

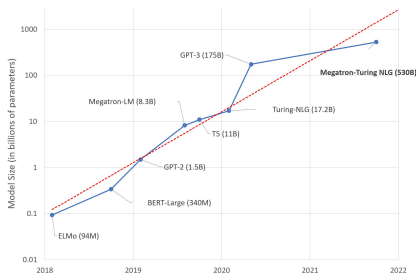
# Overparameterized Linear Regression under Adversarial Attacks

**Antônio H. Ribeiro**

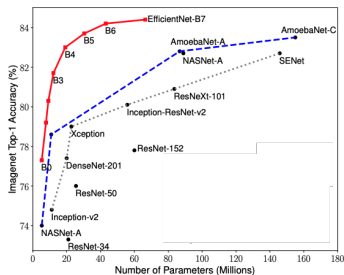
Uppsala University

Online Presentation @ University of British Columbia  
June 3rd, 2022

# Model size in neural networks



(a) Language models



(b) Image classification

Figure: Models number of parameters

Sources: J. Simon (2021) "Large Language Models: A New Moore's Law?". Online (accessed: 2021-11-09). URL: [huggingface.co/blog/large-language-models](https://huggingface.co/blog/large-language-models).

M. Tan and Q. V. Le (2019) "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," ICML

Seminars: overparameterized machine learning models (2021, Fall) — PhD level course - together with Dave Zachariah and Per Mattsson.

All material available in: <https://github.com/uu-sml/seminars-overparam-ml>

Inaugural paper: M. Belkin, D. Hsu, S. Ma, and S. Mandal, “Reconciling modern machine-learning practice and the classical bias–variance trade-off.” PNAS, 2019

PNAS

### Reconciling modern machine-learning practice and the classical bias–variance trade-off

Mikhail Belkin<sup>1,2,\*</sup>, Daniel Hsu<sup>1</sup>, Sreyan Ma<sup>1</sup>, and Soumik Mandal<sup>1</sup>

<sup>1</sup>Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210; <sup>2</sup>Department of Statistics, The Ohio State University, Columbus, OH 43210 and <sup>3</sup>Computer Science Department and Data Science Institute, Columbia University, New York, NY 10027

Edited by Peter J. Bressi, University of California, Berkeley, CA, and approved July 2, 2019 (received for review February 23, 2019)

**Breakthroughs in machine learning are rapidly changing science and society, yet our fundamental understanding of this technology has lagged far behind. Indeed, one of the central tenets of the field, the bias–variance trade-off, appears to be at odds with the observed behavior of modern tools used in machine-learning practice. The bias–variance trade-off implies that a model should balance underfitting and overfitting. Risk enough to express underlying structure in data and simple enough to avoid fitting spurious patterns. However, in modern practice, very rich models such as neural networks are trained to exactly fit (i.e., interpolate) the data. Classically, such models would be considered overfitted, and yet they often obtain high accuracy on new data. This apparent contradiction has raised questions about the mathematical foundations of machine learning and their relevance to practitioners. In this paper, we reconcile the classical understanding and the modern practice within a unified performance curve. This “double-descent” curve subsumes the textbook U-shaped bias–variance trade-off curve by showing how increasing model capacity beyond the point of interpolation results in improved performance. The provided evidence for the existence and ubiquity of double descent for a wide spectrum of models and datasets, and we point a mechanism for its emergence. This connection between the performance and the structure of machine-learning models delineates the biases of classical analysis and has implications for both the theory and the practice of machine learning.**

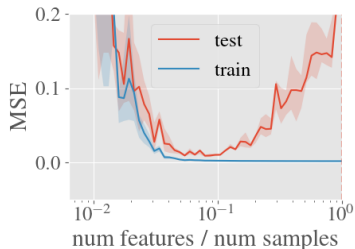
machine learning • bias–variance trade-off • neural networks

**M**achine learning has become key to important applications in science, technology, and commerce. The success of machine learning is on the problem of predicting the value of training examples  $\{x_1, y_1\}, \dots, \{x_n, y_n\}$  from  $\mathbb{R}^d \times \mathbb{R}$ , we learn a predictor  $h$ ,  $\mathbb{R}^d \rightarrow \mathbb{R}$ , so that to predict the label  $y$  of a new point  $x$ , we return  $h(x)$ .

