Optimal Nonlinearities Improve Generalization Performance of Random Features

Samet Demir sdemir20@ku.edu.tr
Machine Learning and Information Processing Group, KUIS AI Center, Koç University, Turkey

Zafer Doğan zdogan@ku.edu.tr
Machine Learning and Information Processing Group, KUIS AI Center, Koç University, Turkey
Electrical and Electronics Engineering, Koç University, Turkey

Editors: Berrin Yanıkoğlu and Wray Buntine

Abstract

Random feature model with a nonlinear activation function has been shown to perform asymptotically equivalent to a Gaussian model in terms of training and generalization errors. Analysis of the equivalent model reveals an important yet not fully understood role played by the activation function. To address this issue, we study the "parameters" of the equivalent model to achieve improved generalization performance for a given supervised learning problem. We show that acquired parameters from the Gaussian model enable us to define a set of optimal nonlinearities. We provide two example classes from this set, e.g., second-order polynomial and piecewise linear functions. These functions are optimized to improve generalization performance regardless of the actual form. We experiment with regression and classification problems, including synthetic and real (e.g., CIFAR10) data. Our numerical results validate that the optimized nonlinearities achieve better generalization performance than widely-used nonlinear functions such as ReLU. Furthermore, we illustrate that the proposed nonlinearities also mitigate the so-called double descent phenomenon, which is known as the non-monotonic generalization performance regarding the sample size and the model size.

Keywords: Random feature model; generalization performance; activation functions; Gaussian equivalence conjecture; universality; double descent phenomenon

1. Introduction

We consider a supervised learning problem of fitting a collection of training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$ using the random feature model (RFM) (Rahimi and Recht, 2007):

$$\hat{y}_{RF} := \boldsymbol{\omega}^T \sigma(\mathbf{F}^T \mathbf{x}), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is an input vector, $\mathbf{F} \in \mathbb{R}^{n \times k}$ is a random feature matrix drawn from some matrix ensembles, $\sigma : \mathbb{R} \to \mathbb{R}$ is an element-wise nonlinear mapping (i.e., activation function), and $\boldsymbol{\omega} \in \mathbb{R}^k$ is called the weight vector. Note that, in the RFM, the feature matrix $\mathbf{F}$ is fixed after random sampling. As such, one can also view this model as a two-layer network with k hidden neurons where the first layer weights are frozen in the learning process. Hence, only the weight vector $\boldsymbol{\omega}$ is learned by solving the following optimization problem:

$$\hat{\boldsymbol{\omega}} = \underset{\boldsymbol{\omega} \in \mathbb{R}^k}{\text{argmin}} \frac{1}{m} \sum_{i=1}^{m} l(y_i, \boldsymbol{\omega}^T \sigma(\mathbf{F}^T \mathbf{x}_i)) + \frac{\lambda}{2} ||\boldsymbol{\omega}||_2^2, \tag{2}$$

where $l : \mathbb{R}^2 \to \mathbb{R}$ is a (convex) loss function, and $\lambda > 0$ is a regularization constant. Then, we measure the performance of the learning process via the generalization error defined as

$$\underset{(\mathbf{x},y)\sim D}{\mathbb{E}} l(y, \hat{\boldsymbol{\omega}}^T \sigma(\mathbf{F}^T \mathbf{x})), \tag{3}$$

where $\hat{\boldsymbol{\omega}}$ denotes the optimal solution of (2), and D refers to the data distribution.

The RFM was initially proposed to approximate kernel methods with linear models (Rahimi and Recht, 2007). This approach allowed for faster and more scalable computations, making kernel methods feasible for larger datasets. Since then, random features, in general, have been used under different settings (Brault et al., 2016; Nishio and Yamane, 2019) including multiple kernel learning (Bektaş et al., 2022). Furthermore, the RFM is also shown to outperform traditional linear models while still being more efficient than kernel methods. Therefore, the RFM has been applied to various problems in machine learning and signal processing (Liu et al., 2022).

The RFM has received considerable interest in the last few years mainly due to its simplicity, empirical performance, and connection to overparameterized neural networks (Bach, 2017; Jacot et al., 2018). Some of that attention has been directed toward characterizing the generalization performance of this model in high-dimensional regimes (Mei and Montanari, 2022; Ba et al., 2020; Mel and Pennington, 2022). In this regard, the asymptotic equivalence of the RFM and a Gaussian model have been observed and validated empirically in several papers in the literature (Mei and Montanari, 2022; Montanari et al., 2019; Gerace et al., 2020).

Theoretically, the asymptotic equivalence of the RFM and the Gaussian model has been initially predicted (Thrampoulidis et al., 2015) by using a non-rigorous method from statistical physics (known as the replica method (Mézard et al., 1986)). Recently, these predictions have been verified rigorously, and an underlying universality theorem for the RFM has been proved (Hu and Lu, 2023). Furthermore, using the equivalent model, the performance of the RFM in the overparameterized regime is precisely characterized by the Gaussian min-max theorem (Dhifallah and Lu, 2020).

Our work is based on the asymptotically equivalent Gaussian formulation of the RFM (1) provided as follows:

$$\hat{y}_{\mathcal{N}} := \boldsymbol{\omega}^T(\mu_0 \mathbf{1} + \mu_1 \mathbf{F}^T \mathbf{x} + \mu_2 \mathbf{z}), \tag{4}$$

where $\mathbf{1}$ is an all-one vector and $\mathbf{z} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \mathbf{I}_k)$ is independent of $\mathbf{x}$. Moreover, for a given $\sigma(\cdot)$, the quantities μ_0, μ_1, and μ_2, which we call "mapping parameters", are defined as $\mu_0 = \mathbb{E}[\sigma(z)], \quad \mu_1 = \mathbb{E}[z\sigma(z)]$, and $\mu_2 = (\mathbb{E}[\sigma(z)^2] - \mu_1^2 - \mu_0^2)^{1/2}$ where $z \sim \mathcal{N}(0, 1)$. Specifically, training and generalization performances (errors) for (1) are asymptotically equivalent to those for (4). This holds for some reasonable feature matrix $\mathbf{F}$, loss function $l(\cdot, \cdot)$, and nonlinearity $\sigma(\cdot)$. In addition, it also requires standard Gaussian inputs and labels that are generated using a typical teacher-student framework, which is a common technique in the theoretical literature (Loureiro et al., 2022; Wang and Abernethy, 2021; Cao et al., 2022).

