To Understand Deep Learning We Need to Understand Kernel Learning

Malo Canu, Yann Chauvard
Theory of Deep Learning / Centrale Supélec

February 4, 2026

Abstract

This report reproduces the findings of the paper "To Understand Deep Learning We Need to Understand Kernel Learning", Belkin et al. (2018), investigating the theory crisis where over-parameterized models achieve zero training error yet generalize well. The authors demonstrate that kernel methods exhibit the same properties. As such, we implement a Kernel Regression framework with both exact interpolation and EigenPro-SGD optimization. We test the properties of kernel machines on two datasets: the standard MNIST digit dataset and the more complex Fashion-MNIST image dataset. Our experiments demonstrate that: (1) kernel methods can interpolate training data while maintaining good performance on the test set, (2) interpolating classifiers are robust to label noise, (3) kernel geometry (smooth vs. spiky) significantly impacts optimization speed and behavior.

The code can be found at this link.

1 Introduction

Deep Neural Networks (DNNs) defy classical learning theory by fitting training data perfectly (including random noise as shown in Zhang et al. (2017)) without catastrophic overfitting. The authors argue that this "Simplicity Bias" is a property of kernel machines in the interpolating regime. The authors also derive a new bound for the RKHS norm of a gaussian kernel interpolating function in the case of a non-zero label noise, as classical bound fail to explain the observations. This bond can be expressed in an exponential fashion as follows:

$$||f||_{\mathcal{H}} > Ae^{Bn^{\frac{1}{d}}}$$

with A and B constants, n the number of samples and d their dimension.

While the original paper tests on many datasets, we focus on **MNIST** and **Fashion-MNIST**. We specifically chose Fashion-MNIST to test the kernel machine's performance on data that is "less clean" and structurally more complex than the simple digits of MNIST, providing a more rigorous test of the model's interpolation capabilities while not being too long to compute.

We focus on three parts:

- **Universality:** Does the generalization performance hold for kernel machines?
- **Algorithm Dynamics:** Comparing the trajectory of EigenPro-SGD against the theoretical limit of exact interpolation.
- **Label Noise Robustness:** Analyzing how models handle corrupted labels across different data complexities.

2

3 Methodology

We implemented a Kernel Regression framework ($\min ||f||_{\mathcal{H}}$ s.t. $f(x_i) = y_i$) using two solvers: Either a direct interpolation from kernel theory by an exact matrix inversion $\alpha = (K + \epsilon I)^{-1} Y$. Or a preconditioned gradient descent optimizer EigenPro-SGD that accelerates convergence along "flat" directions of the loss landscape. This solver is the same as in the paper and is from this EigenPro github [2].

3.1 Dataset Adaptation

We use two datasets with distinct characteristics:

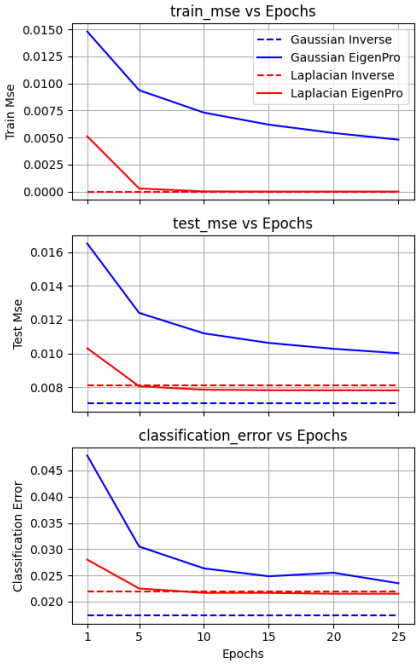
- **MNIST:** 28×28 grayscale images, flattened to $\mathbb{R}^{784}$. Contains simple, clean handwritten digits.
- **Fashion-MNIST:** 28×28 grayscale images, flattened to $\mathbb{R}^{784}$. Contains images of clothing with more complex edges, textures, and variations than digits.

One other key factor in the performance of kernel methods is the choice of the bandwidth σ. Indeed, this hyperparameter controls the scale at which the kernel operates and drastically influences the performance of the models. After tuning, we found that a small value of $\sigma = 2.0$ works well enough for our datasets.