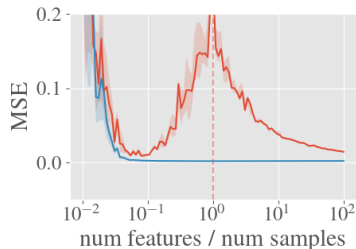
The predictor  $h$  is commonly chosen from some function class  $\mathcal{H}$ , such as neural networks with a certain architecture, using empirical risk minimization (ERM) and its variants. In ERM, the predictor is taken to be a function  $h$  that minimizes the empirical (or training) risk  $\sum_{i=1}^n \ell(y_i, h(x_i))$ , where  $\ell$  is a loss function, such as the squared loss  $\ell(y, \hat{y}) = (y - \hat{y})^2$  or regression or 0–1 loss  $\ell(y, \hat{y}) = \mathbb{I}_{y \neq \hat{y}}$  for classification. The goal of machine learning is to find  $h$ , that performs well on new data, unseen in training. To study performance on new data (such as generalization), we typically assume the training examples are sampled randomly from a probability distribution  $P$  over  $\mathbb{R}^d \times \mathbb{R}$  and evaluate  $h$ , on a new test example  $(x, y)$ .

Figure 1 consists of two subplots, A and B, showing Risk versus Capacity of  $\mathcal{H}$ . Subplot A shows a U-shaped curve for training risk (dashed line) and a test risk curve (solid line) that initially decreases and then increases, forming a U-shape. The minimum of the training risk is labeled 'sweet spot'. Subplot B shows a similar setup but with a different test risk curve that decreases monotonically after the interpolation threshold. Both plots have 'Capacity of  $\mathcal{H}$ ' on the x-axis and 'Risk' on the y-axis. The region to the left of the interpolation threshold is labeled 'underfitting', and the region to the right is labeled 'overfitting'.

# Double-descent



(a) U-Shape



(b) Double-descent

**Figure:** Nonlinear ARX performance in Couple Eletric Drives benchmark.

**A. H. Ribeiro**, J. N. Hendriks, A. G. Wills, T. B. Schön. "Beyond Occam's Razor in System Identification: Double-Descent when Modeling Dynamics". IFAC SYSID (2021) *Honorable mention*: **Young author award**

# Double-descent in linear models

**Estimated parameter:** using train dataset  $(\mathbf{x}_i, y_i)$ ,  $i = 1, \dots, n$ :

- ▶ Underparametrized:

$$\hat{\beta} = \arg \min_{\beta} \sum_i (y_i - \mathbf{x}_i^T \beta)^2$$

- ▶ Overparametrized:

$$\hat{\beta} = \arg \min_{\beta} \|\beta\|_2^2$$

subject to  $y_i = \mathbf{x}_i^T \beta$   
for every  $i$

**Random features:** Belking et.al. (2019) generates the features through the nonlinear mapping:  $\phi : u_i \mapsto \mathbf{x}_i$  obtained from Random Fourier Features.

*Overparametrized models can generalize effectively when train and test come from the **same** distribution...*

*are they robust?*

# Adversarial Attacks

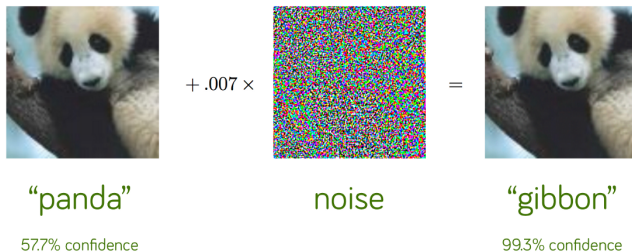


Figure: Illustration of adversarial attack.

Source: I. J. Goodfellow, J. Shlens, C. Szegedy, “*Explaining and Harnessing Adversarial Examples*”, ICLR 2015.

