixture of two zero mean Gaussians with low-rank covariances, which clearly violates assumption A1. The probability of each mixture component is set to $1/2$. We use $p = 2000$ and set rank of covariance of each component to $r = 200$. The covariance of

each component $c = 1, 2$ is generated as

$$\Sigma_c = S_c S_c^\mathsf{T}, \quad S_c \in \mathbb{R}^{p \times r}, [S_c]_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1/\sqrt{p}).$$

Under this model, the resulting covariance matrix of the data would be

$$\Sigma_x = \tfrac{1}{2}\Sigma_1 + \tfrac{1}{2}\Sigma_2,$$

which would have rank $2r$ almost surely. In other words, the data only spans a subspace of dimension 400 of the 2000-dimensional space.

Figure 4.4 shows that the kernel model which is the optimal estimator has a generalization error very close to the expected optimal error over the whole range of number of training samples, whereas the linear model performs worse. In this example, the linear approximation M of the true kernel matrix $K(X_{\text{tr}}, X_{\text{tr}})$ is inaccurate when we deviate from the proportional quasi-uniform data regime and cannot be used to consistently approximate the kernel model.

4.5 Conclusions

This work, of course, *does not* contest the power of neural networks or kernel models relative to linear models. In a tremendous range of practical applications, nonlinear models outperform linear models. The results should interpreted as a limitations of Assumptions A1-A3 as a model for high-dimensional data. While this proportional high-dimensional regime has been incredibly successful in explaining complex behavior of many other ML estimators, it provides degenerate results for kernel models and neural networks that operate in the kernel regime.

As mentioned above, the intuition is that when the data samples are generated as $x = \Sigma_x^{1/2} z$ where z has i.i.d. components and Σ_x is positive definite, so long as the number of samples n only scales linearly with p, it is impossible to learn models more complex than linear models.

This limitation suggests that more complex models for the generated data will be needed

if the high-dimensional asymptotics of kernel methods are to be understood.

4.6 Future Work

The results of this work only apply to a class of kernels that are rotationally invariant. In other words, the output of the kernel function does not change if the inputs are rotated via an orthogonal matrix. As mentioned earlier, this class is quite large and includes many of the widely used kernels such as kernels that are functions of the Euclidean distance or the inner product. However, the neural tangent kernel of many architectures is not a function of the inner product or the Euclidean distance. A very common example is the NTK of convolutional architectures. There are some works that empirically show that convolutional models learn linear models in certain high-dimensional problems. In the future, we plan to extend the results of this work to more general kernels that would also include the NTK of convolutional networks.

Chapter 5

Appendices

A Spectrum of Random Kernel Matrices

In this section we modify the results of [21]. In [21], kernels of the form $K(x_i, x_j) = g(\langle x_i, x_j \rangle / p)$ are considered whereas here we consider a more general form where $K(x_i, x_j) = g(\|x_i\|_2^2/p, \langle x_i, x_j \rangle / p, \|x_j\|_2^2/p)$. Define $\tau = \lim_{p\to\infty} \mathrm{tr}\Sigma_p/p$. Similar to [21] we assume that

- $n/p \to \gamma \in (0, \infty)$ as $p \to \infty$.
- $x_i = \Sigma_x^{1/2} y_i$ where $y_i \in \mathbb{R}^p$ has i.i.d. sub-Gaussian entries with $\mathbb{E}y_{ik} = 0, \mathbb{E}y_{ik}^2 = 1$.
- Σ_x is positive definite with bounded operator norm.
- g is a C^3 function in a neighborhood of $(\tau, 0, \tau)$ and a neighborhood of (τ, τ, τ).
- g is a valid kernel function and hence is symmetric in its first and third argument.

Let $\psi \in \mathbb{R}^n$ be the vector with entries $\psi_i = \|x_i\|_2^2/p - \tau$. We have the following result.

Theorem 4. *Let $x_i \in \mathbb{R}^p$ for $i = 1, \ldots, n$ be n i.i.d. random vectors and form the kernel matrix*

$$K_{ij} = g\left(\frac{\|x_i\|_2^2}{p}, \frac{\langle x_i, x_j \rangle}{p}, \frac{\|x_j\|_2^2}{p}\right), \quad i, j = 1, \ldots, n.$$

Then under the assumptions above we have

$$\lim_{p\to\infty} \|K - M\|_2 \overset{\mathsf{p}}{=} 0,$$

where

$$M = c_0 I + c_1 1 1^\mathsf{T} + c_2 \frac{XX^\mathsf{T}}{p} + c_3(\psi 1^\mathsf{T} + 1\psi^\mathsf{T}) + c_4 \psi\psi^\mathsf{T} + c_5\big((\psi\circ\psi)1^\mathsf{T} + 1(\psi\circ\psi)^\mathsf{T}\big) \quad (1)$$

$$c_0 = g(\tau,\tau,\tau) - g(\tau,0,\tau) - \frac{\partial g}{\partial z_2}(\tau,0,\tau)\frac{\mathrm{tr}\Sigma_p}{p},$$

$$c_1 = g(\tau,0,\tau) + \frac{\partial^2 g}{\partial z_2^2}(\tau,0,\tau)\frac{\mathrm{tr}\Sigma_p^2}{2p^2},$$

$$c_2 = \frac{\partial g}{\partial z_2}(\tau,0,\tau)$$

$$c_3 = \frac{\partial g}{\partial z_1}(\tau,0,\tau)$$

$$c_4 = \frac{\partial^2 g}{\partial z_1 \partial z_3}(\tau,0\tau)$$

$$c_5 = \frac{1}{2}\frac{\partial^2 g}{\partial z_1^2}(\tau,0,\tau)$$

Proof. Define $\tau := \lim_{p\to\infty} \frac{\mathrm{tr}\Sigma}{p}$, $z = [\|x_i\|_2^2/p, \langle x_i, x_j\rangle/p, \|x_j\|_2^2/p]^\mathsf{T}$ and for $i \neq j$ write the second order Taylor expansion of $g(z_1, z_2, z_3)$ around $z_0 = [\tau, 0, \tau]^\top$ for $i \neq j$

$$g(\|x_i\|_2^2/p, \langle x_i, x_j\rangle/p, \|x_j\|_2^2/p) = g(x_0) + \langle \nabla g(x_0), z\rangle + \frac{1}{2}\langle \nabla^2 g(x_0), z^{\otimes 2}\rangle + R_{ij},$$

where R_{ij} is the Lagrange remainder of this Taylor expansion and has the form

$$R_{ij} = \frac{1}{6}\sum_{\substack{\alpha_1,\alpha_2,\alpha_3 \\ \sum_k \alpha_k = 3}} \frac{\partial^{\alpha_1\alpha_2\alpha_3}}{\partial z_1^{\alpha_1}\partial z_2^{\alpha_2}\partial z_3^{\alpha_3}} g(\xi_1^{ij}, \xi_2^{ij}, \xi_3^{ij}) \left(\frac{\|x_i\|_2^2}{p} - \tau\right)^{\alpha_1} \left(\frac{\langle x_i, x_j\rangle}{p}\right)^{\alpha_2} \left(\frac{\|x_j\|_2^2}{p} - \tau\right)^{\alpha_3},$$

where $\alpha_k \geq 0$ for some $\xi_1^{ij}, \xi_2^{ij}, \xi_3^{ij}$

$$\begin{aligned}
\min(\tau, \|x_i\|_2^2/p) \leq \xi_1^{ij} &\leq \max(\tau, \|x_i\|_2^2/p), \\
\min(0, \langle x_i, x_j\rangle/p) \leq \xi_2^{ij} &\leq \max(0, \langle x_i, x_j\rangle/p), \\
\min(\tau, \|x_j\|_2^2/p) \leq \xi_3^{ij} &\leq \max(\tau, \|x_j\|_2^2/p).
\end{aligned}$$

