Rethinking Generalization in Deep Learning:
Double Descent and Grokking Phenomena

Pascal Jr. Tikeng Notsawo [1,2,3]

pascal.tikeng@mila.quebec

[1]Université de Montréal, Montréal, Quebec, Canada

[2]Mila, Quebec Artificial Intelligence Institute

[3]ENSPY alumni (2021)

Deep Learning IndabaX Cameroon:
Machine Learning for Sustainable Development


Mila

Outline

1. Important concepts
2. Classical Bias variance tradeoff
3. What statistical learning doesn't tell us (& PAC learning)
4. Double Descent (model-wise, data-wise and epoch-wise)
5. Grokking (epoch-wise)
6. Why is it important to study such phenomena? (AI safety, OOD generalization ...)

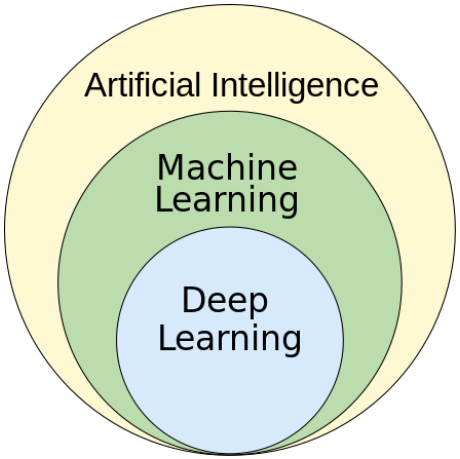
Part I - Machine Learning, Deep learning

ML: Machine Learning (Wikipedia)

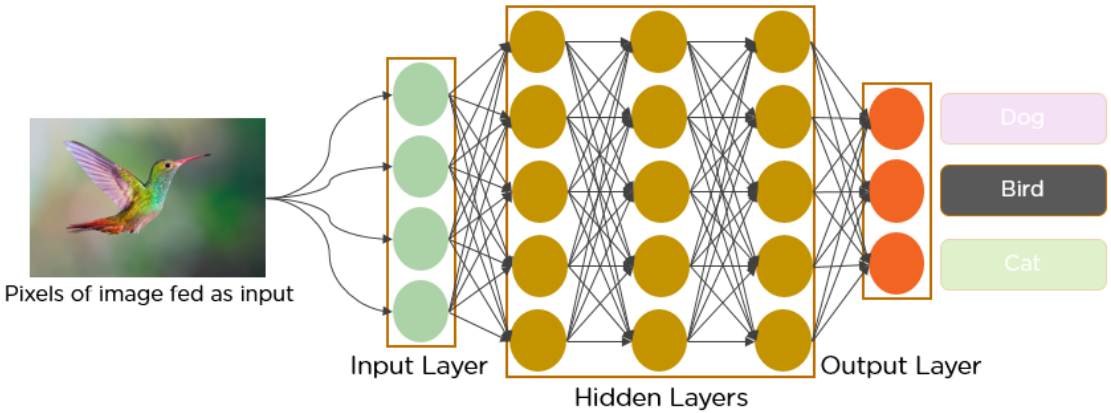
- Train a model on some training data
- This trained model can process additional data to make predictions

DL: Deep Learning

ML with a (deep) neural network as a model



ML as a subfield of AI (Wikipedia)



Deep Neural Network (Multilayer Perceptron)

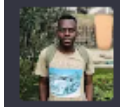
Generalization

Machine Learning (ML) and Deep Learning (DL)

- ML : learning (using a model) from (training) data in other to (among other things) make predictions
- DL : ML with a (deep) neural network as a model

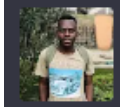
Generalization in Machine Learning

- Acquiring knowledge from some (training) data and effectively applying that acquired knowledge to different (test) data
- Ability of a ML model to perform well on new, unseen data that it hasn't been explicitly trained on (ChatGPT)
- Fundamental goal in machine learning because the ultimate aim is to build models that can provide useful insights and predictions for real-world scenarios beyond the training data



In a few words, what is generalization in machine learning?

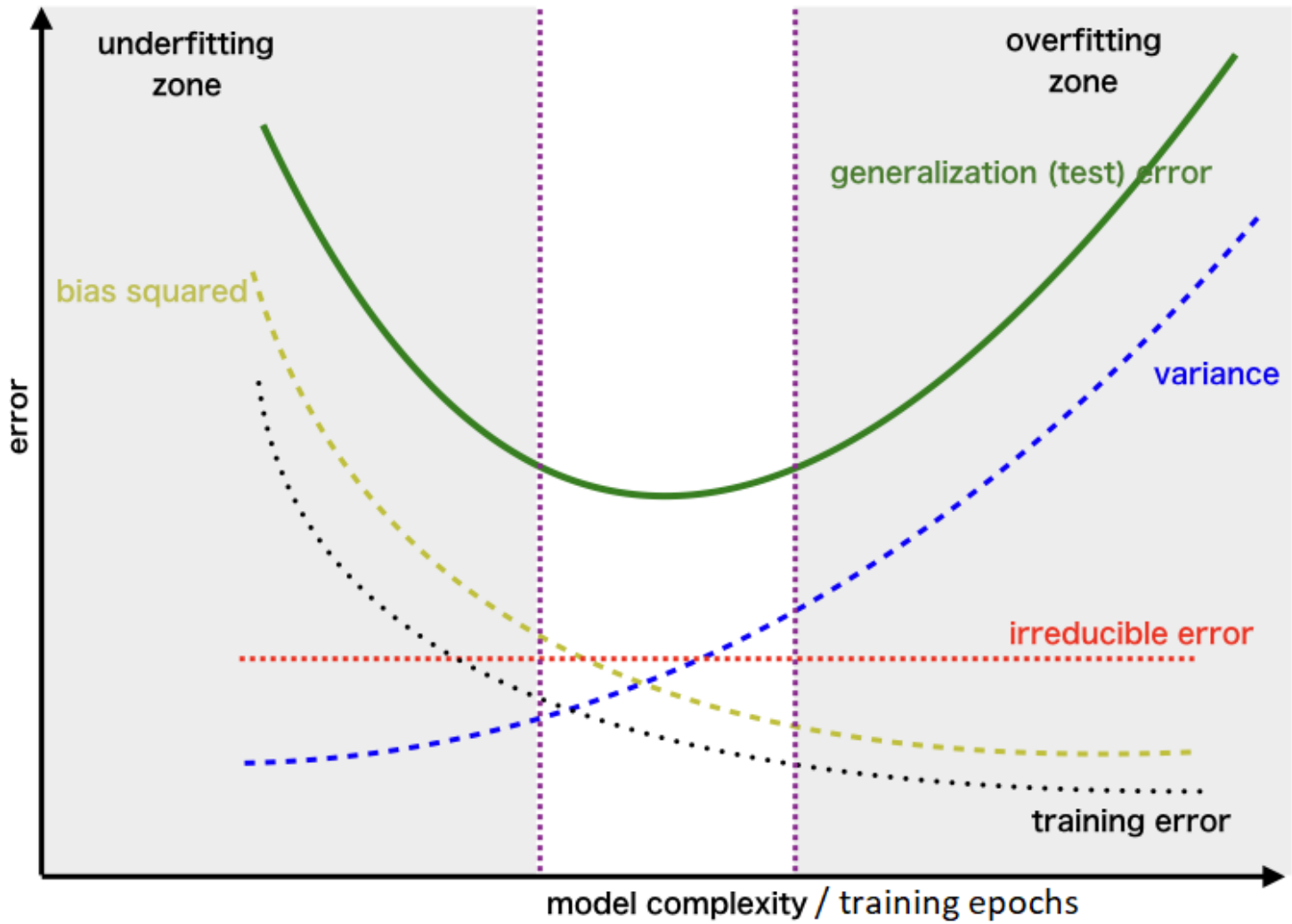
Generalization in machine learning is the model's ability to make accurate predictions or classifications on new, unseen data based on its training experience.



Are you a machine learning model that generalized or did you just memorize your training data?

I am a machine learning model based on the GPT-3.5 architecture. I generalize from the extensive text data I was trained on rather than memorizing it. While I don't have personal experiences or awareness, I generate responses based on patterns and information present in the data up until my last update in September 2021. My responses aim to be contextually relevant and helpful based on that training data, but I don't possess consciousness or memory in the human sense.

Bias-variance tradeoff: U-shaped curve



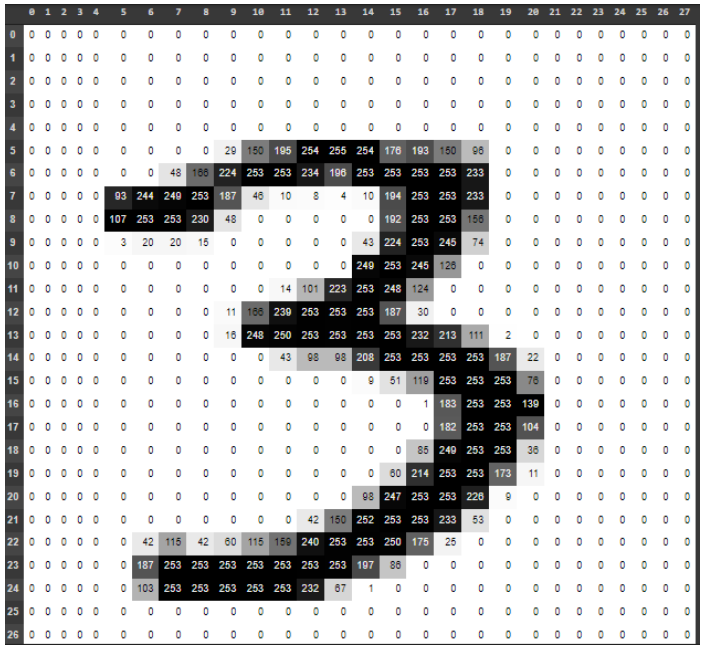
The U-shaped test error curve as a key consequence of the bias-variance tradeoff

Notations (supervised learning)

- $\mathcal{X}$: domain set (input space)

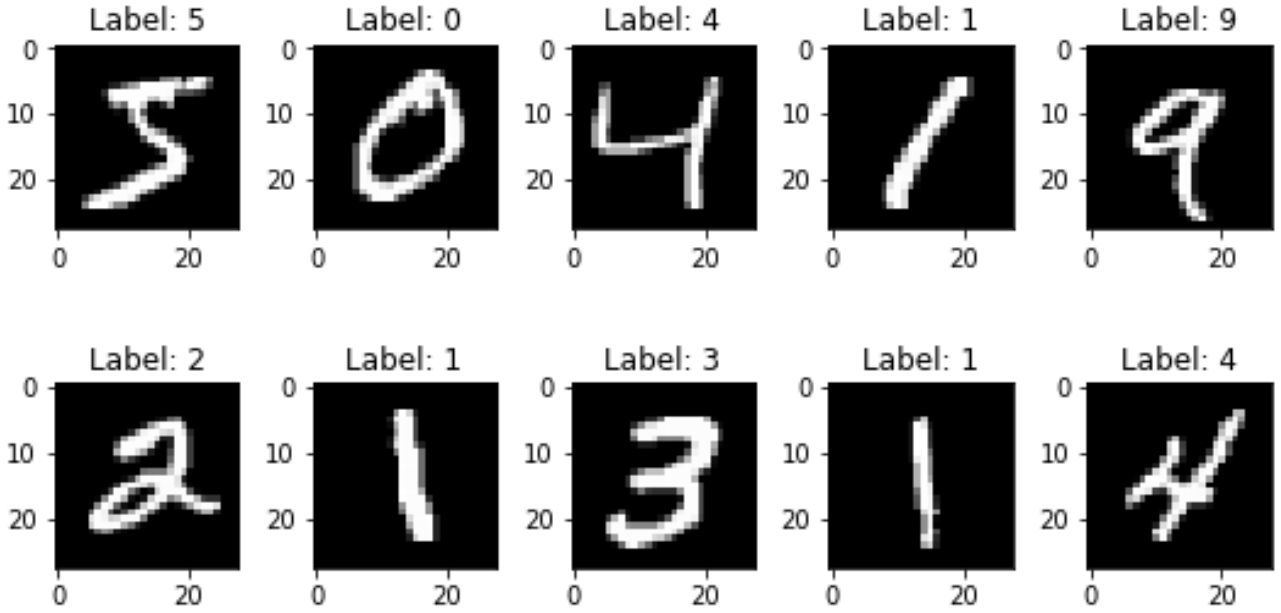
$$\mathcal{X} = \{0, 1, \ldots, 255\}^{784}$$

MNIST : 28×28 (784 pixels) handwritten digit from "0" to "9"
Each pixel value is a grayscale integer between 0 and 255



Notations (supervised learning)

- $\mathcal{X}$: domain set (input space)
- $\mathcal{Y}$: label set (output space)



MNIST : 28×28 (784 pixels) handwritten digit from "0" to "9"
Each pixel value is a grayscale integer between 0 and 255

$$\mathcal{X} = \{0, 1, \ldots, 255\}^{784} \text{ and } \mathcal{Y} = \{0, 1, \ldots, 9\}$$

Notations (supervised learning)

- $\mathcal{X}$: domain set (input space)
- $\mathcal{Y}$: label set (output space)
- $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$: hypothesis class (class of possible models we can learn)
  Choosing $\mathcal{H}$ introduces inductive bias

MNIST : 28×28 (784 pixels) handwritten digit from "0" to "9".
Each pixel value is a grayscale integer between 0 and 255

$$\mathcal{X} = \{0, 1, \ldots, 255\}^{784}$$

$$\mathcal{Y} = \{0, 1, \ldots, 9\}$$

$$\mathcal{H} = \left\{ f(x) = \text{softmax}(Wx + b) \; \forall x \in \mathcal{X} \mid W \in \mathbb{R}^{10 \times 784}, b \in \mathbb{R}^{10} \right\}$$

(set of linear function from $\mathcal{X}$ to "$\mathcal{Y}$")

$$\text{softmax}(h) = \frac{e^{h}}{\sum_k e^{h_k}}$$

Notations (supervised learning)

$$h = Wx + b \in \mathbb{R}^{10} \ \forall x \in \mathcal{X} = \{0, 1, \ldots, 255\}^{784}$$

$$\mathbb{P}\left[x \in \text{ Class } i\right] = \left[\text{softmax}(h)\right]_i = \frac{e^{h_i}}{\sum_k e^{h_k}} \ \forall i \in \mathcal{Y} = \{0, 1, \ldots, 9\}$$

$$h = \begin{pmatrix} 1.0 \\ 0.0 \\ -0.3 \\ -10.1 \\ 1.0 \\ -0.1 \\ -3.5 \\ -0.3 \\ 0.0 \\ 10.1 \end{pmatrix} \Longrightarrow \text{softmax}(h) = \begin{pmatrix} \frac{e^{1.0}}{e^{1.0}+e^{0.0}+\ldots+e^{10.1}} \\ \frac{e^{0.0}}{e^{1.0}+e^{0.0}+\ldots+e^{10.1}} \\ \cdot \\ \cdot \\ \cdot \\ \frac{e^{10.0}}{e^{1.0}+e^{0.0}+\ldots+e^{10.1}} \end{pmatrix} = \begin{pmatrix} 1.1 \times 10^{-04} \\ 4.1 \times 10^{-05} \\ \cdot \\ \cdot \\ \cdot \\ 9.9 \times 10^{-01} \end{pmatrix}$$

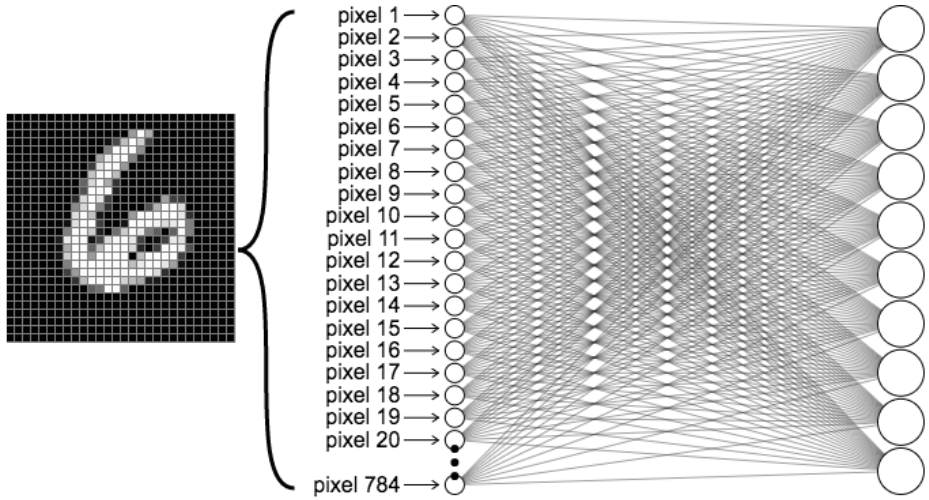
Notations (supervised learning)

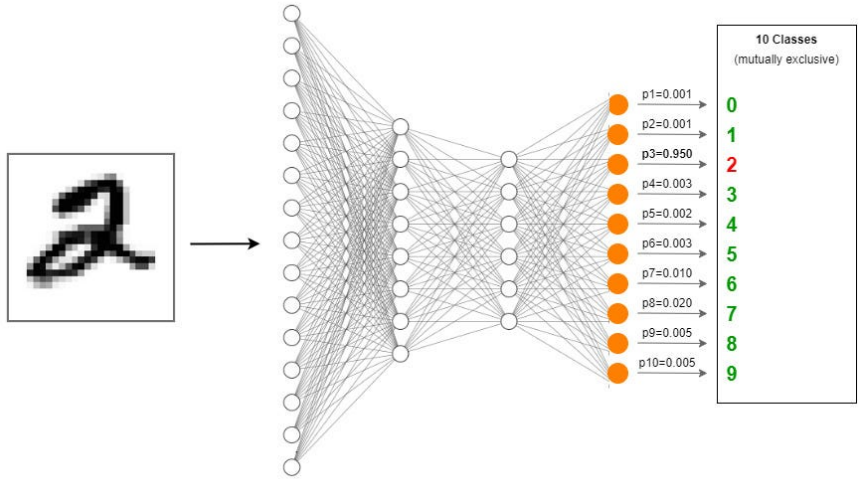
MNIST : 28×28 (784 pixels) handwritten digit from "0" to "9".
Each pixel value is a grayscale integer between 0 and 255

$$\mathcal{X} = \{0, 1, \ldots, 255\}^{784}$$

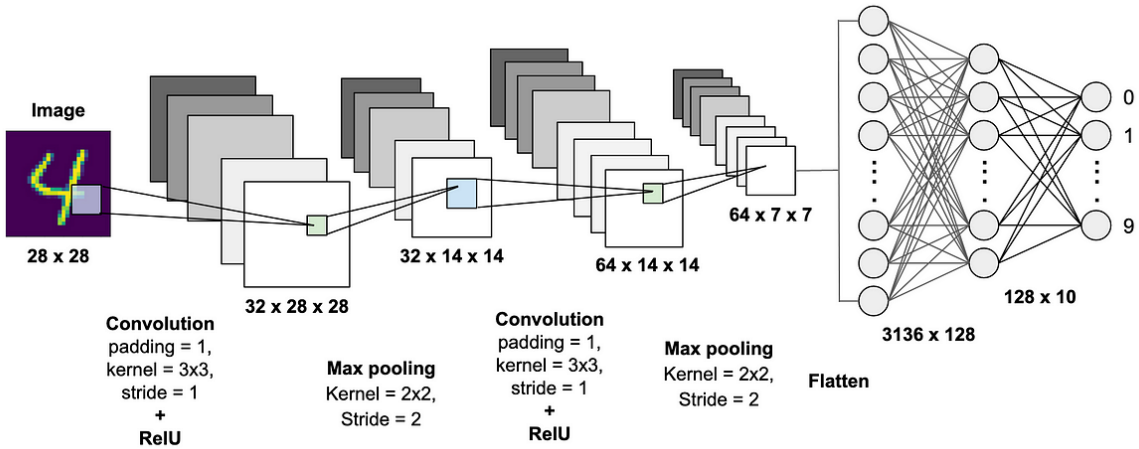
$$\mathcal{Y} = \{0, 1, \ldots, 9\}$$

$$\mathcal{H} = \left\{ f(x) = \text{softmax}(Wx + b) \; \forall x \in \mathcal{X} \mid W \in \mathbb{R}^{10 \times 784}, b \in \mathbb{R}^{10} \right\}$$





$\mathcal{H} \sim$ multilayer perceptron



$\mathcal{H} \sim$ particular convolutional neural network architecture

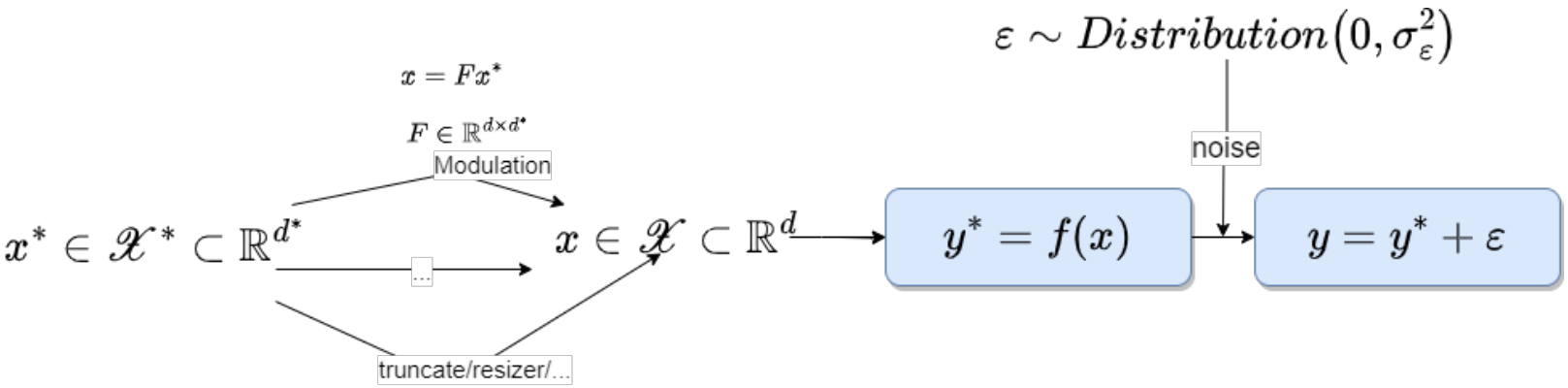
Definitions (supervised learning)

Training set

Set $\mathcal{S}_n$ of n values $z_i = (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, where $x_i \in \mathcal{X}$ represents a feature vector and $y_i = y_i(x) \in \mathcal{Y}$ the label of the i^{th} sample.

$$\mathcal{S}_n = \{z_1, \cdots, z_n\}$$

Assumption (fundamental in statistical learning) : $z_1, \ldots, z_n$, are assumed to be i.i.d. and sampled from an unknown data distribution $\mathcal{D}$.