# The role of high-dimensionality

- ▶ High-dimensionality as a source of vulnerability:
  - ▶ I. J. Goodfellow, J. Shlens, C. Szegedy, “*Explaining and Harnessing Adversarial Examples*”, ICLR 2015
  - ▶ J. Gilmer et al., “*Adversarial Spheres*,” arXiv:1801.02774, Sep. 2018.
  - ▶ D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Ma, “Robustness May Be At Odds with Accuracy,” ICLR, p. 23, 2019.
- ▶ High-dimensionality as a source of robustness:
  - ▶ S. Bubeck and M. Sellke, “A Universal Law of Robustness via Isoperimetry,” Advances in Neural Information Processing Systems, 2021



# Outline

**Paper I** **A. H. Ribeiro** and T. B. Schön, "Overparametrized Linear Regression under Adversarial Attacks,"  
arXiv:2204.06274, April 2022.

**Paper II** **A. H. Ribeiro**, D. Zachariah, and T. B. Schön, "Surprises in adversarially-trained linear regression,"  
arXiv:2205.12695, May 2022.

# Linear regression under adversarial attacks

Given a data point not seen during training ( $\mathbf{x}, y$ ).

**Standard risk:**

$$R = E\left\{(\mathbf{y} - \mathbf{x}^T \hat{\boldsymbol{\beta}})^2\right\}$$

**Adversarial risk:**

$$R_p^{\text{adv}} = E\left\{\max_{\|\Delta \mathbf{x}\|_p \leq \delta} (\mathbf{y} - (\mathbf{x} + \Delta \mathbf{x})^T \hat{\boldsymbol{\beta}})^2\right\}$$

$\Delta \mathbf{x} \rightsquigarrow$  Adversarially generated disturbance



(a)  $\ell_1$



(b)  $\ell_{1.5}$



(c)  $\ell_2$



(d)  $\ell_{20}$



(e)  $\ell_\infty$

# Linear regression is a special case

The original formula

$$R_p^{\text{adv}} = E \left\{ \max_{\|\Delta \mathbf{x}\|_p \leq \delta} (\mathbf{y} - (\mathbf{x} + \Delta \mathbf{x})^\top \hat{\boldsymbol{\beta}})^2 \right\}$$

Can be reformulated. Let  $q$ , such that  $\frac{1}{p} + \frac{1}{q} = 1$

$$R_p^{\text{adv}} = E \left( |\mathbf{y} - \mathbf{x}^\top \hat{\boldsymbol{\beta}}| + \delta \|\hat{\boldsymbol{\beta}}\|_q \right)^2.$$

# Bounds on the adversarial risk

$$R + \delta^2 \|\hat{\beta}\|_q^2 \leq R^{\text{adv}} \leq \left( \sqrt{R} + \delta \|\hat{\beta}\|_q \right)^2$$

- ▶  $R^{\text{adv}}$   $\rightsquigarrow$  Adversarial risk
- ▶  $R$   $\rightsquigarrow$  Risk
- ▶  $\delta$   $\rightsquigarrow$  Adv. disturbance magnitude

**Note:** *in the Gaussian case*

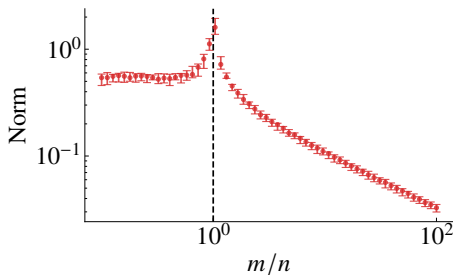
$$R^{\text{adv}}(\beta) = \left( 1 - \sqrt{\frac{2}{\pi}} \right) (\text{Upper bound}) + \sqrt{\frac{2}{\pi}} (\text{Lower bound}).$$

# Decay rate of the $\ell_2$ -norm

- Data model:

$$(x_i, \epsilon_i) \sim P_x \times P_\epsilon, \quad y_i = x_i^\top \beta + \epsilon_i,$$

- $\ell_2$ -norm of the estimated parameter: decays with  $\frac{1}{\sqrt{\# \text{ features}}}$