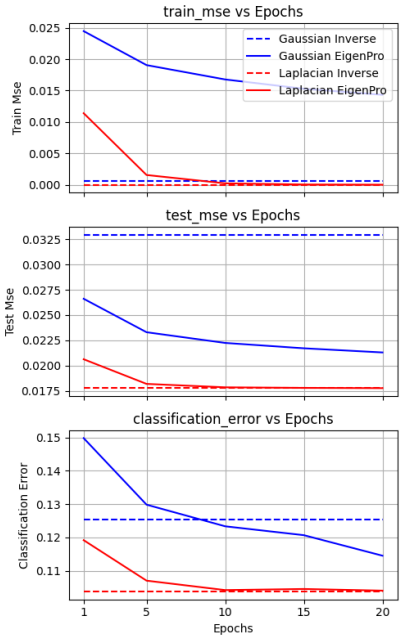
4 Experiments and Results

4.1 Convergence: MNIST vs. Fashion-MNIST

We compared the convergence of the approximate EigenPro solver against the exact solution.



(a) MNIST Convergence



(b) Fashion-MNIST Convergence

Figure 1: Training MSE of EigenPro-SGD vs. Direct Interpolation (dashed line). Fashion-MNIST requires more epochs/preconditioning.

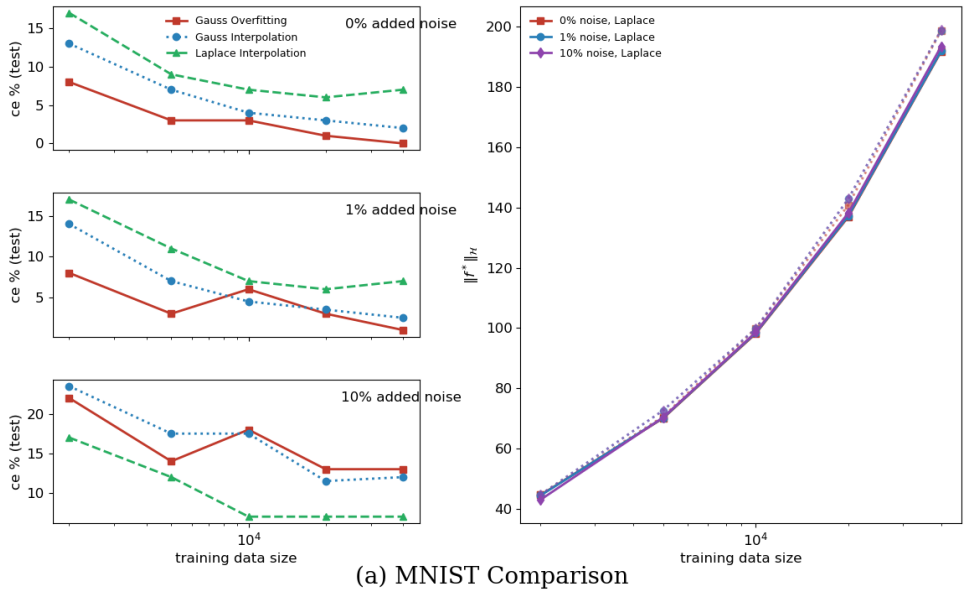
As found in the article, we can see that kernel machines can interpolate the training data with a train MSE of almost zero and still perform well on the test set, achieving approximately 98% accuracy on MNIST.

The Laplacian kernel consistently performs better than the Gaussian kernel in speed. It reaches the interpolating limit defined by the inverse models in only a couple of epochs, whereas the Gaussian model seems to converge to a higher limit at 25 epochs. According to the authors, this is because the spiky nature of the Laplacian allows it to interpolate quicker.

On the Fashion-MNIST dataset, the Gaussian kernel seems to require many more epochs to converge, whereas the Laplacian kernel does not. Furthermore, as predicted, performance is lower on this more complex dataset, with an accuracy of approximately 89%. Interestingly, even for interpolated kernels, the Gaussian one performs worse than the Laplacian (with 2% more error), whereas they are almost equal in performance across all other tests. This suggests that for more complex datasets, the Laplacian's inductive bias is not only faster to optimize but also leads to better generalization.

4.2 Robustness to Label Noise

Like in the article, we inject different label noise in the MNIST dataset to determine the impact of corrupted data on the performance of models. We test for 1% and 10% label noise with a Gaussian kernel.



(a) MNIST Comparison

Figure 2: Performance and RKHS norm of overfitted and interpolated kernels on MNIST.

We see that both inverse and SGD models manage to interpolate and drive train error very low as theorized. On test sets, even though the performance degrades with the random label ratio getting higher, the performance is still remarkable with around 13% error at 10% noise.

Moreover, we can see that the Laplacian kernel is comparatively worse with little noise added and better with higher noise. It manages to maintain a performance of around 92% accuracy regardless of the level of corruption in the labels. This again reinforces the idea that Laplacian kernels are suited for real-world datasets where label noise is omnipresent.

As proved in the article, the Reproducing Kernel Hilbert Space (RKHS) norm of the interpolating function ($\|f^*\|_{\mathcal{H}}$) grows exponentially with the training data size. This growth is logical, as a higher-complexity function is required to fit a larger number of constraints. The linear dependance of previous generalization bounds in the norm is causing them to become vacuous.