For $i = j$, consider the second order Taylor expansion of $g(z_1, z_2, z_3)$ around $z_0 = [\tau, \tau, \tau]^\top$

$$g(\|x_i\|_2^2/p, \|x_i\|_2^2/p, \|x_j\|_2^2/p) = g(x_0) + \langle \nabla g(x_0), z\rangle + \frac{1}{2}\langle \nabla^2 g(x_0), z^{\otimes 2}\rangle + R_{ii},$$

where again R_{ii} is the Lagrange remainder of this Taylor expansion and has the form

$$R_{ii} = \frac{1}{6} \sum_{\substack{\alpha_1,\alpha_2,\alpha_3 \\ \sum_k \alpha_k = 3}} \frac{\partial^{\alpha_1\alpha_2\alpha_3}}{\partial z_1^{\alpha_1} \partial z_2^{\alpha_2} \partial z_3^{\alpha_3}} g(\xi_1^{ii}, \xi_2^{ii}, \xi_3^{ii}) \left(\frac{\|x_i\|_2^2}{p} - \tau \right)^3.$$

where $\alpha_k \geq 0$ for some $\xi_1^{ii}, \xi_2^{ii}, \xi_3^{ii}$

$$\min(\tau, \|x_i\|_2^2/p) \leq \xi_k^{ii} \leq \max(\tau, \|x_i\|_2^2/p), \quad \text{for } k = 1, 2, 3.$$

As is shown in [21], under the Assumptions A1-A3 we have

$$\max_i \left| \|x_i\|_2^2/p - \tau \right| \leq p^{-1/2} \log p, \quad \max_{i \neq j} |\langle x_i, x_j\rangle/p| \leq p^{-1/2} \log p.$$

See Lemma A.3 of [21] for the proof. This inequalities are a result of the sub-Gaussianity assumption on y_is. This assumption can be relaxed to moment assumptions of suitable order. Therefore, for the remainder of the Taylor expansion of off-diagonal entries we have

$$\max_{i \neq j} |\xi_k^{ij} - \tau| \to 0 \quad \text{a.s. for } k = 1, 3,$$

and

$$\max_{i \neq j} |\xi_2^{ij}| \to 0 \quad \text{a.s.}$$

Therefore, by continuity assumptions $\frac{\partial^{\alpha_1\alpha_2\alpha_3}}{\partial z_1^{\alpha_1}\partial z_2^{\alpha_2}\partial z_3^{\alpha_3}} g(\xi_1^{ij}, \xi_2^{ij}, \xi_3^{ij}) \to \frac{\partial^{\alpha_1\alpha_2\alpha_3}}{\partial z_1^{\alpha_1}\partial z_2^{\alpha_2}\partial z_3^{\alpha_3}} g(\tau, 0, \tau)$ and it is also bounded.

Similarly, for the remainder of Taylor expansion of the diagonal entries we have

$$\max_{i \neq j} |\xi_k^{ij} - \tau| \to 0 \quad \text{a.s. for } k = 1, 2, 3.$$

Therefore, again by the continuity assumption we have

$$\frac{\partial^{\alpha_1\alpha_2\alpha_3}}{\partial z_1^{\alpha_1}\partial z_2^{\alpha_2}\partial z_3^{\alpha_3}} g(\xi_1^{ii}, \xi_2^{ii}, \xi_3^{ii}) \to \frac{\partial^{\alpha_1\alpha_2\alpha_3}}{\partial z_1^{\alpha_1}\partial z_2^{\alpha_2}\partial z_3^{\alpha_3}} g(\tau, \tau, \tau)$$

and it is also bounded.

The argument is very similar to the argument made in [21]. We consider the diagonal entries of the kernel matrix and the off-diagonals separately. We also consider terms of the zeroth, first, second, and third order separately. Many of the terms are either exactly the same as the ones considered in [21] or the bounds that are derived therein can be used to bound them consistently in operator norm.

The zeroth order term: this term is exactly the same as the one considered in [21]. This term does not necessarily vanish and we need to keep it in the Taylor expansion to have consistent approximation of the kernel matrix in operator norm.

The first order terms: There are three first order terms. We consider them one by one. First, consider the $n \times n$ matrix M_1^1 with entries

$$[M_1^1]_{ij} = \frac{\|x_i\|_2^2}{p} - \tau, \qquad \text{for } i, j = 1, \ldots, n.$$

This matrix is a rank-one matrix: if we consider a vector $\psi \in \mathbb{R}^n$ with $\psi_i = \|x_i\|_2^2/p - \tau$, we have

$$M_1^1 = \psi 1^\mathsf{T}.$$

Therefore the operator norm of M is bounded by

$$\|M\|_1 = \|\psi\|_2 \|1\|_2 \leq \sqrt{n} p^{-1/2} \log p \sqrt{n} = n p^{-1/2} \log p.$$

Hence, the operator norm of this term does not vanish and we need to keep it in the Taylor expansion to have a consistent approximation of the kernel matrix. However, note that the kernel matrices in our kernel models (see for example Equation (2.4)) are all divided by n. In these cases, in the limit of $n, p \to \infty$ this term can be ignored as the operator norm of M/n is bounded by $p^{-1/2} \log p$ which vanishes in the limit.

The next first order term is a multiple of the matrix M_2^1 with entries $[M_2^1]_{ij} = \langle x_i, x_j \rangle / p$ i.e. $M_2^1 = XX^\mathsf{T}/p$. This term is exactly the same as the term considered in [21], does not vanish, and it is what makes this kernels all similar to an inner product kernel.

The last first order term has is a multiple of the matrix M_3^1 with entries $[M_3^1]_{ij} = \|x_j\|_2^2/p - \tau$. Therefore, $M_3^1 = {M_1^1}^\mathsf{T}$ and we could use the equations that we had for M_1^1. in particular, the operator norm of this matrix does not vanish, but the operator norm of M_3^1/n which appears in kernel model equations does indeed vanish and thus this term can be ignored in those instances. This completes the analysis of the first order terms. This concludes the treatment of the first order terms.

The second order terms: In total, there are nine second order terms in the Taylor expansion some of which are similar due to the symmetries of the kernel function.

First, let us consider the term that is a multiple of M_1^2 with entries

$$[M_1^2]_{ij} = \left(\frac{\|x_i\|_2^2}{p} - \tau \right)^2.$$

Recalling the definition of the vector ψ as the vector with entries $\psi_i = \|x_i\|_2^2/p - \tau$ this matrix can be written as $M_1^2 = (\psi \circ \psi)1^\mathsf{T}$ where $\circ$ denotes the Hadamard product. Therefore, the operator norm of this matrix is

$$\|M_1^2\|_2 = \|\psi \circ \psi\|_2 \|1\|_2 \leq \log p \sqrt{\frac{n}{p}}.$$

Therefore, this term also does not vanish in operator norm, but M_1^2/n vanishes and can be ignored.