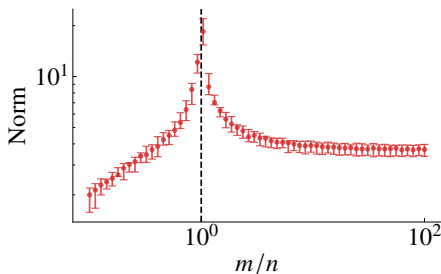


## Decay rate of the $\ell_1$ -norm

- ▶ Relation between  $p$ -norm

$$\|\hat{\beta}\|_2 \leq \|\hat{\beta}\|_1 \leq \sqrt{m} \|\hat{\beta}\|_2.$$

- ▶  $\ell_1$ -norm of the estimated parameter: approaches a constant



- ▶ Hence:

$$\|\hat{\beta}\|_1 \rightarrow c\sqrt{m} \|\hat{\beta}\|_2.$$

# Scaling

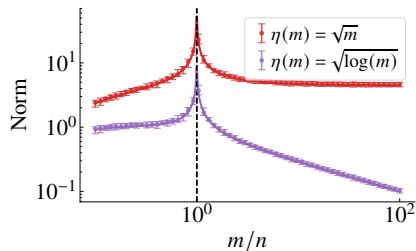
- ▶ Model prediction:  $\hat{\beta}^\top x$ .
- ▶ Equivalent model prediction:  $\tilde{\beta}^\top \tilde{x}$ .

$$\tilde{x} = \frac{1}{\eta} x$$

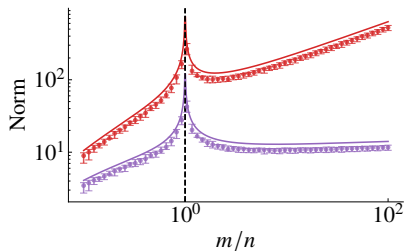
$$\tilde{\beta} = \eta \hat{\beta}$$

- ▶  $x$  be an isotropic  $\rightarrow \mathbb{E} [\|x\|_2^2] = m$ .  
 $\rightarrow \eta(m) = \sqrt{m}$
- ▶  $x$  is a sub-Gaussian  $\rightarrow \mathbb{E} [\|x\|_\infty] = \Theta(\sqrt{\log(m)})$   
 $\rightarrow \eta(m) = \sqrt{\log m}$

# Norm



(a)  $\|\hat{\beta}\|_2$

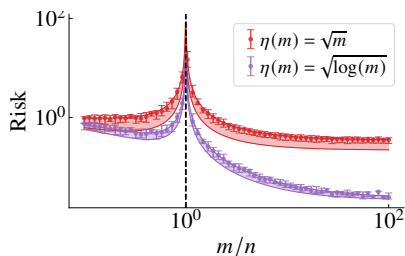


(b)  $\|\hat{\beta}\|_1$

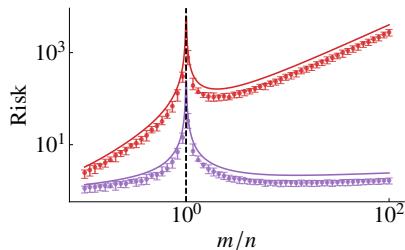


# Adversarial Risk

$$R + \delta^2 \|\hat{\beta}\|_q^2 \leq R^{\text{adv}} \leq \left( \sqrt{R} + \delta \|\hat{\beta}\|_q \right)^2$$



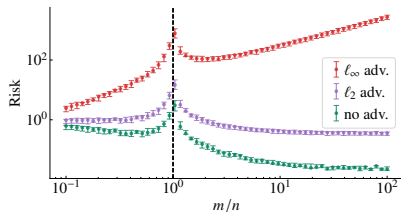
(a)  $\ell_2$  adv. risk



(b)  $\ell_\infty$  adv. risk

# Discussion

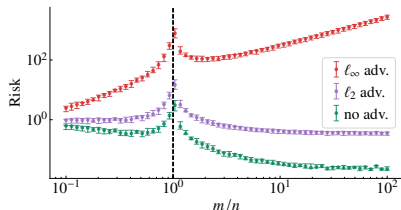
- Different metrics in the input space  $\rightarrow$  different assessments of the robustness.