Definitions (supervised learning)

Loss Function

The loss function $\ell(y, \hat{y})$ is defined as a function that takes two labels and produces a value between 0 and some constant $M \in [0, \infty]$, and measures the cost of predicting $\hat{y}$ when the true value is y.

$$\begin{aligned} \ell \colon \mathcal{Y} \times \mathcal{Y} &\to [0, M] \\ (y, \hat{y}) &\mapsto \ell(y, \hat{y}) \end{aligned}$$

Examples :

- Square loss : $\ell(y, \hat{y}) = \|y - \hat{y}\|^2$
- Absolute loss : $\ell(y, \hat{y}) = \sum_i |y_i - \hat{y}_i|$
- Cross-entropy : $\ell(y, \hat{y}) = \sum_i y_i \log(\hat{y}_i)$
- Zero-one loss : $\ell(y, \hat{y}) = -\mathbb{I}[y \neq \hat{y}]$

Cross-entropy : $\ell(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i)$

$$h = \begin{pmatrix} -1.0 \\ 0.0 \\ . \\ . \\ . \\ 10.1 \end{pmatrix} \Longrightarrow \hat{y} = \text{softmax}(h) = \begin{pmatrix} 1.5 \times 10^{-05} \\ 4.1 \times 10^{-05} \\ . \\ . \\ . \\ 0.9 \end{pmatrix} \textbf{ vs } y = \begin{pmatrix} 0 \\ 1 \\ . \\ . \\ . \\ 0 \end{pmatrix}$$

$$\ell(y, \hat{y}) = -\log\left(4.1 \times 10^{-05}\right) = 1.0 \times 10^{+01} \gg$$

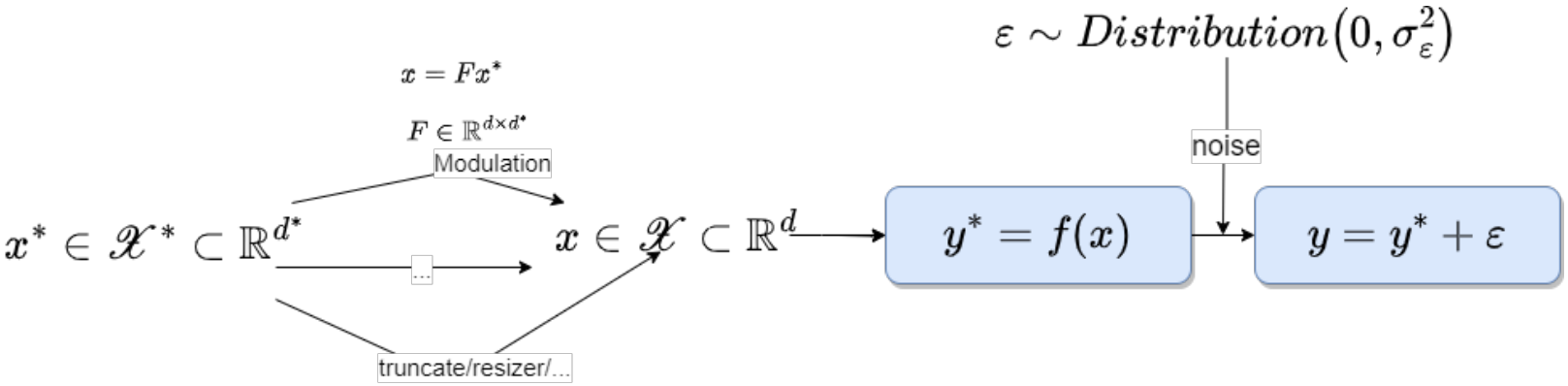
$$h = \begin{pmatrix} -1.0 \\ . \\ . \\ 10.1 \end{pmatrix} \Longrightarrow \hat{y} = \text{softmax}(h) = \begin{pmatrix} 1.5 \times 10^{-05} \\ . \\ . \\ 0.9 \end{pmatrix} \textbf{ vs } y = \begin{pmatrix} 0 \\ . \\ . \\ 1 \end{pmatrix}$$

$$\ell(y, \hat{y}) = -\log\left(0.9\right) = 3.4 \times 10^{-04} \ll$$

Definitions

Assumption : We don't have a joint distribution, but just x being random and $y = y(x)$ being a deterministic or random function of x

$$p\left(z = (x, y)\right) = p(x)p(y|x)$$



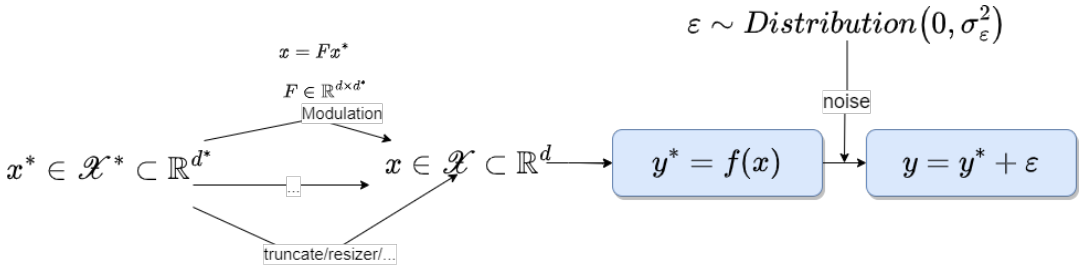
Optimal prediction

$$y^{opt}(x) = \arg\min_{\hat{y}} \mathbb{E}_{y \sim p(y|x)}[\ell(y, \hat{y})]$$

Proposition

Under the square loss, the optimal prediction is the mean of $p(y|x)$.

$$
\begin{aligned}
y^{opt}(x) &= \arg\min_{\hat{y}} \mathbb{E}_{y\sim p(y|x)}[(\hat{y}-y)^2] \\
&= \arg\min_{\hat{y}} \mathbb{E}_{y\sim p(y|x)}[\hat{y}^2 - 2y\hat{y} + y^2] \\
&= \arg\min_{\hat{y}} \ \hat{y}^2 - 2\mathbb{E}_{y\sim p(y|x)}[y]\hat{y} + \mathbb{E}_{y\sim p(y|x)}[y^2] \\
&= \mathbb{E}_{y\sim p(y|x)}[y]
\end{aligned}
$$



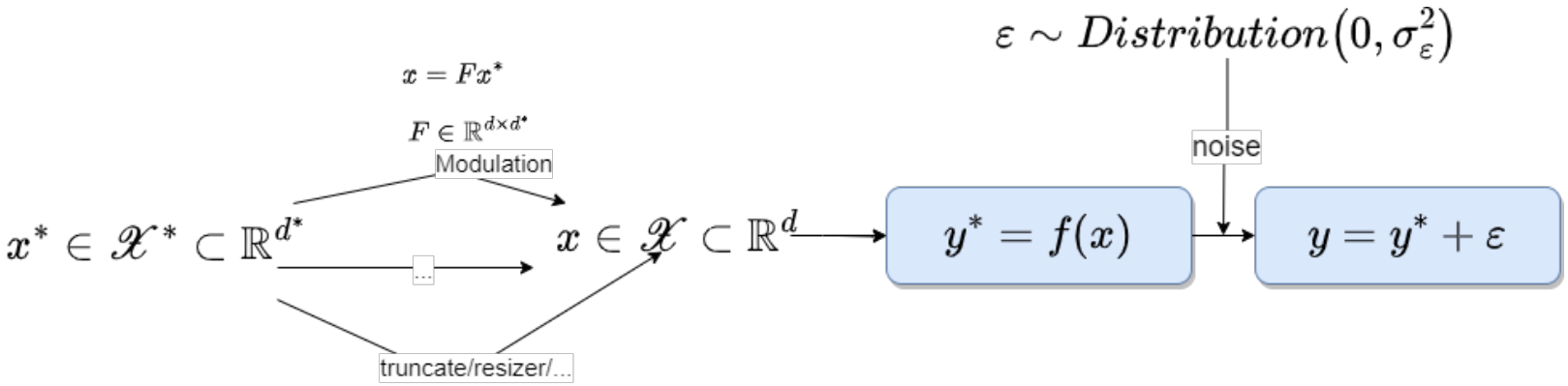
$$y^{opt}(x) = \mathbb{E}_{\epsilon}[f(x) + \epsilon] = f(x)$$

Proposition

Under the absolute loss $\ell(y, \hat{y}) = |y - \hat{y}|$, $y^{opt}(x)$ *is the median* $F^{-1}_{y|x}(1/2)$ *of* $p(y|x)$, *with* $F_{y|x}$ *the cumulative distribution function of* $p(y|x)$.

Proposition

Under the zero-one loss $\ell(y, \hat{y}) = \mathbb{I}[y \neq \hat{y}]$, $y^{opt}(x)$ *is the most frequent prediction* : $\arg\max_y p(y|x) = \arg\max_y p(y, x)$.



Definitions

True Risk

Given $f \in \mathcal{H}$ and the data distribution $\mathcal{D}$.

$$R[f] = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, f(x))] = \int_{z=(x,y)} \ell(y, f(x))p(z)dz$$

Empirical Risk

Given $f \in \mathcal{H}$ and a dataset $\mathcal{S}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$

$$\hat{R}_{\mathcal{S}_n}[f] = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i))$$

Generalization Gap

$$\epsilon^{gen}_{\mathcal{S}_n}[f] = |R[f] - \hat{R}_{\mathcal{S}_n}[f]|$$

Definitions

Learning algorithm (in practice) : Empirical Risk Minimization

$$\begin{aligned} \mathcal{A}\colon (\mathcal{X}\times\mathcal{Y})^n &\to \mathcal{H} \\ \mathcal{S}_n &\mapsto \mathcal{A}(\mathcal{S}_n) = \hat{f} = \arg\min_{f\in\mathcal{H}} \hat{R}_{\mathcal{S}_n}[f] \end{aligned}$$

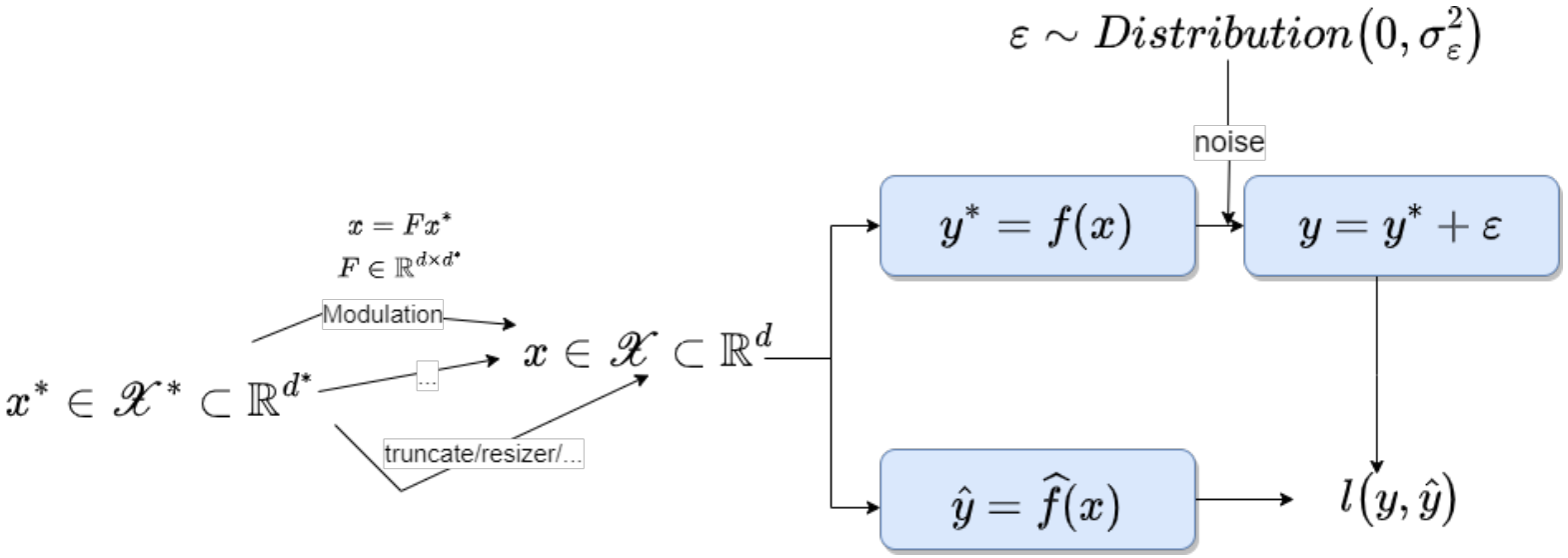
Examples: Gradient Descent (GD) and Stochastic Gradient Descent (SGD), L-shaped method (stochastic programming), stochastic dual dynamic programming ...

$$(\text{GD}) : \hat{f}^{(t+1)} = \hat{f}^{(t)} - \alpha_t \nabla \hat{R}_{\mathcal{S}_n}[\hat{f}^{(t)}]$$

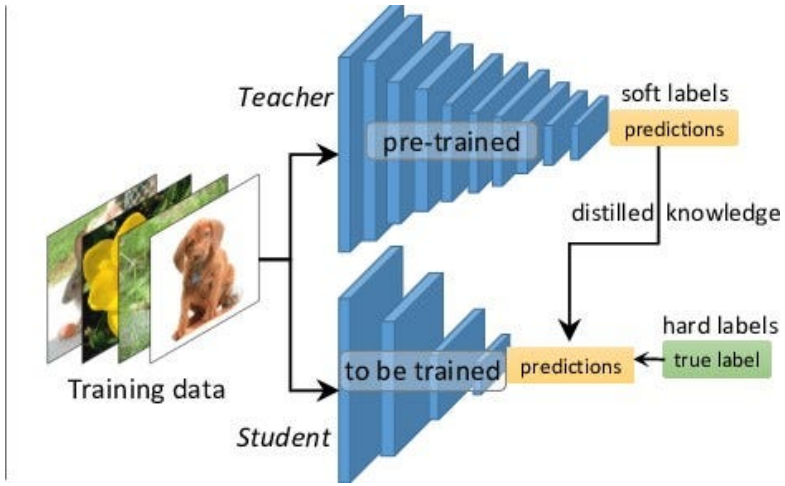
Optimal model

Model for which $\hat{f}(x) = y^{opt}(x)\ \forall x \in \mathcal{X}$

Examples: In the case of zero-one loss, the optimal model is the Bayes classifier, and its loss is called the Bayes rate.



Setup



Example : Knowledge Distillation

We care about these three levels of stochasticity

- Noise ϵ : $\mathbb{E}\epsilon = 0$ and $\text{Var}\,\epsilon = \sigma_\epsilon^2$
- Choice of $\mathcal{S}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\} \subset (\mathcal{X} \times \mathcal{Y})^n$ for a given $n \in \mathbb{N}^*$
- Learning algorithm $\mathcal{A}$, hence $\hat{f}(\mathcal{S}_n) = \mathcal{A}(\mathcal{S}_n)$
  - Gradient Descent (GD)

$$\hat{f}^{(t+1)} = \hat{f}^{(t)} - \alpha_t \nabla \hat{R}_{\mathcal{S}_n}[\hat{f}^{(t)}] = \hat{f}^{(t)} - \frac{\alpha_t}{n} \nabla \sum_{i=1}^{n} \ell(y_i, \hat{f}^{(t)}(x_i))$$

  - Stochastic Gradient Descent :

$$i_t \sim \mathcal{U}(\{1, \ldots, n\})$$
$$\hat{f}^{(t+1)} = \hat{f}^{(t)} - \alpha_t \nabla \hat{R}_{\{(x_{i_t}, y_{i_t})\}}[\hat{f}^{(t)}] = \hat{f}^{(t)} - \alpha_t \nabla \ell\left(y_{i_t}, \hat{f}^{(t)}(x_{i_t})\right)$$

  - Mini-batch gradient descent (with batch-size $m \in \{1, \ldots, n\}$)

$$\mathcal{B}_t \sim \mathcal{U}\left(\mathcal{P}_m\left(\{1, \ldots, n\}\right)\right)$$
$$\hat{f}^{(t+1)} = \hat{f}^{(t)} - \alpha_t \nabla \hat{R}_{\mathcal{B}_t}[\hat{f}^{(t)}] = \hat{f}^{(t)} - \frac{\alpha_t}{|\mathcal{B}_t|} \nabla \sum_{i \in \mathcal{B}_t} \ell(y_i, \hat{f}^{(t)}(x_i))$$

We care about these three levels of stochasticity

- Noise ϵ : $\mathbb{E}\epsilon = 0$ and $\text{Var}\,\epsilon = \sigma_\epsilon^2$
- Choice of $\mathcal{S}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\} \subset (\mathcal{X} \times \mathcal{Y})^n$ for a given $n \in \mathbb{N}^*$
- Learning algorithm $\mathcal{A}$, hence $\hat{f}(\mathcal{S}_n) = \mathcal{A}(\mathcal{S}_n)$
  - Stochastic Gradient Descent, Mini-batch gradient descent, ...
  - Regularizers: weight decay, early stopping, ...
  - Choice of hyperparameters: learning rate (and its scheduler) $\{\alpha_t\}_{t\geq 0}$, weight decay coefficient ...
  - Initialization (iterative method) : choice of $f^{(0)}$

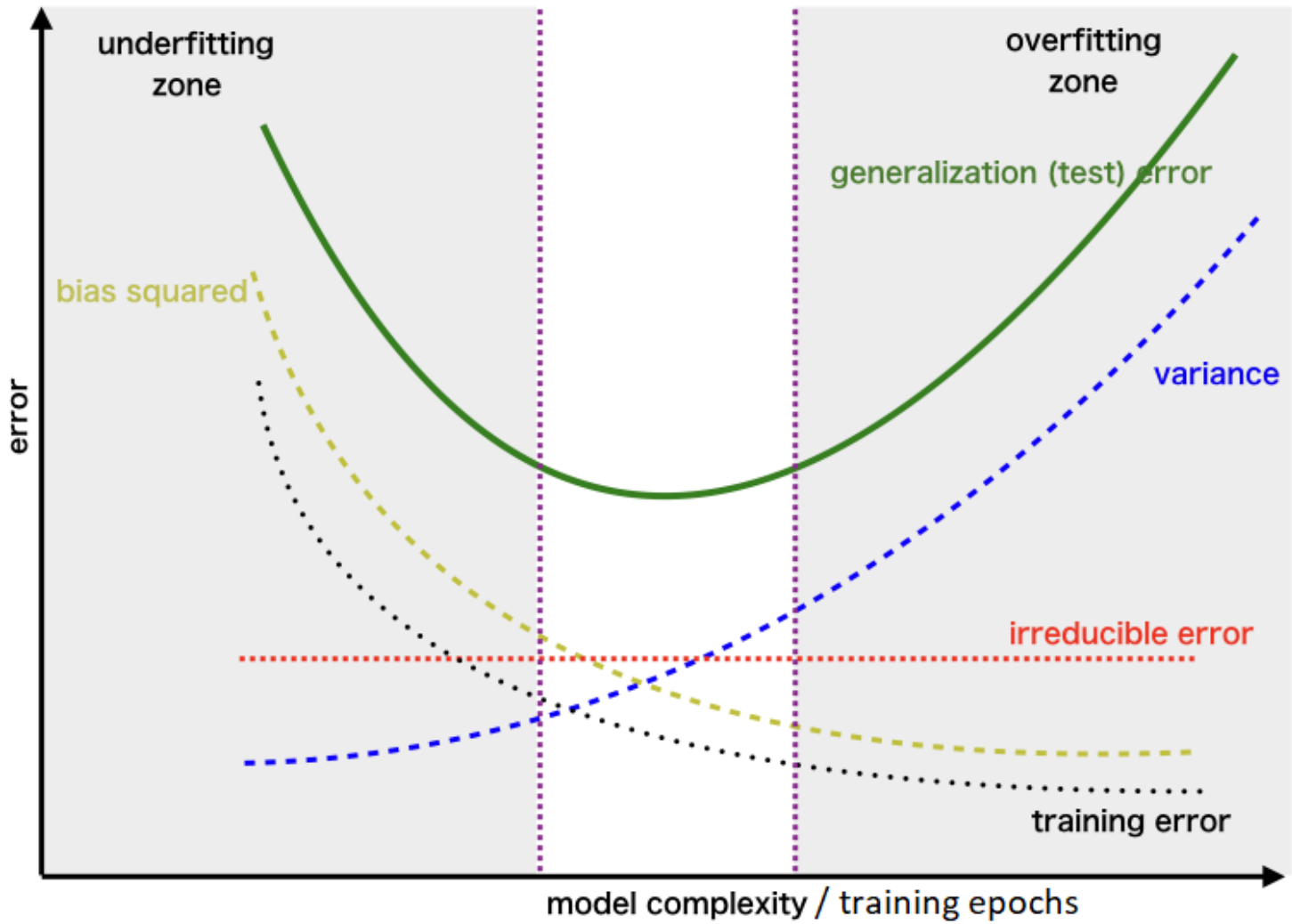
We care about these three levels of stochasticity

- Noise ϵ : $\mathbb{E}\epsilon = 0$ and Var $\epsilon = \sigma_\epsilon^2$
- Choice of $\mathcal{S}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\} \subset (\mathcal{X} \times \mathcal{Y})^n$ for a given $n \in \mathbb{N}^*$
- Learning algorithm $\mathcal{A}$, hence $\hat{f}(\mathcal{S}_n) = \mathcal{A}(\mathcal{S}_n)$

We don't care about

- The choice of $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ (MLP, CNN, RNN, ...)
  Choosing $\mathcal{H}$ introduces inductive bias
- PAC (Probably Approximately Correct) learning:
  - Occam's (Razor) bound put a prior over $\mathcal{H}$ before seeing the training dataset S_n
  - PAC Bayes bound put a prior probability distribution over $\mathcal{H}$ before seeing S_n and a posterior probability distribution over $\mathcal{H}$ after seeing S_n
- etc.