The next second order term to consider is of the form M_2^2 where

$$[M_2^2]_{ij} = \left(\frac{\|x_i\|_2^2}{p} - \tau \right) \frac{\langle x_i, x_j \rangle}{p}.$$

If we denote the diagonal matrix with entries $\|x_i\|_2^2/p - \tau$ on the diagonal with $\mathrm{diag}(\psi)$, we have $M_2^2 = \mathrm{diag}(\psi) X X^\mathsf{T}/p$, therefore,

$$\|M_2^2\|_2 \leq \|\mathrm{diag}(\psi)\|_2 \|X X^\mathsf{T}/p\|_2 \leq p^{-1/2} \log p \|X X^\mathsf{T}/p\|_2 \|Y \Sigma_x Y^\mathsf{T}\|_2,$$

where Y is the matrix of i.i.d. sug-Gaussian random variables that generate X (refer to Assumptions A1-A3). By sub-Gaussianity assumption (in fact finite 4th moment is enough), the operator norm of the matrix Y converges in the limit of $n, p \to \infty$ with n/p finite (Theorem 2.1 of [57]). Therefore, $\|M_2^2\|_2 \to 0$ almost surely in the limit and this term can be ignored.

The next term is of the form M_3^2 where

$$[M_3^2]_{ij} = \left(\frac{\|x_i\|_2^2}{p} - \tau \right) \left(\frac{\|x_j\|_2^2}{p} - \tau \right),$$

which can be rewritten as $M_3^2 = \psi \psi^\mathsf{T}$. Thus, $\|M_3^2\|_2 = \|\psi\|_2^2 \leq \log^2 p$. Therefore, this term also does not vanish but M_3^2/n vanishes in operator norm.

Next up is a term of the form M_4^2 with entries

$$[M_4^2]_{ij} = \left(\frac{\langle x_i, x_j \rangle}{p}\right)^2.$$

This term is exactly the same as the term that is considered in [21]. Its operator norm converges in the limit and operator norm of M_4^2/n vanishes in the asymptotic limit.

All the other second order terms have similar forms to the ones considered here thus far. Therefore, the same results apply to them. This concludes the analysis of the second order terms.

The third order terms: It only remains to show that the remainder of the second order Taylor expansions vanish in operator norm. The remainders consist of the third order terms. here we consider these terms one by one.

First, let us consider the term of the form M_1^3 with entries

$$[M_1^3]_{ij} = \left(\frac{\|x_i\|_2^2}{p} - \tau\right)^3,$$

i.e. $M_1^3 = (\psi \circ \psi \circ \psi)1^\mathsf{T}$. The operator norm of this term can be bounded as

$$\|M_1^3\|_2 = \|\psi \circ \psi \circ \psi\|_2 \|1\|_2 \leq \frac{\log p}{p}\sqrt{n},$$

which goes to zero almost surely in asymptotic limit, and hence can be ignored.

The next term is $M_2^3 = \mathrm{diag}(\psi \circ \psi)(XX^\mathsf{T}/p)$. Recall that as we showed earlier in the analysis of the second order terms, the operator norm of XX^T/p is bounded, and the operator norm of $\mathrm{diag}(\psi \circ \psi)$ goes to zero as $\log^2 p/p$. Therefore, this term also vanishes.

Next, consider the term $M_3^3 = \mathrm{diag}(\psi)\left(XX^\mathsf{T}/p \circ XX^\mathsf{T}/p\right)$. Again, the operator norm of the second matrix is bounded but the operator norm $\mathrm{diag}(\psi)$ vanishes in the limit. Hence, this term can also be ignored.

Next, consider the term of the form $M_4^3 = (\psi \circ \psi)\psi^\mathsf{T}$. The operator norm of this term is bounded by

$$\|M_4^3\|_2 = \|\psi \circ \psi\|_2 \|\psi\|_2 \leq \frac{\log^3 p}{\sqrt{p}},$$

which also goes to zero.

Next up is the term of the form $M_5^3 = \left(XX^\mathsf{T}/p \circ XX^\mathsf{T}/p \circ XX^\mathsf{T}/p\right)$ which is exactly of the form analyzed in [21], and shown therein to vanish in the limit.

All the other third order terms have similar forms to the ones considered here thus far and hence they all can be ignored and still have a consistent approximation of the kernel matrix in operator norm in the limit. This shows that in the limit, the second order Taylor expansion is exact, i.e. it converges in operator norm in probability to the kernel matrix.

Corrections for the diagonal terms: Recall that we used different Taylor expansions for the diagonal and the off-diagonal terms of the kernel matrix. In our analysis, when we considered first, second, and third order terms, we should have made the diagonals of such terms zero. For example, a first order term was of the form $M_1^1 = \psi 1^\mathsf{T}$. The diagonal entries of this matrix can be zeroed by subtracting the matrix $\mathrm{diag}(\psi)$. Notice that the operator norm of diagonal matrices is very easy to control. For example, the operator norm of $\mathrm{diag}(\psi)$ goes to zero as $\max_i |\|x_i\|_2^2/p - \tau| \leq \log p/\sqrt{p}$. All the other diagonal corrections except for the zeroth order terms go to zero in operator norm using a similar argument. Hence, we only need to correct for the diagonals of the zeroth order terms which are constants. This concludes the proof.

□

Corollary 3. *The normalized kernel matrix K/n can be consistently approximated in operator norm by the matrix M/n where*

$$M = c_1 1 1^\mathsf{T} + c_2 \frac{XX^\mathsf{T}}{p}.$$

Proof. This is a direct result of Theorem 4. In the proof, we derived upper bounds for the operator norm of the terms that remain in Equation (1). Once we normalize the matrix M by n, many of these terms vanish in operator norm in the limit. See the proof of Theorem 4 for the details. □

Corollary 4. *The normalized kernel matrix $K/\sqrt{n}$ can be consistently approximated in operator norm by the matrix $M/\sqrt{n}$ where*

$$M = c_1 1 1^\mathsf{T} + c_2 \frac{XX^\mathsf{T}}{p} + c_3(\psi 1^\mathsf{T} + 1\psi^\mathsf{T}).$$

Proof. The proof is the same as the proof of Corollary 3. We use Theorem 4 and only keep the terms which have non-vanishing operator norm when normalized by $\sqrt{n}$. □

These two corollaries will be used in what follows to prove the results of this work.