Interestingly, our results show that the norm growth is primarily driven by the training data size rather than the noise ratio. The curves for 0%, 1%, and 10% noise are nearly indistinguishable in their norm trajectory, suggesting that the adjustments required to interpolate noise do not drastically increase the global norm compared to the effort of interpolating clean samples. Maybe this is due to the "clean" dataset having already some noise, or maybe a need for tuning the σ parameter of the kernels but we could not find a better value.

4.3 Kernel Geometry Analysis

In order to verify the hypothesis that "spiky" kernels (Laplacian) behave more like ReLU networks than "smooth" kernels (Gaussian), we design a small experiment. Using a simple signal, we perturb it with a big outlier then interpolate with both types of kernels.

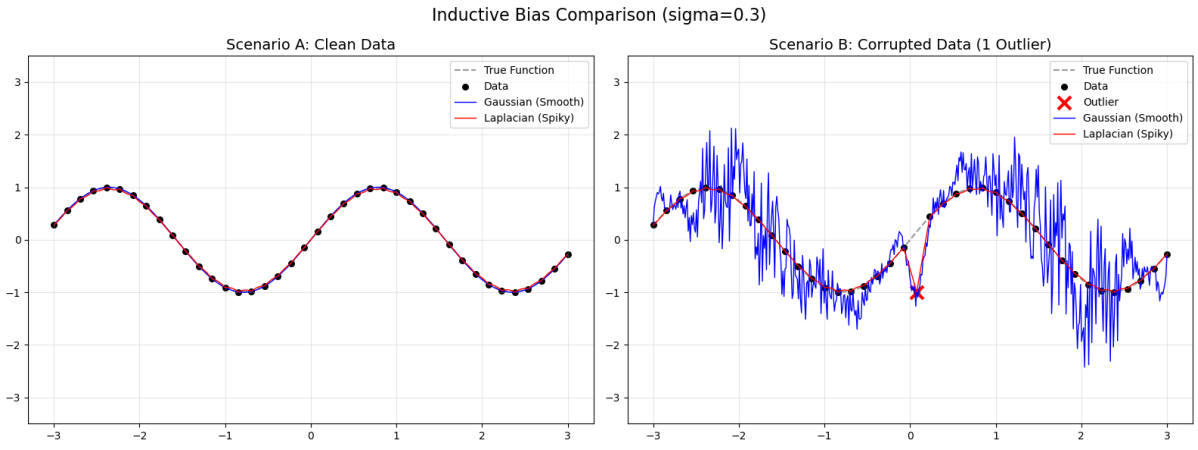


Figure 3: Fitting a perturbed sinusoid with both kernels.

The Gaussian kernel, constrained by its smoothness, is forced to distort the function in a wide neighborhood around the outlier to accommodate it. In contrast, the Laplacian kernel exhibits the "spiky" behavior predicted by Belkin et al.; it essentially memorizes the outlier with a sharp, local peak while preserving the global structure of the true function elsewhere. This is the inductive bias of the Laplacian kernel.

This visualizes why the Laplacian kernel (and by extension, ReLU networks) can achieve zero training error on noisy data without sacrificing generalization performance on clean test points. The authors do basically the same thing by showing that Laplacian kernels can interpolate data much faster than Gaussian ones.

5 Discussion

5.1 The Generalization Gap

While the Generalization issue (zero train error, good test error) holds for both, the absolute performance differs. On MNIST, kernel machines are competitive with DNNs. On Fashion-MNIST, they lag behind Convolutional Neural Networks (CNNs). This highlights a limitation of our reproduction: while we explain the mechanism of generalization (minimum norm), raw Kernel Machines lack the feature learning capabilities (convolutions) required for high performance on complex images.

5.2 Universality of Minimum Norm

Despite the performance difference, the phenomenon is universal; SGD implicitly regularizes the solution to the minimum norm interpolant. This confirms that the "Overfitting is a Myth" claim is not an artifact of simple datasets but a fundamental property of high-dimensional interpolation.

6 Conclusion

We showed that:

1. **Interpolation $\neq$ Overfitting:** Even on complex Fashion-MNIST data with noise, perfect fitting does not destroy generalization.
2. **Optimization Geometry matters:** Spiky kernels (Laplacian) are necessary to efficiently fit high-dimensional complex data, mirroring the success of ReLU in Deep Learning.

References

[1] Belkin, M., Ma, S., & Mandal, S. (2018). *To Understand Deep Learning We Need to Understand Kernel Learning.*

[2] Ma, S. & Belkin, M. (2019). *Kernel machines that adapt to GPUs for effective large batch training*