Part II - Bias-variance tradeoff: U-shaped curve



The U-shaped test error curve as a key consequence of the bias-variance tradeoff

Bias-variance decomposition

True Risk, Empirical Risk

$$R[f] = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, f(x))] \quad \text{and} \quad \hat{R}_{\mathcal{S}_n}[f] = \frac{1}{n}\sum_{(x,y)\in\mathcal{S}_n} \ell(y, f(x))$$

We care about these three levels of stochasticity

- Noise ϵ and the choice of $\mathcal{S}_n \in (\mathcal{X} \times \mathcal{Y})^n$ for a given $n \in \mathbb{N}^*$
- Learning algorithm $\mathcal{A}$, hence $\hat{f}(\mathcal{S}_n) = \mathcal{A}(\mathcal{S}_n)$

Bias-variance decomposition

Given $\hat{f} = \mathcal{A}(\mathcal{S}_n)$

$$R[\hat{f}] = \mathbb{E}[\ell(y, \hat{f}(x))] = \text{Bias}^2\left[\hat{f}\right] + \text{Var}[\hat{f}] + \text{Noise (Irreducible error)}$$

Bias-variance decomposition : example with the Square loss

$$
\begin{aligned}
R[\hat{f}] &= \mathbb{E}[(y - \hat{f}(x))^2] \\
&= \mathbb{E}[y^2 + \hat{f}^2(x) - 2y\hat{f}(x)] \\
&= \mathbb{E}[y^2] + \mathbb{E}[\hat{f}^2(x)] - 2\mathbb{E}[y\hat{f}(x)] \\
&= \mathrm{Var}[y] + \mathbb{E}^2[y] + \mathrm{Var}[\hat{f}] + \mathbb{E}^2[\hat{f}] - 2\mathbb{E}[y]\mathbb{E}[\hat{f}(x)]
\end{aligned}
$$

But

$$\mathbb{E}[y] = \mathbb{E}_\epsilon[f + \epsilon] = \mathbb{E}_\epsilon[f] + \mathbb{E}_\epsilon[\epsilon] = f$$

and

$$\mathrm{Var}[y] = \mathbb{E}\left[(y - \mathbb{E}[y])^2\right] = \mathbb{E}\left[(f + \epsilon - f)^2\right] = \mathbb{E}\left[\epsilon^2\right] = \mathrm{Var}[\epsilon] + \mathbb{E}^2[\epsilon] = \sigma_\epsilon^2$$

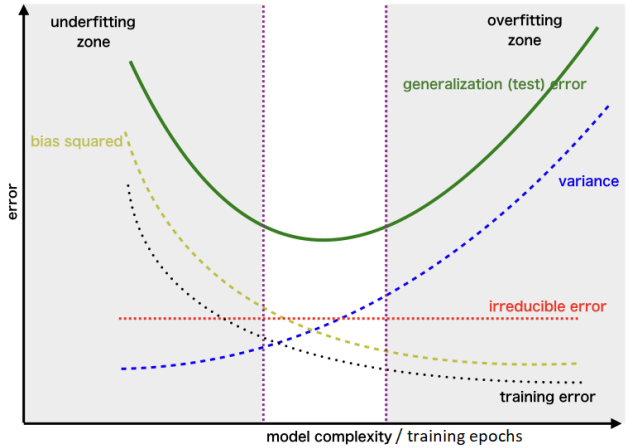
So

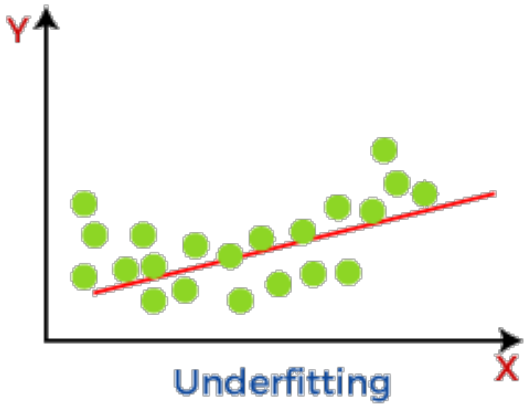
$$R[\hat{f}] = \sigma_\epsilon^2 + f^2 + \mathrm{Var}[\hat{f}] + \mathbb{E}^2[\hat{f}] - 2f\mathbb{E}[\hat{f}(x)]$$

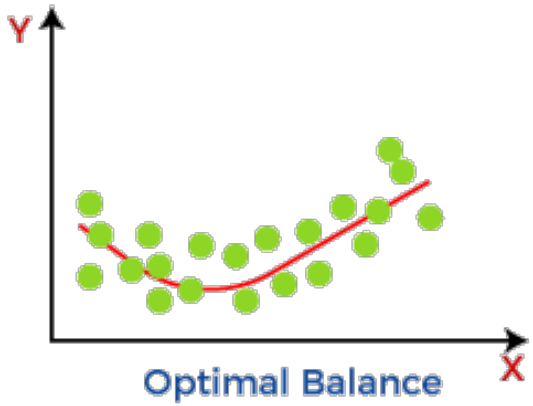
Bias-variance decomposition: example with the square loss

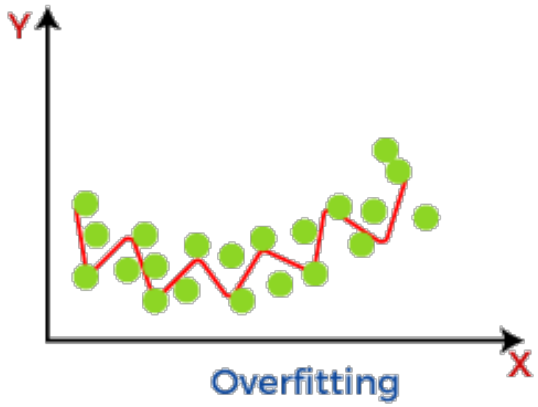
$$
\begin{aligned}
R[\hat{f}] &= \sigma_\epsilon^2 + f^2 + \mathrm{Var}[\hat{f}] + \mathbb{E}^2[\hat{f}] - 2f\mathbb{E}[\hat{f}(x)] \\
&= \sigma_\epsilon^2 + \mathrm{Var}[\hat{f}] + f^2 - 2f\mathbb{E}[\hat{f}(x)] + \mathbb{E}^2[\hat{f}] \\
&= \sigma_\epsilon^2 + \mathrm{Var}[\hat{f}] + \left(f^2 - 2f\mathbb{E}[\hat{f}(x)] + \mathbb{E}^2[\hat{f}]\right) \\
&= \underbrace{\sigma_\epsilon^2}_{\text{Noise (Irreducible error)}} + \underbrace{\mathrm{Var}[\hat{f}]}_{\mathbb{E}\left[\left(\hat{f}-\mathbb{E}[\hat{f}]\right)^2\right]} + \left(\underbrace{\mathbb{E}[\hat{f}] - f}_{\text{Bias}[\hat{f}]}\right)^2
\end{aligned}
$$

$$R[\hat{f}] = \overbrace{\sigma_\epsilon^2}^{\text{Noise (Irreducible error)}} + \overbrace{\mathbb{E}\left[\left(\hat{f} - \mathbb{E}[\hat{f}]\right)^2\right]}^{\text{Var}[\hat{f}]} + \overbrace{\left(\mathbb{E}[\hat{f}] - f\right)^2}^{\text{Bias}^2[\hat{f}]}$$









Bias-variance decomposition: The general recipe for any loss function (Domingos, 2000)

Expected loss for each input feature vector

- $\mathbb{S}_n = \{\mathcal{S}_n \in (\mathcal{X} \times \mathcal{Y})^n\}$: set of training sets of size n
- $\mathbb{Y}_n(x) = \{\mathcal{A}(\mathcal{S}_n)(x) \mid \mathcal{S}_n \in \mathbb{S}_n\}$: multiset of the predictions produced for example x by applying the learner to each training set in $\mathbb{S}_n$

$$R_n(x) = \mathbb{E}_{\hat{y} \sim \mathbb{Y}_n(x),\ y \sim p(y|x)}[\ell(y, \hat{y})]$$

Main prediction

Value whose average loss relative to all the predictions in $\mathbb{Y}_n(x)$ is minimum (i.e., it is the prediction that "differs least" from all the predictions in $\mathbb{Y}_n(x)$ according to ℓ) : "central tendency" of the learner

$$y^{\ell,n}(x) = \arg\min_y \mathbb{E}_{\hat{y} \sim \mathbb{Y}_n(x)}[\ell(y, \hat{y})]$$

Main prediction

- $\mathbb{S}_n = \{\mathcal{S}_n \in (\mathcal{X} \times \mathcal{Y})^n\}$
- $\mathbb{Y}_n(x) = \{\mathcal{A}(\mathcal{S}_n)(x) \mid \mathcal{S}_n \in \mathbb{S}_n\}$
- $y^{\ell,n}(x) = \arg\min_y \mathbb{E}_{\hat{y} \sim \mathbb{Y}_n(x)}[\ell(y, \hat{y})]$
  It is a measure of the "central tendency" of a learner.

Proposition

The main prediction $y^{\ell,n}(x)$ is

- *the mean $\mathbb{E}_{y \sim Y_n(x)}[y]$ of the predictions in $\mathbb{Y}_n(x)$ under squared loss*
- *the median $F^{-1}_{\mathbb{Y}_n(x)}(1/2)$ of $\mathbb{Y}_n(x)$ under absolute loss*
- *the mode $\arg\max_y f_{\mathbb{Y}_n(x)}(y)$ of $\mathbb{Y}_n(x)$ (i.e. the most frequent prediction) under zero-one loss*

Bias, Variance, Noise

- $y^{opt}(x) = \arg\min_{\hat{y}} \mathbb{E}_{y \sim p(y|x)}[\ell(y, \hat{y})]$
- $\mathbb{Y}_n(x) = \{\mathcal{A}(\mathcal{S}_n)(x) \mid \mathcal{S}_n \in (\mathcal{X} \times \mathcal{Y})^n\}$
- $y^{\ell,n}(x) = \arg\min_{y} \mathbb{E}_{\hat{y} \sim \mathbb{Y}_n(x)}[\ell(y, \hat{y})]$
- Bias (square): loss incurred by the main prediction relative to the optimal prediction

$$\text{Bias}^2(x) = \ell(y^{opt}(x), y^{\ell,n}(x))$$

- Variance: average loss incurred by predictions relative to the main prediction

$$\text{Var}(x) = \mathbb{E}_{y \sim \mathbb{Y}_n(x)}[\ell(y^{\ell,n}(x), y)]$$

- Noise : unavoidable component of the loss incurred independently of the learning algorithm

$$\textit{Noise}(x) = \mathbb{E}_{y \sim p(y|x)}[\ell(y, y^{opt}(x))]$$

Task

For a given loss functions ℓ, we are looking for two constants $c_1(x, \ell)$ and $c_2(x, \ell)$ such that

$$\begin{aligned} R_n(x) &= \mathbb{E}_{\hat{y} \sim \mathbb{Y}_n(x),\ y \sim p(y|x)}[\ell(y, \hat{y})] \\ &= \text{Bias}^2(x) + c_1(x, \ell) \cdot \text{Var}(x) + c_2(x, \ell) \cdot \textit{Noise}(x) \end{aligned}$$

Proposition (Domingos (2000))

- *For square loss* $\ell(y, \hat{y}) = (y - \hat{y})^2$, $c_1(x, \ell) = c_2(x, \ell) = 1$
- *For zero-one loss* $\ell(y, \hat{y}) = \mathbb{I}[y \neq \hat{y}]$ *in two class problems,*
  - $c_1(x, \ell) = 2\mathbb{P}_{\mathbb{S}_n}(x) - 1$
  - $c_2(x, \ell) = 2\mathbb{I}[y^{\ell,n}(x) = y^{opt}(x)] - 1$

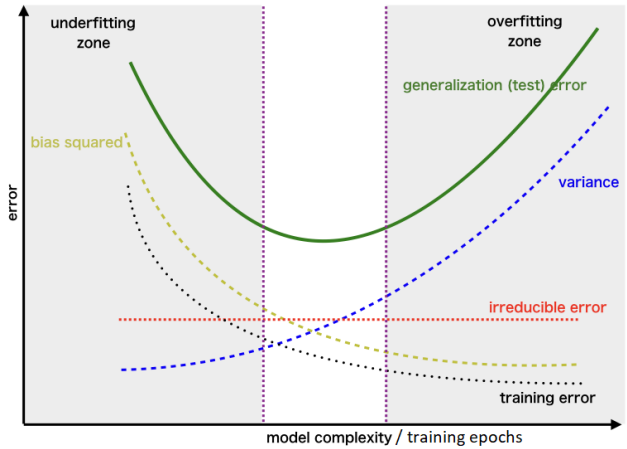
  with

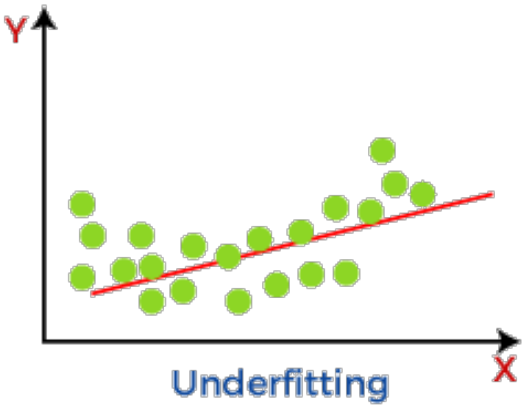
$$\mathbb{P}_{\mathbb{S}_n}(x) = \mathbb{P}[y^{opt}(x) \in \mathbb{Y}_n(x)]$$

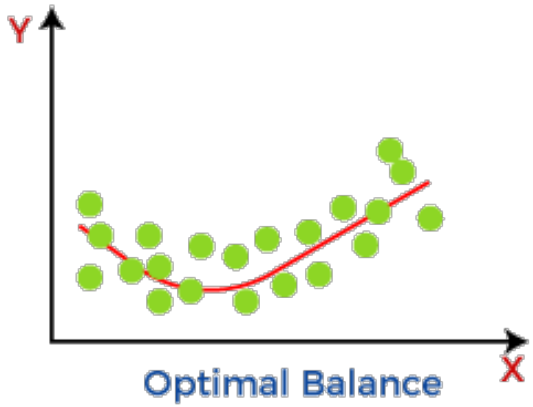
  the probability over training sets in $\mathbb{S}_n$ *that the learner predicts the optimal class for* x.

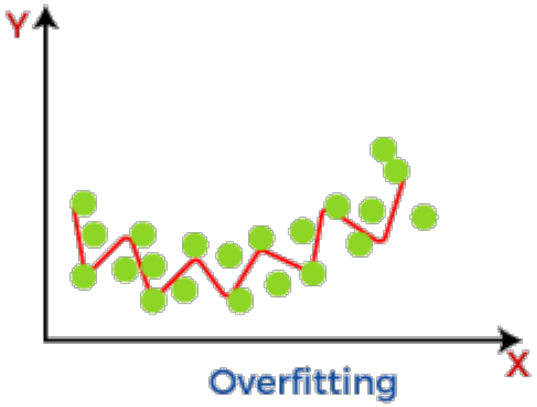
Part III - What statistical learning doesn't tell us

$$R[\hat{f}] = \text{Noise (Irreducible error)} + \text{Var}[\hat{f}] + \text{Bias}^2\left[\hat{f}\right]$$

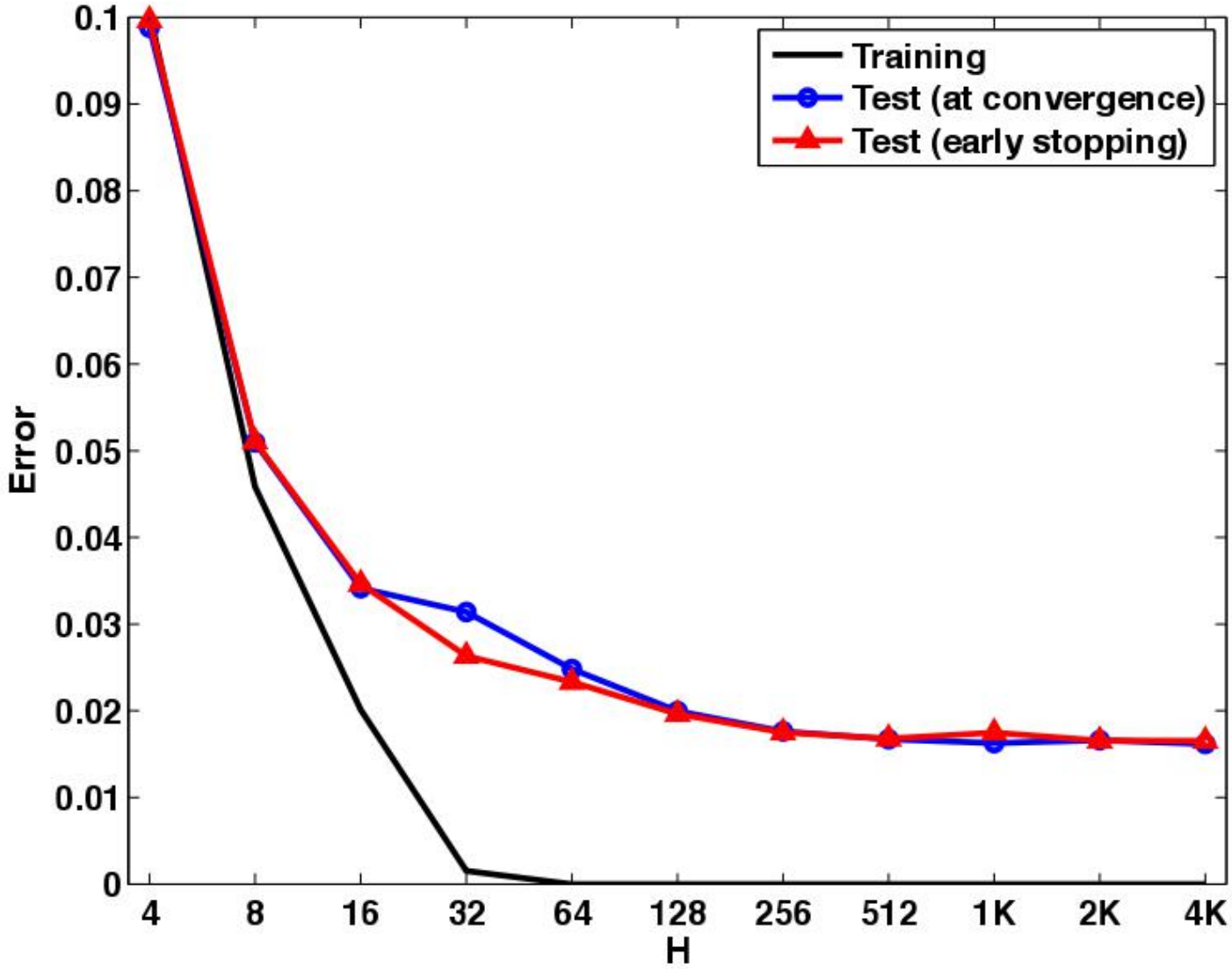








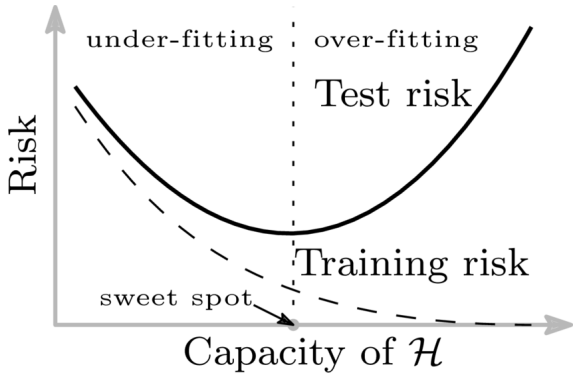
What statistical learning doesn't tell us



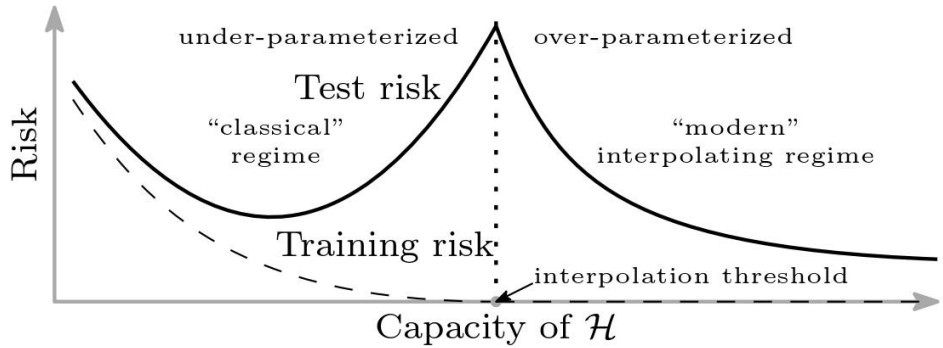
Neyshabur et al. (2014) found that test error simply decreases with network width

What statistical learning doesn't tell us

There is growing evidence that test error acts as the classic U-shaped curve in the under-parameterized regime and monotonically decreases in the over-parameterized regime (Belkin et al., 2018).