B Some Useful Lemmas

Lemma 1. *Let A be an invertible $n \times n$ matrix, and $U \in \mathbb{R}^{d\times n}$ for some d, then*

$$(A + UU^\mathsf{T})^{-1}U = A^{-1} - A^{-1}U(I_n + U^\mathsf{T}A^{-1}U)^{-1}U^\mathsf{T}A^{-1}.$$

Proof. This is a special case of the Woodbury matrix identity. □

Lemma 2. *For any integer $t' \geq 0$ and matrix $A \in \mathbb{R}^{n\times p}$ we have*

$$(\alpha I_{p\times p} - A^\mathsf{T}A)^t A^\mathsf{T} = A^\mathsf{T}(\alpha I_{n\times n} - AA^\mathsf{T})^t.$$

Proof. This result can be easily proved by using the singular value decomposition of $A = U\Sigma V^\mathsf{T}$. □

Lemma 3. *Let $A_i \in \mathbb{R}^{n_1 \times n_2}$ and $B_i \in \mathbb{R}^{n_2 \times n_3}$ be two sequences of random matrices and assume that*

$$\lim_{i\to\infty} \|A_i - A\|_2 \xrightarrow{\mathsf{p}} 0, \quad \lim_{i\to\infty} \|B_i - B\|_2 \xrightarrow{\mathsf{p}} 0.$$

Then if $\|A\|_2, \|B\|_2 < \infty$ we have

$$\lim_{i\to\infty} \|A_i B_i - AB\|_2 \xrightarrow{\mathsf{p}} 0.$$

Proof.

$$\begin{aligned}
\|A_i B_i - AB\|_2 &= \|A_i B_i - AB_i + AB_i - AB\|_2 \\
&\leq \|A_i B_i - AB_i\|_2 + \|AB_i - AB\|_2 \\
&\leq \|A_i - A\|_2 \|B_i\|_2 + \|A\|_2 \|B_i - B\|_2 \\
&\xrightarrow{\mathsf{p}} 0,
\end{aligned}$$

where the last equality follows from the continuous mapping theorem ([45]). This proves the claim.

□

A special case of this theorem is when B_i is a sequence of $n_2 \times 1$ matrices, i.e. a sequence of vectors. In this case, the operator norm is the same as the ℓ_2 norm. Therefore, we have the following corollary.

Corollary 5. *Let $A_i \in \mathbb{R}^{n_1 \times n_2}$ and $x_i \in \mathbb{R}^{n_2 \times n_3}$ be a sequence of random matrices and random vectors respectively and let*

$$\lim_{i\to\infty} \|A_i - A\|_2 \xrightarrow{\mathsf{p}} 0, \quad \lim_{i\to\infty} \|x_i - x\|_2 \xrightarrow{\mathsf{p}} 0.$$

Then if $\|A\|_2, \|x\|_2 < \infty$ *we have*

$$\lim_{i\to\infty} \|A_i x_i - Ax\|_2 \stackrel{\mathsf{p}}{=} 0.$$

The next corollary considers limits of powers of a matrix which can be proven by a simple induction using Lemma 3.

Corollary 6. *Let* $A_i \in \mathbb{R}^{n\times n}$ *be a sequence of random matrices and assume that* $\lim_{i\to\infty} \|A_i - A\|_2 \stackrel{\mathsf{p}}{=} 0$. *Then for any finite* $m \in \mathbb{N}$ *we have* $\lim_{i\to\infty} \|A_i^m - A^m\|_2 \stackrel{\mathsf{p}}{=} 0$.

C Proof of Theorem 1

Let $\overline{X} = [x_{\text{ts}}^{\mathsf{T}}, X_{\text{tr}}^{\mathsf{T}}]^{\mathsf{T}}$ and partition the kernel matrix, $K(\overline{X}, \overline{X})$ as

$$K(\overline{X}, \overline{X}) = \begin{bmatrix} K(x_{\text{ts}}, x_{\text{ts}}) & K(x_{\text{ts}}, X_{\text{tr}}) \\ K(X_{\text{tr}}, x_{\text{ts}}) & K(X_{\text{tr}}, X_{\text{tr}}) \end{bmatrix}.$$

The optimal estimator in (3.8) is

$$\widehat{f}_{\text{ker}}(x_{\text{ts}}) = \frac{K(x_{\text{ts}}, X_{\text{tr}})}{\sqrt{n}} \left(\frac{1}{n} K(X_{\text{tr}}, X_{\text{tr}}) + \lambda I\right)^{-1} \frac{y_{\text{tr}}}{\sqrt{n}}. \tag{2}$$

This is a product of three terms. Our proof relies on Lemma 3. We will show that each of these terms converge in operator norm to a kernel with a simple (almost linear) feature map in probability. Furthermore, they all have bounded operator norms. Therefore, this Lemma implies that for a given test data, the output of the model learned by the kernel model is the same as the model learned by the kernel with the simple feature map in probability. Therefore, the two models are equivalent. We make this argument precise below.

First, note that the kernel shows up with a scaling factor of $1/\sqrt{n}$ for $K(x_{\text{ts}}, X_{\text{tr}})$ and a scaling factor of $1/n$ for $K(X_{\text{tr}}, X_{\text{tr}})$. Hence, we only need to have a consistent approximation of the kernel with these scaling factors. We have the following result.

Proposition 2. *Consider the kernel matrix $K(\overline{X}, \overline{X})$ under assumptions A1-A3. Then, there exists constants γ_1, γ_2 and γ_3 such that for the kernel K_{lin} with feature map $x \mapsto \phi(x)$ (i.e. $K_{\text{lin}}(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle$) where*

$$\phi(x) = \left[\gamma_1 x, \quad \gamma_2, \quad \gamma_3 \left(\frac{\|x\|_2^2/p - \tau - \overline{\psi}}{\|\psi - \overline{\psi} 1\|_2} + \frac{1}{\sqrt{n}} \right), \quad -\gamma_3 \left(\frac{\|x\|_2^2/p - \tau - \overline{\psi}}{\|\psi - \overline{\psi} 1\|_2} - \frac{1}{\sqrt{n}} \right) \right] \tag{3}$$

we have

$$\lim_{p \to \infty} \frac{1}{\sqrt{n}} \|K(\overline{X}, \overline{X}) - K_{\text{lin}}(\overline{X}, \overline{X})\|_2 \overset{\text{p}}{=} 0.$$

Here, ψ is a vector with $\psi_i = \|x_i\|_2^2/p - \tau$, $\overline{\psi} = 1/n \sum_i \psi_i$ and

$$\gamma_1 = \sqrt{c_1 + 2\overline{\psi} c_3}, \quad \gamma_2 = \sqrt{c_2}, \quad \gamma_3 = \left(\sqrt{n} \|\psi - \overline{\psi} 1\|_2 c_3 \right)^{1/2},$$

and c_1, c_2 and c_3 are defined in Theorem 4[1].

Proof. By Corollary 4, the kernel matrix $K(\overline{X}, \overline{X})/\sqrt{n}$ can be approximated in operator norm by $M/\sqrt{n}$ where

$$M = c_1 1 1^{\mathsf{T}} + c_2 \frac{X X^{\mathsf{T}}}{p} + c_3 (\psi 1^{\mathsf{T}} + 1 \psi^{\mathsf{T}}).$$

Therefore, we only need to show that a feature map of the form claimed, can generate M. In other words, if we denote by $\phi(\overline{X})$ the matrix whose ith row is ϕ applied to the ith row of $\overline{X}$, then we need to show that

$$\lim_{p \to \infty} \frac{1}{\sqrt{n}} \|M - \phi(\overline{X}) \phi(\overline{X})^{\mathsf{T}}\|_2 \overset{\text{p}}{=} 0.$$

Or stated otherwise, we need to find a symmetric decomposition of M.

The first two terms are already in the correct form. We next obtain a symmetric decomposition of $\psi 1^{\mathsf{T}} + 1 \psi^{\mathsf{T}}$. First note that this is a rank-two matrix symmetric matrix.

[1]We are using the term feature map here with some abuse of notation. Please refer to Remark 6.