In this work, we focus on the equivalent model (4) to study the effects of the nonlinearity in the RFM (1) to achieve improved generalization performance over a set of learning problems. In the literature, the effects of regularization (Nakkiran et al., 2021) and data-dependent feature selection (Shahrampour et al., 2018) on the generalization performance of RFM have been studied. Furthermore, the effects of different nonlinear activation functions in the RFM have been observed before (Dhifallah and Lu, 2020; Hu and Lu, 2023) not to mention a vast literature (Agostinelli et al., 2015; Ramachandran et al., 2018; Nwankpa et al., 2018) on activation functions in general. However, the precise characterization of the nonlinearity over the generalization performance of the RFM is still required. In a concurrent work on "optimal activation functions", a combination of generalization error and sensitivity of the activation functions is minimized under a regression setting with a synthetic data model (Wang and Bento, 2023). On the other hand, we focus on minimizing the generalization error under various settings with synthetic and real data.

Another line of related work is about the so-called "double descent" phenomenon (Belkin et al., 2019, 2018) observed in the generalization performance of the RFM. The phenomenon is characterized by the non-monotonic decrease of the generalization error regarding the model complexity. In this sense, an optimal choice of the regularization constant λ in (2) has been shown to improve the generalization performance and mitigate the double descent phenomenon (Nakkiran et al., 2021). For the RFM, the equivalent formulation (4) suggests that the optimal nonlinearities can be linked to the optimal choice of the regularization constant.

Overall, the main contributions of this work are two-fold: First, we improve the generalization performance of the RFM by using optimal nonlinearities derived from the mapping parameters of the asymptotically equivalent Gaussian formulation. Second, we show that the proposed nonlinearities also mitigate the double descent phenomenon.

2. Optimal Nonlinearities

Comparing the RFM and the equivalent formulation, the effect of the nonlinearity in the former only appears as the mapping parameters ($\mu_{0,1,2}$) in the latter. As such, we first replace the RFM with the equivalent model and optimize the mapping parameters to minimize the generalization error. Next, we show that for a given set of optimized mapping parameters (optimal in the sense of generalization performance), it is possible to define a set of optimal nonlinearities that perform asymptotically equivalent. Finally, we provide two example classes of such nonlinear functions to be replaced with σ in the RFM to illustrate the performance of the "optimal" nonlinearities.

2.1. Assumptions

Our results are based on the following technical assumptions for the equivalence of the RFM and the Gaussian model (Hu and Lu, 2023):

1. The data vectors $\{\mathbf{x}_i\}_{i=1}^{m}$ are drawn independently from $\mathcal{N}(0, \mathbf{I}_n)$.

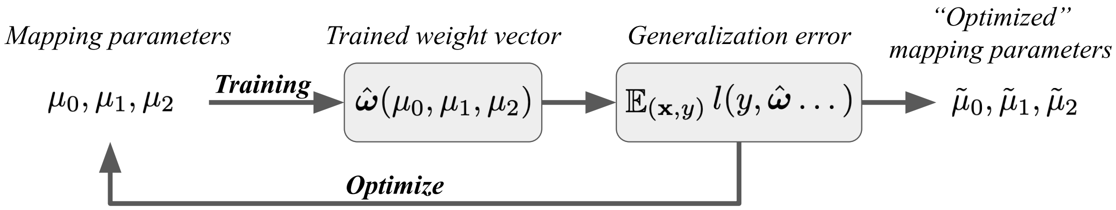

Mapping parameters
μ_0, μ_1, μ_2
Training
Trained weight vector
$\hat{\boldsymbol{\omega}}(\mu_0, \mu_1, \mu_2)$
Generalization error
$\mathbb{E}_{(\mathbf{x},y)}\, l(y, \hat{\boldsymbol{\omega}} \dots)$
"Optimized" mapping parameters
$\tilde{\mu}_0, \tilde{\mu}_1, \tilde{\mu}_2$
Optimize

Figure 1: Overview - minimizing the generalization error w.r.t. the mapping parameters

2. The number of samples m, the input dimension n, and the feature dimension k go to infinity with finite ratios $m/n > 0$ and $k/m > 0$.
3. The labels are generated using a typical teacher-student setup described in the Sec. 3. Furthermore, the unknown signal $\boldsymbol{\xi} \in \mathbb{R}^n$ used in the teacher model is independent of the feature matrix $\mathbf{F}$, where $\|\boldsymbol{\xi}\| = \rho$ is known.
4. The nonlinear function $\sigma(\cdot)$ satisfies the conditions that $\mu_1 = E[z\sigma(z)] > 0$ and $0 < \mu_2 = E[\sigma(z)^2] < \infty$, where $z \sim \mathcal{N}(0, 1)$.
5. The loss function $l(\cdot, \cdot)$ is a proper convex function in $\mathbb{R}^2$.
6. The columns of the feature matrix $\mathbf{F}$ are independent and identically drawn from a Gaussian distribution with zero mean and covariance matrix $\frac{1}{n}\mathbf{I}_n$.

2.2. Optimizing the Mapping Parameters

For a given set of training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, loss function l, feature matrix $\mathbf{F}$, and the regularization constant λ, we learn the *optimal mapping parameters*, i.e., $\tilde{\mu}_{0,1,2}$ as illustrated in Fig. 1. First, we define the trained weight vector $\hat{\boldsymbol{\omega}}$ as a function of the mapping parameters:

$$\hat{\boldsymbol{\omega}}(\mu_0, \mu_1, \mu_2) := \underset{\boldsymbol{\omega} \in \mathbb{R}^k}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} l(y_i, \boldsymbol{\omega}^T(\mu_0 \mathbf{1} + \mu_1 \mathbf{F}^T \mathbf{x}_i + \mu_2 \mathbf{z})) + \frac{\lambda}{2} ||\boldsymbol{\omega}||_2^2, \tag{5}$$

where $\mathbf{1}$ is an all-one vector and $\mathbf{z} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \mathbf{I}_k)$ as in the definition of the equivalent Gaussian model. Then, we can pose the objective of minimizing the generalization error with respect to the mapping parameters:

$$\tilde{\mu}_{0,1,2} = \underset{\mu_0 \in \mathbb{R}, \mu_1, \mu_2 > 0}{\arg\min} \underset{(\mathbf{x},y) \sim D, \mathbf{z} \sim \mathcal{N}(0, \mathbf{I}_k)}{\mathbb{E}} l(y, \hat{\boldsymbol{\omega}}(\mu_0, \mu_1, \mu_2)^T(\mu_0 \mathbf{1} + \mu_1 \mathbf{F}^T \mathbf{x} + \mu_2 \mathbf{z})). \tag{6}$$