Classical U-shaped test error curve



Double Descent

Complexity/Capacity of $\mathcal{H}$

- We use the number of parameters of our model
- Others: VC dimension, Rademacher complexity, etc ...

Some results from statistical learning: PAC learning

https://study.com/academy

Some results from statistical learning: PAC learning

Definition (PAC Learning (Mitliagkas, 2023))

A hypothesis class $\mathcal{H}$ is (Agnostic) PAC learnable if given some arbitrary $\tau > 0$ and $\delta \in [0, 1]$ there exists an $n_{\mathcal{H}}(\tau, \delta)$ such that for any $\mathcal{S}_n$ with $n = n_{\mathcal{H}}(\tau, \delta)$ we have

$$\mathbb{P}_{\mathcal{S}_n}\left[\epsilon^{gen}_{\mathcal{S}_n}[f] \leq \tau\right] \geq 1 - \delta \quad \forall f \in \mathcal{H}$$

Literally speaking, $\epsilon^{gen}_{\mathcal{S}_n}[f] \leq \tau$ holds with probability at least $1 - \delta$ for any $f \in \mathcal{H}$.

- The "probably" (P) part of PAC corresponds to $1 - \delta$ while the "approximately correct" (AC) part corresponds to τ
- The small $n_{\mathcal{H}}(\tau, \delta)$ we can find is known as the Sample Complexity of the hypothesis class $\mathcal{H}$

Some results from statistical learning: PAC learning

Theorem ((Mitliagkas, 2023))

Any finite hypothesis class $\mathcal{H}$ is agnostic PAC learnable

Bound on the generalization gap $\epsilon^{gen}_{\mathcal{S}_n} = |R - \hat{R}_{\mathcal{S}_n}|$ for finite $\mathcal{H}$

In fact, let $\delta,\ \tau \in \mathbb{R}_+$.

- For $\hat{f} = \mathcal{A}(\mathcal{S}_n) \in \mathcal{H}$

$$n \geq \frac{M^2}{2}\frac{1}{\tau^2}\log\left(2\frac{|\mathcal{H}|}{\delta}\right) \Longrightarrow \mathbb{P}_{\mathcal{S}_n}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \leq \tau\right] \geq 1 - \delta$$

- For any $f \in \mathcal{H}$,

$$n \geq \frac{M^2}{2}\frac{1}{\tau^2}\log\left(\frac{2}{\delta}\right) \Longrightarrow \mathbb{P}_{\mathcal{S}_n}\left[\epsilon^{gen}_{\mathcal{S}_n}[f] \leq \tau\right] \geq 1 - \delta$$

Proof. : See below

Generalization Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

Lemma (Markov's Inequality)

Let Z be a non-negative random variable. Then for any $a > 0$,

$$\mathbb{P}[Z \geq a] \leq \frac{\mathbb{E}[Z]}{a}$$

Proof.

$$\begin{aligned}\mathbb{E}[Z] &= \int_0^{+\infty} zp(z)dz = \int_0^{a} zp(z)dz + \int_a^{+\infty} zp(z)dz \\ &\geq \int_a^{+\infty} zp(z)dz \geq \int_a^{+\infty} ap(z)dz = a\int_a^{+\infty} p(z)dz\end{aligned}$$

Generalization Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

Lemma (Chebyshev's Inequality)

Let X be an integrable random variable with finite expectation and finite nonzero variance. Then for any $a > 0$,

$$\mathbb{P}\left[|X - \mathbb{E}[X]| \geq a\right] \leq \frac{\text{Var}[X]}{a^2}$$

Proof.

Let $Z = |X - \mathbb{E}[X]|$. Using the Markov's Inequality, we have

$$\mathbb{P}\left[Z \geq a\right] = \mathbb{P}\left[Z^2 \geq a^2\right] \leq \frac{\mathbb{E}[Z^2]}{a^2} = \frac{\mathbb{E}[(X - \mathbb{E}[X])^2]}{a^2} = \frac{\text{Var}[X]}{a^2}$$

Generalization Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

Lemma (Generic Chernoff's Bound)

Let X be a random variable. Then for any $t \geq 0$,

$$\mathbb{P}[X \geq a] = \mathbb{P}[tX \geq ta] = \mathbb{P}\left[e^{tX} \geq e^{ta}\right] \leq \frac{\mathbb{E}\left[e^{tX}\right]}{e^{ta}}$$

We can minimize the bound with respect to t to get the tightest upper-bound, i.e.

$$\mathbb{P}[X \geq a] \leq \inf_{t \geq 0} e^{-ta}\mathbb{E}\left[e^{tX}\right]$$

Such probabilistic bounds that show some random variable is close to its mean with high probability are called concentration bounds.

Generalization Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

Lemma (Hoeffding's Lemma)

Let X be a random variable taking values in the interval $[a, b]$ such that $\mathbb{E}[X] = 0$. Then for $\lambda > 0$,

$$\mathbb{E}\left[e^{\lambda X}\right] \leq e^{\frac{\lambda^2 (b-a)^2}{8}}$$

Since the exponential function e is convex

$$\frac{b-X}{b-a} e^{\lambda a} + \frac{X-a}{b-a} e^{\lambda b} \geq e^{\frac{b-X}{b-a}\lambda a + \frac{X-a}{b-a}\lambda b} = e^{\lambda X}$$

$$\implies \frac{b-\mathbb{E}[X]}{b-a} e^{\lambda a} + \frac{\mathbb{E}[X]-a}{b-a} e^{\lambda b} = \frac{b}{b-a} e^{\lambda a} - \frac{a}{b-a} e^{\lambda b} \geq \mathbb{E}\left[e^{\lambda X}\right]$$

$$\begin{aligned}\frac{b}{b-a}e^{\lambda a} - \frac{a}{b-a}e^{\lambda b} &= e^{\lambda a}\left(\frac{b}{b-a} - \frac{a}{b-a}e^{\lambda(b-a)}\right) \\ &= e^{\frac{a}{b-a}\lambda(b-a)}e^{\ln\left(\frac{b}{b-a} - \frac{a}{b-a}e^{\lambda(b-a)}\right)} \\ &= e^{\frac{a}{b-a}\lambda(b-a)+\ln\left(\frac{b}{b-a} - \frac{a}{b-a}e^{\lambda(b-a)}\right)} \\ &= e^{L(\lambda(b-a))}\end{aligned}$$

with

$$L(h) = \frac{a}{b-a}h + \ln\left(\frac{b}{b-a} - \frac{a}{b-a}e^{h}\right) \leq \frac{1}{8}h^2 \quad \forall h \in \mathrm{Dom}(L)$$

So

$$\mathbb{E}\left[e^{\lambda X}\right] \leq \frac{b}{b-a}e^{\lambda a} - \frac{a}{b-a}e^{\lambda b} = e^{L(\lambda(b-a))} \leq e^{\frac{\lambda^2(b-a)^2}{8}}$$

$$L(h) = \frac{a}{b-a}h + \ln\left(\frac{b}{b-a} - \frac{a}{b-a}e^h\right)$$

$$L'(h) = \frac{a}{b-a} - \frac{ae^h}{b-ae^h} \quad \text{and} \quad L''(h) = -\frac{abe^h}{(b-ae^h)^2}$$

Using the Lagrange form of the Taylor Theorem, there exists $\xi \in]0, h[$ such that

$$L(h) = L(0) + hL'(0) + \frac{1}{2}h^2L''(\xi) = \frac{1}{2}h^2L''(\xi)$$

On the other hand, since $a \leq 0 \leq b$ (otherwise $\mathbb{E}[X] = \int_a^b xp(x)dx \neq 0$)

$$0 \leq e^{\frac{1}{2}\xi}\sqrt{-ab} = e^{\frac{1}{2}\xi}\sqrt{-ae^{\frac{1}{2}\xi}be^{-\frac{1}{2}\xi}} \leq e^{\frac{1}{2}h}\frac{-ae^{\frac{1}{2}\xi} + be^{-\frac{1}{2}\xi}}{2} = \frac{b-ae^{\xi}}{2}$$

$$\implies -abe^{\xi} \leq \frac{\left(b-ae^{\xi}\right)^2}{4} \implies L''(\xi) = -\frac{abe^{\xi}}{(b-ae^{\xi})^2} \leq \frac{1}{4} \quad \forall \xi \in \text{Dom}(L'')$$

Generalization Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

Lemma (Hoeffding's Inequality)

Let $Z_1, \ldots, Z_n$ be independent random variables such that $\mathbb{P}[a \leq Z_i \leq b] = 1$ for all $i \in [n]$. Let $\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i$. Then, for any $\tau > 0$:

$$\mathbb{P}\left[|\bar{Z} - \mathbb{E}[\bar{Z}]| \geq \tau\right] \leq 2e^{\frac{-2n\tau^2}{(b-a)^2}}$$

This expression says that for any positive τ, the sample mean $\bar{Z}$ will be at least τ away from its expected value $\mathbb{E}[\bar{Z}]$ with a probability that decays exponentially with the number of training examples n we have.

Let $X_i = Z_i - \mathbb{E}[\bar{Z}]$ for all $i \in [n]$, and $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. Then for all $\lambda \geq 0$,

$$\mathbb{P}[\bar{X} \geq \tau] \leq \frac{\mathbb{E}\left[e^{\lambda \bar{X}}\right]}{e^{\lambda \tau}} \text{ (Generic Chernoff's Bound)}$$

$$= e^{-\lambda\tau}\mathbb{E}\left[e^{\frac{\lambda}{n}\sum_{i=1}^{n} X_i}\right] = e^{-\lambda\tau}\mathbb{E}\left[\prod_{i=1}^{n} e^{\frac{\lambda}{n}X_i}\right]$$

$$\leq e^{-\lambda\tau}\mathbb{E}\left[\prod_{i=1}^{n} e^{\frac{\lambda}{n}X_i}\right] = e^{-\lambda\tau}\prod_{i=1}^{n}\mathbb{E}\left[e^{\frac{\lambda}{n}X_i}\right] \text{ (independence)}$$

$$\leq e^{-\lambda\tau}\prod_{i=1}^{n} e^{\frac{\lambda^2\left(\frac{b-\mathbb{E}[\bar{Z}]}{n} - \frac{a-\mathbb{E}[\bar{Z}]}{n}\right)^2}{8}} \text{ (Hoeffding's Lemma)}$$

$$= e^{-\lambda\tau}\prod_{i=1}^{n} e^{\frac{\lambda^2(b-a)^2}{8n^2}} = e^{-\lambda\tau}e^{\sum_{i=1}^{n}\frac{\lambda^2(b-a)^2}{8n^2}}$$

$$= e^{-\lambda\tau + \frac{\lambda^2(b-a)^2}{8n}}$$

Let $X_i = Z_i - \mathbb{E}[\bar{Z}]$ for all $i \in [n]$, and $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. Then,

$$\mathbb{P}[\bar{X} \geq \tau] \leq e^{-\lambda\tau + \frac{\lambda^2(b-a)^2}{8n}} \quad \forall \lambda \geq 0$$

$$\implies \mathbb{P}[\bar{X} \geq \tau] \leq \min_{\lambda \geq 0} e^{-\lambda\tau + \frac{\lambda^2(b-a)^2}{8n}} = e^{-\frac{2n\tau^2}{(b-a)^2}}$$

By the same argument, we can show that $\mathbb{P}[-\bar{X} \geq \tau] \leq e^{-\frac{2n\tau^2}{(b-a)^2}}$ since

$$\mathbb{P}[-\bar{X} \geq \tau] = \mathbb{P}[e^{-\lambda\bar{X}} \geq e^{\lambda\tau}] \leq \frac{\mathbb{E}\left[e^{-\lambda\bar{X}}\right]}{e^{\lambda\tau}}$$

$$= e^{-\lambda\tau} \prod_{i=1}^{n} \mathbb{E}\left[e^{-\frac{\lambda}{n}X_i}\right]$$

$$= e^{-\lambda\tau} \prod_{i=1}^{n} e^{\frac{\lambda^2(b-a)^2}{8n^2}} = e^{-\lambda\tau + \frac{\lambda^2(b-a)^2}{8n}}$$

Let $X_i = Z_i - \mathbb{E}[\bar{Z}]$ for all $i \in [n]$, and $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. We have

$$\begin{aligned}\mathbb{P}[|\bar{X}| \geq \tau] &= \mathbb{P}[\bar{X} \geq \tau] + \mathbb{P}[\bar{X} \leq -\tau] \\ &= \mathbb{P}[\bar{X} \geq \tau] + \mathbb{P}[-\bar{X} \geq \tau] \\ &\leq 2e^{-\frac{2n\tau^2}{(b-a)^2}}\end{aligned}$$

with

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i = \frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}[\bar{Z}] = \bar{Z} - \mathbb{E}[\bar{Z}]$$

Generalization Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

Lemma (Hoeffding's Inequality)

Let $Z_1, \ldots, Z_n$ be independent random variables such that $\mathbb{P}[a \leq Z_i \leq b] = 1$ for all $i \in [n]$. Let $\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i$. Then, for any $\tau > 0$:

$$\mathbb{P}\left[|\bar{Z} - \mathbb{E}[\bar{Z}]| \geq \tau\right] \leq 2e^{\frac{-2n\tau^2}{(b-a)^2}}$$

For a given dataset $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^{n}$ and a hypothesis $f \in \mathcal{H}$ independent of $\mathcal{S}_n$, let $Z_i = \ell(y_i, f(x_i))$.

Then we get

$$\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i)) = \hat{R}_{\mathcal{S}_n}[f]$$

and

$$\mathbb{E}[\bar{Z}] = \mathbb{E}\left[\hat{R}_{\mathcal{S}_n}[f]\right] = R[f]$$

For a given dataset $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$ and a hypothesis $f \in \mathcal{H}$ independent of $\mathcal{S}_n$, let $Z_i = \ell(y_i, f(x_i))$.

Then we get

$$\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i)) = \hat{R}_{\mathcal{S}_n}[f] \quad \text{and} \quad \mathbb{E}[\bar{Z}] = \mathbb{E}\left[\hat{R}_{\mathcal{S}_n}[f]\right] = R[f]$$

So

$$|\bar{Z} - \mathbb{E}[\bar{Z}]| = \epsilon^{gen}_{\mathcal{S}_n}[f]$$

and (since $0 \leq Z_i \leq M$)

$$\mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[f] \geq \tau\right] \leq 2e^{\frac{-2n\tau^2}{M^2}} \Longrightarrow \mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[f] \leq \tau\right] \geq 1 - 2e^{\frac{-2n\tau^2}{M^2}}$$

If we set $2e^{\frac{-2n\tau^2}{M^2}} \leq \delta \leq 1$, we can solve to get $n \geq -\frac{M^2}{2\tau^2} \ln \frac{\delta}{2} = \frac{M^2}{2\tau^2} \ln \frac{2}{\delta}$.
This n guaranteed

$$\mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[f] \leq \tau\right] \geq 1 - \delta$$

Let $\hat{f} = \mathcal{A}(\mathcal{S}_n) \in \mathcal{H}$ and $Z_i = \ell(y_i, \hat{f}(x_i))$. Then we get

$$\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \hat{f}(x_i)) = \hat{R}_{\mathcal{S}_n}[\hat{f}]$$

But

$$\mathbb{E}[\bar{Z}] = \mathbb{E}\left[\hat{R}_{\mathcal{S}_n}[\hat{f}]\right] \neq R[\hat{f}]$$

So

$$|\bar{Z} - \mathbb{E}[\bar{Z}]| \neq \epsilon^{gen}_{\mathcal{S}_n}[\hat{f}]$$

We can't use the Hoeffding's Inequality directly. But we have :

$$\begin{aligned}
\mathbb{P}\left[\left|\hat{R}_{\mathcal{S}_n}[\hat{f}] - R[\hat{f}]\right| \geq \tau\right] &\leq \mathbb{P}\left[\max_{f\in\mathcal{H}}\left|\hat{R}_{\mathcal{S}_n}[f] - R[f]\right| \geq \tau\right] \\
&= \mathbb{P}\left[\cup_{f\in\mathcal{H}}\left\{\left|\hat{R}_{\mathcal{S}_n}[f] - R[f]\right| \geq \tau\right\}\right] \\
&\leq \sum_{f\in\mathcal{H}} \mathbb{P}\left[\left|\hat{R}_{\mathcal{S}_n}[f] - R[f]\right| \geq \tau\right] \text{ (union bound)} \\
&\leq 2|\mathcal{H}|e^{\frac{-2n\tau^2}{M^2}}
\end{aligned}$$

For a given dataset $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$, let $\hat{f} = \mathcal{A}(\mathcal{S}_n) \in \mathcal{H}$ and $Z_i = \ell(y_i, \hat{f}(x_i))$. Then we get

$$\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \hat{f}(x_i)) = \hat{R}_{\mathcal{S}_n}[\hat{f}]$$

But

$$\mathbb{E}[\bar{Z}] = \mathbb{E}\left[\hat{R}_{\mathcal{S}_n}[\hat{f}]\right] \neq R[\hat{f}]$$

So

$$|\bar{Z} - \mathbb{E}[\bar{Z}]| \neq \epsilon^{gen}_{\mathcal{S}_n}[\hat{f}]$$

We can't use the Hoeffding's Inequality directly. But we have :

$$\mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \geq \tau\right] \leq 2|\mathcal{H}|e^{\frac{-2n\tau^2}{M^2}} \Longrightarrow \mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \leq \tau\right] \geq 1 - 2|\mathcal{H}|e^{\frac{-2n\tau^2}{M^2}}$$

Bound for Finite Hypothesis Classes (Mitliagkas, 2023)

We can't use the Hoeffding's Inequality directly. But we have :

$$\mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \geq \tau\right] \leq 2|\mathcal{H}|e^{\frac{-2n\tau^2}{M^2}} \Longrightarrow \mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \leq \tau\right] \geq 1 - 2|\mathcal{H}|e^{\frac{-2n\tau^2}{M^2}}$$

If we set $2|\mathcal{H}|e^{\frac{-2n\tau^2}{M^2}} \leq \delta \leq 1$, we can solve to get $n \geq \frac{M^2}{2\tau^2} \ln \frac{2|\mathcal{H}|}{\delta}$.
This n guaranteed

$$\mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \leq \tau\right] \geq 1 - \delta$$

Consider for example

$$\mathcal{H}_\theta = \left\{ f(x) = Wx + b \; \forall x \in \mathbb{R}^d \mid \theta = \left(W \in \mathbb{R}^{c \times d}, b \in \mathbb{R}^c\right)\right\}$$

In theory, $|\mathcal{H}_\theta| = \infty \; (\aleph_1)$

But if the numbers are stored on $q \geq 2$ bits (for example 32 or 64), we get $|\mathcal{H}_\theta| = 2^{qc(d+1)}$

Non finite $\mathcal{H}$: VC dimension (Mitliagkas, 2023)

Definition (Shattering)

A set of points Ω is shattered by a hypothesis class $\mathcal{H}$ if there are hypotheses in $\mathcal{H}$ that split Ω in all of the $2^{|\Omega|}$ possible ways; i.e., all possible ways of classifying points in Ω are achievable using concepts in $\mathcal{H}$.

Definition (Vapnik-Chervonenkis dimension)

The VC dimension of a hypothesis space $\mathcal{H}$ is the cardinality of the largest set Ω that can be shattered by $\mathcal{H}$. If arbitrarily large finite sets can be shattered by $\mathcal{H}$, then the VC dimension of $\mathcal{H}$ is infinite (∞).

Example

For $\mathcal{X} = \mathbb{R}$ and $\mathcal{Y} = \{0, 1\}$. The VC dimension of $\mathcal{H} = \{\mathbb{I}[x \leq a] \mid a \in \mathbb{R}\}$ is 1.

Non finite $\mathcal{H}$: VC dimension

Definition (Vapnik-Chervonenkis dimension)

The VC dimension of a hypothesis space $\mathcal{H}$ is the cardinality of the largest set Ω that can be shattered by $\mathcal{H}$. If arbitrarily large finite sets can be shattered by $\mathcal{H}$, then the VC dimension of $\mathcal{H}$ is infinite (∞).

