

Lecture 1/2 What is Machine Learning?

IEMS 402 Statistical Learning

Northwestern

Logistics

Logistics

- Course Website: <https://2prime.github.io/teaching/2025-Statistical-Learning>
- Grading: Problem Sets (15%) + Exams (80%) + Scribe Note (5%)
$$\max(HW1, HW8) + \max(HW2, HW3) + \max(HW4, HW5) + \max(HW6, HW7).$$

- [\[Homework 1\]](#) Review of Probability and Optimization
- [\[Homework 2\]](#) Bias and Variance Trade-off 1
- [\[Homework 3\]](#) Bias and Variance Trade-off 2
- [\[Homework 4\]](#) Asymptotic Theory 1
- [\[Homework 5\]](#) Asymptotic Theory 2
- [\[Homework 6\]](#) Non-Asymptotic Theory 1
- [\[Homework 7\]](#) Non-Asymptotic Theory 2
- [\[Homework 8\]](#) Advanced Topics

Review of technical basic
Start early!

Advanced research in OR

- Latex and overleaf (not required)

Logistics

- Course Website: <https://2prime.github.io/teaching/2025-Statistical-Learning>
- Grading: Problem Sets (15%) + Exams (80%) + Scribe Note (5%)

$$\max(HW1, HW8) + \max(HW2, HW3) + \max(HW4, HW5) + \max(HW6, HW7).$$

- [\[Homework 1\]](#) Review of Probability and Optimization
 - [\[Homework 2\]](#) Bias and Variance Trade-off 1
 - [\[Homework 3\]](#) Bias and Variance Trade-off 2
 - [\[Homework 4\]](#) Asymptotic Theory 1
 - [\[Homework 5\]](#) Asymptotic Theory 2
 - [\[Homework 6\]](#) Non-Asymptotic Theory 1
 - [\[Homework 7\]](#) Non-Asymptotic Theory 2
 - [\[Homework 8\]](#) Advanced Topics
- Easy } Start early!
- Easy
- Easy

- Latex and overleaf (not required)

Logistics

- Course Website: <https://2prime.github.io/teaching/2025-Statistical-Learning>
- Grading: Problem Sets (15%) + Exams (80%) + Scribe Note (5%)

Exams

- [Practice Mid-Term Exam]   The same technique as the exam
 - Modern Machine Learning Concepts, Bias and Variance Trade-off
 - Kernel Smoothing, Asymptotic Theory, Influence Function Concentration Inequality, Uniform Bound
- [Practice Final Exam] 
 - Rademacher complexity, Covering Number, Dudley's theorem
 - RKHS, Optimal Transport, Robust Learning

Logistics

- Course Website: <https://2prime.github.io/teaching/2025-Statistical-Learning>
- Grading: Problem Sets (15%) + Exams (80%) + Scribe Note (5%)

The screenshot shows a LaTeX editor interface. On the left, the 'Code Editor' displays the LaTeX source code for a document. The code includes various packages like `\documentclass[twoside]{article}`, `\usepackage{amsmath, amssymb, amsthm}`, and `\usepackage{geometry}`. It also defines a new theorem style `\newtheorem[problem]{Problem}` and sets up a counter for lecnum. The right side shows a 'Visual Editor' with a preview of the document. The preview title is 'IEMS 402: Statistical Learning' and the subtitle is 'Lecture 15: Optimal Transport'. Below the title, it says 'Scribe:'. The preview content includes a disclaimer about the notes being unscrubbed and a section titled '15.1 Introduction to Optimal Transport' which discusses the theory's history and applications.

Refine my note

Logistics

- Course Website: <https://2prime.github.io/teaching/2025-Statistical-Learning>
- Grading: Problem Sets (15%) + Exams (80%) + Scribe Note (5%)
- Textbook: Bach, Francis. Learning theory from first principles. MIT press, 2024.
 - https://www.di.ens.fr/~fbach/ltpf_book.pdf

Gradescope
Campuswire
ChatGPT Tutor!

Late Work Policy

- For your first late assignment within 12 hours after the deadline (as indicated on Gradescope), no point deductions.
- All subsequent assignments submitted within 12 hours after the deadline will convert to a zero at the end of semester.
- In all cases, work submitted 12 hours or more after the deadline will not be accepted.

Preliminary

Review Document:

<https://2prime.github.io/files/IEMS402/IEMS402ProbOptReview.pdf>

Calculus, Linear Algebra

IEMS 302 Probability Probability and Statistics: Strong Law of Large Numbers, Central Limit Theorem, Big-O, little-o notation,

Optimization Theory: **Lagrangian Duality Theory IEMS 450-2: Mathematical Optimization II**
(Interestingly, IEMS 450-1 is not required)

You **need** to know

Law of strong numbers, Central Limit Theorem, Continuous Map Theorem, Slutsky Theorem, Markov's Inequality

You **don't need** to distinguish Convergence in Probability/Covergence in distribution, you just need to write →

Online Calibration with Human Feedback

The screenshot shows a digital feedback interface. At the top, there's a navigation bar with Chinese characters: 问题 (Question), 回复 (Reply), and 设置 (Settings). Below this, the title "Feedback for IEMS402 Lecture 2" is displayed, with "Lecture 2" highlighted by a blue rounded rectangle and an arrow pointing to the text "Feedback for each lecture". To the right of the title is a vertical toolbar with icons for adding (+), saving (disk), and other functions. The main content area contains a question about lecture pace with a 5-point scale from "Much too slow" to "Much too fast". There are also sections for identifying confusing parts and surprising/interesting aspects, each with a "Detailed Answer Text" input field.

Feedback for IEMS402 Lecture 2

This feedback will help calibrate future lectures. Feel free to answer any subset of the questions (it is encouraged to at least answer the first question on pace).

The pace of material was

1 2 3 4 5

Much too slow Much too fast

What parts were confusing?

详答文本

What was most surprising/interesting?

详答文本

Other Course

Stats 300b - Stanford

1. Introduction
2. Convergence of random variables (January 14)
3. Delta method (January 14)
4. Basics of asymptotic normality (January 18 and 20)
5. Moment method (January 20)
6. Uniform laws of large numbers (January 26)
7. Basics of concentration (January 28 and February 2)
8. Sub Gaussian processes and chaining (February 2 and February 4)
9. VC Dimension (February 4)
10. Uniform central limit theorems and convergence in distribution (February 9 and February 11)
11. Applications of Uniform Central Limit Theorems (February 16 and February 18)
12. Relative efficiency and basic tests (February 18 and February 23)
13. Asymptotic level and relative efficiency in testing (February 23 and 25)
14. Contiguity and Asymptotics (February 25)
15. Local Asymptotic Normality (March 2 and 4)
16. Regular estimators and consequences (March 8 and 10)
17. U statistics (March 11 and 16)
18. Parting thoughts (March 18)

Date	Lecture Topic
August 31	Review
September 2	Concentration Inequalities
September 4	Concentration Inequalities
September 7	No Class (Labor Day)
September 9	Convergence
September 11	Convergence
September 14	Central Limit Theorem
September 18	Uniform Laws and Empirical Process Theory
September 18	Uniform Laws and Empirical Process Theory
September 21	Uniform Laws and Empirical Process Theory
September 23	Uniform Laws and Empirical Process Theory
September 25	Review
	TEST 1
September 28	Likelihood and Sufficiency
September 30	Point Estimation (MLE)
October 2	Point Estimation (Method of Moments, Bayes)
October 5	Decision Theory
October 7	Decision Theory
October 9	Asymptotic Theory
October 12	Asymptotic Theory
October 14	Hypothesis Testing
October 16	NO CLASS (Community Engagement)
October 19	Goodness-of-fit, two-sample, independence
October 21	Multiple testing
October 23	NO CLASS (Mid-Semester Break)
October 26	Multiple testing
October 28	Confidence Intervals
October 30	Confidence Intervals
November 2	Confidence Intervals
November 4	Review
November 6	TEST 2
November 9	Bootstrap
November 11	Bootstrap
November 13	Bayesian Inference
November 16	Bayesian Inference
November 18	Linear Regression
November 20	Non-parametric Regression
November 23	NO CLASS
November 25	NO CLASS (Thanksgiving)
November 27	NO CLASS
November 30	Minimax Lower Bounds
December 2	Minimax Lower Bounds
December 4	High-dimensional Statistics
December 7	High-dimensional Statistics
December 9	Model Selection
December 11	Model Selection

Stats 705 - CMU

Other Course

Stanford: Stats 300b/ CS229T

Berkeley: Stats 241/Stats 241B

MIT IDS.160/9.521/18.656/6.S988

CMU Stat705, 10-072

Princeton COS 511

Cornell CS6783, ORIE 7790

Umich EECS598, UW Madison CS 839, UofT STA3000F

Good machine learning courses are open source!

Why IEMS402?

Deep learning is eating the world — Jorge Nocedal

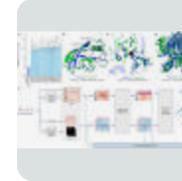


Nature

[https://www.nature.com › articles](https://www.nature.com/articles) · 翻译此页 · :

Highly accurate protein structure prediction with AlphaFold

作者: J Jumper · 2021 · 被引用次数: 30628 — AlphaFold greatly improves the accuracy of structure prediction by incorporating novel neural network architectures



and t



Nature

[https://www.nature.com › articles](https://www.nature.com/articles) · 翻译此页 · :

Magnetic control of tokamak plasmas through deep ...