Therefore, it has an eigenvalue decomposition. Also, the eigenvectors corresponding to the non-zero eigenvalues are in the span of $\{\psi, 1\}$. We first orthogonilize these vectors. Let $\overline{\psi} = 1/n \sum_i \psi_i$ be the empirical average of the components in ψ. Then the vectors $\{\psi - \overline{\psi}1, 1\}$ are orthogonal. We can also rewrite

$$\psi 1^\mathsf{T} + 1\psi^\mathsf{T} = (\psi - \overline{\psi}1)1^\mathsf{T} + (1(\psi - \overline{\psi}1)^\mathsf{T}) + 2\overline{\psi}11^\mathsf{T},$$

which decomposes this matrix in an orthogonal basis. Therefore,

$$M = (c_1 + 2\overline{\psi}c_3)11^\mathsf{T} + c_2\frac{XX^\mathsf{T}}{p} + c_3\left((\psi - \overline{\psi}1)1^\mathsf{T} + 1(\psi - \overline{\psi})^\mathsf{T}\right),$$

and we need to find a decomposition of $A := \big((\psi - \overline{\psi}1)1^\mathsf{T} + 1(\psi - \overline{\psi}1)^\mathsf{T}\big)$.

Now, let $u = \alpha(\psi - \overline{\psi}1) + \beta 1$ be an eigenvector of the matrix A with corresponding eigenvalue λ, i.e.

$$A\big(\alpha(\psi - \overline{\psi}1) + \beta 1\big) = \lambda\big(\alpha(\psi - \overline{\psi}1) + \beta 1\big).$$

Simplifying this equation we get

$$n\beta = \lambda\alpha$$
$$\lambda\beta = \alpha\|\psi - \overline{\psi}1\|_2^2.$$

If we set $\alpha = 1/\|\psi - \overline{\psi}1\|_2$ (as the the norm of eigenvectors are arbitrary) we get

$$\lambda = \pm\sqrt{n}\|\psi - \overline{\psi}1\|_2, \qquad \alpha = \frac{1}{\|\psi - \overline{\psi}1\|_2}, \qquad \beta = \pm\frac{1}{\sqrt{n}}$$

which gives us the decomposition

$$\begin{aligned} A =& \sqrt{n}\|\psi - \overline{\psi}1\|_2 \left(\frac{\psi - \overline{\psi}1}{\|\psi - \overline{\psi}1\|_2} + \frac{1}{\sqrt{n}}1\right)\left(\frac{\psi - \overline{\psi}1}{\|\psi - \overline{\psi}1\|_2} + \frac{1}{\sqrt{n}}1\right)^{\mathsf{T}} \\ & - \sqrt{n}\|\psi - \overline{\psi}1\|_2 \left(\frac{\psi - \overline{\psi}1}{\|\psi - \overline{\psi}1\|_2} - \frac{1}{\sqrt{n}}1\right)\left(\frac{\psi - \overline{\psi}1}{\|\psi - \overline{\psi}1\|_2} - \frac{1}{\sqrt{n}}1\right) \end{aligned}$$

From this factorization, the Proposition easily follows. □

Remark 6. Note that the feature map defined in Equation (3) is not actually a feature map as it $\overline{\psi}$ and $\|\psi - \overline{\psi}1\|_2$ are both functions of all the data points. However, what we need for the proof is only a consistent factorization of the kernel matrix in operator norm and not a feature map is its strict sense. Therefore, for the purpose of the proof, we can treat these as constants and with some abuse of notation still call it a feature map.

Next, we show that each of the $K(x_{\text{ts}}, X_{\text{tr}})/\sqrt{n}$ and $K(X_{\text{tr}}, X_{\text{tr}})/n$ can be obtained using Proposition 2. First, another application of Proposition 2 to X_{tr} instead of $\overline{X}$ shows that $K(X_{\text{tr}}, X_{\text{tr}})/\sqrt{n}$ and hence $K(X_{\text{tr}}, X_{\text{tr}})/n$ can be obtained from the feature map in (3). Also, observe that $\overline{\psi}$ and $\|\psi - \overline{\psi}1\|_2$ computed over X_{tr} and over $\overline{X}$ are the same in the limit. Hence, that same feature map as of the one used on $\overline{X}$ gives the correct normalized kernel matrix in operator norm in probability. In other words, with $\phi(\cdot)$ given in (3) we have

$$lim_{p\to\infty}\frac{1}{n}\|K(X_{\text{tr}}, X_{\text{tr}}) - \phi(X_{\text{tr}})\phi(X_{\text{tr}})^{\mathsf{T}}\|_2 \xrightarrow{\mathsf{p}} 0.$$

Now, consider the term $K(x_{\text{ts}}, X_{\text{tr}})/\sqrt{n}$. By Proposition 2 we have

$$\lim_{p\to\infty}\frac{1}{\sqrt{n}}\|K(\overline{X}, \overline{X}) - \phi(\overline{X})\phi(\overline{X})^{\mathsf{T}}\|_2 \xrightarrow{\mathsf{p}} 0.$$

Let $e_1 = [1, 0, 0, ..., 0]^\mathsf{T} \in \mathbb{R}^{n+1}$. We have

$$\begin{aligned}\lim_{p\to\infty} \frac{1}{\sqrt{n}} \|(K(x_{\text{ts}}, \overline{X}) - \phi(x_{\text{ts}})\phi(\overline{X})^\mathsf{T}\|_2 &= \lim_{p\to\infty} \frac{1}{\sqrt{n}} \|(K(\overline{X}, \overline{X}) - \phi(\overline{X})\phi(\overline{X})^\mathsf{T})e_1\|_2 \\ &\leq \lim_{p\to\infty} \frac{1}{\sqrt{n}} \|K(\overline{X}, \overline{X}) - \phi(\overline{X})\phi(\overline{X})^\mathsf{T}\|_2 \|e_1\|_2 \\ &\overset{\mathsf{p}}{=} 0.\end{aligned}$$

Hence, we have

$$\lim_{p\to\infty} \frac{1}{\sqrt{n}} \|K(x_{\text{ts}}, X_{\text{tr}}) - \phi(x_{\text{ts}})\phi(X_{\text{tr}})^\mathsf{T}\|_2 \overset{\mathsf{p}}{=} 0$$

Now, consider the model parameterized as $\tilde{f}(x) = \langle \phi(x), \widehat{\theta} \rangle$, where

$$\widehat{\theta} = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} (y_i - \langle \phi(x_i), \theta \rangle)^2 + \lambda \|\theta\|_2^2,$$

and $\phi(\cdot)$ is the feature map defined in (3). As we showed in Section 2.1, this model is equivalent to the kernel model with kernel $K(x_1, x_2) = \langle \phi(x_1, \phi(x_2) \rangle$ and has the form

$$\begin{aligned}\tilde{f}_{\mathsf{krr}}(x_{\text{ts}}) &= \phi(x_{\text{ts}})\widehat{\theta} \\ &= \frac{\phi(x_{\text{ts}})\phi(X_{\text{tr}})^\mathsf{T}}{\sqrt{n}} \left(\frac{1}{n} \phi(X_{\text{tr}})\phi(X_{\text{tr}})^\mathsf{T} + \lambda I \right)^{-1} \frac{y_{\text{tr}}}{\sqrt{n}}. \end{aligned} \tag{4}$$

First note that as we mentioned in the proof of Theorem 4, we have

$$\max_i \left| \|x_i\|_2^2/p - \tau \right| \leq p^{-1/2} \log p.$$

Therefore, we can set $|\|x_i\|_2^2/p - \tau| = 0$ for any finite number of training or test samples. In particular, for x_{ts} we could set $\|x_{\text{ts}}\|_2^2/p - \tau = 0$ which gives us

$$\phi(x_{\text{ts}}) = \left[\gamma_1 x, \quad \gamma_2, \quad \gamma_3 \left(\frac{-\overline{\psi}}{\|\psi - \overline{\psi}1\|_2} + \frac{1}{\sqrt{n}} \right), \quad -\gamma_3 \left(\frac{-\overline{\psi}}{\|\psi - \overline{\psi}1\|_2} - \frac{1}{\sqrt{n}} \right) \right] := [\alpha_1 x, \alpha_2, \alpha_3, \alpha_4].$$

Therefore,

$$\tilde{f}_{\mathsf{krr}} = \widehat{\theta}_1 \alpha_1 x + \widehat{\theta}_2 \alpha_2 + \widehat{\theta}_3 \alpha_3 + \widehat{\theta}_4 \alpha_4,$$

which is indeed a linear model.

