Frequency Principle

Yaoyu Zhang, Tao Luo

- **Background**
- **Frequency Principle**
- **Implication of F-Principle**
- **Quantitative theory for F-Principle**

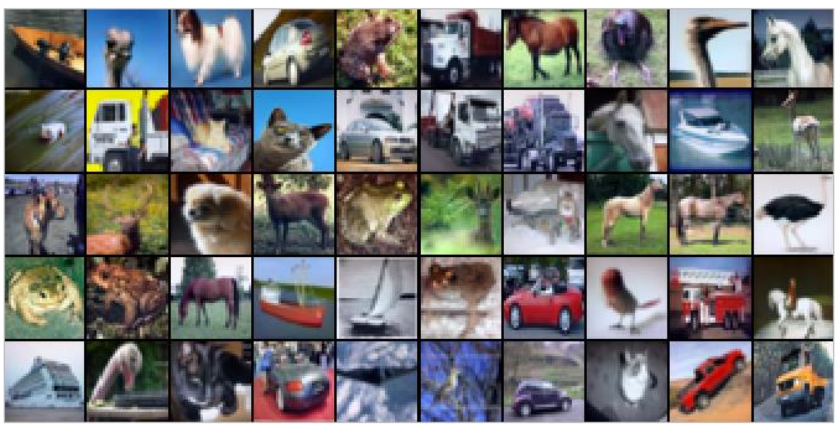
Background—Why DNN remains a mystery?

Supervised Learning Problem

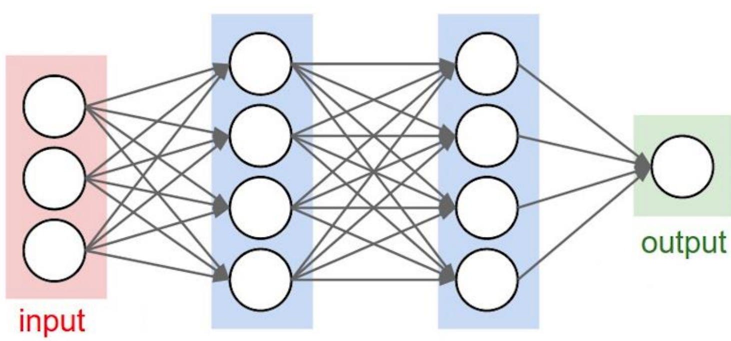
Given $\mathcal{D}$: $\{(x_i, y_i)\}_{i=1}^{n}$ and $\mathcal{H}$: $\{f(\cdot;\Theta) | \Theta \in \mathbb{R}^m\}$, find $f \in \mathcal{H}$ such that $f(x_i) = y_i$ for $i = 1, \cdots, n$.

Example 1 (mystery)—**Deep Learning**

$\mathcal{D}$



$\mathcal{H}$



$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{W}^{[L]}\sigma \circ (\cdots \boldsymbol{W}^{[2]}\sigma \circ (\boldsymbol{W}^{[1]}\boldsymbol{x} + \boldsymbol{b}^{[1]}) + \cdots) + \boldsymbol{b}^{[L]}$$

find

$$\dot{\Theta} = -\nabla_{\Theta} L(\Theta)$$

Initialized by special Θ_0

$$L(\Theta) = \Sigma_{i=1}^{n}(h(x_i;\Theta) - y_i)^2/2n$$

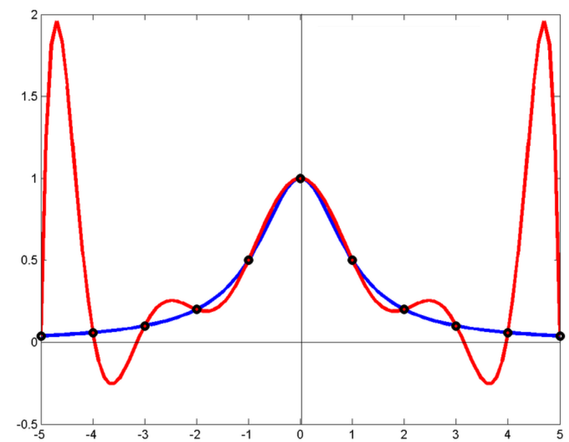
Supervised Learning Problem

Given $\mathcal{D}: \{(x_i, y_i)\}_{i=1}^{n}$ and $\mathcal{H}: \{f(\cdot;\Theta) | \Theta \in \mathbb{R}^m\}$, find $f \in \mathcal{H}$ such that $f(x_i) = y_i$ for $i = 1, \cdots, n$.

Example 2 (well understood)—**polynomial interpolation**

$\mathcal{D}$

$$\{(x_i \in \mathbb{R}, y_i \in \mathbb{R})\}_{i=1}^{n}$$



$\mathcal{H}$

$$h(x;\Theta) = \theta_1 + \cdots + \theta_M x^{m-1}$$

with $m = n$

find

Newton's interpolation formula

Q: Why we think polynomial interpolation is well understood?

Supervised Learning Problem

Given $\mathcal{D}$: $\{(x_i, y_i)\}_{i=1}^{n}$ and $\mathcal{H}$: $\{f(\cdot;\Theta)|\Theta \in \mathbb{R}^m\}$, find $f \in \mathcal{H}$ such that $f(x_i) = y_i$ for $i = 1, \cdots, n$.