**Figure:** Adv. risk.

# Discussion

- ▶ Can be seen as one aspect of the curse of dimensionality.
- ▶ Most pathological results for mismatched setup:  $\mathbb{E}_x [\|x\|_2^2]$  const. attack while  $\ell_\infty$  attack

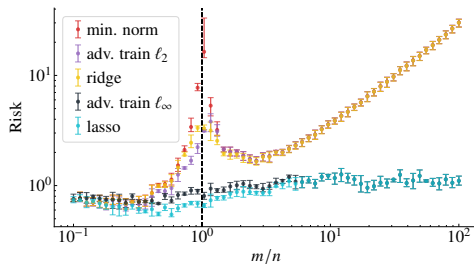


**Figure:** Adv. risk. Fixed  $\eta(m) = \sqrt{n}$

- ▶ Brittleness to adversarial examples in linear models is highly influential. The mismatch usually appears hidden in the examples.

I. J. Goodfellow, J. Shlens, C. Szegedy, "Explaining and Harnessing Adversarial Examples", ICLR 2015  
D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Ma, "Robustness May Be At Odds with Accuracy," ICLR, p. 23, 2019.

# The effect of regularization and adversarial training



**Figure:** Adversarial  $\ell_\infty$  risk.

## Concentration of the norm

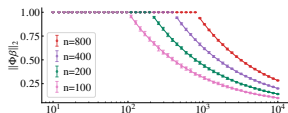
- ▶ The parameter estimated is

$$\begin{aligned}\hat{\beta} &= (X^T X)^\dagger X^T y, \\ &= (X^T X)^\dagger X^T (X\beta + \epsilon), \\ &= \underbrace{(X^T X)^\dagger X^T X}_{\Phi} \beta + (X^T X)^\dagger X^T \epsilon\end{aligned}$$

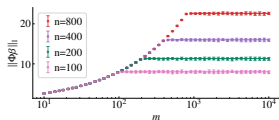
- ▶  $\Phi \in \mathbb{R}^{m \times m}$  is an orthogonal projector into a subspace of dimension  $n$ .
- ▶ If the entries of  $X$  are Gaussian, then  $\Phi$  projects onto a random subspace uniformly sampled from Grassmannian  $G(m, n)$ .
- ▶ It is well known (Vershynin 2018, High-Dimensional Probability, Lemma 5.3.2) probability greater than  $1 - 2 \exp(-ct^2 n)$ :

$$(1 - t) \sqrt{\frac{n}{m}} \|\beta\|_2 \leq \|\Phi\beta\|_2 \leq (1 + t) \sqrt{\frac{n}{m}} \|\beta\|_2 \quad (1)$$

# Concentration of the norm



(a)  $\ell_2$  norm of projection



(b)  $\ell_1$  norm of projection

Figure: Random projection and norms.

$\|\Phi\beta\|_1$  concentrate with high-probability around  $c\sqrt{m}\|\beta\|_2$ .

# Outline

**Paper I** **A. H. Ribeiro** and T. B. Schön, "Overparametrized Linear Regression under Adversarial Attacks,"  
arXiv:2204.06274, April 2022.

**Paper II** **A. H. Ribeiro**, D. Zachariah, and T. B. Schön, "Surprises in adversarially-trained linear regression,"  
arXiv:2205.12695, May 2022.

# Adversarial Training

- ▶ Empirical risk minimization (ERM). Minimizes:

$$\hat{R}(\beta) = \frac{1}{n} \sum_{i=1}^n (y_i - \mathbf{x}_i^T \beta)^2,$$

- ▶ Adversarial training, minimizes *empirical adversarial risk*:

$$\hat{R}_p^{\text{adv}}(\beta) = \frac{1}{n} \sum_{i=1}^n \max_{\|\Delta \mathbf{x}_i\|_p \leq \delta} (y_i - (\mathbf{x}_i + \Delta \mathbf{x}_i)^T \beta)^2$$



# Adversarial Training in linear regression

- ▶ The same simplification applies:

$$\hat{R}_p^{\text{adv}}(\beta) = \frac{1}{n} \sum_{i=1}^n \left( |y_i - x_i^T \beta| + \delta \|\beta\|_q \right)^2$$

- ▶ The above expression is **convex**

# Lasso and $\ell_\infty$ -adversarial training

- ▶  $\ell_\infty$ -adversarial training:

$$\hat{R}_\infty^{\text{adv}}(\beta) = \frac{1}{n} \sum_{i=1}^n \left( |y_i - x_i^\top \beta| + \delta \|\beta\|_1 \right)^2$$