(6) specifies our general objective. We observe that μ_0 plays the role of the so-called "bias" term. Therefore, we may drop μ_0 from the objective in favor of a bias term b:

$$\tilde{\mu}_1, \tilde{\mu}_2 = \underset{\mu_1, \mu_2}{\arg\min} \underset{(\mathbf{x},y) \sim D, \mathbf{z} \sim \mathcal{N}(0, \mathbf{I})}{\mathbb{E}} l(y, \hat{\boldsymbol{\omega}}(\mu_1, \mu_2)^T(\mu_1 \mathbf{F}^T \mathbf{x} + \mu_2 \mathbf{z}) + \hat{b}(\mu_1, \mu_2)), \tag{7}$$

where $\hat{\boldsymbol{\omega}}(\mu_1, \mu_2)$ and $\hat{b}(\mu_1, \mu_2)$ are obtained, similar to (5), as follows:

$$\hat{\boldsymbol{\omega}}(\mu_1, \mu_2), \hat{b}(\mu_1, \mu_2) := \arg\min_{\boldsymbol{\omega}, b} \frac{1}{m} \sum_{i=1}^{m} l(y_i, \boldsymbol{\omega}^T(\mu_1 \mathbf{F}^T \mathbf{x}_i + \mu_2 \mathbf{z}_i) + b) + \frac{\lambda}{2} ||\boldsymbol{\omega}||_2^2, \tag{8}$$

where $\mathbf{z} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \mathbf{I}_k)$, $\mathbf{z}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \mathbf{I}_k)$ for $i \in \{1, \ldots, m\}$ and $\tilde{\mu}_0 = \hat{b}(\tilde{\mu}_1, \tilde{\mu}_2) / \hat{\boldsymbol{\omega}}(\tilde{\mu}_1, \tilde{\mu}_2)^T \mathbf{1}$. For the experimental results, (7) is solved using grid search, and we use the closed-form solution for (8) whenever it is available (depending on the loss). Note that for each feature matrix $\mathbf{F}$, (7) is solved again. Therefore, the optimal mapping parameters are specific to the feature matrix, dataset, and loss function. The complete algorithm for the optimization is provided in Sec. 3.1.1.

2.3. Mapping Parameters to Nonlinear Mappings

Given mapping parameters $(\mu_{0,1,2})$, we next define a set of nonlinear mappings:

$$\mathcal{F}_{\mu_{0,1,2}} = \left\{ \sigma_f : \mathbb{R} \to \mathbb{R} \;\middle|\; \begin{array}{l} \mu_0 = \mathbb{E}_z[\sigma_f(z)], \\ \mu_1 = \mathbb{E}_z[z \sigma_f(z)], \\ \mu_2 = \sqrt{\mathbb{E}_z[\sigma_f(z)^2] - \mu_1^2 - \mu_0^2} \end{array} \right\}, \tag{9}$$

where $z \sim \mathcal{N}(0, 1)$. Note that constraints in (9) do not determine a unique mapping without further assumptions on σ_f. Instead, the constraints specify a set of performance-wise equivalent mappings. Moreover, any linear function cannot satisfy these constraints when $\mu_2 \neq 0$. Hence, (9) defines a set of nonlinear mappings.

Here, we provide two example classes of functions that enable us to uniquely determine the nonlinear mappings assuming specific function families. First, we consider a naive second-order polynomial with three coefficients to be determined. Using the optimal mapping parameters from (7) and the set of equations in (9), we obtain the following form:

$$\sigma_{polynomial}(z) = \tilde{\mu}_0 + \tilde{\mu}_1 z + \frac{\tilde{\mu}_2}{\sqrt{2}} \left(z^2 - 1 \right), \tag{10}$$

which is equivalent to the orthonormal Hermite polynomial expansion with the three mapping parameters.

Second, we consider a nonlinear mapping in piecewise linear form as a generalization of (leaky) ReLU:

$$\sigma_{piecewise}(z) = \begin{cases} az + c, & \text{if } z \geq 0, \\ bz + c, & \text{otherwise}, \end{cases} \tag{11}$$

where the three parameters a, b, c are to be determined. By solving (9), we obtain the parameters as $a = \tilde{\mu}_1 + \sqrt{\frac{\pi}{\pi-2}} \tilde{\mu}_2$, $b = \tilde{\mu}_1 - \sqrt{\frac{\pi}{\pi-2}} \tilde{\mu}_2$, and $c = \tilde{\mu}_0 - \sqrt{\frac{2}{\pi-2}} \tilde{\mu}_2$.

The proposed nonlinearities can be directly used in the RFM, providing improved generalization performance by definition. Unlike widely used nonlinear activation functions such as ReLU, and Softplus, the proposed ones are “task-optimized” in the sense that it takes the data, the loss, and the regularization into account. As such, the results can be extended to various other applications for better generalization performance.