Example 3 (well understood)—**linear spline**

$\mathcal{D}$

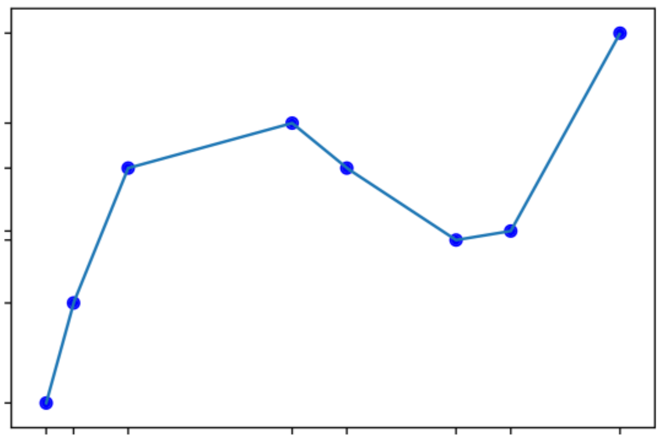
$$\{(x_i \in \mathbb{R}, y_i \in \mathbb{R})\}_{i=1}^{n}$$

$\mathcal{H}$

piecewise linear functions

find

explicit solution

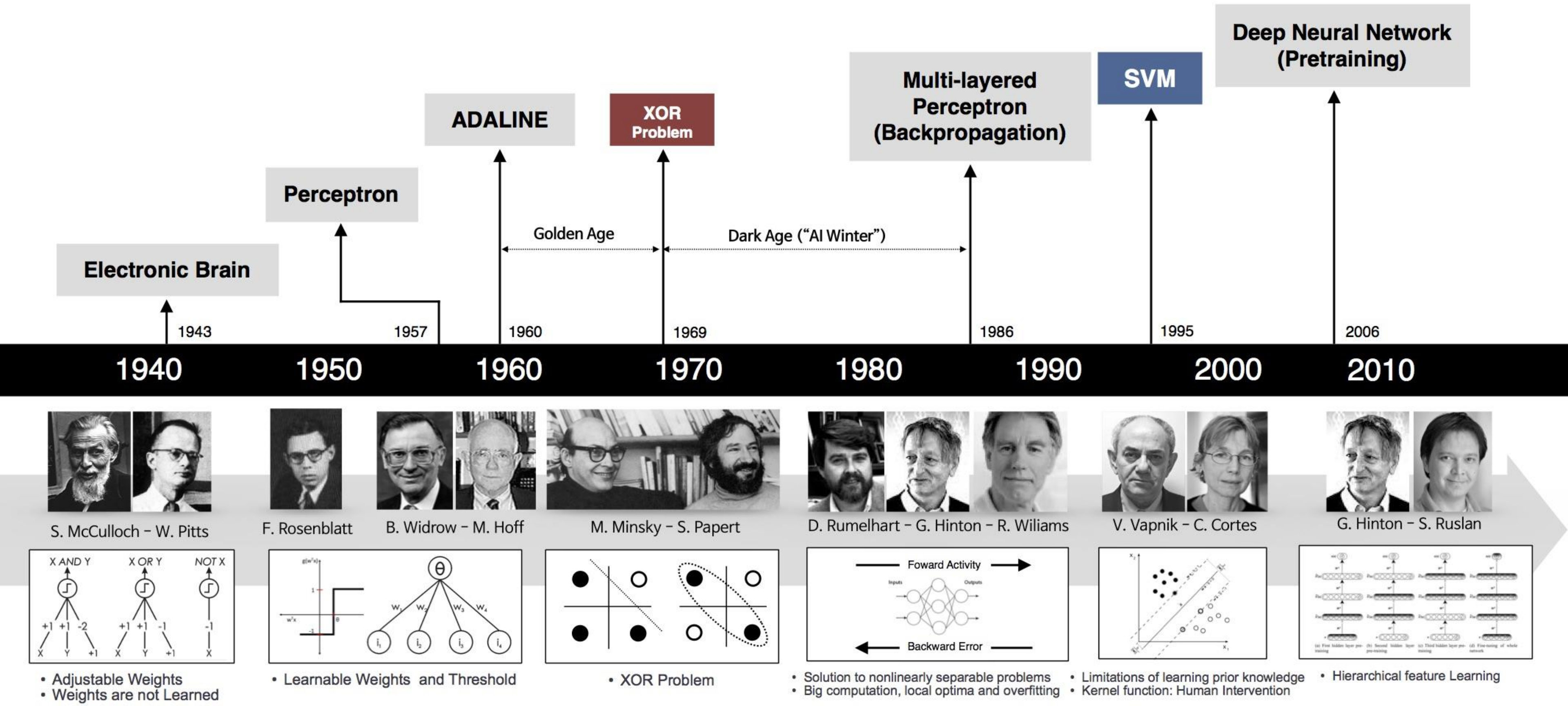


Why deep learning is a mystery?

Given $\mathcal{D}$: $\{(x_i, y_i)\}_{i=1}^n$ and $\mathcal{H}$: $\{f(\cdot;\Theta)|\Theta \in \mathbb{R}^m\}$, find $f \in \mathcal{H}$ such that $f(x_i) = y_i$ for $i = 1, \cdots, n$.