作者: J Degraeve · 2022 · 被引用次数: 897 — In this work, we introduce a previously undescribed architecture for t



Nature

[https://www.nature.com › news](https://www.nature.com/news) · 翻译此页 · :

DeepMind AI weather forecaster beats world-class system

2024年12月4日 — Google DeepMind has developed the first artificial intelligence (AI) model of its kind to predict the weather more accurately than the best ...

Why IEMS402?



Nature

[https://www.nature.com › articles](https://www.nature.com/articles) · 翻译此页 · :



Highly accurate protein structure prediction with AlphaFold

作者: J Jumper · 2021 · 被引用次数: 30628 — AlphaFold greatly improves the accuracy of structure prediction by incorporating novel neural network architectures and ti



Nature

[https://www.nature.com › articles](https://www.nature.com/articles) · 翻译此页 · :

Magnetic control of tokamak plasmas through deep ...

作者: J Degraeve · 2022 · 被引用次数: 897 — In this work, we introduce a previously undescribed architecture for t



Nature

[https://www.nature.com › news](https://www.nature.com/news) · 翻译此页 · :

DeepMind AI weather forecaster beats world-class system

2024年12月4日 — Google DeepMind has developed the first artificial intelligence (AI) model of its kind to predict the weather more accurately than the best ...

Why IEMS402?

I Don't have a cat

The screenshot shows a dark-themed ChatGPT interface. At the top, it says "ChatGPT o1". Below that is a profile icon with a question mark and the text "Thought about IEMS 402 course introduction for 7 seconds". The main message is: "Subject: Welcome to IEMS 402 – Let the Statistical Learning Adventure Begin! Hello everyone, Welcome to the **brand-new** (and dare I say, *improved*) version of **IEMS 402**! I'm excited, you're excited, and even my cat is excited (though she's mostly excited for treats). This course is going to be a lot of fun, full of intriguing discoveries, many "aha!" moments, and, well, a few puffs of chalk dust here and there." A blue arrow points from the text "I Don't have a cat" to the word "cat" in the message. At the bottom, there's another section titled "A New Version of 402 – Your Feedback is Golden" with a "ChatGPT" icon, followed by text about collecting anonymous opinions after each course session, and links to the course website and year.

ChatGPT o1

Thought about IEMS 402 course introduction for 7 seconds

Subject: Welcome to IEMS 402 – Let the Statistical Learning Adventure Begin!

Hello everyone,

Welcome to the **brand-new** (and dare I say, *improved*) version of **IEMS 402**! I'm excited, you're excited, and even my cat is excited (though she's mostly excited for treats). This course is going to be a lot of fun, full of intriguing discoveries, many "aha!" moments, and, well, a few puffs of chalk dust here and there.

ChatGPT

A New Version of 402 – Your Feedback is Golden

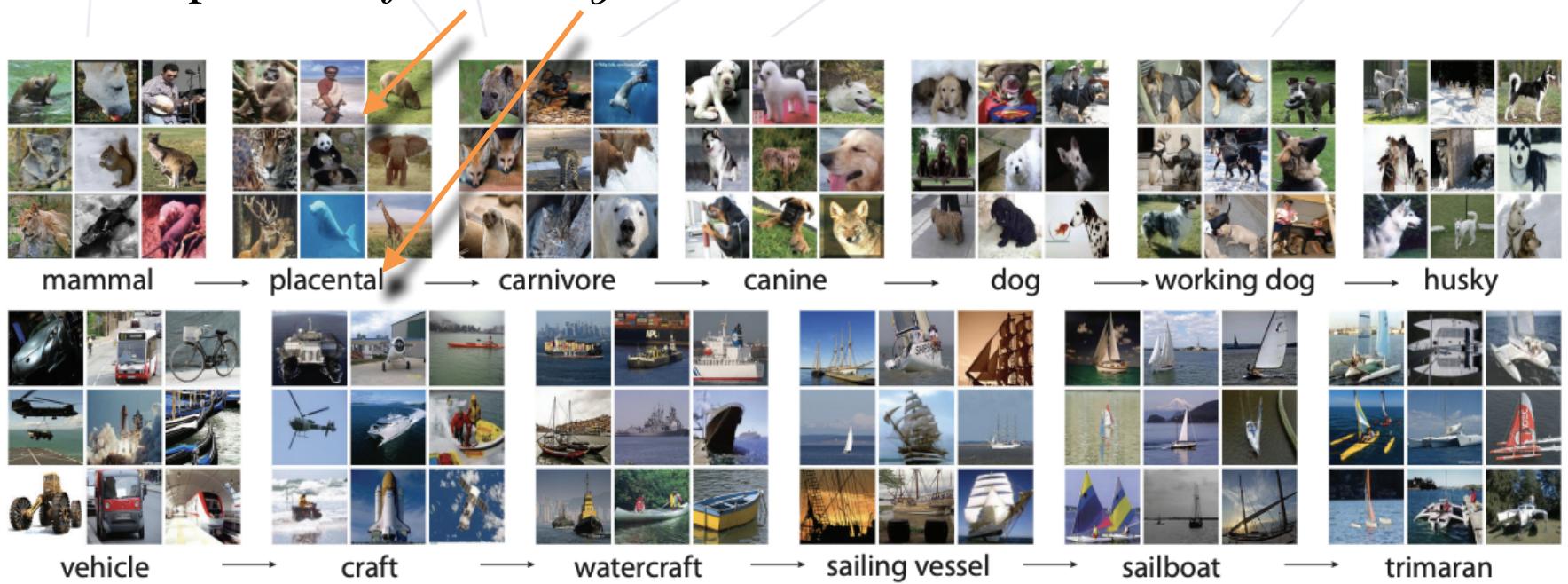
Because this is a fresh, sparkling-new take on **402**, I will be collecting **anonymous opinions after each course session**. I genuinely want to hear your thoughts—both the accolades and the grievances (go easy on me!). Your feedback will help shape how I refine and tweak the course going forward. So please speak up, share your impressions, and let me know what's working well or what you'd like to see improved.

Course Website:
2025-Statistical-Learning

Supervised Learning

Supervised Learning

- Aim: learn a predictor $f: \mathcal{X} \rightarrow \mathcal{Y}$



PAC Learning Model

- **Input: Training Data.** $S = \{(x_1, y_1), \dots (x_m, y_m)\}$ is a finite set of pairs in $\chi \times \mathcal{Y}$. This is the *input* that the learner has access to. Such labeled examples are also referred to as *training examples* or *labeled sample set*. The size of the sample set m is the *sample size*. We will generally assume that the sample S was generated by drawing m IID samples from the distribution D .
 - **Output: Hypothesis.** A Hypothesis class consists of a subset of target functions $\mathcal{H} = \{h : h : \chi \rightarrow \mathcal{Y}\}$ that turns unlabeled samples to labels. Each learning algorithm outputs a hypothesis, the class of hypotheses the learner may return is the algorithms hypothesis class.

ML Alg: input : m data output : a function $h: x \rightarrow y$

PAC Learning Model

- **Input: Training Data.** $S = \{(x_1, y_1), \dots, (x_m, y_m)\}$ is a finite set of pairs in $\chi \times \mathcal{Y}$. This is the *input* that the learner has access to. Such labeled examples are also referred to as *training examples* or *labeled sample set*. The size of the sample set m is the *sample size*. We will generally assume that the sample S was generated by drawing m IID samples from the distribution D .
- **Output: Hypothesis.** A Hypothesis class consists of a subset of target functions $\mathcal{H} = \{h : h : \chi \rightarrow \mathcal{Y}\}$ that turns unlabeled samples to labels. Each learning algorithm outputs a hypothesis, the class of hypotheses the learner may return is the algorithms hypothesis class.



Our Goal

Definition 1.1 ((realizable) PAC Learning). A concept class C of target functions is PAC learnable (w.r.t to \mathcal{H}) if there exists an algorithm A and function $m_C^A : (0, 1)^2 \rightarrow \mathbb{N}$ with the following property:

Assume $S = ((x_1, y_1), \dots, (x_m, y_m))$ is a sample of IID examples generated by some arbitrary distribution D such that $y_i = h(x_i)$ for some $h \in C$ almost surely. If S is the input of A and $m > m_C^A(\epsilon, \delta)$ then the algorithm returns a hypothesis $h_S^A \in \mathcal{H}$ such that, with probability $1 - \delta$ (over the choice of the m training examples):

$$\text{err}(h_S^A) < \epsilon$$

How to define error?

The function $m_C^A(\epsilon, \delta)$ is referred to as the sample complexity of algorithm A .