Theorem (Bound base on the VC dimension)

Let $\delta \in [0, 1]$. *For* $f \in \mathcal{H}$

$$\mathbb{P}_{\mathcal{S}_n}\left[R[f] - \hat{R}_{\mathcal{S}_n}[f] \leq \sqrt{\frac{n\left[VC(\mathcal{H})\left(\log \frac{2n}{VC(\mathcal{H})} + 1\right) + \log \frac{4}{\delta}\right]}{2n}}\right] = 1 - \delta$$

Some results from statistical learning: PAC learning

Theorem (Occam's Razor bound (McAllester, 2013; Mitliagkas, 2023))

Let $\delta \in [0, 1]$. Given a prior distribution p over $\mathcal{H}$, we have for $\hat{f} = \mathcal{A}(\mathcal{S}_n) \in \mathcal{H}$

$$\tau \geq M\sqrt{\frac{\log \frac{1}{p(\hat{f})} + \log \frac{2}{\delta}}{2n}} \Longrightarrow \mathbb{P}_{\mathcal{S}_n}\left[R[\hat{f}] - \hat{R}_{\mathcal{S}_n}[\hat{f}] \leq \tau\right] \geq 1 - \delta$$

- If our prior distribution gives more probability to $\hat{f}$ then $\log \frac{1}{p(\hat{f})}$ will decrease, therefore giving a tighter bound and vice versa
- If, however, we don't give any probability to a hypothesis $\hat{f}$ (i.e $p(\hat{f}) = 0$) then $\log \frac{1}{p(\hat{f})}$ will be undefined, which provides a vacuous bound

Occam's (Razor) bound (McAllester, 2013; Mitliagkas, 2023)

For a given dataset $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$, let $\hat{f} = \mathcal{A}(\mathcal{S}_n) \in \mathcal{H}$. We have :

$$\begin{aligned}
\mathbb{P}\left[\epsilon^{gen}_{\mathcal{S}_n}[\hat{f}] \geq \tau\right] &\leq \sum_{f\in\mathcal{H}} \mathbb{P}\left[\left|\hat{R}_{\mathcal{S}_n}[f] - R[f]\right| \geq \tau\right] \\
&\leq \sum_{f\in\mathcal{H}} \delta p(f) = \delta \\
&\text{if } \mathbb{P}\left[\left|\hat{R}_{\mathcal{S}_n}[f] - R[f]\right| \geq \tau\right] \leq \delta p(f)\ \forall f \in \mathcal{H}
\end{aligned}$$

This last condition is satisfied is we set $2e^{\frac{-2n\tau^2}{M^2}} \leq \delta p(f) \leq 1$, that is $\tau \geq M\sqrt{\frac{\log \frac{2}{\delta p(f)}}{2n}} = M\sqrt{\frac{\log \frac{1}{p(f)} + \log \frac{2}{\delta}}{2n}}$.

Occam's (Razor) bound : Example

Consider

$$\mathcal{H}_\theta = \left\{ f(x) = Wx + b \; \forall x \in \mathbb{R}^d \mid \theta = \left(W \in \mathbb{R}^{c \times d}, b \in \mathbb{R}^c \right) \right\}$$

In theory, $|\mathcal{H}_\theta| = \infty$ ($\aleph_1$). But if the numbers are stored on $q \geq 2$ bits (for example 32 or 64), we get $|\mathcal{H}_\theta| = 2^{qc(d+1)}$

Using a regularizer $\frac{\gamma}{2}\|\theta\|$ is equivalent to putting a prior on $\mathcal{H}$:

- Laplacian prior for ℓ_1-norm $|\theta| = \sum_{i,j} |W_{i,j}| + \sum_i |b_i|$
- Gaussian prior for ℓ_2-norm $\|\theta\|_2^2 = \sum_{i,j} W_{i,j}^2 + \sum_i b_i^2$

$$p(\theta) = \frac{1}{\sqrt{(2\pi)^k |\Sigma|}} e^{-\frac{1}{2}(\mathrm{vec}\,\theta)^T \Sigma^{-1} \mathrm{vec}\,\theta}$$

$$\theta^{(0)} \sim p(\theta)$$

Some results from statistical learning : PAC learning

Theorem (PAC Bayes bound (Shalev-Shwartz et al., 2014; Mitliagkas, 2023))

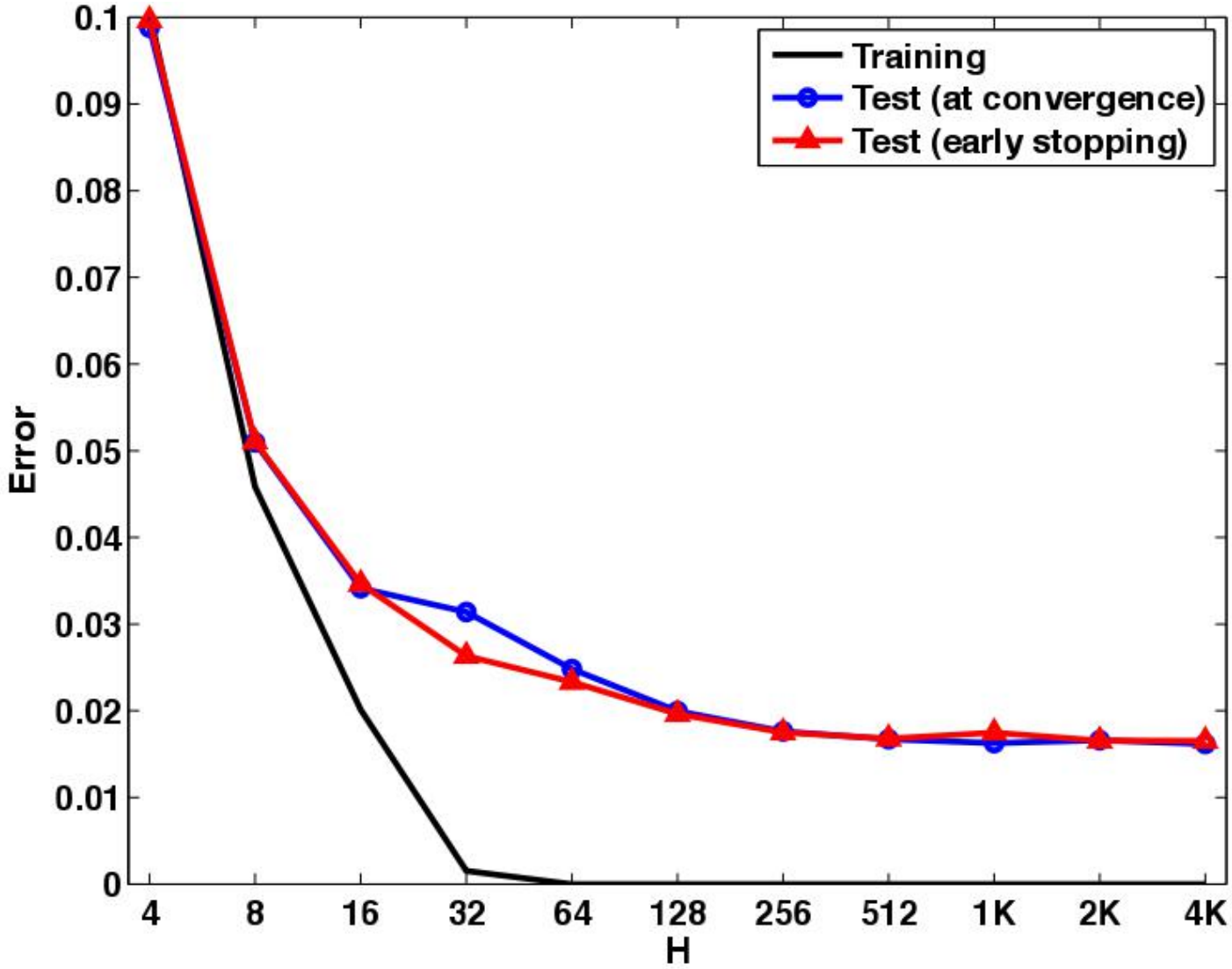
Let $\delta \in [0,1]$. Given a prior distribution p over $\mathcal{H}$ and a posterior probability distribution q over $\mathcal{H}$, we have for all $f \in \mathcal{H}$

$$\mathbb{P}_{\mathcal{S}_n}\left[\mathbb{E}_{f\sim q}\left[R[f] - \hat{R}_{\mathcal{S}_n}[f]\right] \leq \sqrt{\frac{KL\left(q||p\right) + \log\frac{n}{\delta}}{2(n-1)}}\right] \geq 1-\delta$$

If we update our distribution over hypotheses using the posterior, q, so that f performs well on the empirical risk, it ensures that we can reduce $\mathbb{E}_{f\sim q}\hat{R}_{\mathcal{S}_n}[f]$ more than if we just sampled from the prior p. Here are some examples of posteriors:

- $q(\hat{f}) = 1 \Longrightarrow KL\left(q||p\right) = \infty$ and the bound explodes
- $q = p \Longrightarrow KL\left(q||p\right) = 0$ and the bound becomes tight.

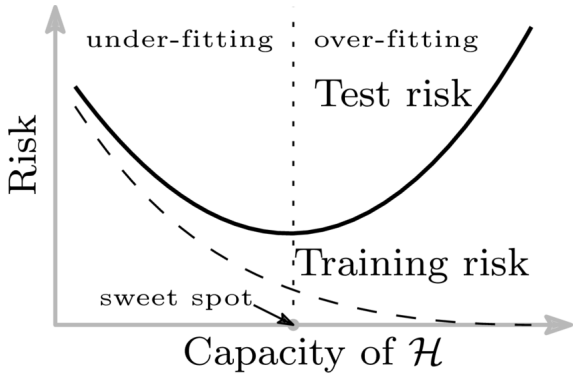
Part IV - Double descent



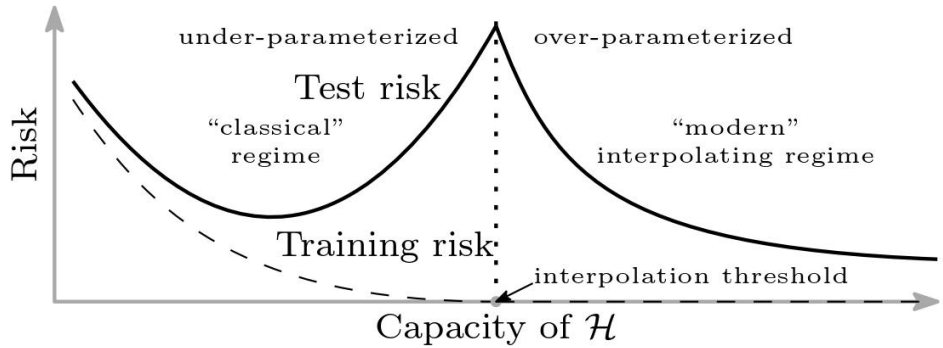
Neyshabur et al. (2014) found that test error simply decreases with network width

Part IV - Double descent

There is growing evidence that test error acts as the classic U-shaped curve in the under-parameterized regime and monotonically decreases in the over-parameterized regime (Belkin et al., 2018).



Classical U-shaped test error curve



Double Descent

Complexity/Capacity of $\mathcal{H}$

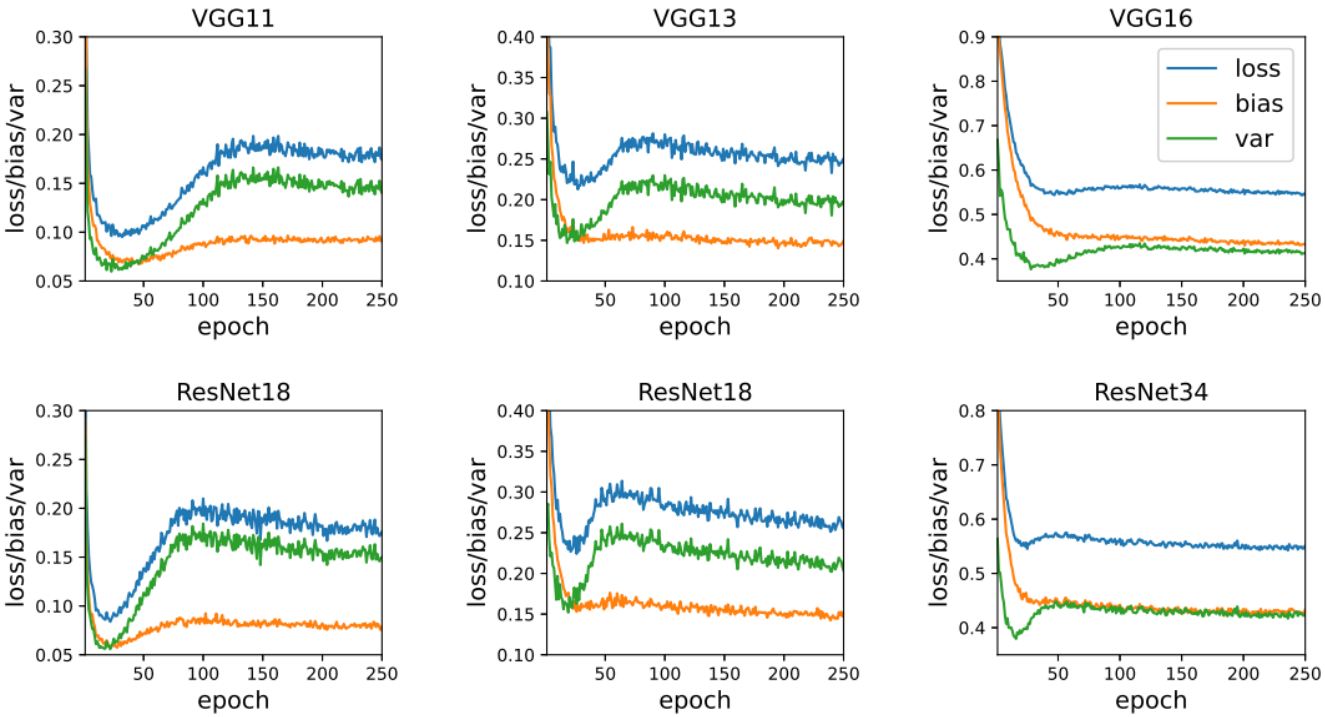
- We use the number of parameters of our model
- Others: VC dimension, Rademacher complexity, etc ...

Epoch-wise Double-Descent

$$\mathbb{E}_{\mathcal{T}}[\mathcal{L}(\boldsymbol{t}, \boldsymbol{y})] = \underbrace{\mathcal{L}(\boldsymbol{t}, \bar{\boldsymbol{y}})}_{\text{Bias}} + \beta \underbrace{\mathbb{E}_{\mathcal{T}}[\mathcal{L}(\bar{\boldsymbol{y}}, \boldsymbol{y})]}_{\text{Variance}},$$

where β takes different values for different loss functions, and $\bar{\boldsymbol{y}}$ is the expected output:

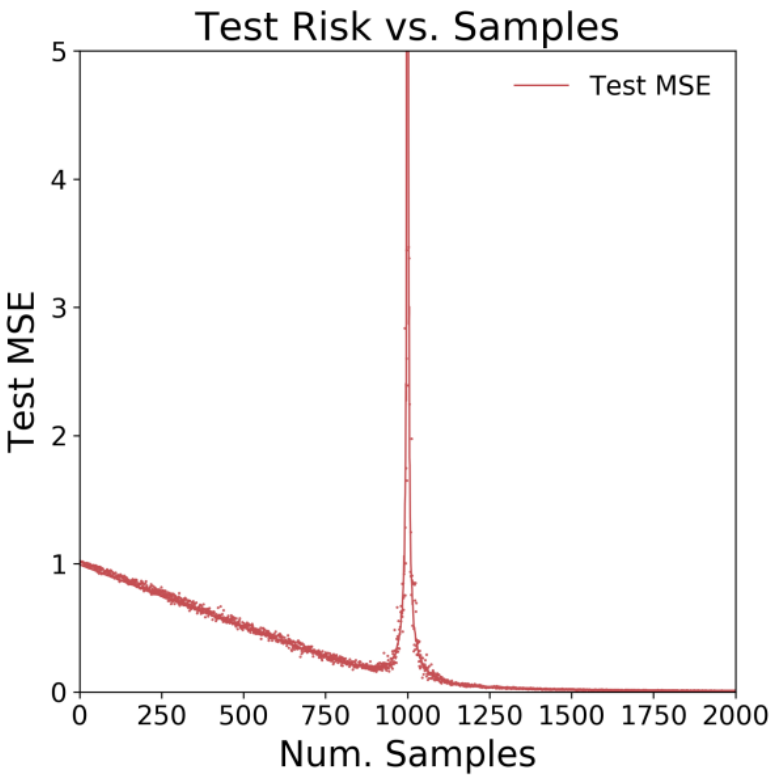
$$\bar{\boldsymbol{y}} = \underset{\boldsymbol{y}^* \in \mathbb{R}^c \,\middle|\, \sum_{k=1}^{c} \boldsymbol{y}_k^* = 1, \boldsymbol{y}_k^* \geq 0}{\arg\min} \mathbb{E}_{\mathcal{T}}[\mathcal{L}(\boldsymbol{y}^*, \boldsymbol{y})].$$



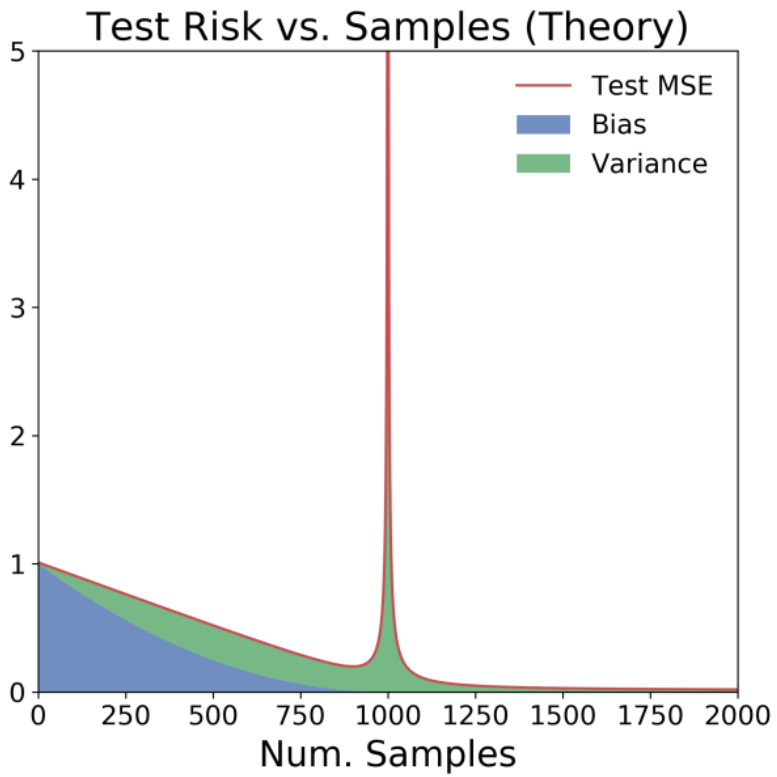
(a) SVHN (b) CIFAR10 (c) CIFAR100

Epoch-wise Double-Descent (Zhang et al., 2021)

Sample-wise Double descent



(a) Test MSE for $d = 1000, \sigma = 0.1$.



(b) Test MSE in theory for $d = 1000, \sigma = 0.1$

Test MSE vs. Num. Train Samples for the min-norm ridgeless regression estimator in 1000 dimensions (Nakkiran, 2019)

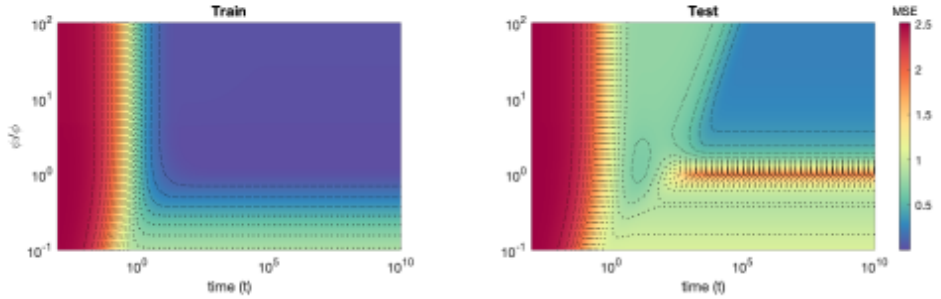


Figure 3: *Model-wise double descent.* Analytical training error and test error evolution with parameters $(\mu, \nu, \phi, r, s, \lambda) = (0.5, 0.3, 3, 2., 0.4, 0.001)$. Note that we vary the number of model parameters (ψ).

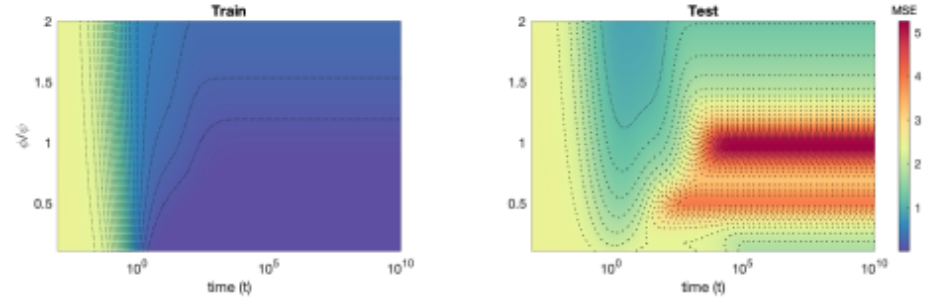


Figure 4: *Sample-wise descents.* Analytical training error and test error evolution with parameters $(\mu, \nu, \psi, r, s, \lambda) = (0.9, 0.1, 2, 1, 0.8, 0.0001)$. Note that we vary the number of samples (ϕ).