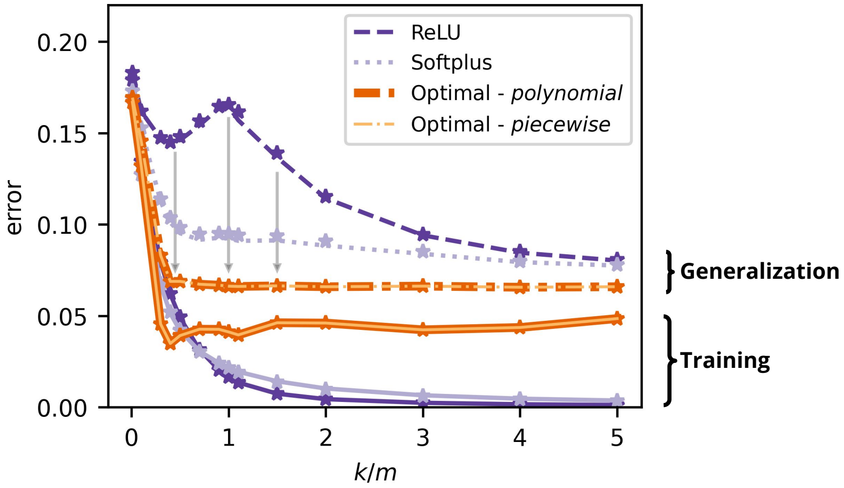


Figure 2: Regression (Sec. 3.2) - training (solid lines) and generalization (dashed lines) errors are provided for the RFM and the equivalent Gaussian model with ReLU, Softplus, and the proposed optimal nonlinearities. $\star$ denotes the error for the equivalent Gaussian model (4). The downward arrows show that the generalization error has been improved across the full range of the model complexities we consider. The numerical results are averaged over 50.

Before moving to the results, we want to mention a few points about the "optimal" nonlinearities. Regardless of their form, they all provide asymptotically equivalent yet improved generalization errors. However, the optimal mapping parameters reveal more information on optimal nonlinearity. For example, a non-zero $\tilde{\mu}_0$ states that the optimal nonlinearity cannot be an odd function, although this assumption is often used in the simplification of the theoretical development of the universality laws for learning with the RFM (Hu and Lu, 2023). $\tilde{\mu}_1$ plays an important role controlling the scale of $\hat{\boldsymbol{\omega}}$. Hence, the optimal nonlinearities remove the need for tuning optimal λ. $\tilde{\mu}_2$ controls the noise level of the RFM. Furthermore, we can show that (11) can be reduced to (a scaled-version of) widely-used ReLU only if the optimal mapping parameters satisfy a special relationship such that $\tilde{\mu}_0^2 = \tilde{\mu}_1^2(\frac{\pi}{2}) = \tilde{\mu}_2^2(\frac{2}{\pi-2})$. It suggests that ReLU will naturally arise as the optimal nonlinearity for a specific set of learning problems depending on the set of training samples, loss function, $\mathbf{F}$, and λ.

3. Results

In this section, we provide experimental results showing that generalization performance is improved when the proposed set of nonlinearities is used in the RFM. To illustrate our results, we consider two applications: nonlinear regression and binary classification problems with different loss functions on synthetic data. We also consider binary classification on CIFAR10 (Krizhevsky et al., 2009) and Tiny ImageNet (Wu et al., 2017; Deng et al., 2009) datasets. All results present the corresponding training and generalization performances of the underlying the RFM with specific choices of the nonlinear activation functions and the proposed optimal nonlinearities as a function of model complexity (k/m).

3.1. Experimental Setup

In this section, we provide the full details of our experimental setup. Specifically, we discuss the optimization algorithm, the data model, the feature matrix, the dimensions, and the regularization constant, which are essential components of the experimental design.

3.1.1. The Optimization Algorithm

To optimize the mapping parameters, we use grid search, which is a widely used technique for hyperparameter optimization in machine learning (Bergstra and Bengio, 2012). We have two reasons to use grid search: 1) our optimization objective is observed to be nonconvex, which limits the application of gradient-based optimization techniques; 2) the number of parameters to search for is two (μ_1, μ_2), which makes the grid search computationally cheap. One might also use more advanced approaches, such as randomized search, Bayesian optimization, or evolutionary algorithms (Bergstra et al., 2011). However, grid search is simple and efficient enough to show our claims in this paper. The complete algorithm is provided in Algorithm 1. We first calculate the "bias term" from the labels and initialize the variables for the grid search. For each (μ_1, μ_2) pair in the search grid, we train the weight vector $\hat{\boldsymbol{\omega}}$ and calculate generalization error with $\hat{\boldsymbol{\omega}}$. If there is an improvement in the generalization error, we update the parameters. Finally, $\tilde{\mu}_0$ is calculated. Note that there is a difference between synthetic and real data experiments. We use the error calculated on the validation set instead of the generalization error for the real data, while the test set is reserved for calculating the generalization error.

3.1.2. The Data model

We consider the classical teacher-student framework (Loureiro et al., 2022) for the numerical simulations. Specifically, we assume the following data generation model:

$$\mathbf{x}_i \sim \mathcal{N}(0, \mathbf{I}_n),\ y_i = \psi(\boldsymbol{\xi}^T \mathbf{x}_i) + \Delta \epsilon_i, \forall i \in \{1, \cdots, m\}, \tag{12}$$

where $\mathbf{x}_i \in \mathbb{R}^n$ and $y_i \in \mathbb{R}$ are input-output pairs, $\boldsymbol{\xi} \in \mathbb{R}^n$ is an unknown fixed vector, $\epsilon_i \sim \mathcal{N}(0, 1)$, $\Delta \geq 0$ is a constant controlling the noise level, and $\psi : \mathbb{R} \to \mathbb{R}$ is a nonlinear function. Furthermore, we use a fixed $\boldsymbol{\xi}$ with unit norm that is sampled from $\mathcal{N}(0, (1/n)\mathbf{I}_n)$ for the experiments.

3.1.3. The Feature Matrix and The Dimensions

Without loss of generality, k columns of the feature matrix $\mathbf{F}$ are drawn independently from $\mathcal{N}(0, (1/n)\mathbf{I}_n)$ in our experiments. Note that results can be extended to alternative feature matrix $\mathbf{F}$ that satisfy the regularity assumptions introduced in (Hu and Lu, 2023). The number of samples m, the feature dimension k, and the input dimension n are set to satisfy the assumptions in Sec. 2.1.