Supervised Learning

- Aim: learn a predictor $f : \mathcal{X} \rightarrow \mathcal{Y}$
- What is a good predictor? -> evaluation criteria

$$\mathcal{R}(f) = \mathbb{E}[\ell(y, f(x))] = \int_{\mathcal{X} \times \mathcal{Y}} \ell(y, f(x)) dp(x, y).$$

Assume data sample from a distribution p

Ex. classification 0-1 loss

$$\ell(y, f(x)) = \begin{cases} 0 & y = f(x) \\ 1 & y \neq f(x) \end{cases}$$

Ex. Regression

$$\ell(y, f(x)) = (f(x) - y)^2$$

RMSE

Supervised Learning

- Aim: learn a predictor $f : \mathcal{X} \rightarrow \mathcal{Y}$
- What is a good predictor? -> evaluation criteria

$$\mathcal{R}(f) = \mathbb{E}[\ell(y, f(x))] = \int_{\mathcal{X} \times \mathcal{Y}} \ell(y, f(x)) dp(x, y).$$

Evaluate the error of label and prediction

Good habit: " $\hat{\lambda}$ " means empirical

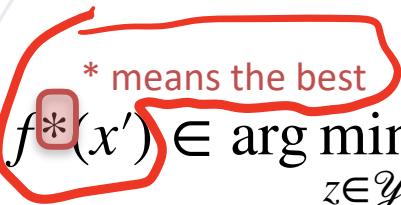
If I want to know the risk, I need to have all the data in the univers?

Empirical Risk: $\hat{\mathcal{R}}(f) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, f(x_i))$, where $\{(x_i, y_i)\}_{i=1}^n$ is a collected dataset

Conditional Risk

$$\mathcal{R}(f) = \mathbb{E}_{x' \sim p} \left[\mathbb{E} \left[\ell(y, f(x')) \mid x = x' \right] \right] = \int_{\mathcal{X}} \mathbb{E} \left[\ell(y, f(x')) \mid x = x' \right] dp(x').$$

Conditional Risk: $r(z \mid x') = \mathbb{E} \left[\ell(y, z) \mid x = x' \right]$

- Bayes Predictor: $f^*(x') \in \arg \min_{z \in \mathcal{Y}} \mathbb{E} \left[\ell(y, z) \mid x = x' \right] = \arg \min_{z \in \mathcal{Y}} r(z \mid x')$.


Conditional Risk

$$\mathcal{R}(f) = \mathbb{E}_{x' \sim p} \left[\mathbb{E} \left[\ell(y, f(x')) \mid x = x' \right] \right] = \int_{\mathcal{X}} \underbrace{\mathbb{E} \left[\ell(y, f(x')) \mid x = x' \right]}_{\text{Conditional Risk: } r(z|x')} dp(x').$$

Conditional Risk: $r(z|x') = \mathbb{E} \left[\ell(y, z) \mid x = x' \right]$

$$r(z|x') = \mathbb{E}[(y - z)^2 \mid x = x'] \leftarrow z^* = \mathbb{E}[y \mid x = x']$$

- Bayes Predictor: $f^*(x') \in \arg \min_{z \in \mathcal{Y}} \mathbb{E} \left[\ell(y, z) \mid x = x' \right] = \arg \min_{z \in \mathcal{Y}} r(z|x') .$



What is the Bayes Predictor of ℓ_2 loss or ℓ_1 loss?

mean

median

Homework,
pinball loss
→ quartile

How to design a loss function

- Method 1: Know what is your Bayes Predictor! [Homework 1 Question 1.](#)

How to design a loss function

- Method 1: Know what is your Bayes Predictor! [Homework 1 Question 1.](#)
- Method 2: Use Max likelihood
 - Step 1: understand what is your $p(y|x)$, e.g. Gaussian, heavy tail distribution
 - Step 2: What is the log-likelihood of dataset $\{(x_i, y_i)\}_{i=1}^n$?

How to design a loss function

- Method 1: Know what is your Bayes Predictor! [Homework 1 Question 1.](#)
- Method 2: Use Max likelihood
 - Step 1: understand what is your $p(y|x)$, e.g. Gaussian, heavy tail distribution
 - Step 2: What is the log-likelihood of dataset $\{(x_i, y_i)\}_{i=1}^n$?
 - $\log \prod_{i=1}^n p(y_i|x_i) = \sum_{i=1}^n \log p(y_i|x_i)$
 - Step 3: use $\log p(\cdot|x_i)$ as your loss function!



How can I get the ℓ_2 loss using this methods?

Sum! like Empirical Risk

$$\log \prod_{i=1}^n p(y_i|x_i) = \sum_{i=1}^n \log p(y_i|x_i)$$

ℓ_2 loss means Gaussian Mle

$$p(y|x) = N(f(x), \sigma^2 I)$$

Example: Logistic Regression

Consider a binary classification with $p(y_i = 1 \mid \mathbf{x}_i, \theta) = \sigma(\mathbf{x}_i^\top \theta) = \frac{1}{1 + e^{-\mathbf{x}_i^\top \theta}}$

Example: Gaussian with Learned Variance

Example (*Gaussian with Learned Variance Leads to Sparsity*)

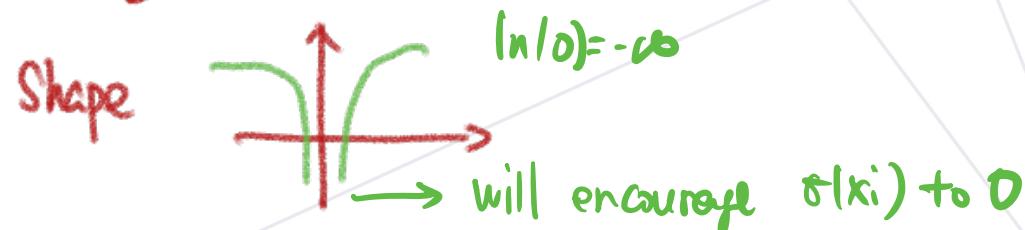
Not Required

$$\ell(\mu, \sigma^2) = \sum_{i=1}^n \log P(y_i | \mu(x_i), \sigma(x_i)^2)$$

$$\begin{aligned} P(x) &= \frac{1}{\sigma \sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \\ &= \sum_{i=1}^n \left(-\frac{1}{2} \log(2\pi) - \frac{1}{2} \log(\sigma(x_i)^2) - \frac{(y_i - \mu(x_i))^2}{2\sigma(x_i)^2} \right) \end{aligned}$$

ℓ_2 loss
weight by variance

$$= -\frac{n}{2} \ln(2\pi) - \underbrace{\frac{n}{2} \ln(\sigma(x_i)^2)}_{\text{sparse regularization}} - \underbrace{\sum_{i=1}^n \frac{(y_i - \mu(x_i))^2}{2\sigma(x_i)^2}}_{\text{weighted } \ell_2 \text{ loss}}$$



Empirical Risk Minimization



I want an estimator to minimize the risk, but I can only get the empirical risk? What's the best thing I can do?

- Consider a parameterized family of prediction functions (often referred to as models) $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$, e.g.
 - Linear prediction $f(x) = \langle x, \theta \rangle$
 - Neural Network
- Empirical Risk Minimization: $\hat{\theta} \in \hat{\mathcal{R}}(f_\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, f_\theta(x_i))$.

θ : parameter

$f(x) = \langle x, \theta \rangle$

$\hat{\mathcal{R}}$ means empirical

$\hat{\mathcal{R}}$

Pro and Con of ERM

- Pro:
 - Flexible
 - Algorithms are available (e.g. SGD)
- Con:
 - can be relatively hard to optimize when the optimization formulation is not convex (e.g.,neural networks); *(? in real application)*
 - the dependence on parameters can be complex (e.g., neural networks);
 - need some capacity control to avoid overfitting *(? in real application)*

Our course is about overfitting!

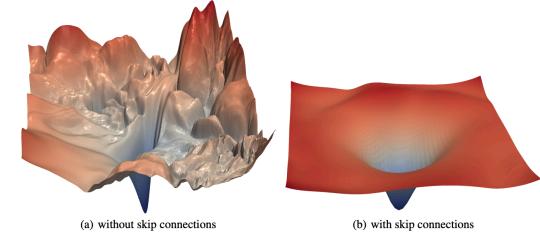


Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

The only theorem: Risk Decomposition

$$\mathcal{R}(f_{\hat{\theta}}) - \mathcal{R}^* = \left\{ \mathcal{R}(f_{\hat{\theta}}) - \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) \right\} + \left\{ \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) - \mathcal{R}^* \right\}$$

The equation is shown with two blue curly braces underlining the terms. The first brace underlines the expression $\mathcal{R}(f_{\hat{\theta}}) - \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'})$. The second brace underlines the expression $\inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) - \mathcal{R}^*$. Below the first brace is the label "Estimation error" and below the second is "Approximation error".

The only theorem: Risk Decomposition

$$\mathcal{R}(f_{\hat{\theta}}) - \mathcal{R}^* = \left\{ \mathcal{R}(f_{\hat{\theta}}) - \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) \right\} + \left\{ \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) - \mathcal{R}^* \right\}$$

Estimation error Approximation error

$\theta^* = \arg \min_{\theta} R(f_{\theta}) \rightarrow \text{best parameter}$

For an ERM Estimator: II

$$\mathcal{R}(f_{\hat{\theta}}) - \hat{\mathcal{R}}(f_{\hat{\theta}}) + \hat{\mathcal{R}}(f_{\hat{\theta}}) - \hat{\mathcal{R}}(f_{\theta^*}) + \hat{\mathcal{R}}(f_{\theta^*}) - \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'})$$

Generalization error Optimization error Generalization error

≤ 0

Question:

$$\Theta_1 \subseteq \Theta_2$$

Approximation Error
of Θ_1 is larger!

