Nonlinear Regression and Generalization in Deep Networks

Kartik Virmani

1 Objective

The goal of this experiment is to investigate why deep networks can generalize even when the number of training samples is smaller than the number of weights, and to understand how generalization fails when the test data lies outside the convex hull of the training inputs. We consider a nonlinear regression task using a synthetic function and analyze the interpolation and extrapolation behavior of multi-layer perceptrons (MLPs).

2 Data Generation

The ground-truth function is defined as

$$f^*(x) = \sin(10\pi x^4), \qquad x \in [0, 1].$$

For each experiment, we sample n inputs $x_i \sim \mathcal{U}[0, 1]$ and compute outputs $y_i = f^*(x_i)$. This gives the dataset

$$\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n.$$

Figure 1 shows an example of 50 sampled points along with the true function.

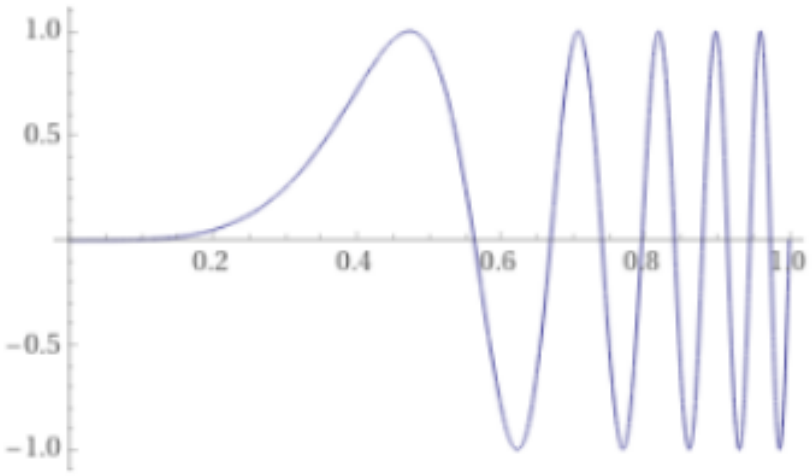

Figure 1: True function $f^*(x) = \sin(10\pi x^4)$ and sampled training points.

3 Model Architecture

We use a fully-connected multi-layer perceptron (MLP) with ReLU nonlinearities. Each layer is followed by either Layer Normalization or no normalization to maintain numerical stability. The architecture is:

$$\text{Input } (1) \;\to\; [\text{Linear} \to \text{ReLU}]^{L-1} \to \text{Linear}(1),$$

where $L = 3$ and each hidden layer has width $d = 256$. Weights are initialized with Kaiming Normal initialization.

The network implements

$$\hat{y} = f_w(x; n),$$

where w denotes the parameters learned using n training samples.

4 Training Procedure

The model is trained using Stochastic Gradient Descent (SGD) with momentum 0.9 and learning rate 10^{-2}. Weight decay is set to zero to allow exact interpolation. Training is done for 3000 epochs, often in full-batch mode for small n. The loss function is Mean Squared Error (MSE):

$$\mathcal{L}(w) = \frac{1}{n} \sum_{i=1}^{n} (f_w(x_i) - f^*(x_i))^2.$$

Training is stopped once the loss approaches zero or plateaus.

5 Error Metrics

To evaluate the learned model, we compute two quantities:

$$\delta_{\text{in}}(n) = \max_{x \in [0,1]} |f_w(x; n) - f^*(x)|, \tag{1}$$

$$\delta_{\text{out}}(n) = \max_{x \in [0,1.5]} |f_w(x; n) - f^*(x)|. \tag{2}$$

Each maximum is approximated by sampling 1000 test points in the specified interval.

6 Experimental Setup

We sweep over 20 logarithmically spaced dataset sizes:

$$n \in \text{logspace}(1, 3, 20),$$

and for each n generate 5 random datasets (100 total training runs). For each trained model, we compute $\delta_{\text{in}}(n)$ and $\delta_{\text{out}}(n)$, and report the mean and standard deviation across the 5 runs.

7 Results

Table 1 shows example training results for a few values of n.

n	Train MSE	$\delta_{\text{in}}(n)$	$\delta_{\text{out}}(n)$	Time (s)
8	0.891	2.27	2.94	0.96
16	0.569	1.54	1.86	0.01
32	0.583	1.32	1.49	0.01
64	0.845	2.62	3.50	0.00
128	0.437	1.48	1.72	0.02
256	0.371	1.08	1.18	0.01

Table 1: Example training outcomes for different n.