Algorithm 1: The algorithm for the optimization

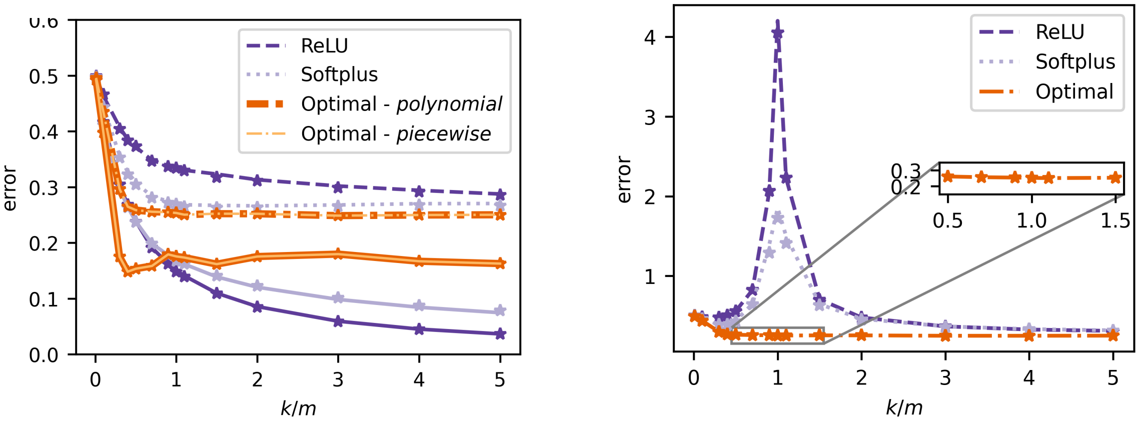
```
Data: ${(\mathbf{x}_i, y_i)}_{i=1}^m$, SearchGrid
Result: $\tilde{\mu}_0, \tilde{\mu}_1, \tilde{\mu}_2$
$\hat{b} \leftarrow \frac{1}{m}\sum_{i=1}^m y_i$;
$\tilde{\mu}_1 \leftarrow 1$;
$\tilde{\mu}_2 \leftarrow 0$;
$\hat{\boldsymbol{\omega}}_{best} \leftarrow \mathbf{1}$;
$best\_error \leftarrow \infty$;
for $(\mu_1, \mu_2)$ in SearchGrid do
    $\hat{\boldsymbol{\omega}} \leftarrow \underset{\boldsymbol{\omega}\in\mathbb{R}^k}{\operatorname{argmin}} \frac{1}{m}\sum_{i=1}^m l(y_i, \boldsymbol{\omega}^T(\mu_1\mathbf{F}^T\mathbf{x}_i + \mu_2\mathbf{z}_i) + \hat{b}) + \frac{\lambda}{2}||\boldsymbol{\omega}||_2^2$ ;    // Eq. (8)
    if Synthetic Data then
        $error \leftarrow \underset{(\mathbf{x},y)\sim D, \mathbf{z}\sim\mathcal{N}(0,\mathbf{I})}{\mathbb{E}} l(y, \hat{\boldsymbol{\omega}}^T(\mu_1\mathbf{F}^T\mathbf{x} + \mu_2\mathbf{z}) + \hat{b})$ ;    // Eq. (7)
    else
        error ← Calculate Error on the Validation Set ;    // Eq. (7) on real data
    end
    if $error < best\_error$ then
        $\tilde{\mu}_1 \leftarrow \mu_1$;
        $\tilde{\mu}_2 \leftarrow \mu_2$;
        $\hat{\boldsymbol{\omega}}_{best} \leftarrow \hat{\boldsymbol{\omega}}$;
        $best\_error \leftarrow error$;
    end
end
$\tilde{\mu}_0 \leftarrow \hat{b}/\hat{\boldsymbol{\omega}}_{best}\mathbf{1}$;
```

3.1.4. The Regularization Constant

The choice of the regularization constant λ in (2) is known to play a key role in generalization performance, and the optimal λ has been shown to mitigate the double descent behavior (Nakkiran et al., 2021). Essentially, the regularization constant controls the scale of the weight vector $\boldsymbol{\omega}$. In the equivalent Gaussian model (4), we can achieve the same control over the scale of $\boldsymbol{\omega}$ by scaling the mapping parameters $(\mu_{0,1,2})$. This observation has two direct consequences. First, we do not need to tune the regularization constant λ when using the optimal nonlinear mappings. Second, similar to the effect of optimal regularization (Nakkiran et al., 2021), we show that the optimized nonlinear mappings also mitigate the double descent phenomenon.

3.2. Regression

Consider a nonlinear regression problem where the labels $\{y_i\}_{i=1}^m$ are generated according to (12) with $\psi(z) = \max(z, 0)$, and $\Delta = 0.05$. Moreover, we assume that the loss function is the squared loss: $l(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$. Note that this is a standard setting for typical regression problems. Also, we set $n = 400$, $m = 1200$, and $\lambda = 10^{-2}$. Fig. 2 shows the training and the generalization performance of learning with the RFM and the equivalent



(a) Classification (Sec. 3.3)

(b) Classification with smaller λ

Figure 3: Numerical simulations - training (solid lines) and generalization (dashed lines) errors are provided for the RFM and the equivalent Gaussian model with ReLU, Softplus, and the proposed optimal nonlinearities. $\star$ denotes the error for the equivalent Gaussian model (4). The numerical results are averaged over 50.

Gaussian model for the regression problem (similarly Fig. 3(a) for the classification problem below). The numerical results show that for a given activation function (ReLU or Softplus), the two model's errors are in excellent agreement, aligned with the previous results in the literature (Dhifallah and Lu, 2020; Hu and Lu, 2023). Note that it was also shown that the training and the generalization errors of the Gaussian formulation converge in probability to deterministic limiting functions as the dimensions m, n, k tend to infinity. Moreover, the parameters of the limiting functions can be determined by some fixed point equations (Dhifallah and Lu, 2020). For brevity, we omit discussions of the analytical predictions. Instead, we focus on improving the generalization performance of the RFM using the proposed optimal nonlinearities. First, the two proposed nonlinear mappings, i.e., $\sigma_{polynomial}$, and $\sigma_{piecewise}$, provide equivalent performance on training and generalization errors by definition. Second, they outperform Softplus and ReLU activation functions in the sense that they provide a lower generalization error for the range of model complexity we consider. This result confirms the key role played by the activation function and also validates the improved generalization performance by using optimal nonlinearities for the RFM.