The only theorem: Risk Decomposition

$$\mathcal{R}(f_{\hat{\theta}}) - \mathcal{R}^* = \underbrace{\left\{ \mathcal{R}(f_{\hat{\theta}}) - \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) \right\}}_{\text{Estimation error}} + \underbrace{\left\{ \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) - \mathcal{R}^* \right\}}_{\text{Approximation error}}$$

For an ERM Estimator: $\|$

$$\mathcal{R}(f_{\hat{\theta}}) - \hat{R}(f_{\hat{\theta}}) + \hat{R}(f_{\hat{\theta}}) - \hat{\mathcal{R}}(f_{\theta^*}) + \hat{\mathcal{R}}(f_{\theta^*}) - \inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'})$$

Generalization error Optimization error Generalization error

≤ 0

$\leq 2 \sup_{\theta \in \Theta} |R(f_{\theta}) - \hat{R}(f_{\theta})|$

Uniform Bound!

Question:
 $\mathbb{Q}_1, \mathbb{E} \mathbb{Q}_2$
 Generalization Bound
 of \mathbb{Q}_2 is bigger.

No Free Lunch Theorem

Let A be any learning algorithm for the task of binary classification with respect to the 0/1-loss function over a domain \mathcal{X} . Let $m < \frac{|\mathcal{X}|}{2}$ be a number representing a training set size.

There exists a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ such that:

- there exists a function $f : \mathcal{X} \rightarrow \{0, 1\}$ with $L_{\mathcal{D}}(f) = 0$;
- with probability at least $1/7$ over the choice of a sample $S \sim \mathcal{D}^m$ (of size m) we have that $L_{\mathcal{D}}(A(S)) \geq 1/8$.

Need Assumption over data .

<https://www.cs.cornell.edu/courses/cs6783/2015fa/lec3.pdf>

No Free Lunch Theorem

Let A be any learning algorithm for the task of binary classification with respect to the 0/1-loss function over a domain \mathcal{X} . Let $m < \frac{|\mathcal{X}|}{2}$ be a number representing a training set size.

There exists a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ such that:

- there exists a function $f : \mathcal{X} \rightarrow \{0, 1\}$ with $L_{\mathcal{D}}(f) = 0$;
- with probability at least $1/7$ over the choice of a sample $S \sim \mathcal{D}^m$ (of size m) we have that $L_{\mathcal{D}}(A(S)) \geq 1/8$.



How to formulate $A(S)$ in math?

<https://www.cs.cornell.edu/courses/cs6783/2015fa/lec3.pdf>

No Free Lunch Theorem

$$\max_{1 \leq i \leq |T|} E_{S \sim \mathcal{D}^m}(L_{D_i}(A(S))) \geq \frac{1}{4}.$$

This means that for every \mathcal{A}' that receives a training set of m examples from $\mathcal{X} \times \{0, 1\}$ there exists $f : \mathcal{X} \rightarrow \{0, 1\}$ and a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ such that $L_{\mathcal{D}}(f) = 0$ and $E_{S \sim \mathcal{D}^m}(L_{\mathcal{D}}(\mathcal{A}'(S))) \geq \frac{1}{4}$.

No Free Lunch Theorem

$$\max_{1 \leq i \leq |T|} E_{S \sim \mathcal{D}^m}(L_{D_i}(A(S))) \geq \frac{1}{4}.$$

This means that for every \mathcal{A}' that receives a training set of m examples from $\mathcal{X} \times \{0, 1\}$ there exists $f : \mathcal{X} \rightarrow \{0, 1\}$ and a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ such that $L_{\mathcal{D}}(f) = 0$ and $E_{S \sim \mathcal{D}^m}(L_{\mathcal{D}}(\mathcal{A}'(S))) \geq \frac{1}{4}$.

No Free Lunch Theorem

Let A be any learning algorithm for the task of binary classification with respect to the 0/1-loss function over a domain \mathcal{X} . Let $m < \frac{|\mathcal{X}|}{2}$ be a number representing a training set size.

There exists a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ such that:

- there exists a function $f : \mathcal{X} \rightarrow \{0, 1\}$ with $L_{\mathcal{D}}(f) = 0$;
- with probability at least $1/7$ over the choice of a sample $S \sim \mathcal{D}^m$ (of size m) we have that $L_{\mathcal{D}}(A(S)) \geq 1/8$.



Important to know what's the implicit assumption on target function

<https://www.cs.cornell.edu/courses/cs6783/2015fa/lec3.pdf>

Review

bayes Risk

$$R(f_{\hat{\theta}}) - R^* = \{R(f_{\hat{\theta}}) - R(f_{\theta^*})\} + \{R(f_{\theta^*}) - R^*\}$$

$$\theta^* = \operatorname{argmin}_{\theta \in \Theta} R(f_\theta)$$

Estimation

Approximation

$$R(f_{\hat{\theta}}) - \hat{R}(f_{\hat{\theta}}) + \hat{R}(f_{\hat{\theta}}) - \hat{R}(f_{\theta^*}) + \hat{R}(f_{\theta^*}) - R(f_{\theta^*})$$

Difference between Empirical and population

uniform Bound

$$\sup_{\theta \in \Theta} |\hat{R}(f_{\theta}) - R(f_{\theta})|$$

generalization

	\mathcal{H}_1	\subseteq	\mathcal{H}_2
Approx	large		Small
Gen	small		large

"bias - Var" Trade-off

Difference between 401 and 402

Statistics

Learning

- Difference 1: Parameter Convergence and Risk Convergence

$$\hat{\theta} \rightarrow \theta^*$$

Stats

$$R(f_{\hat{\theta}}) \rightarrow R(f_{\theta^*})$$

machine learning



- Difference 2: Parametric and Non-parametric

① Assumption over data

ex. $y = \langle \theta, x \rangle + \epsilon$, $\epsilon \sim N(0, 1)$

"minimum Ans"

ex. $y = f(x)$, f is smooth

f is learnable

②

→ ① 1 data, $y = \langle \theta, x \rangle$

10^5 data

$y = \langle \theta, x \rangle$

↪ ② 10^5 data

Small NN
Large NN

You use a parameterized family in Empirical risk minimization, why you call "non-parametric"?

Hardness of ERM

Error of ERM

IEMS 402 Focus

Assume to be 0

Approximation Error + Generalization Error + Optimization Error

$$\inf_{\theta' \in \Theta} \mathcal{R}(f_{\theta'}) - R^*$$

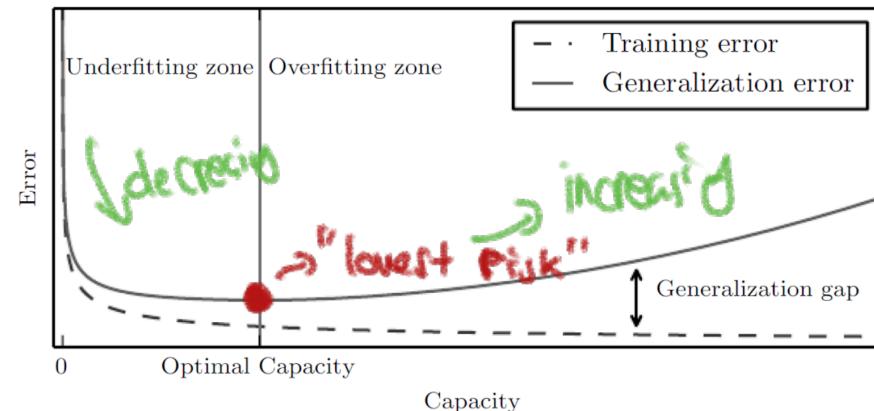
$$\sup_{\theta \in \Theta} |R(f_\theta) - \hat{R}(f_{\hat{\theta}})| \rightarrow \text{"Empirical Process"}$$