- ▶ Lasso:

$$\hat{R}^{\text{lasso}}(\beta) = \frac{1}{n} \sum_{i=1}^n \left( |y_i - x_i^\top \beta| \right)^2 + \delta \|\beta\|_1$$

# Ridge regression and $\ell_2$ -adversarial training

- ▶  $\ell_2$ -adversarial training:

$$\hat{R}_2^{\text{adv}}(\beta) = \frac{1}{n} \sum_{i=1}^n \left( |y_i - \mathbf{x}_i^T \beta| + \delta \|\beta\|_2 \right)^2$$

- ▶ Ridge:

$$\hat{R}^{\text{ridge}}(\beta) = \frac{1}{n} \sum_{i=1}^n \left( |y_i - \mathbf{x}_i^T \beta| \right)^2 + \delta \|\beta\|_2^2$$

# Diabetes example

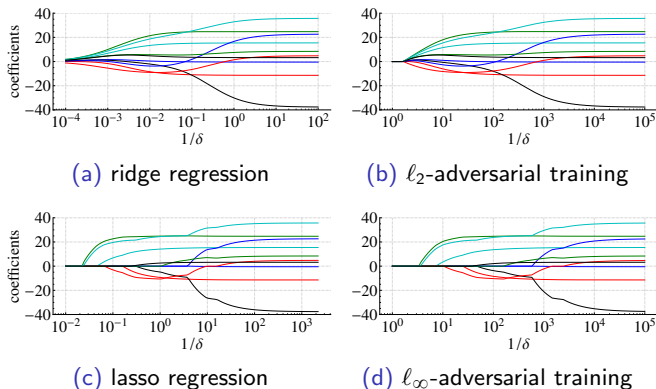
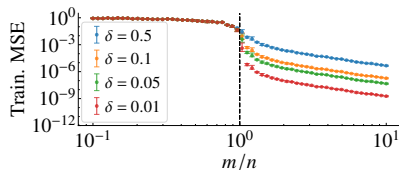
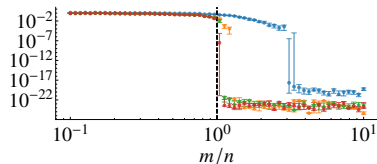


Figure: Regularization paths.

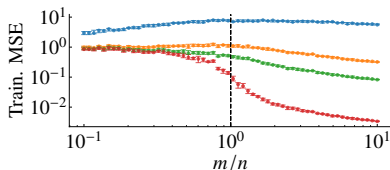
# Differences in the overparametrized region



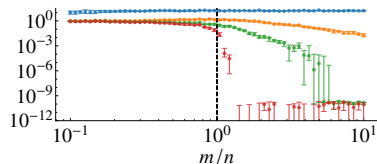
(a) ridge regression



(b)  $\ell_2$ -adversarial training



(c) lasso regression



(d)  $\ell_\infty$ -adversarial training

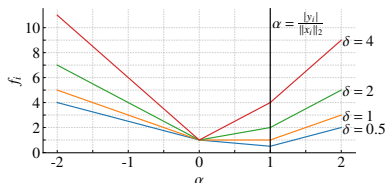
Figure: Mean square error in training data.

# Discussion

- ▶ Adversarial training can go through abrupt transitions in behavior.

- ▶ Looking at one point can be instructive:

$$f_i(\beta) = |y_i - x_i^T \beta| + \delta \|\beta\|_2$$



- ▶ Related work

H. Xu, C. Caramanis, and S. Mannor, "Robust regression and lasso," Advances in neural information processing systems, vol. 21, 2008


- ▶ Robust regression → feature-wise perturbation
- ▶ Adversarial training → sample-wise perturbation

# Thank you!

Contact info:

 [antonio.horta.ribeiro@it.uu.se](mailto:antonio.horta.ribeiro@it.uu.se)

 [@ahortaribeiro](https://twitter.com/ahortaribeiro)

 [antonior92.github.io](https://antonior92.github.io)

 [github.com/antonior92](https://github.com/antonior92)