	Deep learning (black box!)	**Conventional methods**
$\mathcal{D}$	High dimensional real data (e.g., d=32*32*3)	Low dimensional data ($d \leq 3$)
$\mathcal{H}$	Deep neural network (#para>>#data)	Spanned by simple basis functions (#para≤#data)
find	Gradient-based method with proper initialization	explicit formula

Is deep learning alchemy?



https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html

Golden ages of neural network

1960-1969

- Simple (#data small)
- Single-layer NN (cannot solve XOR)
- Non-Gradient based (nondiff activation)

1984-1996

- Moderate (e.g., MNIST)
- Multi-layer NN (universal approx)
- Gradient based (BP)

2010-now

- Complex real data (e.g., ImageNet)
- Deep NN
- Gradient based (BP) with good initialization

NN is still a black box!

Leo Breiman 1995

1. **Why don’t heavily parameterized neural networks overfit the data?**

2. What is the effective number of parameters?

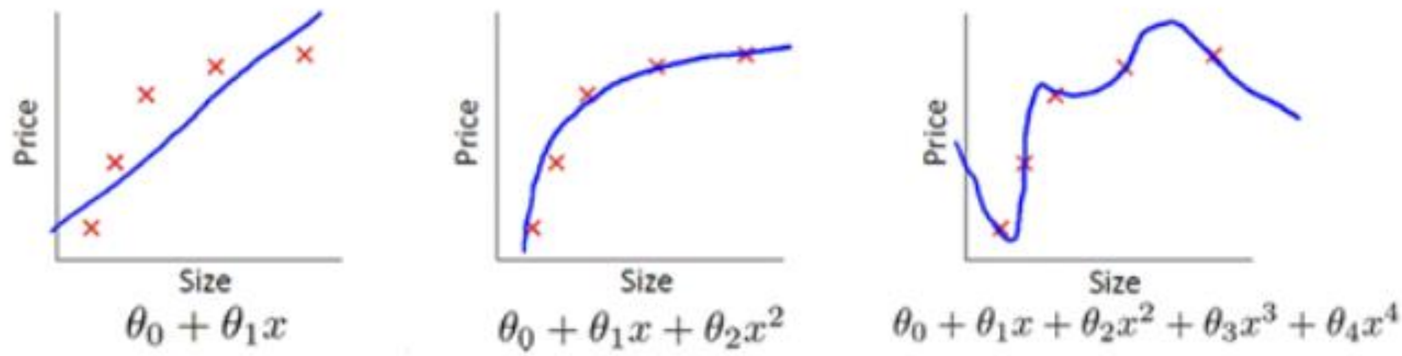
3. Why doesn’t backpropagation head for a poor local minima?

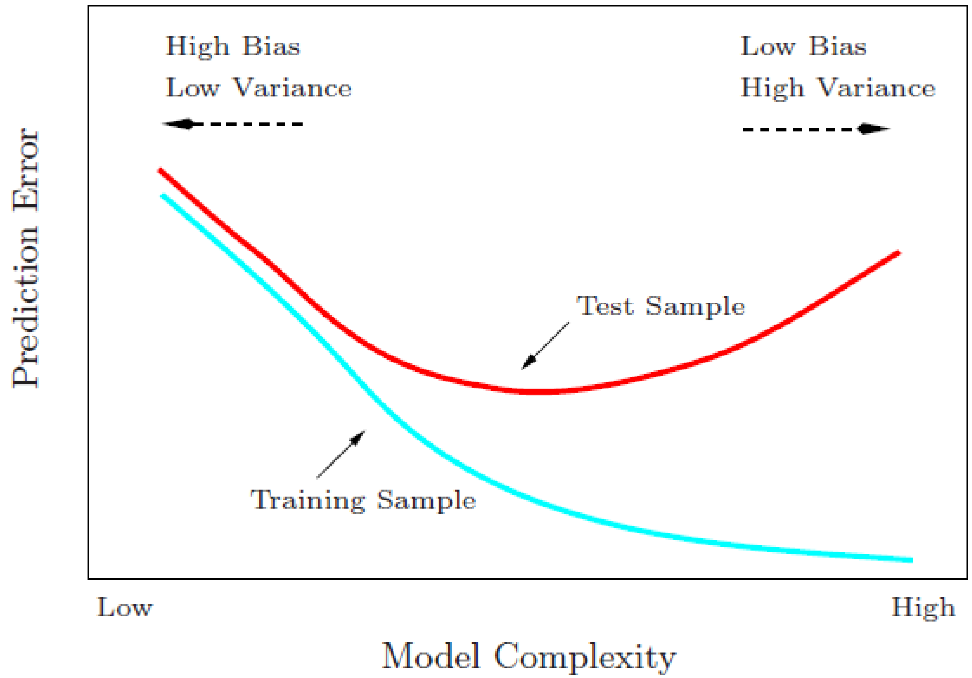
4. When should one stop the backpropagation and use the current parameters?