Weak convergence and Empirical Process

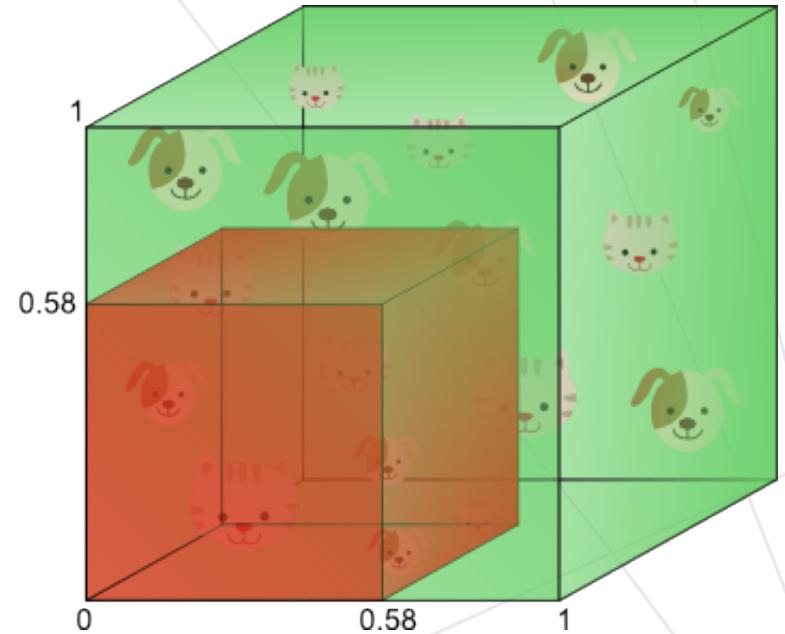
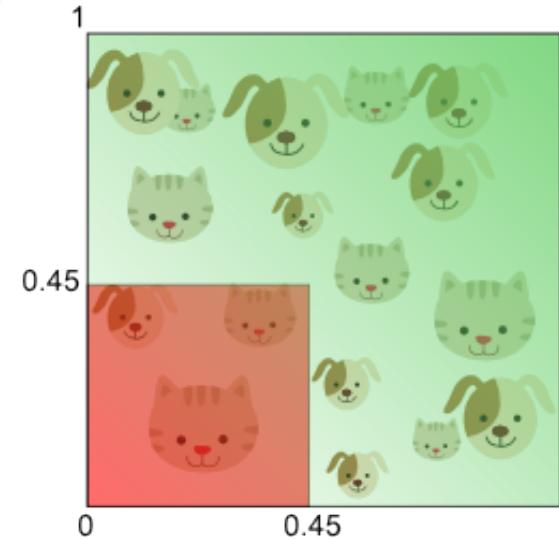
When we use more powerful parameterized family, e.g. Θ is larger:

- Approximation error is smaller!
- Generalization error is larger!

Bias-Variance Trade-off



Approximation: Curse of Dimensionality



Formulation: Approximate a smooth function

Fact. The number of parameters N required to achieve an approximation error of at most ϵ can be estimated by:

s -th order Smoothness:

$$f(x) = f(x_0) + f'(x_0)(x-x_0) + \dots + f^{(s)}(x_0)(x-x_0)^s + O(\|x-x_0\|^{s+1})$$



$$N \approx \left(\frac{1}{\epsilon}\right)^{\frac{d}{s}}$$

Dimension
smoothness

$$N = d^s \cdot \left(\frac{1}{\epsilon}\right)^{\frac{d}{s}}$$

- ① $(0, y_t)^d$ How many I have. $+^d$
- ② on each cube, replace the function by s -order Taylor Expansion
 - ① How many parameter?
 - ② Error? $\left(\frac{1}{\epsilon}\right)^s = \epsilon$

$$d^s \cdot +^d = N$$

- Another Formulation see Homework 1 Question 3.

Formulation: Approximate a smooth function

Fact. The number of parameters N required to achieve an approximation error of at most ϵ can be estimated by:

$$N \approx \left(\frac{1}{\epsilon} \right)^{\frac{d}{s}}$$

Dimension
smoothness

$$N = d^s \left(\frac{1}{\epsilon} \right)^{\frac{d}{s}}$$

① s is a constant.

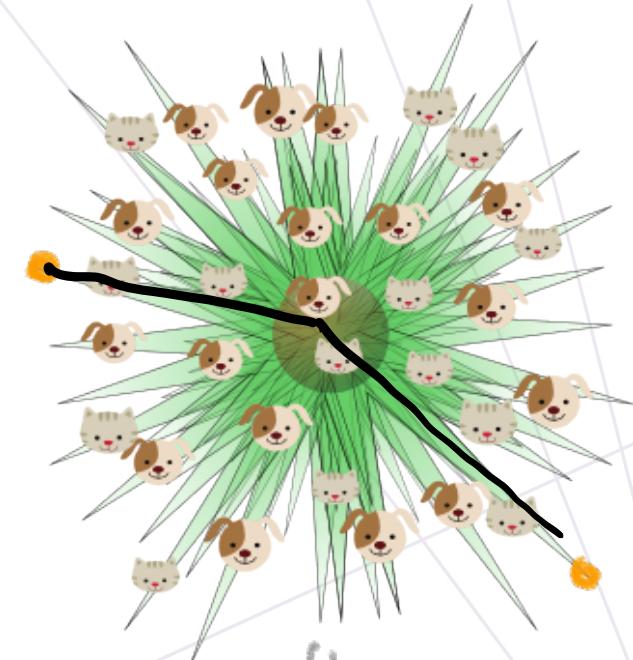
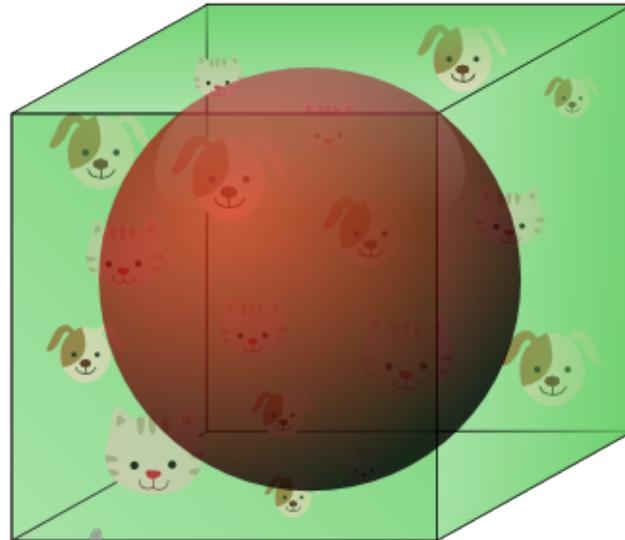
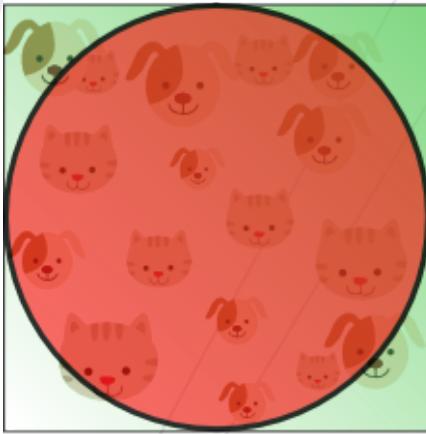
$$N \propto \left(\frac{1}{\epsilon} \right)^d$$

$s \propto d$, " d^s is a exp to d
but it's a good poly respect
to ϵ "

- Another Formulation see Homework 1 Question 3.

How to think about High Dimension

2^n

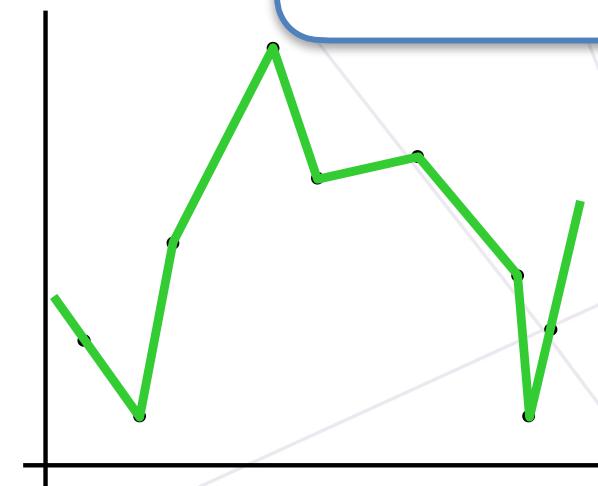
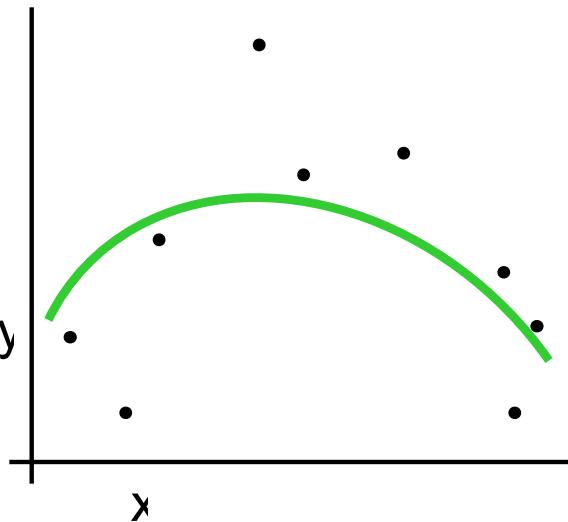
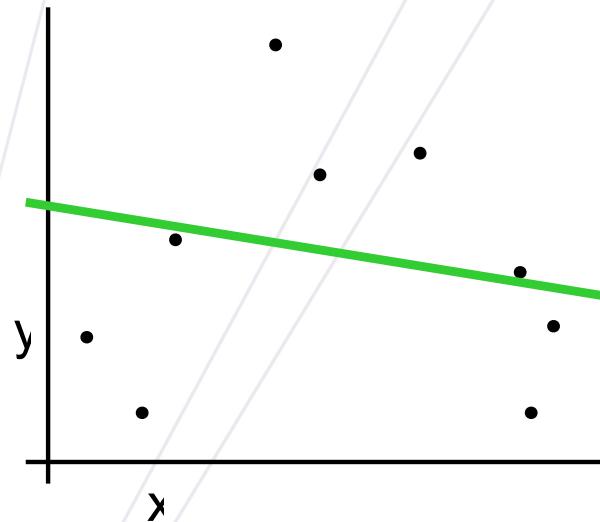


"Two point's distance is a constant"

Generalization: Overfitting?

$$y = f(x) + \text{noise}$$

Can we learn f from this data?