So far we have shown that the model learned in the feature space is linear, and each of the terms in the model in (4) converge in operator norm in probability to the corresponding term in (2). All that remains to show is that each of the terms has bounded operator norm. Then Lemma 3 would give us the desired result.

It remains to prove that all the terms in this product will have bounded operator norms. Let us begin by considering the simplest term: $y_{\text{tr}}/\sqrt{n}$. For this term we have

$$\|y_{\text{tr}}/\sqrt{n}\|_2 = \frac{1}{n} \sum_{i=1}^{n} y_{\text{tr},i}^2.$$

Therefore, if we have assumed that the training data distribution has finite second moments over the labels. Thus, using strong law of large numbers we obtain

$$\|y_{\text{tr}}/\sqrt{n}\|_2 \overset{a.s.}{\to} \sigma_{y_{\text{tr}}}^2 < \infty \qquad \text{as } n \to \infty.$$

Next, by positive semi-definiteness of kernels, we have that $K(X_{\text{tr}}, X_{\text{tr}}) \succeq 0$. Therefore, $\|K(X_{\text{tr}}, X_{\text{tr}})/n + \lambda I\|_{\text{op}} \geq \lambda$, and thus for any $\lambda > 0$, we obtain $\|(M_{22}/n + \lambda I)^{-1}\|_{\text{op}} \leq 1/\lambda$. Therefore, the middle term in both models also has bounded operator norm. Finally, the approximation of the first term is

$$K(x_{\text{ts}}, X_{\text{tr}}) = c_1 1^{\mathsf{T}} + c_2 x_{\text{ts}} X_{\text{tr}}^{\mathsf{T}} p + c_3 \psi^{\mathsf{T}}.$$

Using the bounds derived in the proof of Theorem 4, each of these terms when normalized by $1/\sqrt{n}$ have bounded operator norm.

$$\lim_{p \to \infty} |\tilde{f}_{\mathsf{krr}}(x_{\text{ts}}) - \widehat{f}_{\mathsf{krr}}(x_{\text{ts}})| \overset{\mathsf{p}}{=} 0.$$

In fact, we can prove a stronger result. So long as the number of test samples, n_{ts}, satisfy $n_{\text{ts}} = n_{\text{tr}}^{\gamma}$ where $\gamma < 1$, the result that we have proven holds, using the same bounds that we have derived thus far. Therefore, the number of test samples can grow to infinity, but at a slower rate than the number of training samples.

In order to make the result look simpler, we could go one step further and combine some of the features. In order to do so, we use the fact that the models learned by ridge regression are invariant under orthogonal transformation of features. More specifically, if $O \in \mathbb{R}^{p \times p}$ is an orthogonal matrix, i.e. $OO^{\mathsf{T}} = I$, and we have two features maps $\phi_1(x)$ and $\phi_2(x)$ such that, $\phi_2(x) = \phi_1(x)O$, then if we learn two models f_1 and f_2 using ridge regression with the same regularization parameter over the features ϕ_1 and ϕ_2 respectively, then the two models are exactly the same. This result can be shown very easily by a change of variables in the ridge optimization problem.

Using this fact, we could transform the last two features in Equation (3) (let us call them ϕ_3, ϕ_4) to

$$\phi_3' = \frac{\sqrt{2}}{2}(\phi_3 + \phi_4), \qquad \phi_4' = \frac{\sqrt{2}}{2}(\phi_3 - \phi_4).$$

Doing this would result in the feature map

$$\phi'(x) = \left[\gamma_1 x, \quad \gamma_2, \quad \gamma_3\sqrt{\frac{1}{n}}, \quad \gamma_3 \frac{\|x\|_2^2/p - \tau - \overline{\psi}}{\|\psi - \overline{\psi}1\|_2}\right],$$

where

$$\gamma_1 = \sqrt{c_1 + 2\overline{\psi}c_3}, \quad \gamma_2 = \sqrt{c_2}, \quad \gamma_3 = \left(2\sqrt{n}\|\psi - \overline{\psi}1\|_2 c_3\right)^{1/2},$$

Therefore, we could use this feature map to learn a linear model that is equivalent to the kernel model in the asymptotic regime. This completes the proof.

D Proof of Theorem 2

Here we show that if the kernel model and linear model are learned by gradient descent, they are equivalent to each other throughout the training.

Consider a kernel model parameterized in the feature space

$$\widehat{f}_{\ker}(x) = \langle \phi(x), \widehat{\theta} \rangle$$
$$\widehat{\theta} = \arg\min_{\theta} \frac{1}{n} \|(y_{\rm tr} - \phi(X_{\rm tr})\theta\|_2^2 + \lambda \|\theta\|_{L^2}^2,$$

where $\phi(X_{\rm tr})$ is a matrix with $\phi(x_i)^\mathsf{T}$ as its ith row. The gradient descent update for this problem is

$$\theta^{t+1} = (I - \rho(\frac{1}{n}(\phi(X_{\rm tr})^\mathsf{T}\phi(X_{\rm tr}) + \lambda I))\theta^t + \frac{\rho}{n}\phi(X_{\rm tr})^\mathsf{T} y_{\rm tr},$$

where ρ is the learning rate. Therefore, if initialized with $\theta_0 = 0$, after t steps of gradient descent we obtain

$$\theta^t = \sum_{t'=0}^{t-1} \rho(I - \rho((\frac{1}{n}\phi(X_{\rm tr})^\mathsf{T}\phi(X_{\rm tr}) + \lambda I))^{t'} \phi(X_{\rm tr})^\mathsf{T} \frac{y_{\rm tr}}{n}. \tag{5}$$

Using Lemma 2 we have

$$\theta^t = \sum_{t'=0}^{t-1} \rho\phi(X_{\rm tr})^\mathsf{T}(I - \rho((\frac{1}{n}\phi(X_{\rm tr})\phi(X_{\rm tr})^\mathsf{T} + \lambda I))^{t'} \frac{y_{\rm tr}}{n}.$$