3.3. Classification

As a second example, we consider a binary classification problem where the labels $\{y_i\}_{i=1}^m$ are generated according to (12) with $\psi(z) = \text{sign}(z)$ and $\Delta = 0$. Here, we use $n = 200$, $m = 600$ and $\lambda = 10^{-1}$ for the experiments. In Fig. 3(a), the training and generalization errors for the RFM and the equivalent Gaussian model are plotted for the classification problem with the squared loss. Similar to the regression case, the proposed nonlinear mappings ($\sigma_{polynomial}$, and $\sigma_{piecewise}$) provide equivalent performance while they provide improved generalization performance compared to Softplus and ReLU.

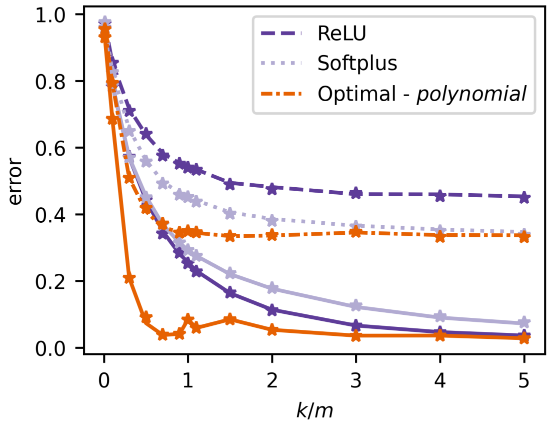


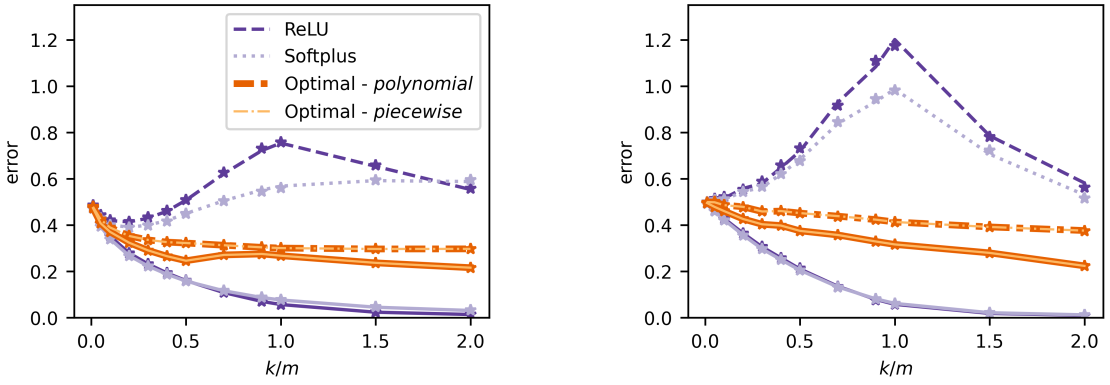
Figure 4: Classification with hinge loss - training (solid lines) and generalization (dashed lines) errors are provided for the RFM and the equivalent Gaussian model with ReLU, Softplus, and the proposed optimal nonlinearities. $\star$ denotes the error for the equivalent Gaussian model (4). The numerical results are averaged over 20.

3.3.1. Double Descent Phenomenon

The generalization error is known to follow a U-shaped curve for small model complexity until reaching a peak known as the interpolation threshold. After the peak, the generalization error decreases monotonically as a function of the model complexity. This behavior is known as the "double descent" phenomenon (Belkin et al., 2019, 2018). To highlight it, we consider binary classification with the squared loss and set $\lambda = 10^{-4}$, $n = 400$, $m = 1200$. In Fig. 3(b), while ReLU and Softplus experience a steep double descent with a peak at $k/m = 1$, optimal nonlinear mappings lead to a monotonically decreasing generalization error. Specifically, comparing the generalization performance in Fig. 3(a) and Fig. 3(b) for ReLU or Softplus, the results confirm that optimal regularization plays a key role in mitigating the double descent phenomenon, matching the results stated in (Nakkiran et al., 2021). However, regardless of the choice of regularization constant, the optimized nonlinear mappings always achieve a monotonically decreasing generalization error.

3.3.2. Impact of Loss Functions:

To see the impact of different loss functions, we next consider the hinge loss: $l(y, \hat{y}) = \max(1 - y\hat{y}, 0)$. Fig. 4 shows the training and generalization errors for the classification case with the hinge loss and $\lambda = 10^{-1}$. Note that we use a smaller search grid for this plot since there is no closed-form solution for $\hat{\boldsymbol{\omega}}$ in this case. The optimal nonlinearity provides lower training and generalization errors compared to Softplus and ReLU. Unlike the results in Fig. 3(a), the optimal nonlinearity achieves improved generalization performance without deteriorating the training performance. This result suggests that an optimized loss function can improve the two errors without worsening the other.



(*a*) CIFAR10

(*b*) Tiny ImageNet

Figure 5: Binary image classification - training (solid lines) and generalization (dashed lines) errors are illustrated for the RFM and the equivalent Gaussian model (denoted by $\star$). The average of 50 Monte Carlo trials is plotted.

3.4. Real Image Classification

The equivalence of the RFM and the Gaussian model is only valid under assumptions in 2.1. For a real dataset, such assumptions will not hold in general. However, these requirements can be satisfied (at least partially) after proper preprocessing. In this setting, we propose to use a pretrained normalizing flow model (Rezende and Mohamed, 2015) (e.g., RealNVP (Dinh et al., 2016)) for the preprocessing. Normalizing Flow models are invertible generative models that consist of a sequence of invertible nonlinear transformations. Similar to the popular generative adversarial networks (GAN), the latent space has the standard Gaussian distribution. By taking advantage of the invertibility and the Gaussian latent space, we map real images to Gaussian-distributed latent space. We use real labels in contrast to the teacher-student framework. Another important point regarding our real data experiments is that we optimize the mapping parameters on a validation set and use the test set to calculate the generalization error.