Repeated Parrot
vs
understanding

Degree of Freedom

Suppose that we observe $y_i = r(x_i) + \epsilon_i (i = 1, \dots, n)$, where the errors ϵ_i are uncorrelated with common variance $\sigma^2 > 0$

Now consider the fitted values $\hat{y}_i = \hat{r}(x_i)$ from a regression estimator \hat{r} .

Degree of freedom is defined as
$$df(\hat{y}) = \frac{1}{\sigma^2} \sum_{i=1}^n \text{Cov}(\hat{y}_i, y_i).$$

“How much I remember the label”

Degree of freedom

Fact. $\mathbb{E} \left[\frac{1}{n} \sum_{i=1}^n (y'_i - \hat{y}_i)^2 \right] - \mathbb{E} \left[\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \right] = \frac{2\sigma^2}{n} \text{df}(\hat{y}).$

Generalization error

$$\mathbb{E} \left[\frac{1}{n} \sum_{i=1}^n \underbrace{2(y'_i - y_i) (\hat{y}_i - y_i)}_{2(e_i - e'_i)(\hat{y}_i - y_i)} + \underbrace{(y'_i - y_i)^2}_{2\sigma^2} \right]$$

$$y'_i = r(x_i) + e'_i$$

$$y_i = r(x_i) + e_i$$

independent

$$2e_i(\hat{y}_i - y_i) = 2\text{cov}(y_i, \hat{y}_i) \xrightarrow{2\sigma^2}$$

n labels for n data

Example of DOF 1

① $\hat{Y}_i = Y_i$ $df = \frac{1}{\sigma^2} \sum_{i=1}^n \text{Gv}(\hat{Y}_i, Y_i) = n$

② $\hat{Y}_i = \bar{Y} = \frac{1}{n} \sum_{i=1}^n Y_i$ $df = \frac{1}{\sigma^2} \sum_{i=1}^n \text{Gv}(\hat{Y}_i, Y_i) = 1$

1 label for n data.

"DOF is the number of parameters we are using"

Example of DOF 2

③ linear Regression

$$\mathbb{R}^n \rightarrow \hat{\gamma} = X\beta = X \left[\begin{matrix} (X^T X)^{-1} X^T Y \end{matrix} \right]$$

Not Required

$$\frac{1}{\sigma^2} \text{tr} \left[\text{cov} \left(\underbrace{X(X^T X)^{-1} X^T Y}_{\mathbb{R}^n}, Y \right) \right]$$

$$= \frac{1}{\sigma^2} \text{tr} \left[X(X^T X)^{-1} X^T \text{cov}(Y, Y) \right]$$

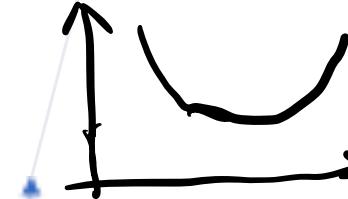
$$= \text{tr} \left(X(X^T X)^{-1} X^T \right) = \text{tr} \left(X^T X (X^T X)^{-1} \right) = \text{tr} \left(I_p \right)$$

$\text{tr}(AB) = \text{tr}(BA)$

"number of parameters"

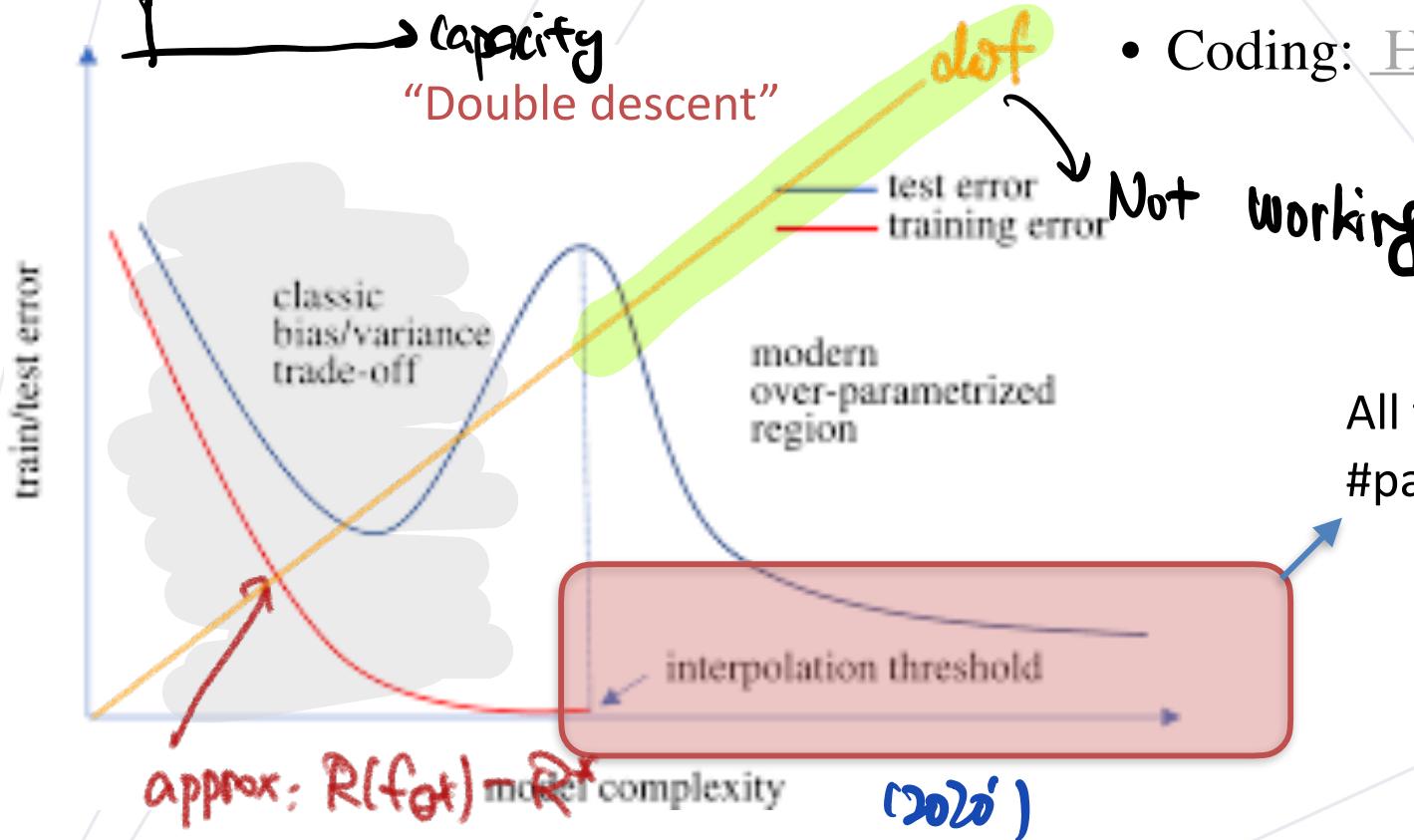
P
"

U-shape curve



However...

- Coding: [Homework 2 Question 3.](#)

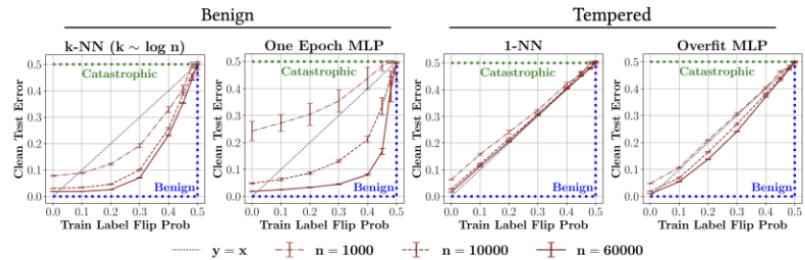
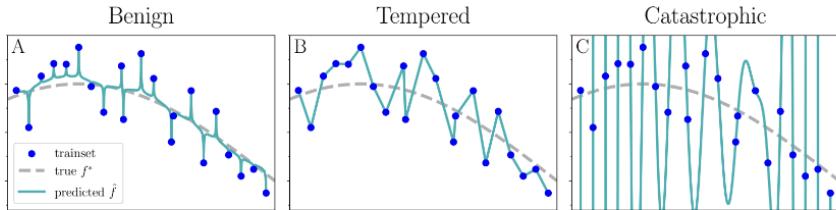


All the data can be remembered
#parameter > #data

Taxonomy of (over)fitting

	Regression	Classification
Benign	$\lim_{n \rightarrow \infty} \mathcal{R}_n = R^*$	$\lim_{n \rightarrow \infty} \mathcal{R}_n = R^*$
Tempered	$\lim_{n \rightarrow \infty} \mathcal{R}_n \in (R^*, \infty)$	$\lim_{n \rightarrow \infty} \mathcal{R}_n \in (R^*, 1 - \frac{1}{K})$
Catastrophic	$\lim_{n \rightarrow \infty} \mathcal{R}_n = \infty$	$\lim_{n \rightarrow \infty} \mathcal{R}_n = 1 - \frac{1}{K}$

Table -1.1: Taxonomy of (over)fitting.