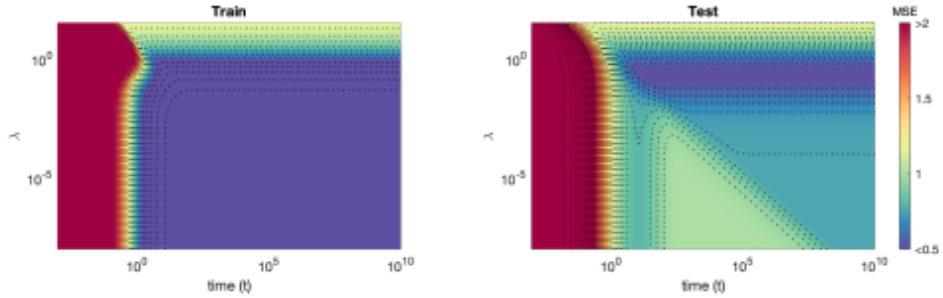


Figure 5: *Epoch-wise descent structures.* Analytical test error evolution with respect to different values of λ $(\mu, \nu, \psi, \phi, r, s) = (0.5, 0.3, 6, 3, 2.0, 0.5)$. Here the ratio of number of parameters and samples is fixed.

Model, sample, epoch-wise descents in random feature model (Bodin et al., 2021)

Double Descent : Linear teacher-student setup

$$\varepsilon \sim \mathcal{N}\left(0, \sigma_\varepsilon^2\right)$$

$$x^* \sim \mathcal{N}(0, \Sigma)$$

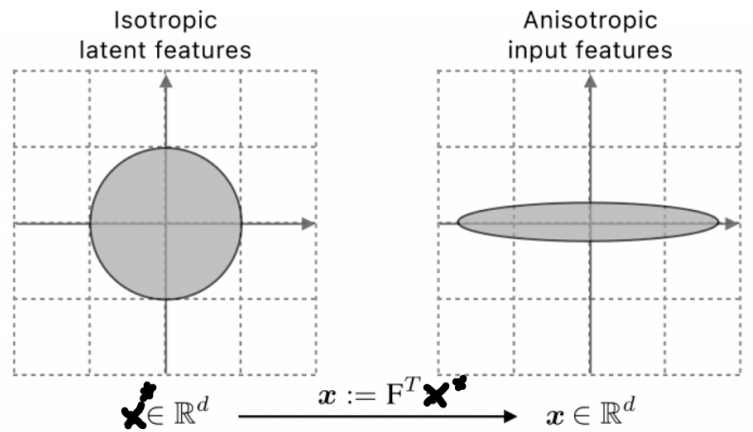
noise

$$\Sigma = diag(\sigma_1^2, ..., \sigma_d^2)$$

$$y^* = f(x^*) = w^T x^* \rightarrow y = y^* + \varepsilon$$

Modulation $F \in \mathbb{R}^{d \times d^*}$

$$x = F^T x^* \sim \mathcal{N}\left(0, F\Sigma F^T\right) \rightarrow \hat{y} = \hat{f}(x) = \hat{w}^T x = \hat{w}^T F^T x^* \rightarrow \frac{1}{2n} \sum_{i=1}^{n} \left(\hat{y}^i - y^i\right)^2 + \frac{\gamma}{2} ||\hat{w}||^2$$



Double Descent : Linear teacher-student setup

$$y = f(x^*) = w^T x^* + \epsilon \quad \text{with} \quad x_i^* \sim \mathcal{N}(0, \sigma_i^2),\ \epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$$

$$\hat{y} = \hat{f}(x) = \hat{w}^T x \quad \text{with} \quad x = F^T x^*$$

True Risk : generalization error

$$\begin{aligned} R[\hat{w}] &= \mathbb{E}_{x^*,\epsilon}\left[\left(\hat{w}^T F^T x^* - w^T x^* - \epsilon\right)^2\right] \\ &= \mathbb{E}_{x^*}\left(\hat{w}^T F^T x^* - w^T x^*\right)^2 - 2\mathbb{E}_{x^*,\epsilon}\left[\left(\hat{w}^T F^T x^* - w^T x^*\right)\epsilon\right] + \mathbb{E}_\epsilon \epsilon^2 \\ &= \mathbb{C}ov\left[(F\hat{w} - w)^T x^*\right] + \sigma_\epsilon^2 \\ &= (F\hat{w} - w)^T \Sigma (F\hat{w} - w) + \sigma_\epsilon^2 \end{aligned}$$

$$R[\hat{w}] = (F\hat{w} - w)^T \Sigma (F\hat{w} - w) + \sigma_\epsilon^2 = \|F\hat{w} - w\|_\Sigma^2 + \sigma_\epsilon^2$$

Theorem ($F = \mathbb{I}_d$)

- *Optimal prediction :* $y^{opt}(x) = w^T x$
- *Main prediction :* $y^{\ell,n}(x) = \hat{w}^T x$
- *Average bias :*

$$\text{Bias}^2[\hat{w}] = \mathbb{E}_x \, \text{Bias}^2(x) = (w - \mathbb{E}[\hat{w}])^T \, \Sigma \, (w - \mathbb{E}[\hat{w}]) = \|w - \mathbb{E}\hat{w}\|_\Sigma^2$$

- *Average variance :*

$$\text{Var}[\hat{w}] = \mathbb{E}_x \, \text{Var}(x) = \text{tr}(\mathbb{C}\text{ov}(\hat{w})\Sigma)$$

$$\mathbb{C}\text{ov}(\hat{w}) = \mathbb{E}\left[(\hat{w} - \mathbb{E}\hat{w})\,(\hat{w} - \mathbb{E}\hat{w})^T\right]$$

- *Noise :* $Noise(x) = \sigma_\epsilon^2$

Training : Empirical risk ($F = \mathbb{I}_d$)

$$\mathcal{S}_n = (X, Y) \quad \text{with} \quad \begin{cases} X = [x_i]_{i=1}^n \in \mathbb{R}^{n \times d}, & x_i \sim \mathcal{N}(0, \Sigma) \\ \epsilon = [\epsilon_i]_{i=1}^n \in \mathbb{R}^n, & \epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2) \\ Y = Xw + \epsilon \in \mathbb{R}^n \end{cases}$$

$$\hat{Y} = X\hat{w} \in \mathbb{R}^n$$

$$\begin{aligned} \hat{R}_{\mathcal{S}_n}[\hat{w}] &= \frac{1}{2n} \sum_{i=1}^{n} (\hat{Y}_i - Y_i)^2 + \frac{\lambda}{2} \|\hat{w}\|^2 \\ &= \frac{1}{2n} \|\hat{Y} - Y\|_2^2 + \frac{\lambda}{2} \hat{w}^T \hat{w} \\ &= \frac{1}{2n} (X\hat{w} - Y)^T (X\hat{w} - Y) + \frac{\lambda}{2} \hat{w}^T \hat{w} \\ &= \frac{1}{2} \hat{w}^T \left(\frac{X^T X}{n} + \gamma \mathbb{I}_d \right) \hat{w} - \frac{1}{n} Y^T X \hat{w} + \frac{1}{2n} Y^T Y \end{aligned}$$

Training : min-norm least squares estimator ($F = \mathbb{I}_d$)

$$\hat{R}_{S_n}[\hat{w}] = \frac{1}{2}\hat{w}^T\left(\frac{X^TX}{n} + \gamma\mathbb{I}_d\right)\hat{w} - \frac{1}{n}Y^TX\hat{w} + \frac{1}{2n}Y^TY$$

$$\nabla_{\hat{w}}\hat{R}_{S_n}[\hat{w}] = \left(\frac{X^TX}{n} + \gamma\mathbb{I}_d\right)\hat{w} - \frac{X^TY}{n} = 0$$

$$\iff \hat{w}^{LS} = \left(\frac{X^TX}{n} + \gamma\mathbb{I}_d\right)^{\dagger}\frac{X^TY}{n}$$

$$\iff \hat{w}^{LS} = \left(\frac{X^TX}{n} + \gamma\mathbb{I}_d\right)^{\dagger}\left(w^T\frac{X^TX}{n} + \frac{\epsilon^TX}{n}\right)$$

$$\frac{X^TX}{n} \underbrace{\to}_{n\to\infty} \mathbb{E}\left[\frac{X^TX}{n}\right] = \Sigma \quad \text{and} \quad \frac{\epsilon^TX}{n} \underbrace{\to}_{n\to\infty} \mathbb{E}\left[\frac{\epsilon^TX}{n}\right] = 0$$

Training : Gradient Descent ($F = \mathbb{I}_d$)

$$\hat{w}^{(t+1)} = \hat{w}^{(t)} - \alpha_t \nabla \hat{R}_{\mathcal{S}_n}[\hat{w}^{(t)}]$$

$$= \underbrace{\left[(1 - \alpha_t \gamma \mathbb{I}_d) - \alpha_t \frac{X^T X}{n}\right]}_{A_t} \hat{w}^{(t)} - \underbrace{\alpha_t \frac{X^T Y}{n}}_{b_t}$$

$$= \left(\prod_{k=0}^{t} A_t\right) \hat{w}^{(0)} - \sum_{k=t}^{1} \left(\prod_{i=0}^{k-1} A_{t-i}\right) b_{t-k} - b_t$$

$$A_t = (1 - \alpha_t \gamma \mathbb{I}_d) - \alpha_t \frac{X^T X}{n} \underbrace{\rightarrow}_{n \to \infty} (1 - \alpha_t \gamma \mathbb{I}_d) - \alpha_t \Sigma$$

$$b_t = \alpha_t \frac{X^T Y}{n} = \alpha_t \left(w^T \frac{X^T X}{n} + \frac{\epsilon^T X}{n}\right) \underbrace{\rightarrow}_{n \to \infty} \alpha_t w^T \Sigma$$

Linear teacher-student setup : training

Training : Least Square Solution and Gradient Descent ($F = \mathbb{I}_d$)

$$\hat{w}^{LS} = \left(\frac{X^T X}{n} + \gamma \mathbb{I}_d\right)^{\dagger} \left(w^T \frac{X^T X}{n} + \frac{\epsilon^T X}{n}\right)$$

If $\alpha_t = \alpha \quad \forall t \in \mathbb{N}$, $A = (1 - \alpha\gamma\mathbb{I}_d) - \alpha\frac{X^T X}{n}$ and

$$\hat{w}^{(t+1)} = A^{t+1}\hat{w}^{(0)} - \alpha\left(\sum_{k=0}^{t} A^k\right)\left(w^T \frac{X^T X}{n} + \frac{\epsilon^T X}{n}\right)$$

Informal theorem : Under certain conditions on the value α (small enough), the rank of $\frac{X^T X}{n}$, ... :

$$\lim_{t\to\infty} \hat{w}^{(t)} = \hat{w}^{LS}$$

Model (d) and Sample (n) wise double descent : Linear teacher-student setup

$F = \mathbb{I}_d$

- Generalization error

$$R[\hat{w}] = (\hat{w} - w)^T \Sigma (\hat{w} - w) + \sigma_\epsilon^2 = \|\hat{w} - w\|_\Sigma^2 + \sigma_\epsilon^2$$

- Average bias :

$$\text{Bias}^2[\hat{w}] = \mathbb{E}_x \, \text{Bias}^2(x) = (w - \mathbb{E}[\hat{w}])^T \, \Sigma \, (w - \mathbb{E}[\hat{w}]) = \|w - \mathbb{E}\hat{w}\|_\Sigma^2$$

- Average variance :

$$\text{Var}[\hat{w}] = \mathbb{E}_x \, \text{Var}(x) = \text{tr}(\mathbb{C}\text{ov}(\hat{w})\Sigma)$$

- Noise : $Noise(x) = \sigma_z^2$

Model (d) and Sample (n) wise double descent

Theorem (Informal (Hastie et al., 2019))

- *Let* $\alpha = d/n$ *be the overparametrization ratio*
- *Assume* $F = \mathbb{I}_d$, $\Sigma = \mathbb{I}_d$
- *Assume* $\|w\|_2^2 = r^2$ *for all n and d*

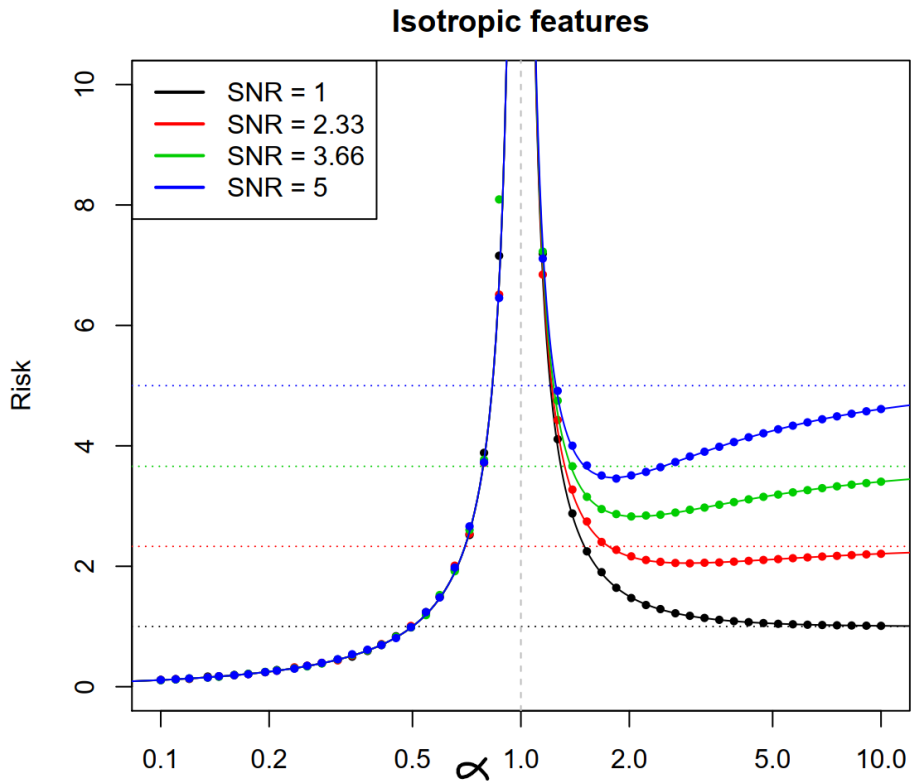
As $n, d \to \infty$ *such that* $d/n \to \alpha \in (0, \infty)$, *it holds almost surely that*

$$\text{Bias}^2[\hat{w}^{LS}] = r^2\left(1 - \frac{1}{\alpha}\right) \quad \textit{and} \quad \text{Var}[\hat{w}^{LS}] = \sigma_\epsilon^2 \frac{1}{1-\alpha}$$

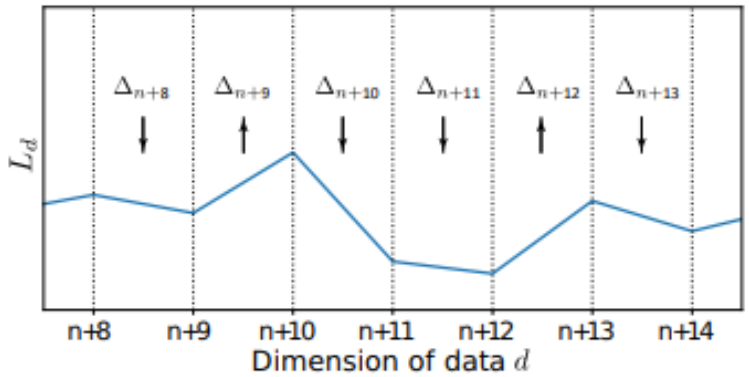
$$R[\hat{w}^{LS}] = \begin{cases} \sigma_\epsilon^2 \frac{1}{1-\alpha} & \textit{if } \alpha < 1 \\ r^2\left(1 - \frac{1}{\alpha}\right) + \sigma_\epsilon^2 \frac{1}{1-\alpha} = \sigma_\epsilon^2 \left[\overbrace{\frac{r^2}{\sigma_\epsilon^2}}^{SNR} \left(1 - \frac{1}{\alpha}\right) + \frac{1}{1-\alpha} \right] & \textit{if } \alpha > 1 \end{cases}$$

$$\text{Bias}^2[\hat{w}^{LS}] = r^2\left(1 - \frac{1}{\alpha}\right) \quad \text{and} \quad \text{Var}[\hat{w}^{LS}] = \sigma_\epsilon^2 \frac{1}{1-\alpha}$$

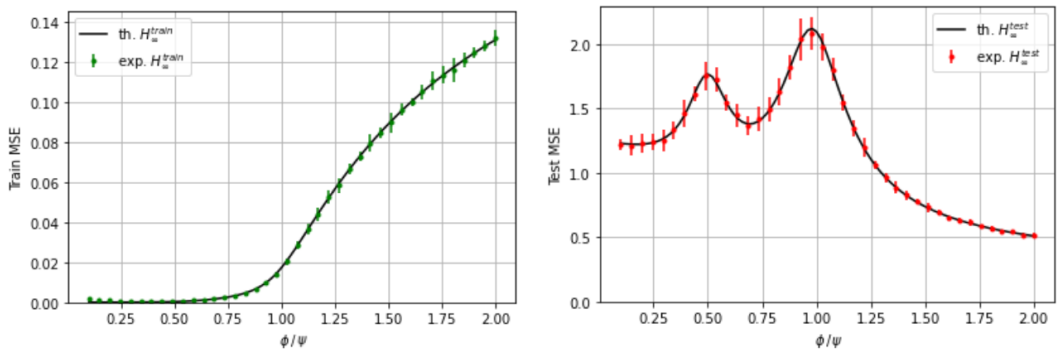
$$R[\hat{w}^{LS}] = \sigma_\epsilon^2 \left[\frac{1}{1-\alpha}\mathbb{I}[\alpha < 1] + \left[SNR\left(1 - \frac{1}{\alpha}\right) + \frac{1}{1-\alpha}\right]\mathbb{I}[\alpha > 1]\right]$$



Multiple descent



Multiple descent phenomenon for the generalization loss L_d versus the dimension of data d in the overparametrized regime starting from $d = n + 8$ (Chen et al., 2021)



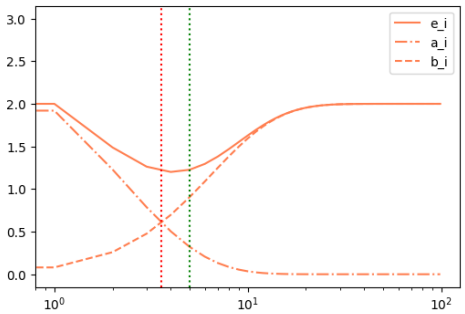
Multiple descent phenomenon for random feature model (Bodin et al., 2021)

Epoch (t) wise double descent

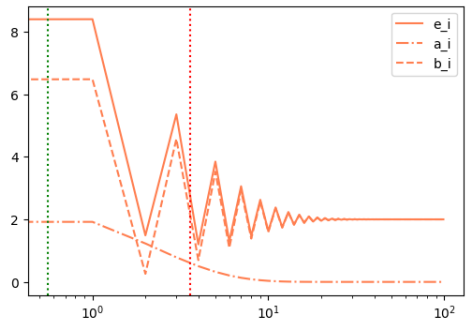
For $a, b, c \in \mathbb{R}$, let $f(\alpha) = \sum_{i=1}^{d} e_i(t)$ for all $\alpha \in [-1, 1]^d$ with

$$e_i(t) = a(\alpha_i^t)^2 + b\left(1 - \alpha_i^t\right)^2 + c\alpha_i^t\left(1 - \alpha_i^t\right)$$

Let's assume without loss of generality that $c = 0$. If $\alpha_i \in [0, 1]$, $e_i(t)$ is a superposition of two U-curves and reaches its minimum at $t^* = \frac{\log\left(\frac{b-\sqrt{ab}}{b-a}\right)}{\log|\alpha_i|}$. If $\alpha_i \in [-1, 0]$, $e_i(t)$ converges by oscillating.