Leo Breiman Reflections After Refereeing Papers for NIPS

Frequency Principle

Conventional view of generalization

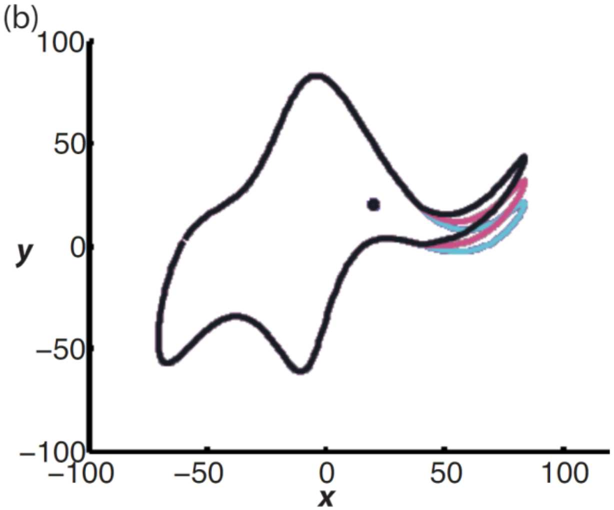




Conventional view of generalization

"With four parameters you can fit an elephant to a curve; with five you can make him wiggle his trunk."

-- John von Neumann



Mayer et al., 2010

A model that can fit anything likely overfits the data.

airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks

UNDERSTANDING DEEP LEARNING REQUIRES RE-THINKING GENERALIZATION

Chiyuan Zhang*
Massachusetts Institute of Technology
chiyuan@mit.edu

Samy Bengio
Google Brain
bengio@google.com

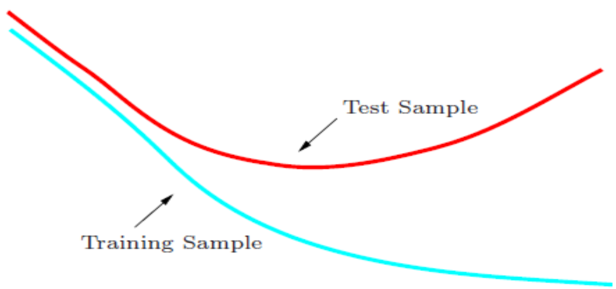
Moritz Hardt
Google Brain
mrtz@google.com

Benjamin Recht†
University of California, Berkeley
brecht@berkeley.edu

Oriol Vinyals
Google DeepMind
vinyals@google.com

Cifar10: 60,000 training data

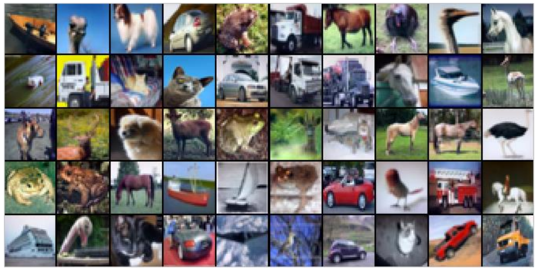
model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
(fitting random labels)		no	no	100.0	9.78

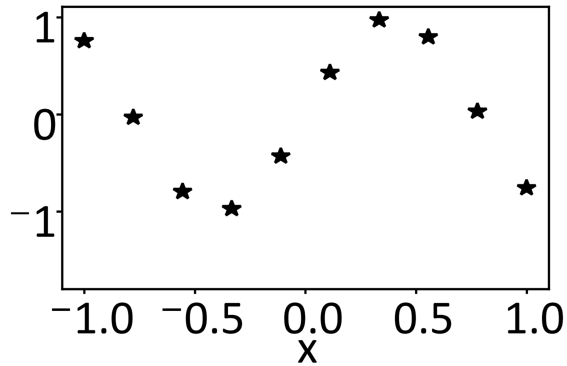


Overparameterized DNNs often generalize well.

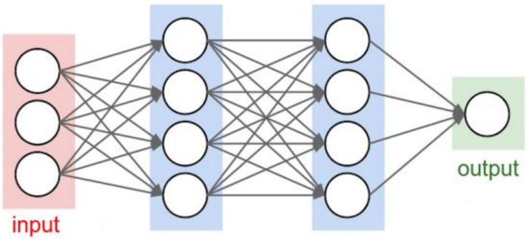
★ Problem simplification

$\mathcal{D}$





$\mathcal{H}$



$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{W}^{[L]}\sigma \circ (\cdots \boldsymbol{W}^{[2]}\sigma \circ (\boldsymbol{W}^{[1]}\boldsymbol{x} + \boldsymbol{b}^{[1]}) + \cdots) + \boldsymbol{b}^{[L]}$$