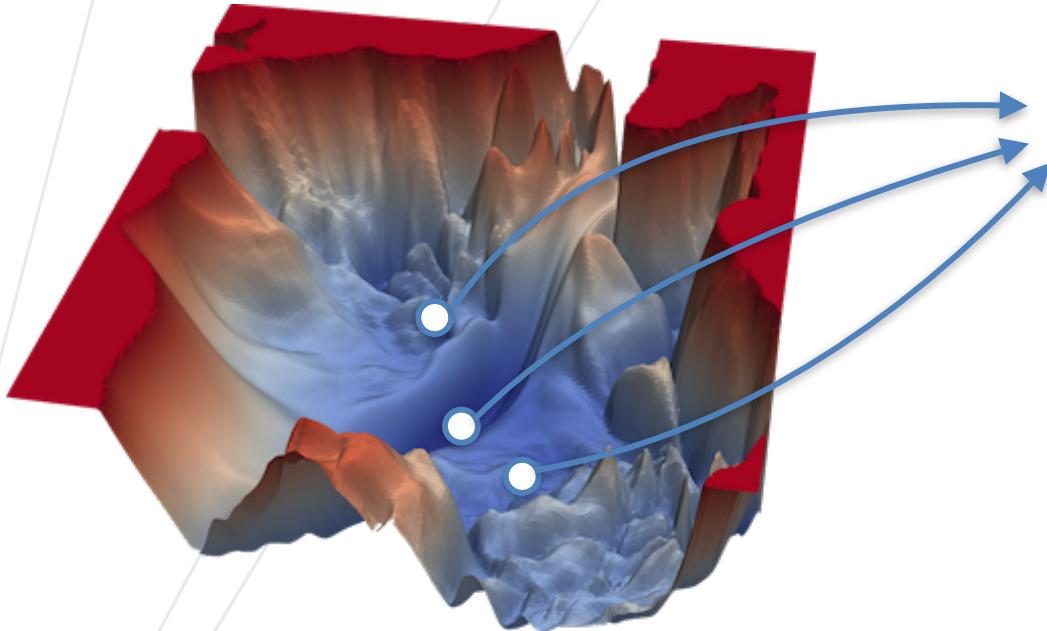
Mallinar, Neil, et al. "Benign, tempered, or catastrophic: A taxonomy of overfitting (2022)." arXiv preprint arXiv:2207.06569.

CIFAR100 : 6000

ResNe xt : #parameters
68.1M

Implicit bias

“Multiple Minima”

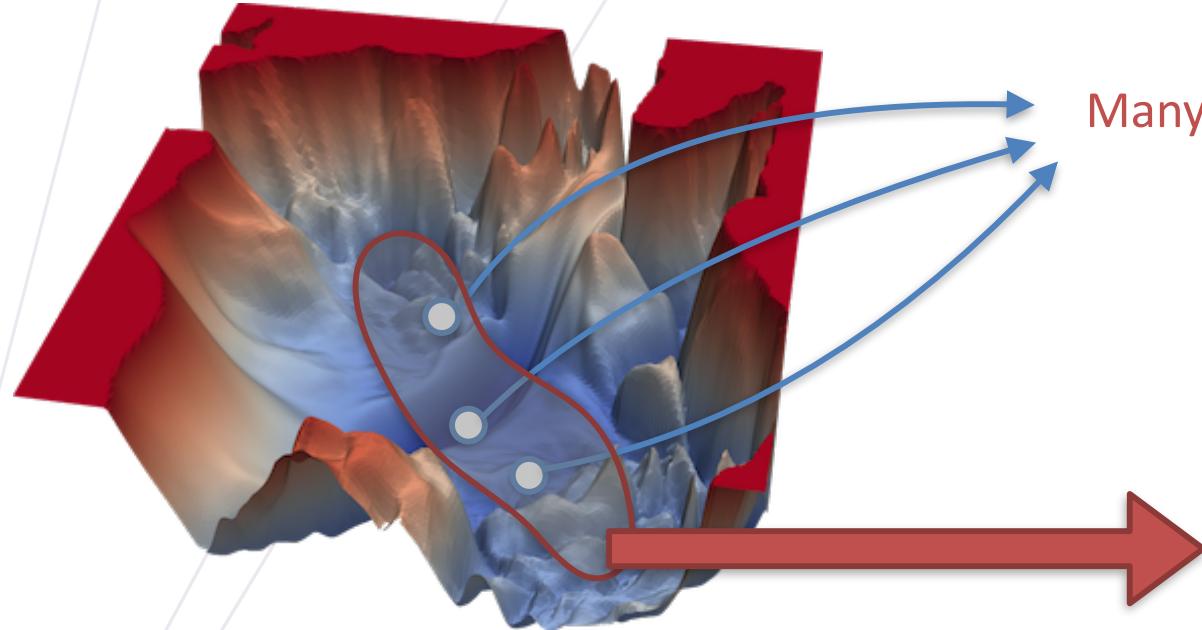


Many models can achieve low training loss

Loss landscape of VGG on CIFAR

Implicit bias

“Multiple Minima”



Loss landscape of VGG on CIFAR

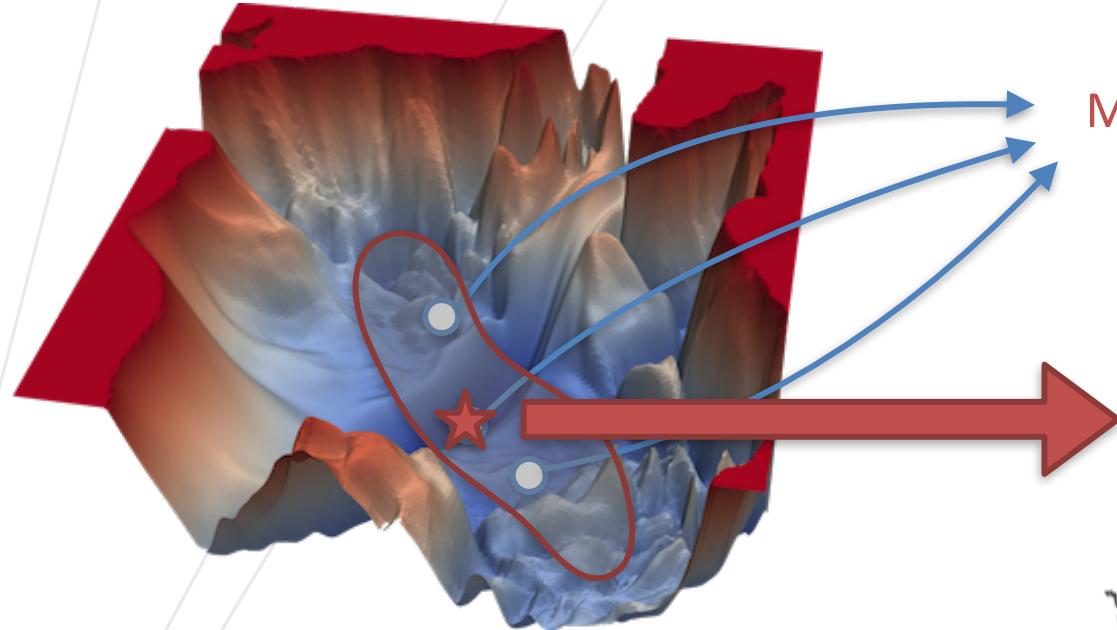
Traditional bounds:

$$\sup_{\theta \in \Theta} |R(f_\theta) - \hat{R}(f_{\hat{\theta}})|$$

too pessimistic

Implicit bias

“Multiple Minima”



Loss landscape of VGG on CIFAR

Many models can achieve low training loss

CORE PRINCIPLES IN RESEARCH



OCCAM'S RAZOR

"WHEN FACED WITH TWO POSSIBLE EXPLANATIONS, THE SIMPLER OF THE TWO IS THE ONE MOST LIKELY TO BE TRUE."