(a) $\alpha_i = 0.8$



(b) $\alpha_i = -0.8$

$$a = 3, b = 2.$$

Epoch (t) wise double descent

Theorem (Informal *(Notsawo, work in progress)*)

- *Let* $\alpha = d/n$ *be the overparametrization ratio*
- *Assume* $F = \mathbb{I}_d$, $\Sigma = \mathbb{I}_d$
- *We also assume* $\mathbb{E}w_i = 0$ *and* $\text{Var}\, w_i = \sigma_w^2$ *for each* $i \in [d]$
- $\mathbb{E}\hat{w}_i^{(0)} = 0$ *and* $\text{Var}\, \hat{w}_i^{(0)} = \sigma^2_{\hat{w}^{(0)}}$ *for each* $i \in [d]$
- $SNR = \sigma_w^2/\sigma_\epsilon^2$ *: signal-to-noise ratio*
- $INR = \sigma^2_{\hat{w}^{(0)}}/\sigma_w^2$ *: initialization noise ratio*

$$R[\hat{w}^{(t)}] = \sigma_\epsilon^2 + \mathbb{E}_X \sum_{i=1}^{d} e_i(t)$$

$$\hat{R}_{S_n}[\hat{w}^{(t)}] = \frac{1}{2}\left[\frac{\sigma_\epsilon^2}{\alpha} + \gamma\sigma_w^2 d + \mathbb{E}_X \sum_{i=1}^{d} (\lambda_i + \gamma)\, e_i(t)\right] + \gamma \mathbb{E}_X \sum_{i=1}^{d} g_i(t)$$

Epoch (t) wise double descent

$$R[\hat{w}^{(t)}] = \sigma_\epsilon^2 + \mathbb{E}_X \sum_{i=1}^{d} e_i(t)$$

$$\hat{R}_{S_n}[\hat{w}^{(t)}] = \frac{1}{2}\left[\frac{\sigma_\epsilon^2}{\alpha} + \gamma\sigma_w^2 d + \mathbb{E}_X \sum_{i=1}^{d} (\lambda_i + \gamma)\, e_i(t)\right] + \gamma \mathbb{E}_X \sum_{i=1}^{d} g_i(t)$$

$$e_i(t) = \begin{cases} \sigma_w^2 + \sigma_{\hat{w}(0)}^2 & \text{if } \lambda_i = \\ \alpha_i^{2t}\left(\sigma_w^2 + \sigma_{\hat{w}(0)}^2\right) + \frac{2\gamma\sigma_w^2}{\lambda_i+\gamma}\alpha_i^t\left(1-\alpha_i^t\right) + \frac{\left(\frac{\lambda_i\sigma_z^2}{n}+\gamma^2\sigma_w^2\right)}{(\lambda_i+\gamma)^2}\left(1-\alpha_i^t\right)^2 & \text{if } \lambda_i \neq \end{cases}$$

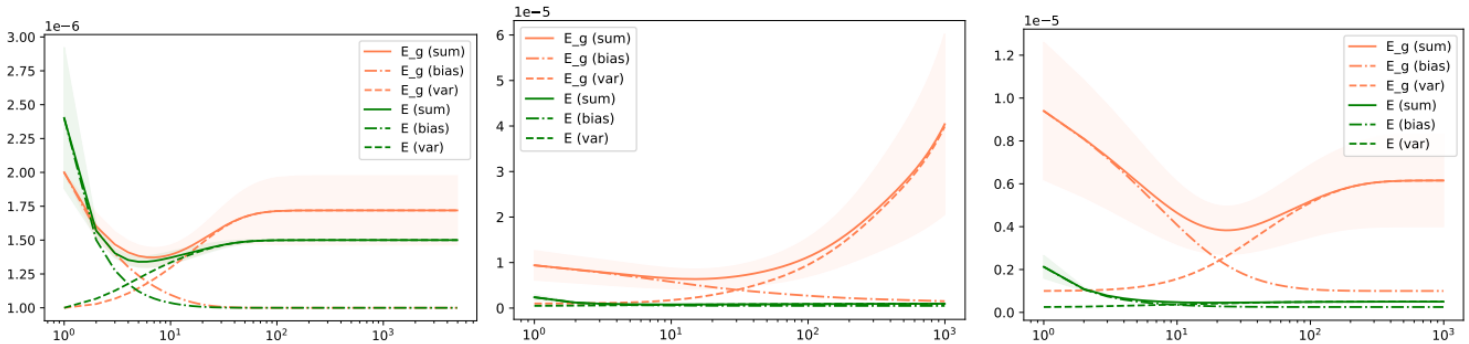
$$= \begin{cases} \sigma_w^2 + \sigma_{\hat{w}(0)}^2 & \text{if } \lambda_i = 0 \wedge \gamma = 0 \\ (1-\eta\lambda_i)^{2t}\left(\sigma_w^2 + \sigma_{\hat{w}(0)}^2\right) + \frac{\sigma_z^2}{n\lambda_i}\left(1-(1-\eta\lambda_i)^t\right)^2 & \text{if } \lambda_i \neq 0 \vee \gamma \neq 0 \end{cases}$$

$\gamma = 0$ and $\frac{X^T X}{n}$ full rank (i.e. $\lambda_i \neq 0 \; \forall \lambda_i \in Spectral\left(\frac{X^T X}{n}\right)$)

$$R[\hat{w}^{(t)}] = \sigma_\epsilon^2 + \mathbb{E}_X \sum_{i=1}^{d} e_i(t) \quad \text{and} \quad \hat{R}_{S_n}[\hat{w}^{(t)}] = \frac{1}{2}\left[\frac{\sigma_\epsilon^2}{\alpha} + \mathbb{E}_X \sum_{i=1}^{d} \lambda_i e_i(t)\right]$$

$$e_i(t) = \left(\sigma_w^2 + \sigma_{\hat{w}(0)}^2\right)\left((1-\eta\lambda_i)^t\right)^2 + \frac{\sigma_z^2}{n\lambda_i}\left(1-(1-\eta\lambda_i)^t\right)^2$$

$$= \sigma_w^2\left[\left(1 + \frac{\sigma_{\hat{w}(0)}^2}{\sigma_w^2}\right)\left((1-\eta\lambda_i)^t\right)^2 + \frac{\sigma_z^2}{\sigma_w^2}\frac{1}{n\lambda_i}\left(1-(1-\eta\lambda_i)^t\right)^2\right]$$



(a) $\alpha = 0.5$ (b) $\alpha = 1$ (c) $\alpha = 2.0$

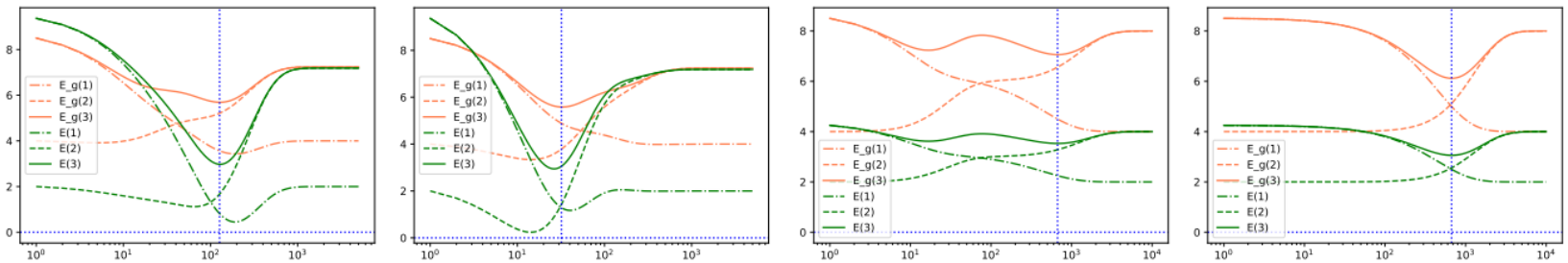
$\gamma = 0$ and $\frac{X^T X}{n}$ full rank (i.e. $\lambda_i \neq 0 \ \forall \lambda_i \in Spectral\left(\frac{X^T X}{n}\right)$)

As $n, d \to \infty$, $\frac{X^T X}{n}$ converge to Σ, so the $\lambda_i \to \sigma_i^2$ for all $i \in [d]$.

$$R[\hat{w}^{(t)}] = \sigma_\epsilon^2 + \mathbb{E}_X \sum_{i=1}^{d} e_i(t) \quad \text{and} \quad \hat{R}_{S_n}[\hat{w}^{(t)}] = \frac{1}{2}\left[\frac{\sigma_\epsilon^2}{\alpha} + \mathbb{E}_X \sum_{i=1}^{d} \sigma_i^2 e_i(t)\right]$$

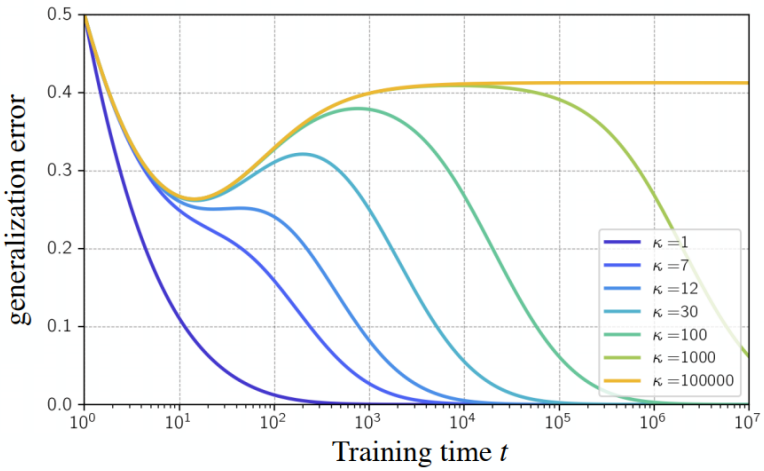
$$e_i(t) = \left(\sigma_w^2 + \sigma_{\hat{w}(0)}^2\right)\left(\left(1-\eta\sigma_i^2\right)^t\right)^2 + \frac{\sigma_z^2}{n\sigma_i^2}\left(1-\left(1-\eta\sigma_i^2\right)^t\right)^2$$

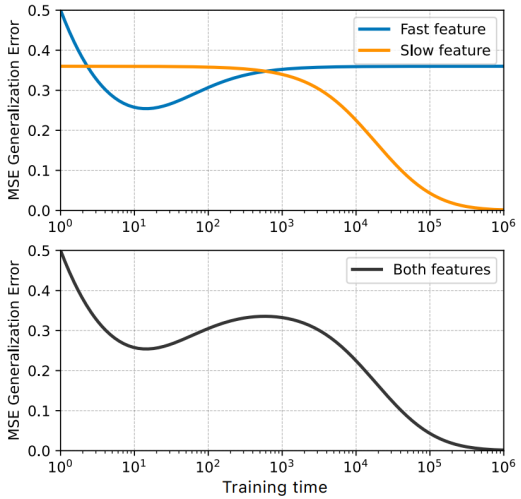
$$= \sigma_w^2\left[\left(1+\frac{\sigma_{\hat{w}(0)}^2}{\sigma_w^2}\right)\left(\left(1-\eta\sigma_i^2\right)^t\right)^2 + \frac{\sigma_z^2}{\sigma_w^2}\frac{1}{n\sigma_i^2}\left(1-\left(1-\eta\sigma_i^2\right)^t\right)^2\right]$$



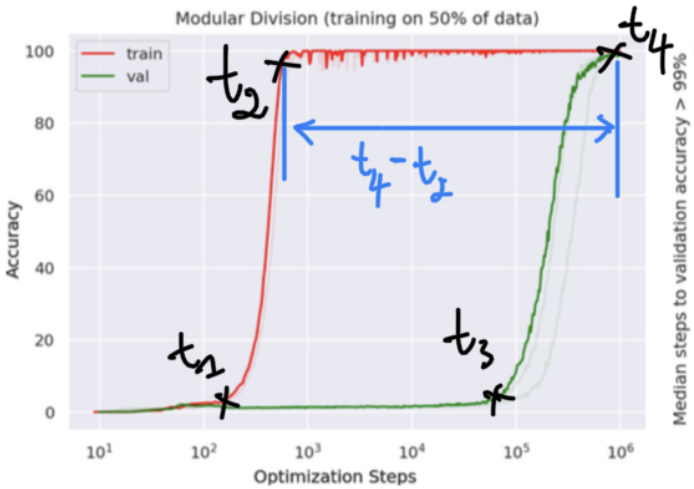
Epoch-wise Double-Descent: feature learning base explanation (Pezeshki et al., 2021)

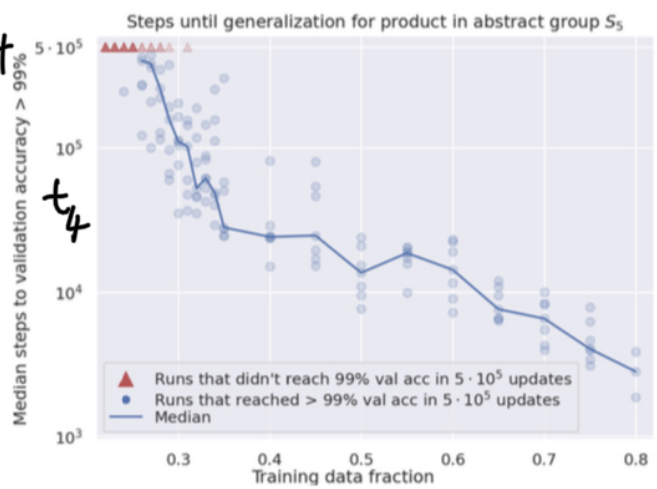
- $\Sigma = \mathbb{I}_d$ and $F = U_F \Lambda_F^{\frac{1}{2}} V_F^T \neq \mathbb{I}_d$ under singular values decomposition
- $\mathbb{C}\text{ov}(F^T x) = FF^T = V_F \Lambda_F V_F^T$, thus creating a correlation between the input features and allowing a second descent.
- The modulation matrix, $F \in \mathbb{R}^{d \times d}$, under a singular value decomposition has 2 singular values : σ_1 and σ_2, with $\sigma_2 \geq \sigma_1$
- condition number of F : $\kappa = \frac{\sigma_2}{\sigma_1} > 1$





Part V - Grokking

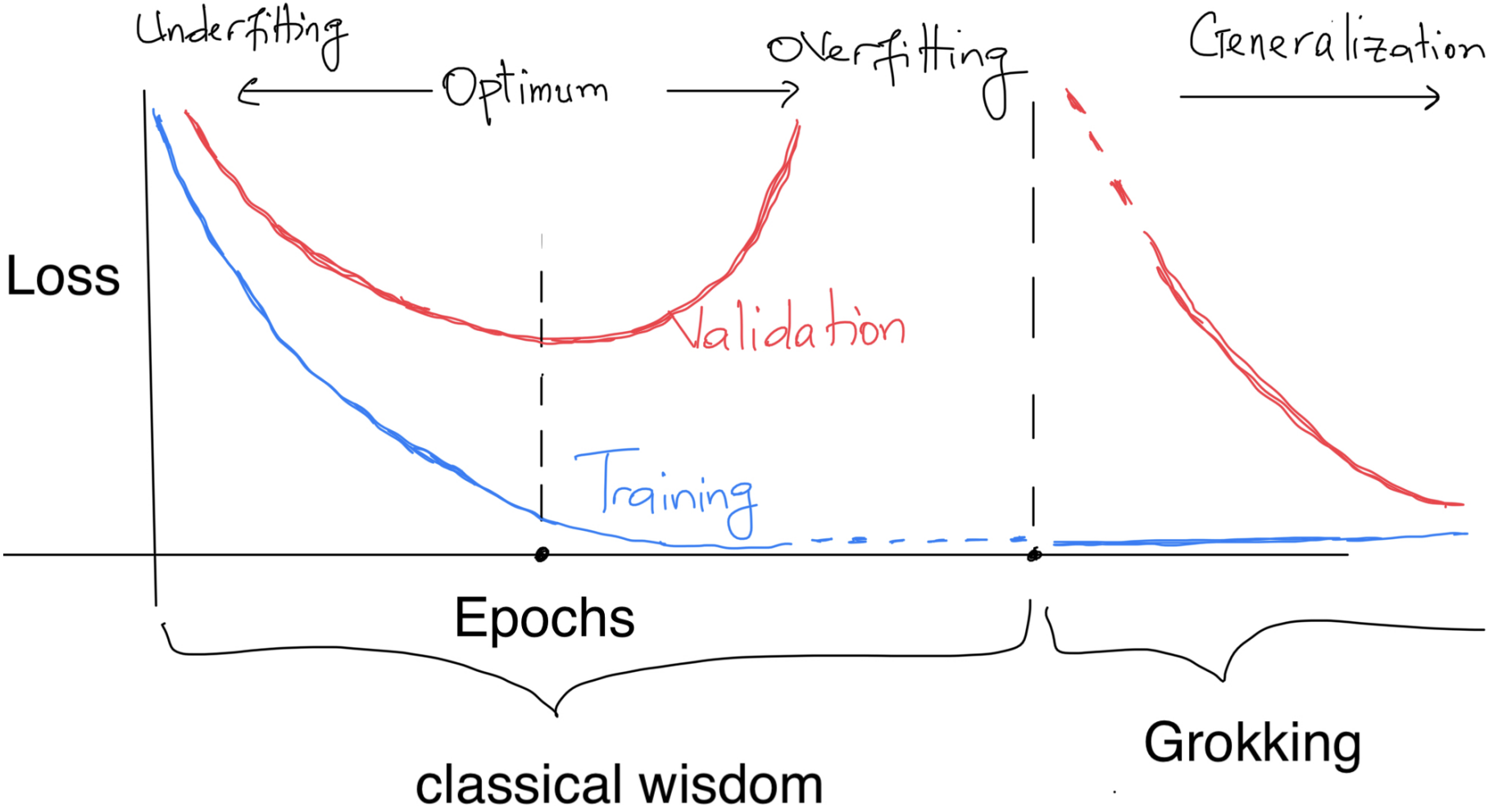




★	a	b	c	d	e
a	a	d	?	c	d
b	c	d	d	a	c
c	?	e	d	b	d
d	a	?	?	b	c
e	b	b	c	?	a

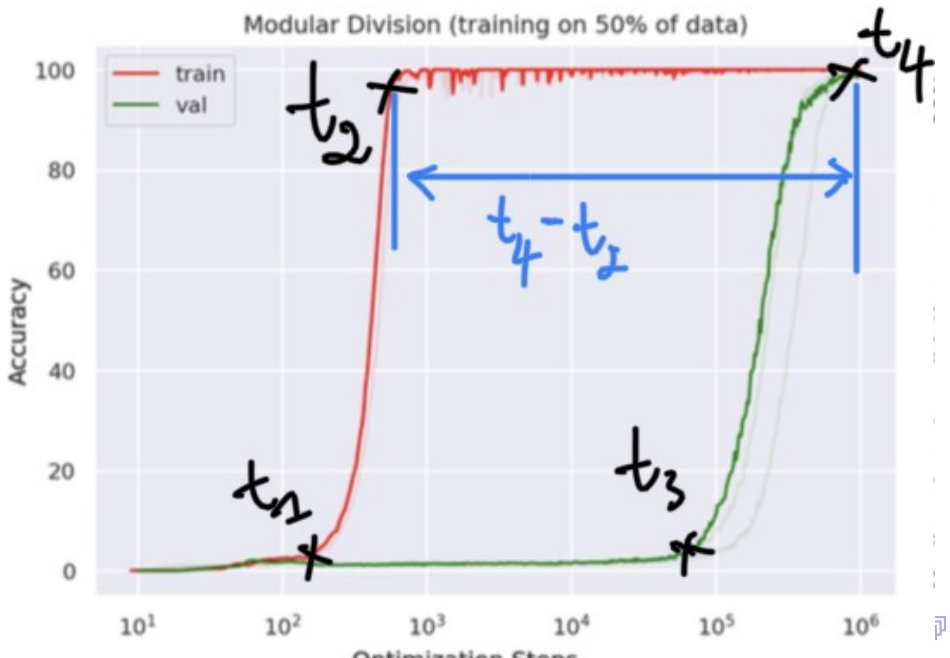
Generalization after overfitting (Power et al., 2022), training and validation accuracies. Training accuracy becomes close to perfect at $t_2 < 1k$ optimization steps, but it takes close to $t_4 \approx 1000k$ steps for validation accuracy to reach that level.