To extend our results to real data, we consider binary image classification (the first class vs. the second class) on CIFAR10 and Tiny ImageNet. Specifically, for CIFAR10, we pick 2000 samples and 500 samples from each class for the training set and validation sets, respectively. We use the complete test set (1000 samples for each class). For Tiny ImageNet, we use the complete training (500 samples for each class) and validation (50 samples for each class) sets. The validation set is used as the test set since the available test set is not labeled. Furthermore, the training set is split into training (400 samples for each class) and validation (100 samples for each class) sets. We convert the labels such that $y_i \in \{-1, 1\}$. In order to achieve the asymptotic equivalence of the RFM and the Gaussian model on real data, the inputs are preprocessed with a pretrained RealNVP model to map them to the latent space, which has Gaussian distribution (see Appendix A for details). The RealNVP model trained in (Goldt et al., 2022) is used for CIFAR10, while we trained another RealNVP model for Tiny ImageNet. We use the squared loss and set $\lambda = 10^{-2}$. Fig. 5 illustrates the results on the real data. The results validate that the proposed optimal nonlinearities achieve improved generalization performance and also mitigate the double descent phenomenon. Furthermore, we provide accuracy plots for the same setting in Appendix D, which suggests our results on the error metrics can be extended to other metrics such as accuracy.

Note that we observe an improved generalization performance with all real image classification problems we consider, provided that a proper normalizing flow model is available. As far as we know, this proof-of-concept result is the first to demonstrate the practical usage of equivalent Gaussian models of RFM to improve the generalization performance with real data. However, such a normalizing flow model might not always be available in practice, which is a current limitation of this work, and alternative solutions are left for future work.

4. Conclusion

In this work, we studied the role played by the nonlinear activation function in the random feature model, which has been shown to perform asymptotically equivalent to a linear Gaussian model. By studying the mapping parameters of the equivalent model, we define a set of optimal nonlinearities that provide equivalent yet improved generalization performance. The proposed nonlinearities achieve better generalization performance than widely-used nonlinear functions. Additionally, we show that the proposed nonlinearities also achieve a monotonic decrease in generalization error. The results are valid for a large family of feature matrices, activation functions, and convex loss functions. Experimental results on classification and regression problems validate the effectiveness of the proposed optimal nonlinearities in achieving better generalization performance.

Acknowledgments

We acknowledge that this work was supported in part by TUBITAK 2232 International Fellowship for Outstanding Researchers Award (No. 118C337) and an AI Fellowship provided by Koç University & İş Bank Artificial Intelligence (KUIS AI) Research Center.

References

Forest Agostinelli, Matthew D. Hoffman, Peter J. Sadowski, and Pierre Baldi. Learning activation functions to improve deep neural networks. In *International Conference on Learning Representations (Workshop)*, 2015.

Jimmy Ba, Murat Erdogdu, Taiji Suzuki, Denny Wu, and Tianzong Zhang. Generalization of two-layer neural networks: An asymptotic viewpoint. In *International Conference on Learning Representations (ICLR)*, 2020.

Francis Bach. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research*, 18(21):1–38, 2017.

Ayyüce Begüm Bektaş, Çiğdem Ak, and Mehmet Gönen. Fast and interpretable genomic data analysis using multiple approximate kernel learning. *Bioinformatics*, 38 (Supplement_1):i77–i83, 2022.

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, 2018.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias-variance trade-off. In *Proc. Natl. Acad. Sci.*, volume 116, pages 15849–15854, 2019. doi: 10.1073/pnas.1903070116.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(2), 2012.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems*, 24, 2011.

Romain Brault, Markus Heinonen, and Florence Buc. Random fourier features for operator-valued kernels. In *Asian Conference on Machine Learning*, pages 110–125. PMLR, 2016.

Jiezhang Cao, Jincheng Li, Xiping Hu, Xiangmiao Wu, and Mingkui Tan. Towards interpreting deep neural networks via layer behavior understanding. *Machine Learning (ACML 2021 - Journal Track)*, 111(3):1159–1179, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comp. Vis. Patt. Recogn*, pages 248–255, 2009.

Oussama Dhifallah and Yue M Lu. A precise performance analysis of learning with random features. *arXiv preprint arXiv:2008.11904*, 2020.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

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, 2020.

Sebastian Goldt, Bruno Loureiro, Galen Reeves, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. The gaussian equivalence of generative models for learning with shallow neural networks. In *Math. Sci. Mach Learn.*, pages 426–471, 2022.

Hong Hu and Yue M Lu. Universality laws for high-dimensional learning with random features. *IEEE Trans. Inf. Theory*, 69(3):1932–1964, Mar. 2023.

Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, pages 8580–8589, 2018.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Fanghui Liu, Xiaolin Huang, Yudong Chen, and Johan A. K. Suykens. Random features for kernel approximation: A survey on algorithms, theory, and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(10):7128–7148, Oct. 2022. doi: 10.1109/TPAMI.2021.3097011.

Bruno Loureiro, Cedric Gerbelot, Hugo Cui, Sebastian Goldt, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. Learning curves of generic features maps for realistic datasets with a teacher-student model. *J. Stat. Mech. Theory Exp.*, 2022(11):114001, Nov. 2022. doi: 10.1088/1742-5468/ac9825.

Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Commun. Pure Appl. Math.*, 75(4): 667–766, 2022.

Gabriel Mel and Jeffrey Pennington. Anisotropic random feature regression in high dimensions. In *International Conference on Learning Representations*, 2022.

M. Mézard, G. Parisi, and M.A. Virasoro. *Spin Glass Theory And Beyond: An Introduction To The Replica Method And Its Applications.* World Scientific, 1986.

Andrea Montanari, Feng Ruan, Youngtak Sohn, and Jun Yan. The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime. *arXiv preprint arXiv:1911.01544*, 2019.

Preetum Nakkiran, Prayaag Venkat, Sham M. Kakade, and Tengyu Ma. Optimal regularization can mitigate double descent. In *International Conference on Learning Representations (ICLR)*, 2021.

Daichi Nishio and Satoshi Yamane. Random projection in neural episodic control. In *Asian Conference on Machine Learning*, pages 1–15. PMLR, 2019.

Chigozie Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2007.

Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions. In *International Conference on Learning Representations (Workshop)*, 2018.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538, 2015.

Shahin Shahrampour, Ahmad Beirami, and Vahid Tarokh. On data-dependent random features for improved generalization in supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Christos Thrampoulidis, Samet Oymak, and Babak Hassibi. Regularized linear regression: A precise analysis of the estimation error. In *Proc. Conf. Learn. Theory*, pages 1683–1709, 2015.