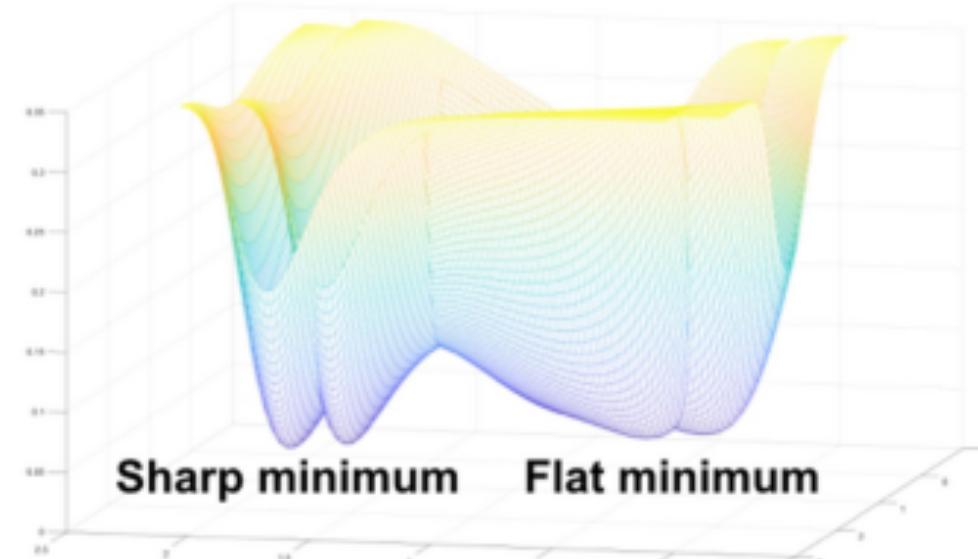
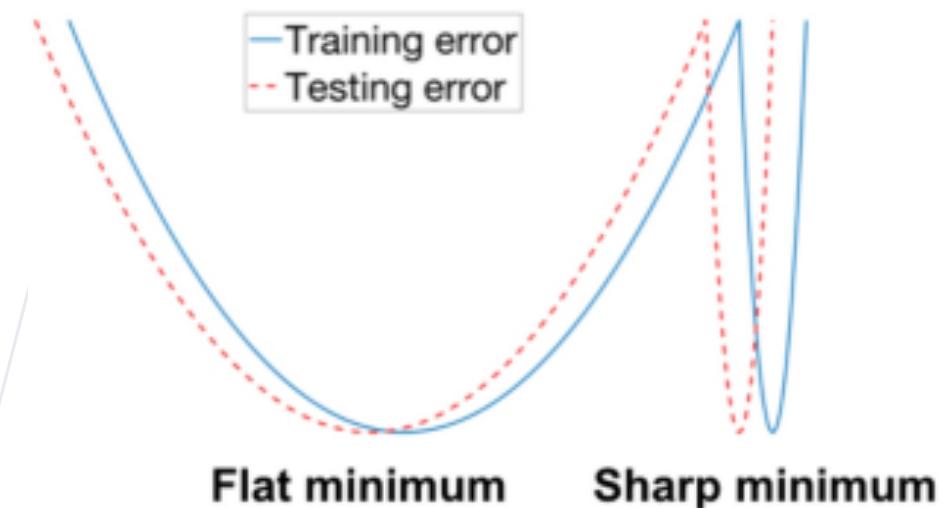


OCCAM'S PROFESSOR

"WHEN FACED WITH TWO POSSIBLE WAYS OF DOING SOMETHING, THE MORE COMPLICATED ONE IS THE ONE YOUR PROFESSOR WILL MOST LIKELY ASK YOU TO DO."

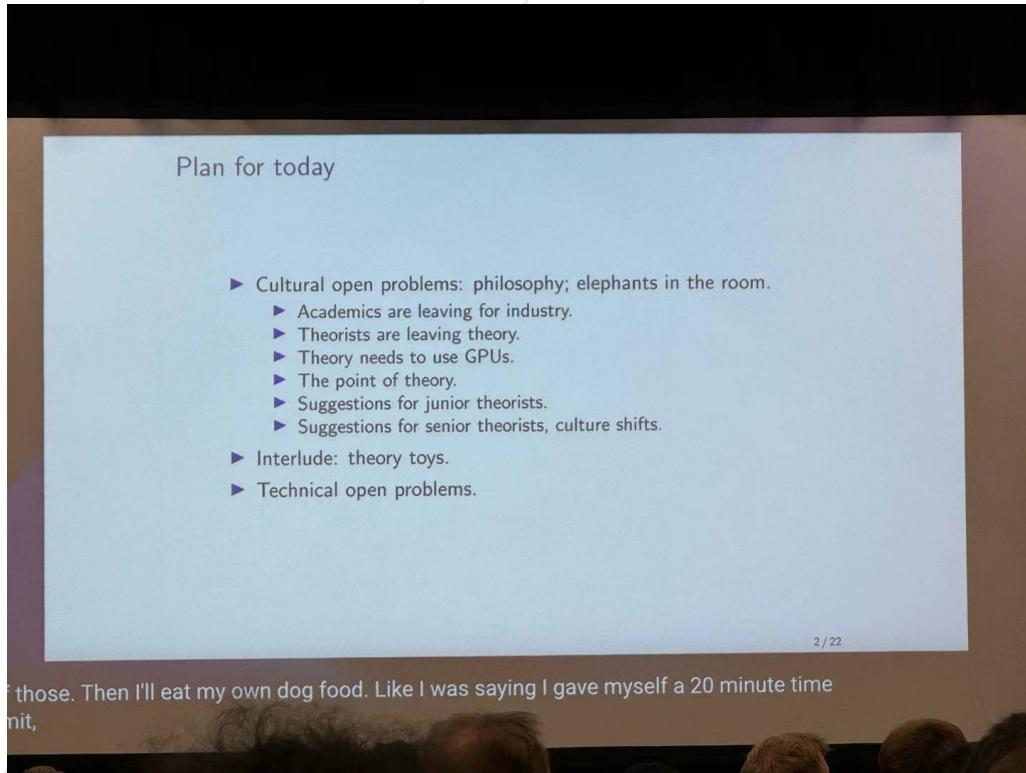
What's special about over-parametrization?

“Multiple Minima”



Last Note on Learning Theory

ML Theory workshop @Neurips24



- ▶ Cultural open problems: philosophy; elephants in the room.
 - ▶ Academics are leaving for industry.
 - ▶ Theorists are leaving theory.
 - ▶ Theory needs to use GPUs.
 - ▶ The point of theory.
 - ▶ Suggestions for junior theorists.
 - ▶ Suggestions for senior theorists, culture shifts.
- ▶ Interlude: theory toys.
- ▶ Technical open problems.

2 / 22

if those. Then I'll eat my own dog food. Like I was saying I gave myself a 20 minute time
limit,

[https://cims.nyu.edu/~matus/
neurips.2024.workshop/talk.pdf](https://cims.nyu.edu/~matus/neurips.2024.workshop/talk.pdf)

Math-physics-ethology

Theory of Language Models

math

Pros: Learning
math
L: linear space
H: Hilbert space
A: abstract algebra
D: differential calc
O: optimization
S: statistics
C: complex analysis
E: error analysis
N: numerical methods
P: probability theory
R: real analysis
T: topology
F: functional analysis
G: group theory
M: measure theory
I: information theory
B:博弈论 (Game Theory)

mathematics + learning theory
(concept class, data, model, assumptions, learnability theorems)

Pros: rigorous, theorem!

Cons:
assumptions might be too *idealistic*;
networks may be too *shallow*;
only in rare cases theorems *connect to practice*;
even if...people may not read your paper...
(e.g., "none" of the LoRA users knew we had a FOCS paper before it to study lora-rankness in feature learning...)

"ethology"

animal behavior science

GPT4
GPT4-mini

(chain-of-thought, tree-of-thought, etc.)

the theorems that you prove really do connect to practice, and even if it does people may not read

ICML 2024 International Conference On Machine Learning

1:59 / 1:53:42 • Prelude >

1316 | 分享 | 下载 | 感谢 | ...

ICML 2024 Tutorial: Physics of Language Models

Zeyuan Allen-Zhu, Sc.D.
4040位订阅者

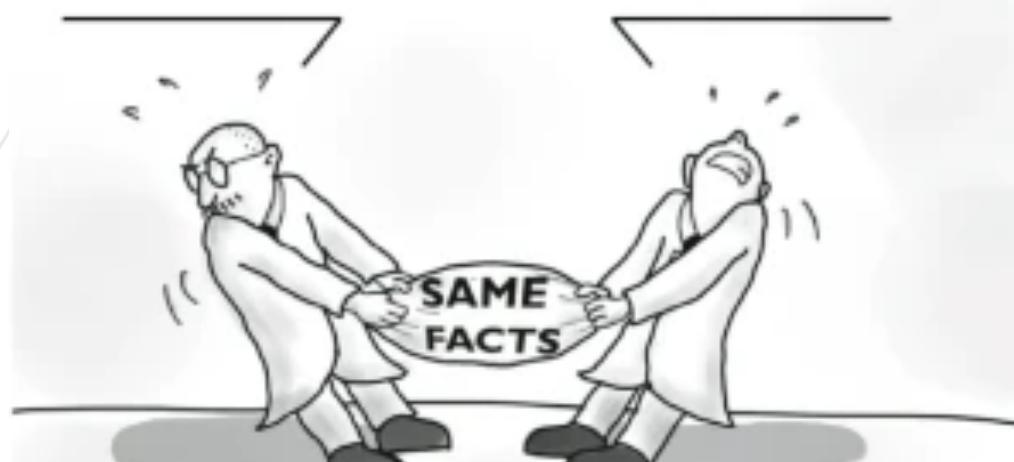
订阅

Physics of language model
ICML 2024
<https://shorturl.at/ZDwQE>

Learning Theory Today

These facts prove
MY theory!

No! These facts
prove MY theory!



MD²