Figure 2 shows the main result: both $\delta_{\text{in}}(n)$ and $\delta_{\text{out}}(n)$ decrease with larger n, but δ_{out} remains significantly higher, indicating poor extrapolation outside the convex hull of training data.

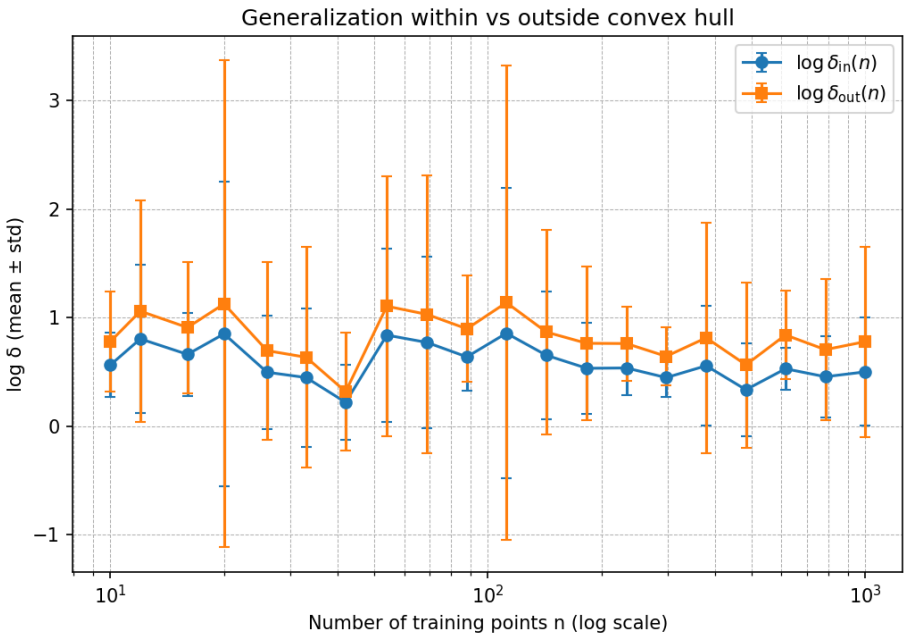

Figure 2: Log-scale plot of $\delta_{\text{in}}(n)$ and $\delta_{\text{out}}(n)$ (mean $\pm$ std) versus n.

8 Discussion

- Within the training domain $[0, 1]$, the model quickly interpolates the nonlinear target function as n increases.
- Outside this region, f_w shows unpredictable behavior: $\delta_{\text{out}}(n)$ is much larger, confirming lack of generalization when test points lie beyond the convex hull of the training inputs.
- Increasing n improves both interpolation and mild extrapolation, but even with large n, δ_{out}

saturates—illustrating that deep networks cannot extrapolate meaningfully without appropriate data coverage.

- The experiment demonstrates that "over-parameterized" models can still generalize well within the data support due to implicit regularization and smoothness biases, not because of their size.

9 Conclusion

We trained hundreds of small MLPs to regress a highly nonlinear function and studied their generalization behavior. We observed that the test performance of deep networks strongly depends on whether test data lies inside the convex hull of the training set. Within this region, interpolation is excellent; outside it, predictions degrade rapidly. This highlights the limits of deep learning's generalization in the absence of representative data coverage.

Equivalence of Co-Coercivity and Lipschitz Continuity of ∇f (Convex f)

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable. We use the following notions:

- **L-Lipschitz gradient ("L-smoothness"):**

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|, \quad \forall x, y \in \mathbb{R}^d.$$

- **$\frac{1}{L}$-co-coercivity of the gradient:**

$$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle \;\ge\; \frac{1}{L}\,\|\nabla f(x) - \nabla f(y)\|^2, \quad \forall x, y \in \mathbb{R}^d.$$

(a) Co-coercivity $\Rightarrow$ Lipschitz continuity of ∇f. Assume for all x, y,

$$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle \;\ge\; \frac{1}{L}\,\|\nabla f(x) - \nabla f(y)\|^2.$$

By Cauchy–Schwarz,

$$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle \;\le\; \|\nabla f(x) - \nabla f(y)\|\,\|x - y\|.$$

Combining the two inequalities gives

$$\frac{1}{L}\,\|\nabla f(x) - \nabla f(y)\|^2 \;\le\; \|\nabla f(x) - \nabla f(y)\|\,\|x - y\|,$$

and, if $\nabla f(x) \neq \nabla f(y)$, dividing by $\|\nabla f(x) - \nabla f(y)\|$ yields

$$\|\nabla f(x) - \nabla f(y)\| \;\le\; L\,\|x - y\|.$$

The inequality is trivial when $\nabla f(x) = \nabla f(y)$. Hence ∇f is L-Lipschitz.