Note that the identity matrix in this equation has a different size from the one in (5) and the sizes can be inferred from the number of samples as well as dimension of the feature space as in the Lemma 2. Therefore, by observing that $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$ the model at time t

represented by $\widehat{f}^t_{\text{ker}}$ evaluated on test data point x_{ts} has the form

$$\begin{aligned}\widehat{f}^t_{\text{ker}}(x_{\text{ts}}) &= \rho\frac{K(x_{\text{ts}},X_{\text{tr}})}{\sqrt{n}}\sum_{t'=0}^{t-1}(I-\rho((\frac{1}{n}K(X_{\text{tr}},X_{\text{tr}})+\lambda I))^{t'}\frac{y_{\text{tr}}}{\sqrt{n}} \\ &= \frac{K(x_{\text{ts}},X_{\text{tr}})}{\sqrt{n}}(\frac{1}{n}K(X_{\text{tr}},X_{\text{tr}})+\lambda I)^{-1}\left(I-\left(I-\rho((\frac{1}{n}K(X_{\text{tr}},X_{\text{tr}})+\lambda I)\right)^t\right)\frac{y_{\text{tr}}}{\sqrt{n}} \quad (6)\end{aligned}$$

The series in the equation above and hence the gradient descent converges if all the eigenvalues of $I-\rho((1/nK(X_{\text{tr}},X_{\text{tr}})+\lambda I)$ lie inside the unit circle which is always possible by choosing ρ that is small enough and the limiting solution is the kernel regression solution

$$\widehat{y}_{\text{ts}} = \frac{K(x_{\text{ts}},X_{\text{tr}})}{\sqrt{n}}\left(\frac{1}{n}K(X_{\text{tr}},X_{\text{tr}})+\lambda I\right)^{-1}\frac{y_{\text{tr}}}{\sqrt{n}}.$$

The rest of the proof is very similar to the proof of Theorem 1 in Appendix C. In particular, under Assumptions A1-A3, we showed in Proposition 2 that the kernels that we have considered can be computed using the feature map

$$\varphi(x) = \left[\gamma_1 x, \quad \gamma_2, \quad \gamma_3\sqrt{\frac{1}{n}}, \quad \gamma_3\frac{\|x\|_2^2/p-\tau-\overline{\psi}}{\|\psi-\overline{\psi}1\|_2}\right],$$

where

$$\gamma_1 = \sqrt{c_1+2\overline{\psi}c_3}, \quad \gamma_2 = \sqrt{c_2}, \quad \gamma_3 = \left(2\sqrt{n}\|\psi-\overline{\psi}1\|_2 c_3\right)^{1/2}.$$

Assume that we run the gradient descent on the model parameterized by this feature map

$$\begin{aligned}\tilde{f}_{\mathsf{krr}}(x) &= \langle\varphi(x),\widehat{\theta}\rangle \\ \widehat{\theta} &= \arg\min_{\theta}\frac{1}{n}\|(y_{\text{tr}}-\varphi(X_{\text{tr}})\theta\|_2^2+\lambda\|\theta\|_{L^2}^2.\end{aligned}$$

Then, if the same optimization step ρ is used, at step t of the gradient descent, we have the model

$$\tilde{f}^t_{\mathsf{krr}}(x_{\mathrm{ts}}) = \frac{\varphi(x_{\mathrm{ts}})\varphi(X_{\mathrm{tr}})^{\mathsf{T}}}{\sqrt{n}} \left(\frac{1}{n}\varphi(X_{\mathrm{tr}})\varphi(X_{\mathrm{tr}})^{\mathsf{T}} + \lambda I\right)^{-1} \left(I - \left(I - \rho((\frac{1}{n}\varphi(X_{\mathrm{tr}})\varphi(X_{\mathrm{tr}})^{\mathsf{T}} + \lambda I)\right)^t\right) \frac{y_{\mathrm{tr}}}{\sqrt{n}} \tag{7}$$

First, since $\|x_{\mathrm{ts}}\|_2^2/p - \tau = 0$ almost surely in limit, the exact same argument as in proof of Theorem 1 shows that $\tilde{f}^t_{\mathsf{krr}}$ is linear at each step of the gradient descent. Furthermore, suing Proposition 2, we have that each of the product terms in (7) is equal in operator in probability to the corresponding term in the last equality of (6) in the asymptotic limit. Also, for small enough values of ρ, all the terms have bounded operator norm. Then applying Lemma 3 completes the proof.

Bibliography

[1] Madhu Advani and Surya Ganguli. Statistical mechanics of optimal convex inference in high dimensions. *Physical Review X*, 6(3):031034, 2016.

[2] Sina Alemohammad, Zichao Wang, Randall Balestriero, and Richard Baraniuk. The recurrent neural tangent kernel. *arXiv preprint arXiv:2006.10246*, 2020.

[3] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. In *Advances in Neural Information Processing Systems*, pages 8139–8148, 2019.

[4] Benjamin Aubin, Antoine Maillard, Jean Barbier, Florent Krzakala, Nicolas Macris, and Lenka Zdeborová. The committee machine: Computational to statistical gaps in learning a two-layers neural network. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124023, 2019.

[5] Peter Bartlett, Dylan J Foster, and Matus Telgarsky. Spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1706.08498*, 2017.

[6] Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

[7] Mohsen Bayati and Andrea Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 57(2):764–785, 2011.

[8] Derek Bean, Peter J Bickel, Noureddine El Karoui, and Bin Yu. Optimal m-estimation in high-dimensional regression. *Proceedings of the National Academy of Sciences*, 110(36):14563–14568, 2013.

[9] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[10] Mikhail Belkin, Siyuan Ma, and Soumik Mandal. To understand deep learning we need to understand kernel learning. In *International Conference on Machine Learning*, pages 541–549. PMLR, 2018.

[11] Alberto Bietti and Julien Mairal. On the inductive bias of neural tangent kernels. *arXiv preprint arXiv:1905.12173*, 2019.

[12] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[13] Alon Brutzkus and Amir Globerson. Why do larger models generalize better? a theoretical perspective via the xor problem. In *International Conference on Machine Learning*, pages 822–830. PMLR, 2019.

[14] Xiuyuan Cheng and Amit Singer. The spectrum of random inner-product kernel matrices. *Random Matrices: Theory and Applications*, 2(04):1350010, 2013.

[15] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. *arXiv preprint arXiv:1812.07956*, 2018.

[16] Hugo Cui, Bruno Loureiro, Florent Krzakala, and Lenka Zdeborová. Generalization error rates in kernel regression: The crossover from the noiseless to noisy regime. *arXiv preprint arXiv:2105.15004*, 2021.

[17] Amit Daniely, Roy Frostig, and Yoram Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *Advances In Neural Information Processing Systems*, pages 2253–2261, 2016.

[18] Stéphane D'Ascoli, Maria Refinetti, Giulio Biroli, and Florent Krzakala. Double trouble in double descent: Bias and variance(s) in the lazy regime. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 2280–2290. PMLR, 13–18 Jul 2020.

[19] Edgar Dobriban and Stefan Wager. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247–279, 2018.

[20] David L Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914–18919, 2009.

[21] Noureddine El Karoui et al. The spectrum of kernel random matrices. *Annals of statistics*, 38(1):1–50, 2010.

[22] Melikasadat Emami, Mojtaba Sahraee-Ardakan, Parthe Pandit, Sundeep Rangan, and Alyson Fletcher. Generalization error of generalized linear models in high dimensions. In *International Conference on Machine Learning*, pages 2892–2901. PMLR, 2020.

[23] Marylou Gabrié, Andre Manoel, Clément Luneau, Jean Barbier, Nicolas Macris, Florent Krzakala, and Lenka Zdeborová. Entropy and mutual information in models of deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124014, 2019.