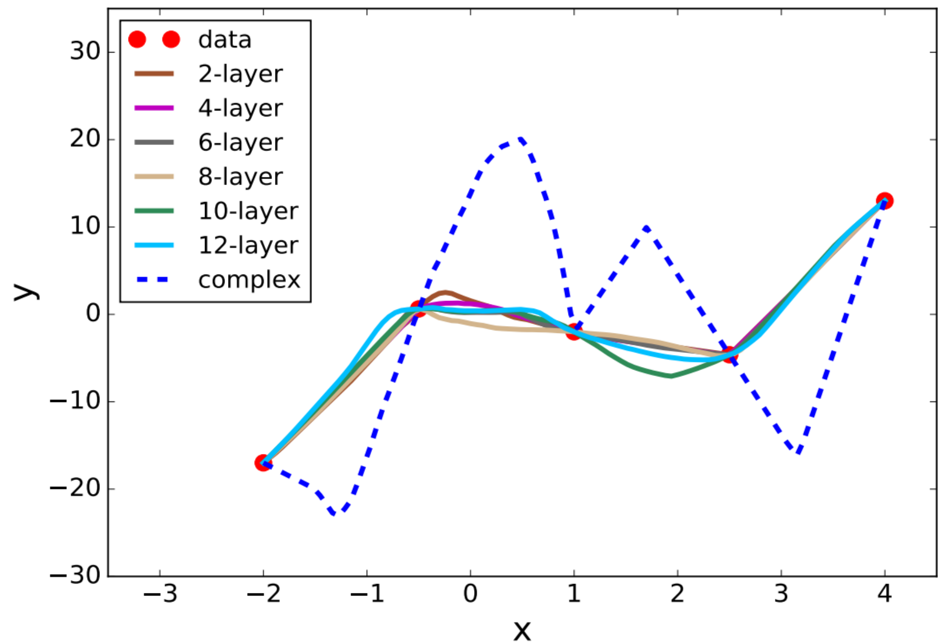
only observe

$$f(x,t) := f(x;\Theta(t))$$

find $\dot{\Theta} = -\nabla_{\Theta} L(\Theta)$

Initialized by special Θ_0

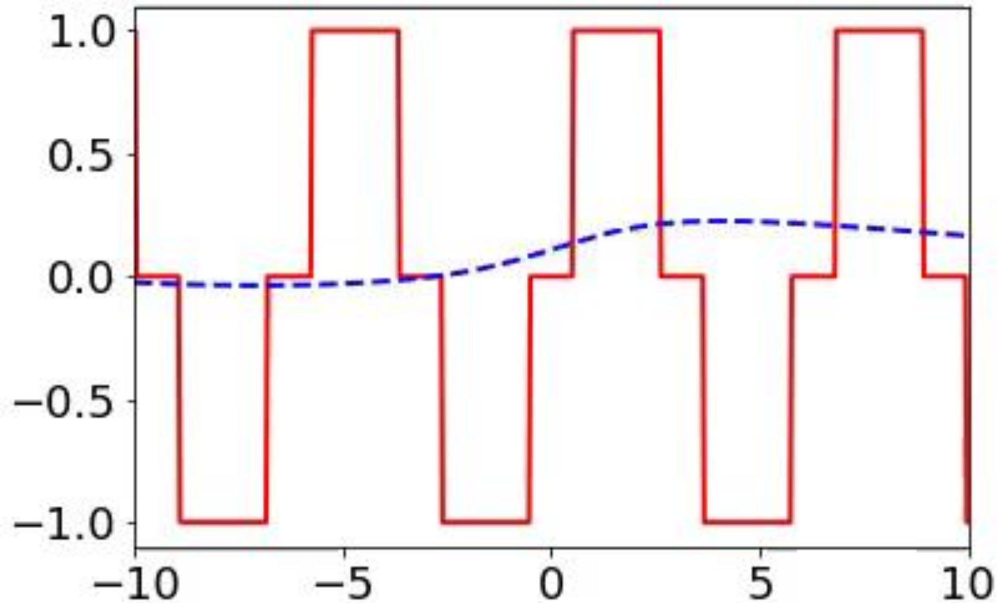
Overparameterized DNNs still generalize well