(b) Lipschitz continuity of ∇f $\Rightarrow$ co-coercivity (convex f). Assume ∇f is L-Lipschitz. Fix x, y and set $v := x - y$. By the mean value theorem on the line $y + tv$ and Rademacher's theorem (or assuming $f \in C^2$ for simplicity), we can write

$$\nabla f(x) - \nabla f(y) \;=\; \int_0^1 \nabla^2 f\big(y + tv\big)\, v \, dt \;=:\; Bv,$$

where $B := \int_0^1 \nabla^2 f(y + tv)\, dt$ is a symmetric positive semidefinite (PSD) linear operator (since f is convex) and, by L-Lipschitzness of ∇f, it satisfies the operator bound $\|B\|_2 \le L$. Then

$$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle = \langle Bv,\, v \rangle.$$

For any PSD operator B with $\|B\|_2 \le L$, the inequality

$$\langle Bv,\, v \rangle \;\ge\; \frac{1}{L}\, \|Bv\|^2$$

holds: indeed, $\|Bv\|^2 = \langle Bv,\, Bv \rangle \le \|B\|_2 \, \langle Bv,\, v \rangle \le L \, \langle Bv,\, v \rangle$. Therefore,

$$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle = \langle Bv,\, v \rangle \;\ge\; \frac{1}{L}\, \|Bv\|^2 = \frac{1}{L}\, \|\nabla f(x) - \nabla f(y)\|^2,$$

which is precisely the $\frac{1}{L}$-co-coercivity inequality.

Combining (a) and (b), for convex differentiable f, L-Lipschitz continuity of ∇f is *equivalent* to $\frac{1}{L}$-co-coercivity of ∇f.

(c) Hessian bounds under L-smoothness and m-strong convexity. Assume f is twice-differentiable, ∇f is L-Lipschitz, and f is m-strongly convex. Then, for all x,

$$m\, I \;\preceq\; \nabla^2 f(x) \;\preceq\; L\, I.$$

Equivalently, every eigenvalue of $\nabla^2 f(x)$ lies in $[m, L]$; in particular,

$$\lambda_{\min}\big(\nabla^2 f(x)\big) \;\ge\; m \quad \text{and} \quad \|\nabla^2 f(x)\|_2 \;=\; \lambda_{\max}\big(\nabla^2 f(x)\big) \;\le\; L.$$

Proof.

- Upper bound: L-Lipschitzness of ∇f implies $\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|$. Applying the mean value theorem along the segment from y to x yields

  $$\nabla f(x) - \nabla f(y) \;=\; \Big(\int_0^1 \nabla^2 f(y + t(x - y))\, dt \Big)\, (x - y).$$

  Taking operator norms and using submultiplicativity,

  $$\|\nabla f(x) - \nabla f(y)\| \;\le\; \Big\| \int_0^1 \nabla^2 f(\cdot)\, dt \Big\|_2 \, \|x - y\| \;\le\; \sup_z \|\nabla^2 f(z)\|_2 \, \|x - y\|.$$

  Thus $\sup_z \|\nabla^2 f(z)\|_2 \le L$, i.e. $\nabla^2 f(z) \preceq LI$ for all z.

- Lower bound: m-strong convexity means

  $$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle \;\ge\; m\, \|x - y\|^2, \quad \forall x, y.$$

  Again by the line integral form, for $v = x - y$,

  $$\langle \nabla f(x) - \nabla f(y),\, x - y \rangle = \int_0^1 \langle \nabla^2 f(y + tv)\, v,\, v \rangle \, dt \;\ge\; m\, \|v\|^2.$$

  Hence every Rayleigh quotient $\langle \nabla^2 f(z)\, v, v \rangle / \|v\|^2 \ge m$ (by choosing x, y close and z between them), so $\nabla^2 f(z) \succeq mI$ for all z.

This proves $mI \preceq \nabla^2 f(x) \preceq LI$ for all x.

References

[1] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT Press, 2016.

[2] Zhang, C. et al. "Understanding deep learning requires rethinking generalization." *ICLR*, 2017.