Jianxin Wang and José Bento. Optimal activation functions for the random features regression model. In *International Conference on Learning Representations*, 2023.

Jun-Kun Wang and Jacob Abernethy. Understanding how over-parametrization leads to acceleration: A case of learning a single teacher neuron. In *Asian Conference on Machine Learning*, pages 17–32. PMLR, 2021.

Jiayu Wu, Qixiang Zhang, and Guoxi Xu. Tiny imagenet challenge. *Technical report*, 2017.

Appendix A. Preprocessing with Normalizing Flow

Normalizing Flow is a likelihood-based generative model that models the data distribution by applying a series of invertible (possibly nonlinear) transformations to multivariate standard Gaussian distribution (Rezende and Mohamed, 2015). A Normalizing Flow models consist of a sequence of invertible mappings as illustrated in Fig. 6. Overall, it learns an invertable mapping $f : \mathbb{R}^n \to \mathbb{R}^n$ with which a data sample can be generated by $\mathbf{x} = f(\mathbf{z})$ where $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I}_n)$. We are interested in the inverse mapping $f^{-1}(\mathbf{x})$ since our goal is to map real data to Gaussian distribution so that the Gaussian equivalence holds. We process the samples of the dataset as $\hat{\mathbf{x}}_i = f^{-1}(\mathbf{x}_i)$. Then, the collection of preprocessed samples $\{(\hat{\mathbf{x}}_i, y_i)\}$ are used to train and test the random feature model.

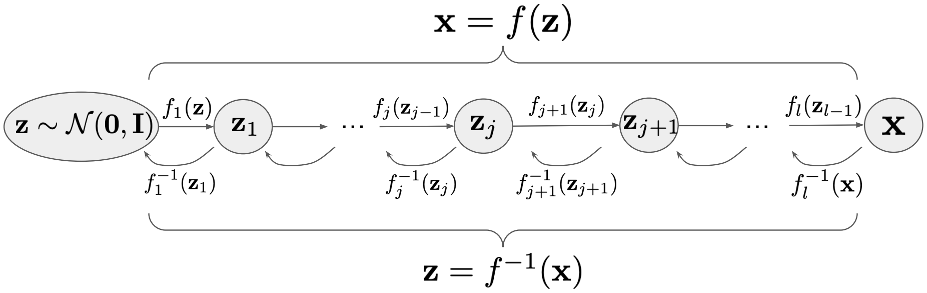
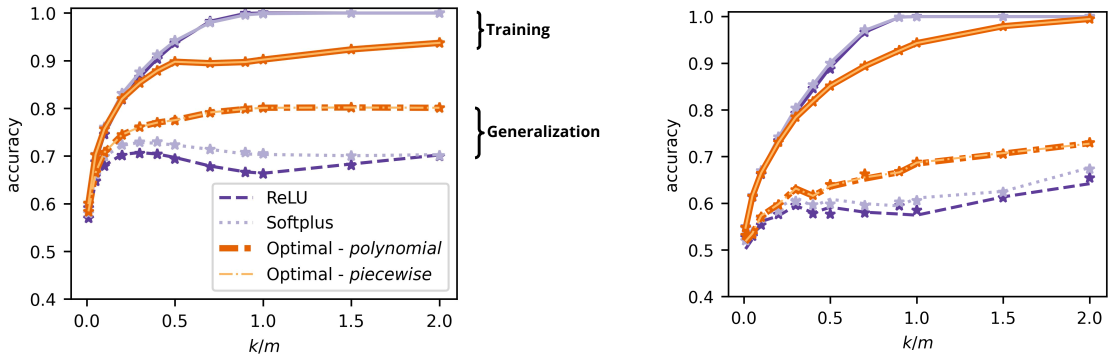


Figure 6: Normalizing flows

Appendix B. CIFAR10 Experiments

We use a subset of the CIFAR10 dataset (Krizhevsky et al., 2009). The CIFAR10 dataset includes colored 32x32 images with labels for ten classes. For the training set, we pick 2000 samples from each of the first two classes to form a binary classification problem (airplane vs. automobile). Also, we convert the labels such that $y_i \in \{-1, 1\}$. For the test set, we use all the test samples for these two classes. Then, we consider a binary classification problem with the squared loss on the 4000 training, 1000 validation, and 2000 test samples. Furthermore, we use RealNVP (Dinh et al., 2016) model that is pretrained and used in (Goldt et al., 2022) as the Normalizing Flow model. The model is pretrained on 46000 samples (including all ten classes) from the CIFAR10 dataset, and it is reported to achieve 3.5 bits/dim on a validation set of 4000 samples (Goldt et al., 2022).



(a) CIFAR10

(b) Tiny ImageNet

Figure 7: Accuracy plots for binary image classification - training (solid lines) and generalization (dashed lines) accuracy values are illustrated for the RFM and the equivalent Gaussian model (denoted by $\star$). The same setting as Fig. 5 is used. The average of 50 Monte Carlo runs is plotted.

Appendix C. Tiny ImageNet Experiments

We also experiment with a small version of ImageNet (Wu et al., 2017; Deng et al., 2009) dataset called “Tiny ImageNet”. This dataset consists of 64x64 colorful images. Similar to our previous settings, we focus on the binary classification problem (goldfish vs. fire salamander) with squared loss when the data is inverted with RealNVP. In this case, the training set has 800 samples; the validation set contains 200 samples, while the test set includes 100. We trained the RealNVP model on the ImageNet dataset using the same code used for CIFAR10 and used it in our experiments. The rest of the setting is the same as CIFAR10 experiments.

Appendix D. Accuracy Plots

In Fig. 7, we provide an additional plot for the accuracy of the random feature model for different model complexity (k/m) values under the same setting as Fig. 5 (in the main text). We observe two points. First, there is an agreement between the accuracy values of the RFM and those of the equivalent Gaussian model. Second, the proposed nonlinearity significantly outperforms the other nonlinearities for the complete range of k/m values we consider. Note that there is an upside-down version of the double descent phenomenon (especially for ReLU) in Fig. 7(a). Although Fig. 7(a) looks smooth, we observe fluctuations in Fig. 7(b), which is related to the fact that there exist a small number of samples for Tiny ImageNet experiments.