Lei Wu, Zhanxing Zhu, Weinan E, 2017

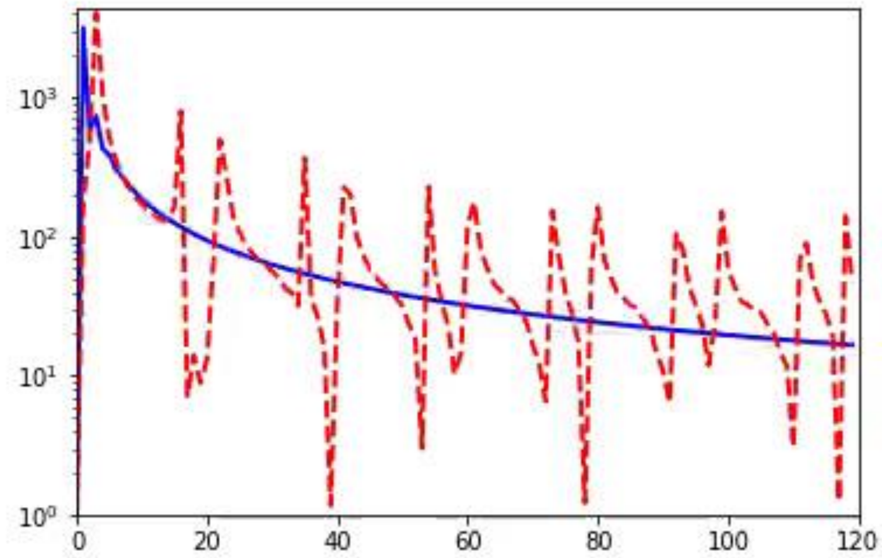
#para(~1000)>>#data: 5

evolution of $f(x,t)$



tanh-DNN, 200-100-100-50

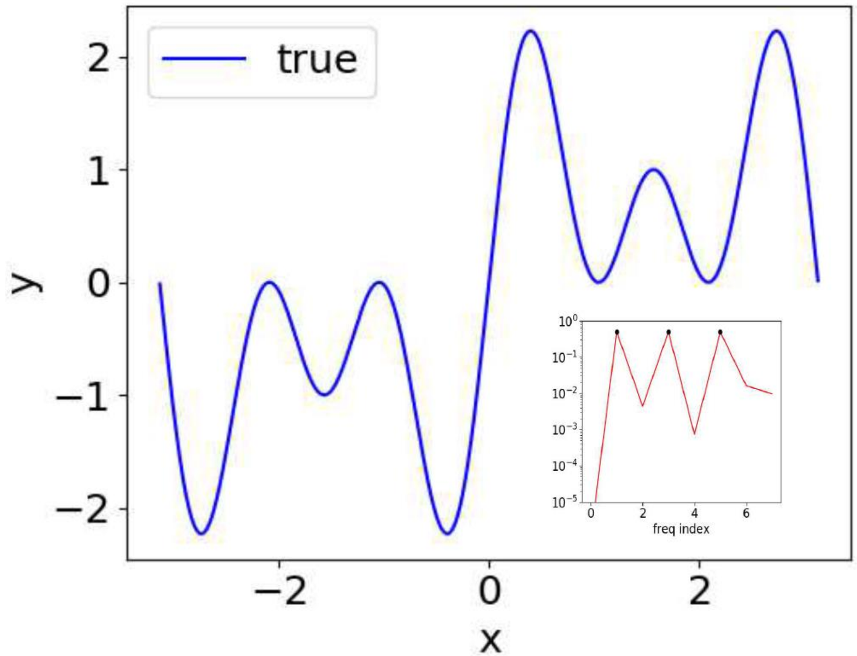
Through the lens of Fourier transform $\widehat{h}(\xi, t)$

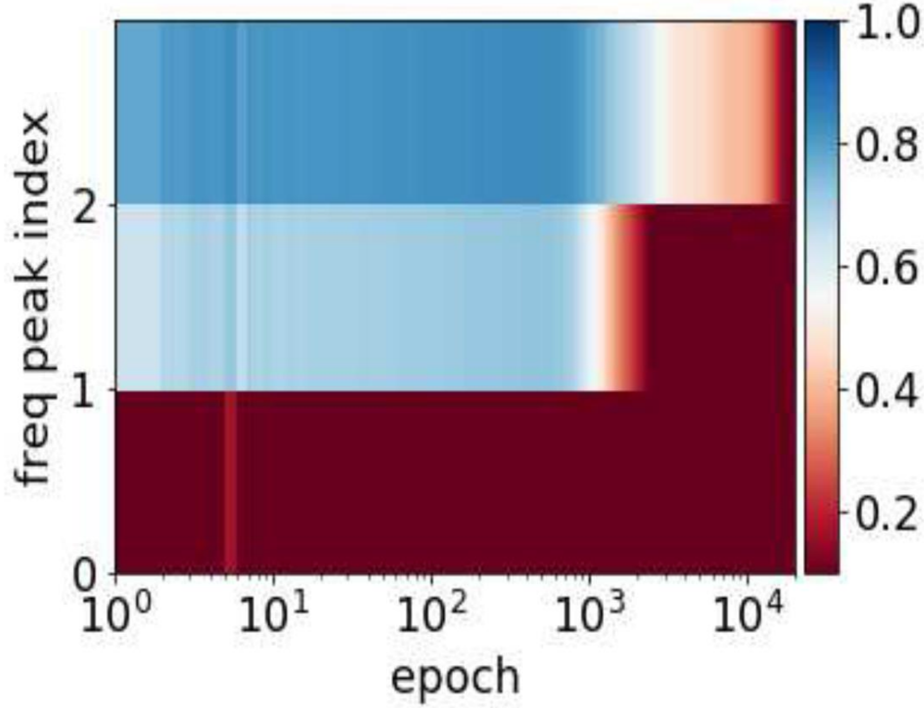


Frequency Principle (F-Principle):
DNNs often fit target functions from low to high frequencies during the training.

Xu, Zhang, Xiao, *Training behavior of deep neural network in frequency domain,* 2018

Synthetic curve with equal amplitude





(a) True image

How DNN fits a 2-d image?

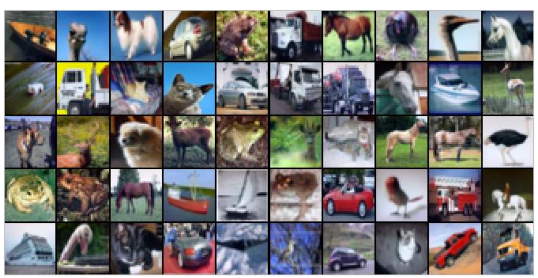
Target: image $I(\mathbf{x})$: $\mathbb{R}^2 \rightarrow \mathbb{R}$

$\mathbf{x}$: location of a pixel

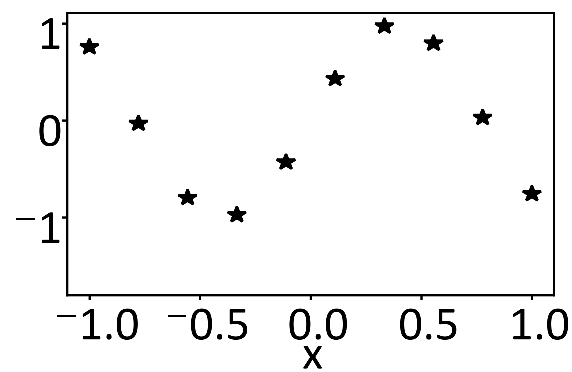
$I(\mathbf{x})$: grayscale pixel value

(b) DNN output

High-dimensional real data?