[24] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[25] Elizabeth Gardner and Bernard Derrida. Three unfinished works on the optimal storage capacity of networks. *Journal of Physics A: Mathematical and General*, 22(12):1983, 1989.

[26] Federica Gerace, Bruno Loureiro, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. Generalisation error in learning with random features and the hidden manifold model. In *International Conference on Machine Learning*, pages 3452–3462. PMLR, 2020.

[27] Cedric Gerbelot, Alia Abbara, and Florent Krzakala. Asymptotic errors for teacher-student convex generalized linear models (or: How to prove kabashima's replica formula). *arXiv preprint arXiv:2006.06581*, 2020.

[28] Behrooz Ghorbani, Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Linearized two-layers neural networks in high dimension. *The Annals of Statistics*, 49(2):1029–1054, 2021.

[29] Sebastian Goldt, Madhu S Advani, Andrew M Saxe, Florent Krzakala, and Lenka Zdeborová. Dynamics of stochastic gradient descent for two-layer neural networks in the teacher–student setup. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12):124010, 2020.

[30] Sebastian Goldt, Marc Mézard, Florent Krzakala, and Lenka Zdeborová. Modeling the influence of data structure on learning in neural networks: The hidden manifold model. *Physical Review X*, 10(4):041044, 2020.

[31] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*, 2019.

[32] Wei Hu, Lechao Xiao, Ben Adlam, and Jeffrey Pennington. The surprising simplicity of the early-time learning dynamics of neural networks. *arXiv preprint arXiv:2006.14599*, 2020.

[33] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580, 2018.

[34] Yoshiyuki Kabashima, Florent Krzakala, Marc Mézard, Ayaka Sakata, and Lenka Zdeborová. Phase transitions and sample complexity in bayes-optimal matrix factorization. *IEEE Transactions on information theory*, 62(7):4228–4265, 2016.

[35] Dimitris Kalimeris, Gal Kaplun, Preetum Nakkiran, Benjamin Edelman, Tristan Yang, Boaz Barak, and Haofeng Zhang. Sgd on neural networks learns functions of increasing complexity. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[36] Ganesh Ramachandra Kini and Christos Thrampoulidis. Analytic study of double descent in binary classification: The impact of loss. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2527–2532. IEEE, 2020.

[37] Florent Krzakala, Marc Mézard, Francois Sausset, Yifan Sun, and Lenka Zdeborová. Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and

threshold achieving matrices. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(08):P08009, 2012.

[38] Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.

[39] Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In *Advances in neural information processing systems*, pages 8570–8581, 2019.

[40] Tengyuan Liang and Alexander Rakhlin. Just interpolate: Kernel "ridgeless" regression can generalize. *The Annals of Statistics*, 48(3):1329–1347, 2020.

[41] Chaoyue Liu, Libin Zhu, and Mikhail Belkin. On the linearity of large non-linear models: when and why the tangent kernel is constant. *Advances in Neural Information Processing Systems*, 33, 2020.

[42] Fanghui Liu, Xiaolin Huang, Yudong Chen, and Johan AK Suykens. Random features for kernel approximation: A survey on algorithms, theory, and beyond. *arXiv preprint arXiv:2004.11154*, 2020.

[43] Fanghui Liu, Zhenyu Liao, and Johan Suykens. Kernel regression in high dimensions: Refined analysis beyond double descent. In *International Conference on Artificial Intelligence and Statistics*, pages 649–657. PMLR, 2021.

[44] Bruno Loureiro, Gabriele Sicuro, Cédric Gerbelot, Alessandro Pacco, Florent Krzakala, and Lenka Zdeborová. Learning gaussian mixtures with generalised linear models: Precise asymptotics in high-dimensions. *arXiv preprint arXiv:2106.03791*, 2021.

[45] Henry B Mann and Abraham Wald. On stochastic limit and order relationships. *The Annals of Mathematical Statistics*, 14(3):217–226, 1943.

[46] Alexander G de G Matthews, Mark Rowland, Jiri Hron, Richard E Turner, and Zoubin Ghahramani. Gaussian process behaviour in wide deep neural networks. *arXiv preprint arXiv:1804.11271*, 2018.

[47] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Generalization error of random feature and kernel methods: hypercontractivity and kernel matrix concentration. *Applied and Computational Harmonic Analysis*, 59:3–84, 2022.

[48] Elchanan Mossel. Deep learning and hierarchal generative models. *arXiv preprint arXiv:1612.09057*, 2016.

[49] Vidya Muthukumar, Adhyyan Narang, Vignesh Subramanian, Mikhail Belkin, Daniel Hsu, and Anant Sahai. Classification vs regression in overparameterized regimes: Does the loss function matter? *Journal of Machine Learning Research*, 22(222):1–69, 2021.

[50] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[51] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, pages 5947–5956, 2017.

[52] Parthe Pandit, Mojtaba Sahraee-Ardakan, Sundeep Rangan, Philip Schniter, and Alyson K Fletcher. Inference with deep generative priors in high dimensions. *arXiv preprint arXiv:1911.03409*, 2019.

[53] Parthe Pandit, Mojtaba Sahraee-Ardakan, Sundeep Rangan, Philip Schniter, and Alyson K. Fletcher. Matrix inference and estimation in multi-layer models. In *NeurIPS*, 2020.

[54] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[55] Sundeep Rangan, Philip Schniter, and Alyson K Fletcher. Vector approximate message passing. *IEEE Transactions on Information Theory*, 65(10):6664–6684, 2019.

[56] Maria Refinetti, Sebastian Goldt, Florent Krzakala, and Lenka Zdeborová. Classifying high-dimensional gaussian mixtures: Where kernel methods fail and neural networks succeed. *arXiv preprint arXiv:2102.11742*, 2021.

[57] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. In *Proceedings of the International Congress of Mathematicians 2010 (ICM 2010) (In 4 Volumes) Vol. I: Plenary Lectures and Ceremonies Vols. II–IV: Invited Lectures*, pages 1576–1602. World Scientific, 2010.

[58] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

[59] Mojtaba Sahraee-Ardakan, Tung Mai, Anup B. Rao, Ryan A. Rossi, Sundeep Rangan, and Alyson K. Fletcher. Asymptotics of ridge regression in convolutional models. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 9265–9275. PMLR, 2021.

[60] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

[61] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola. A generalized representer theorem. In *International conference on computational learning theory*, pages 416–426. Springer, 2001.

[62] Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. *arXiv preprint arXiv:2201.11990*, 2022.

[63] Hossein Taheri, Ramtin Pedarsani, and Christos Thrampoulidis. Sharp asymptotics and optimal performance for inference in binary models. In *International Conference on Artificial Intelligence and Statistics*, pages 3739–3749. PMLR, 2020.

[64] Greg Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *arXiv preprint arXiv:1902.04760*, 2019.

[65] Greg Yang. Wide feedforward or recurrent neural networks of any architecture are gaussian processes. *arXiv*, 2019.

[66] Lenka Zdeborová and Florent Krzakala. Statistical physics of inference: Thresholds and algorithms. *Advances in Physics*, 65(5):453–552, 2016.

[67] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[68] Yaoyu Zhang, Zhi-Qin John Xu, Tao Luo, and Zheng Ma. A type of generalization error induced by initialization in deep neural networks. In *Mathematical and Scientific Machine Learning*, pages 144–164. PMLR, 2020.