Grokking: late generalization

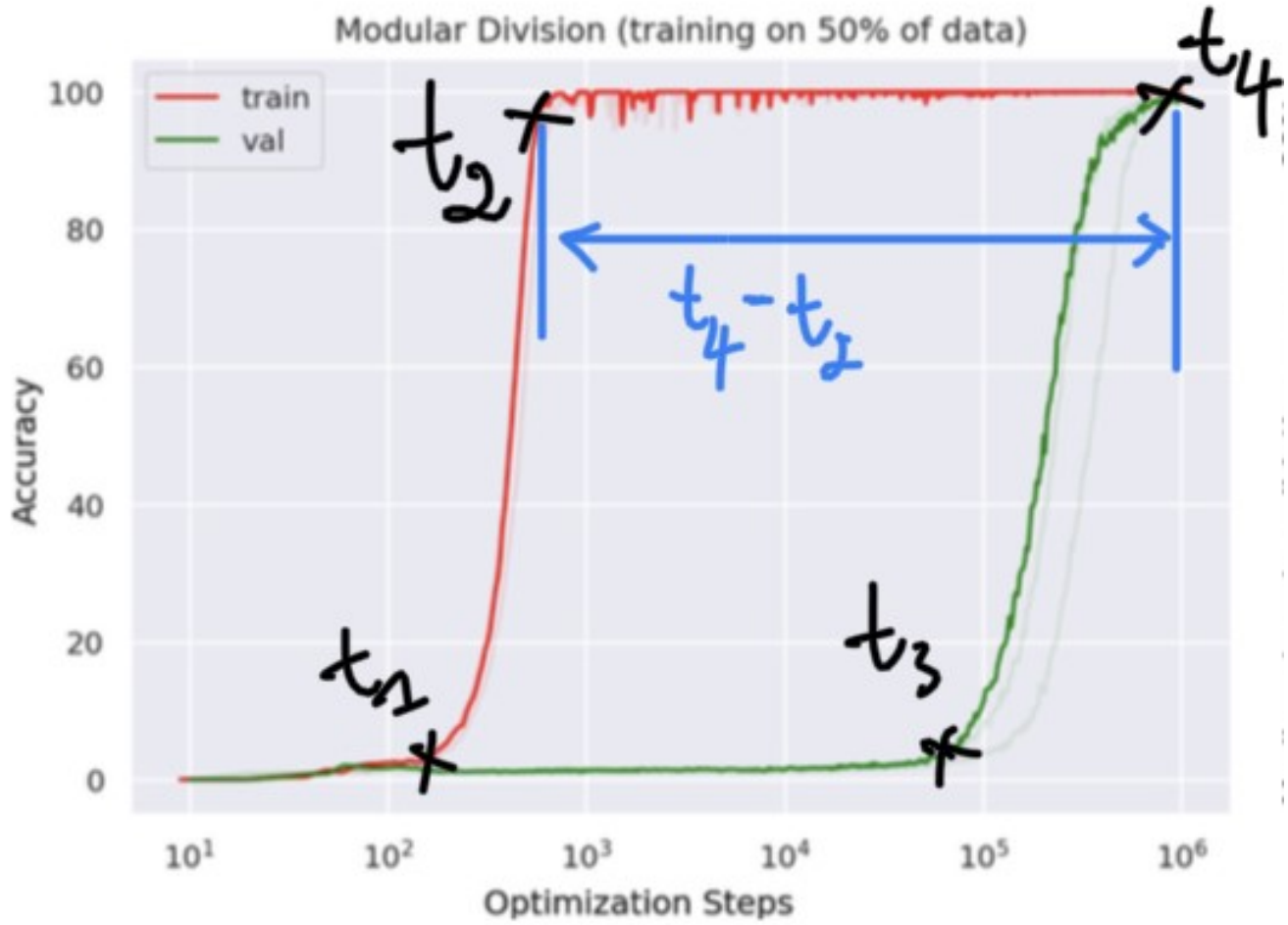


Four learning phases (Liu, Kitouni, et al., 2022)

- Confusion : $t \in [0, t_1]$
- Memorization : $t \in [t_2, t_3]$
- Comprehension : $t \in [t_3, \infty]$
- Generalization : $\mathbb{P}[t_4 < \infty] = 1$
  The measure $\mathbb{P}$ captures randomness in choice of training and validation points, noise in data, learning algorithm (initialization, noise in optimization...)



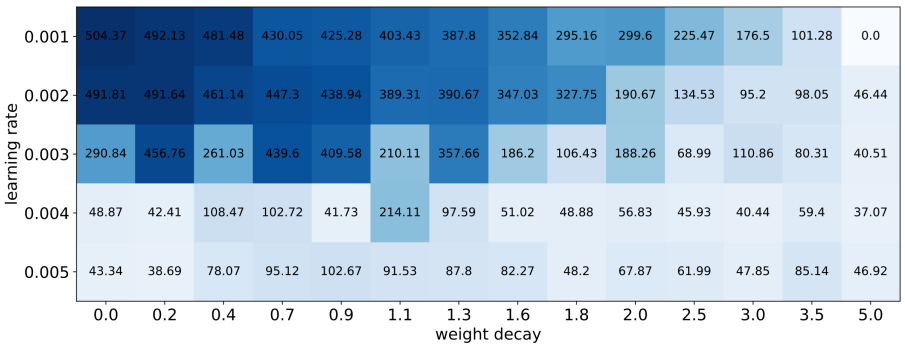
- Generalization : $t_4 < \infty$ almost surely (wrt the randomness in initialization, choice of training and validation points, noise in optimization...)
- Grokking $\approx$ a generalization with $t_4 \gg t_2$.



Spectral Signature of the loss is correlated to generalization (Notsawo et al., 2023)

Spectral Energy (Hjorth's activity)

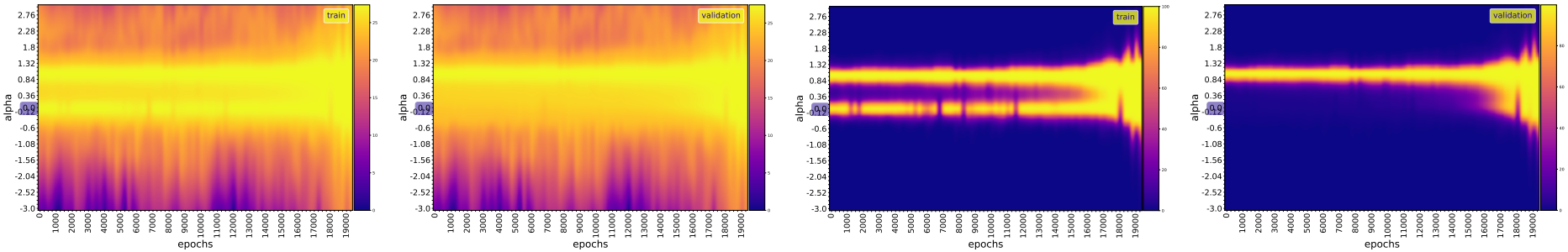
- $\theta(t)$: parameter update at time t given the optimization algorithm
- $L(t)$: loss at $\theta(t)$
- $\mathcal{F}(L)$: Fourier transform of $L(t)$
- $m_n(L) = \int \omega^n \|\mathcal{F}(L)(\omega)\|^2 d\omega$: the n^{th} moment of $\mathcal{F}^2(L)$
- $\|\mathcal{F}(L)(\omega)\|^2$: energy spectral density present in the pulse ω
- Hjorth's activity $m_0(L)$: signal power



(a) Energy: 400 steps (train loss)

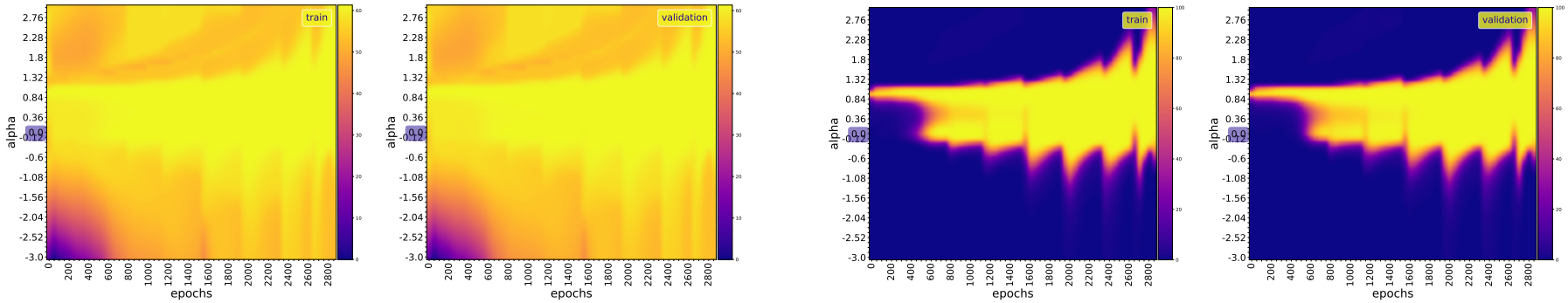
(b) Final test accuracy (10K steps)

Grokking loss landscape (Notsawo et al., 2023)



(a) Loss surface : $f_t(\alpha) = Loss(\theta_t + \alpha\vec{\delta}_t)$ (b) Accuracy : $f_t(\alpha) = Acc(\theta_t + \alpha\vec{\delta}_t)$

$r = 0.30$ (less data), $\vec{\delta}_t \propto \theta^* - \theta_t$



$r = 0.85$ (more data), $\vec{\delta}_t \propto \theta^* - \theta_t$

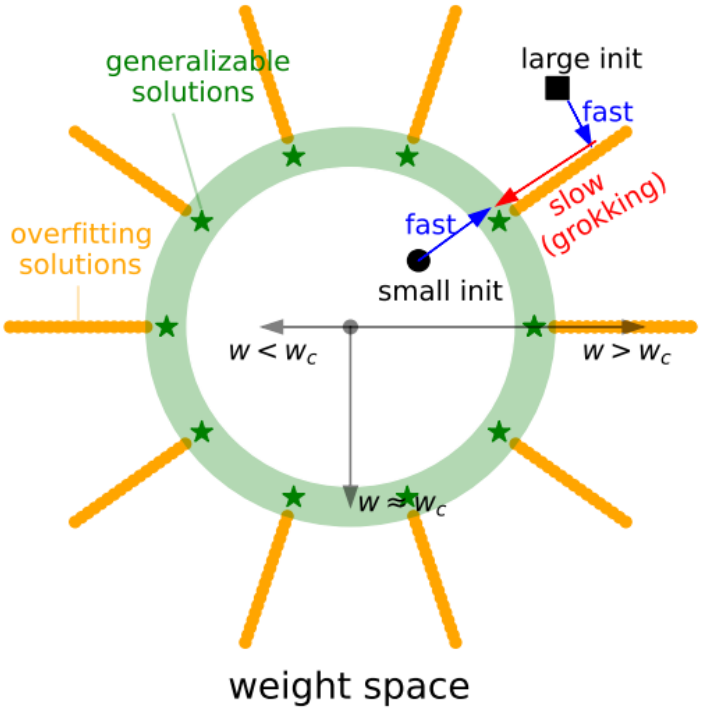
Others observations (Notsawo et al., 2023)

- Larger condition numbers $\lambda_{max}/\lambda_{min}$ of the hessian of the grokking loss: leading to a slower convergence of gradient descent.
- The optimization dynamics is embedded in a low-dimensional space: more than 98% of the total variance in the parameter space occurs in the first 2 PCA modes much smaller than the total number of weights,
- The model remains in a lazy training regime most of the time: the cosine distance between the model weights from one training step to the next remains almost constant, except at the slingshot location.

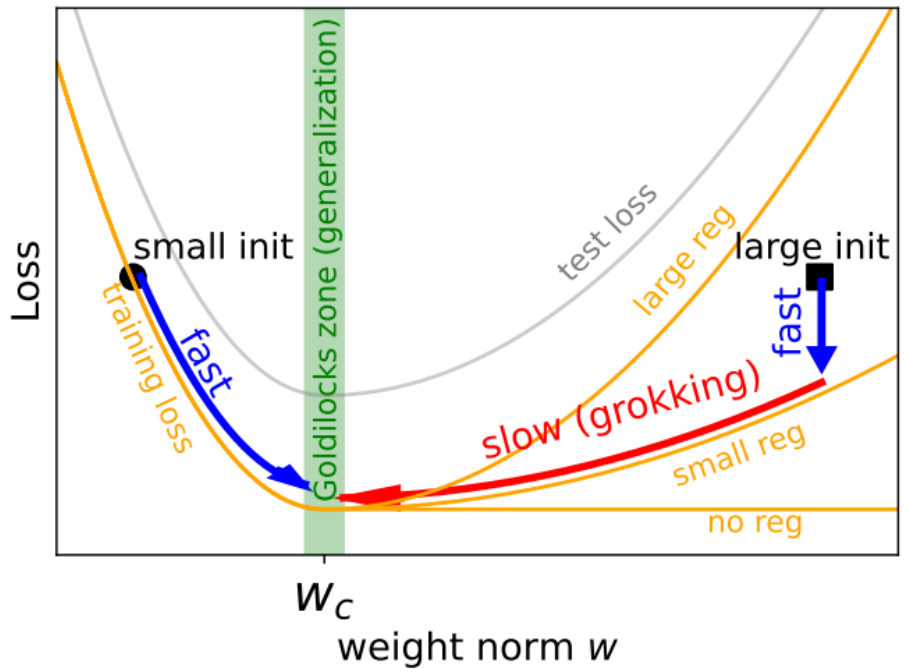
Under realistic hypotheses (Dziugaite et al., 2017) :

- SGD finds good solutions only if they are surrounded by a relatively large volume of solutions that are nearly as good
- SGD performs implicit regularization or tends to find solutions that possess some particular structural property that we already know to be connected to generalization, like widder minima

Grokking : "LU mechanism" (Liu, Michaud, et al., 2023)



(a)



(b)

Figure 1: (a) w: L_2 norm of model weights. Generalizing solutions (green stars) are concentrated around a sphere in the weight space where $w \approx w_c$ (green). Overfitting solutions (orange) populate the $w \gtrsim w_c$ region. (b) The training loss (orange) and test loss (gray) have the shape of L and U, respectively. Their mismatch in the $w > w_c$ region leads to fast-slow dynamics, resulting in grokking.

Grokking : Good Representation (Liu, Kitouni, et al., 2022)

- Generalization can be attributed to learning a good representation of the input embeddings
- The critical training set size corresponds to the least amount of training data that can determine such a representation

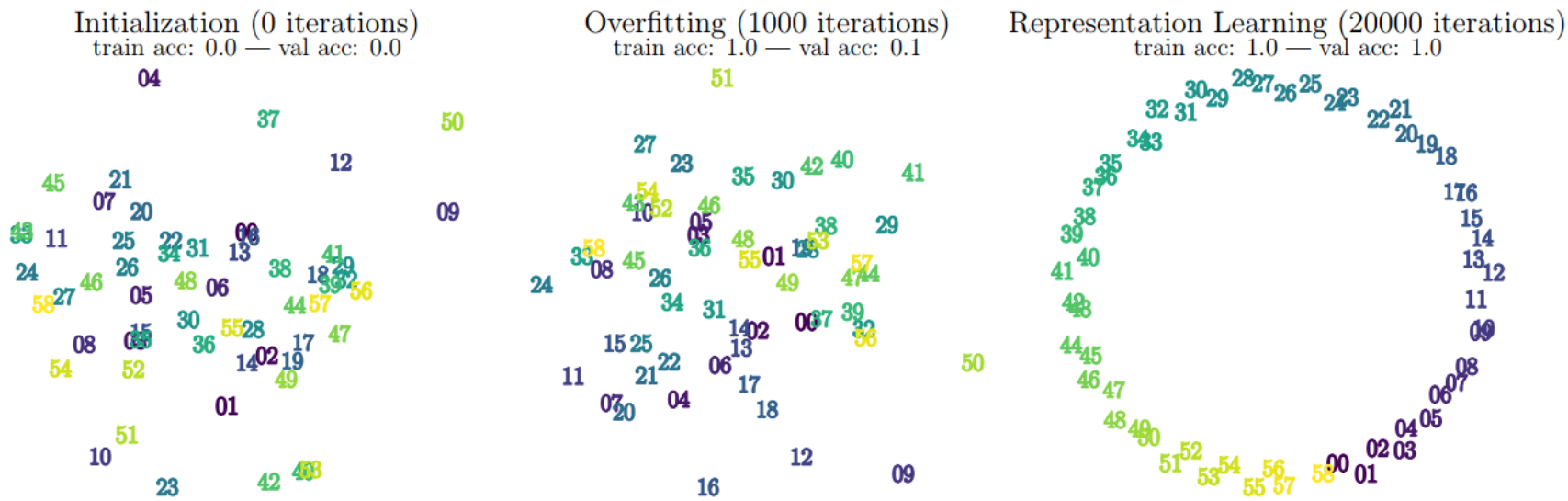
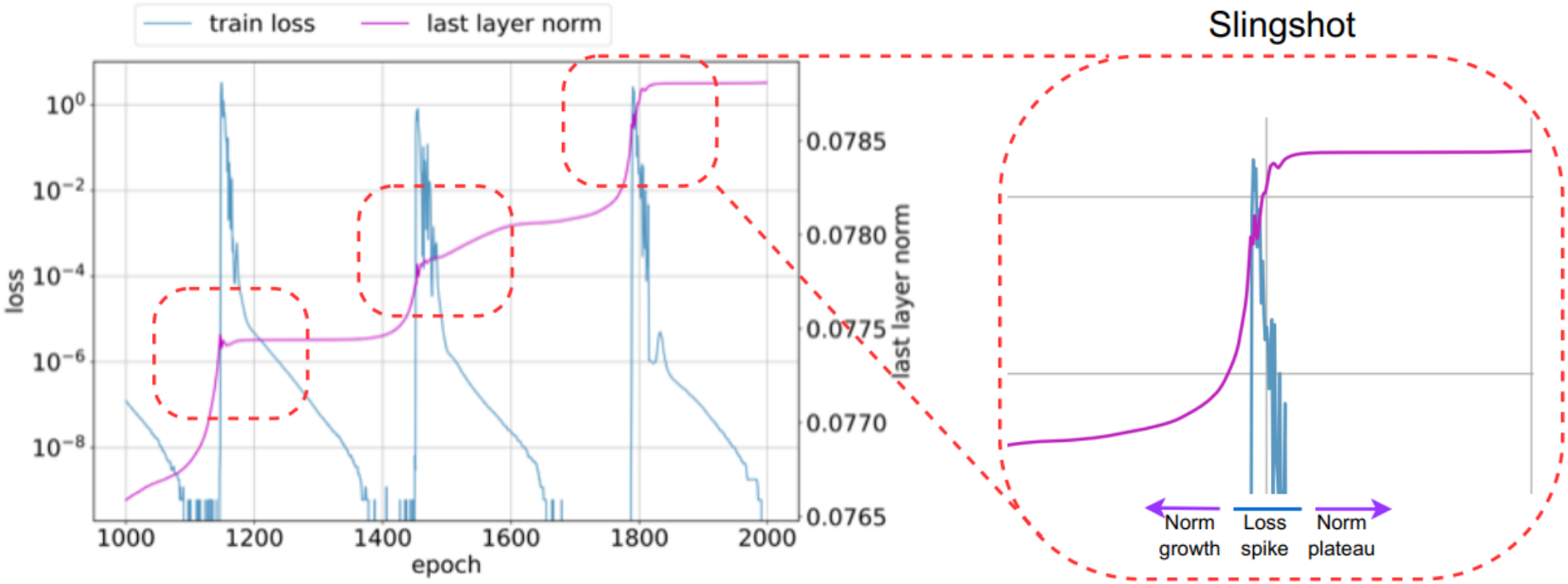


Figure 1: Visualization of the first two principal components of the learned input embeddings at different training stages of a transformer learning modular addition. We observe that generalization coincides with the emergence of structure in the embeddings. See Section 4.2 for the training details.

Grokking: Slingshot mechanism (Thilak et al., 2022)



Slingshot mechanism generally come in tandem with grokking, i.e. grokking almost exclusively happens at the onset of slingshots and is absent without it (Thilak et al., 2022)

Part VI - Why is it important to study such phenomena? *(Grokking, Double descent, emergent behavior, phase transition, etc)*

- Understanding all these behaviours and how they affect the predictive performance of neural networks (at scale or out-of-distribution) is relevant to safety or may have potential safety consequences.
- We need to be certain of a model's safety before we scale it to a capability level beyond which we cannot control it
- Out-of-distribution generalization behaviour of deep learning models is known to be challenging to control or foresee.

[1] Mikhail Belkin et al. "Reconciling modern machine learning practice and the bias-variance trade-off". In: *arXiv preprint arXiv: Arxiv-1812.11118* (2018).

[2] Antoine Bodin et al. "Model, sample, and epoch-wise descents: exact solution of gradient flow in the random feature model". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 21605–21617.

[3] Lin Chen et al. "Multiple descent: Design your own generalization curve". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 8898–8912.

[4] Pedro M. Domingos. "A Unified Bias-Variance Decomposition for Zero-One and Squared Loss". In: *AAAI/IAAI*. 2000. URL: https://api.semanticscholar.org/CorpusID:2063488.

[5] Gintare Karolina Dziugaite et al. "Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data". In: ed. by Gal Elidan et al. AUAI Press, 2017. URL: http://auai.org/uai2017/proceedings/papers/173.pdf.

[6] T. Hastie et al. "Surprises in High-Dimensional Ridgeless Least Squares Interpolation". In: *Annals Of Statistics* (2019). DOI: 10.1214/21-aos2133.

[7] Ziming Liu, Ouail Kitouni, et al. "Towards understanding grokking: An effective theory of representation learning". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 34651–34663.

[8] Ziming Liu, Eric J Michaud, et al. "Omnigrok: Grokking Beyond Algorithmic Data". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=zDiHoIWa0q1.

[9] David McAllester. "A PAC-Bayesian Tutorial with A Dropout Bound". In: *arXiv preprint arXiv: Arxiv-1307.2118* (2013).

[10] Ioannis Mitliagkas. *IFT6169: Theoretical principles for deep learning*. Winter 2023.

[11] Preetum Nakkiran. "More Data Can Hurt for Linear Regression: Sample-wise Double Descent". In: *arXiv preprint arXiv: 1912.07242* (2019).

[12] Behnam Neyshabur et al. "In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning". In: *International Conference On Learning Representations* (2014).

[13] Pascal Jr. Tikeng Notsawo et al. "Predicting Grokking Long Before it Happens: A look into the loss landscape of models which grok". In: *arXiv preprint arXiv: 2306.13253* (2023).

[14] M. Pezeshki et al. "Multi-scale Feature Learning Dynamics: Insights for Double Descent". In: *icml* (2021).

[15] Alethea Power et al. "Grokking: Generalization beyond overfitting on small algorithmic datasets". In: *arXiv preprint arXiv:2201.02177* (2022).

[16] Shai Shalev-Shwartz et al. *Understanding Machine Learning: From Theory to Algorithms*. USA: Cambridge University Press, 2014. ISBN: 1107057132.

[17] Vimal Thilak et al. "The Slingshot Mechanism: An Empirical Study of Adaptive Optimizers and the {Grokking Phenomenon}". In: 2022.

[18] Xiao Zhang et al. “Optimization Variance: Exploring Generalization Properties of DNNs”. In: *arXiv preprint arXiv: Arxiv-2106.01714* (2021).