?



Xu, **Zhang**, Luo, Xiao, Ma, *Frequency Principle: Fourier Analysis Sheds Light on Deep Neural Networks, 2019*

Frequency

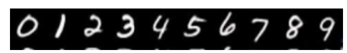
Image frequency (NOT USED)

- This frequency corresponds to the rate of change of intensity across neighboring pixels.

Response frequency

- Frequency of a general Input-Output mapping f.

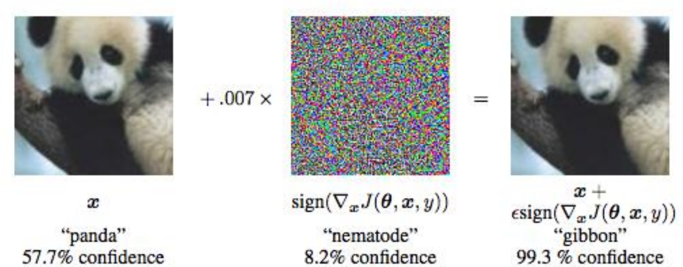
$$\hat{f}(\mathbf{k}) = \int f(\mathbf{x}) \mathrm{e}^{-\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{x}} \, \mathrm{d}\mathbf{x}$$



MNIST: $\mathbb{R}^{784} \to \mathbb{R}^{10}$, $\mathbf{k} \in \mathbb{R}^{784}$

Zero freq
Same color

high freq
Sharp edge



Goodfellow et al.

high freq
Adversarial example

Examining F-Principle for high dimensional real problems

Nonuniform Discrete Fourier transform (NUDFT) for training dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$:

$$\hat{y}_{\mathbf{k}} = \frac{1}{n}\sum_{i=1}^{n} y_i \mathrm{e}^{-\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{x}_i}, \quad \hat{h}_{\mathbf{k}}(t) = \frac{1}{n}\sum_{i=1}^{n} h(\mathbf{x}_i, t)\mathrm{e}^{-\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{x}_i}$$

Difficulty:

- Curse of dimensionality, i.e., $\#\mathbf{k}$ grows exponentially with dimension of problem d.

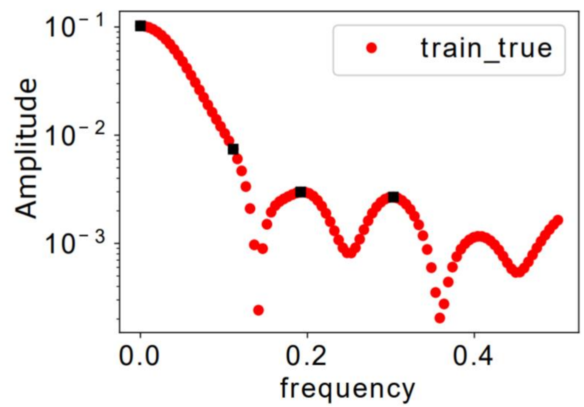
Our approaches:

- **Projection**, i.e., choose $\mathbf{k} = k\mathbf{p}_1$
- **Filtering**

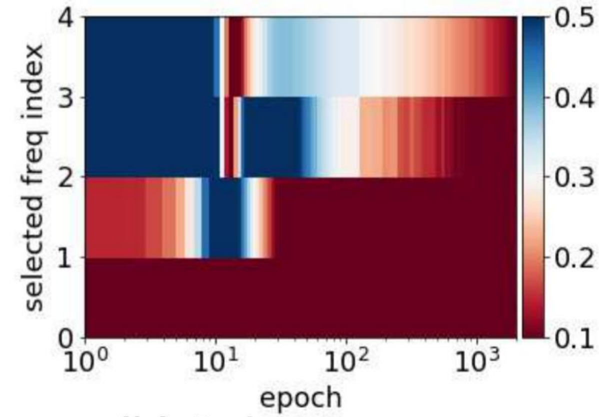
Projection approach

Relative error: $\Delta_F(k) = \left|\hat{h}_k - \hat{y}_k\right| / |\hat{y}_k|$

MNIST

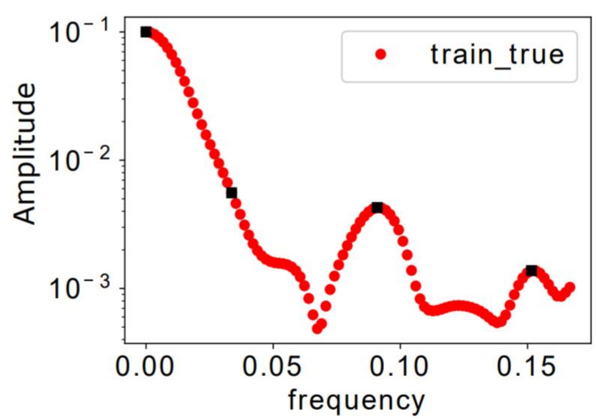


(a) Fourier domain

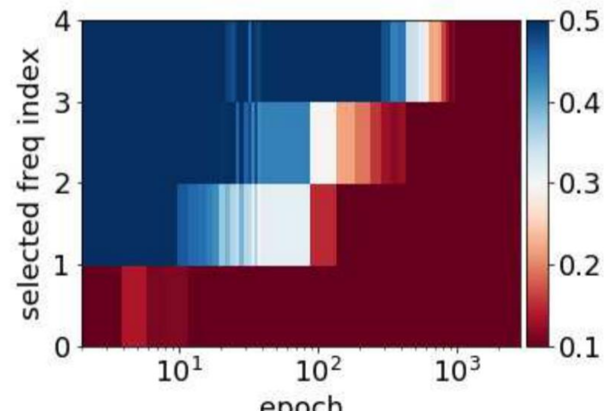


(b) Relative error

CIFAR10

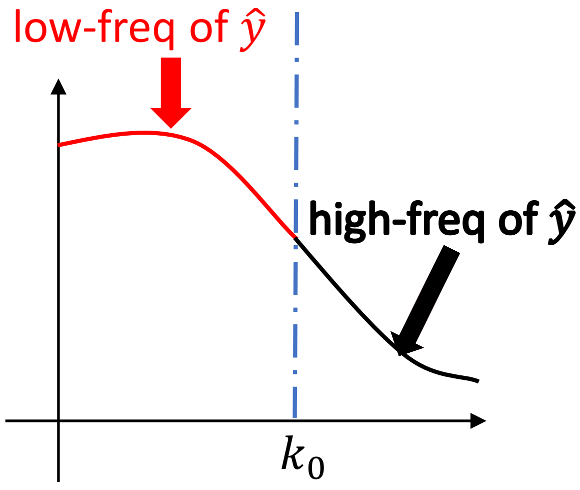


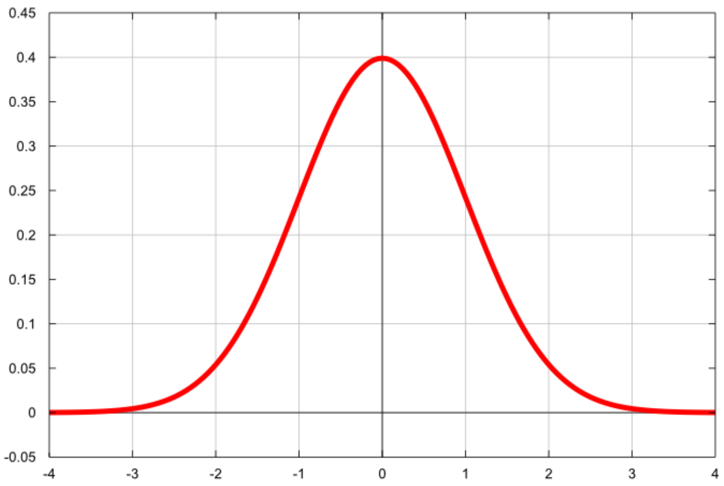
(c) Fourier domain



(d) Relative error

Decompose frequency domain by filtering





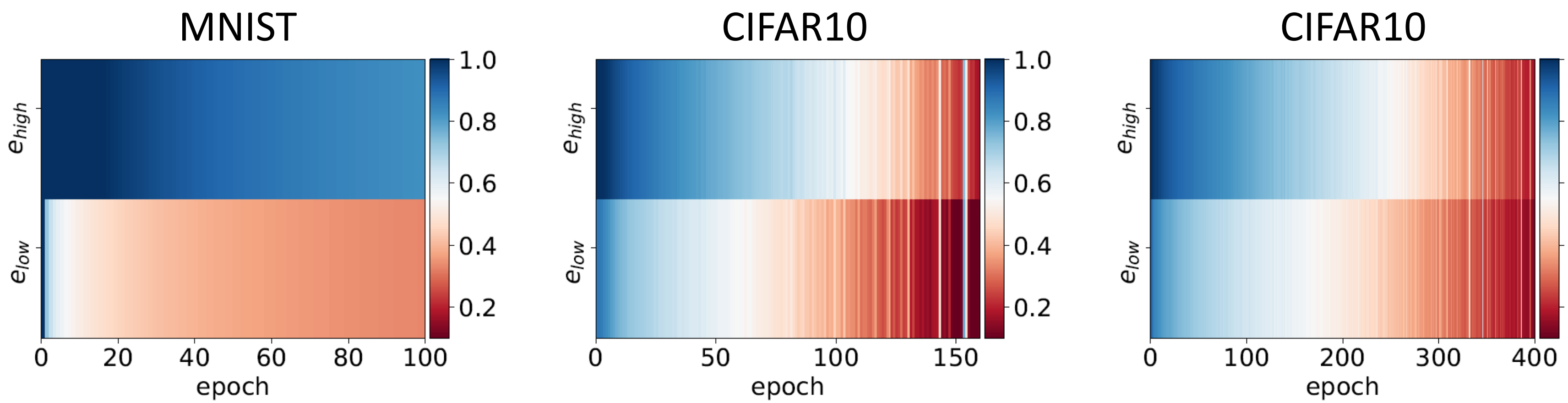
$$\mathbf{y}_i^{\mathrm{low},\delta} = (\mathbf{y} * G^{\delta})_i$$

$$\mathbf{y}_i^{\mathrm{high},\delta} \triangleq \mathbf{y}_i - \mathbf{y}_i^{\mathrm{low},\delta}$$

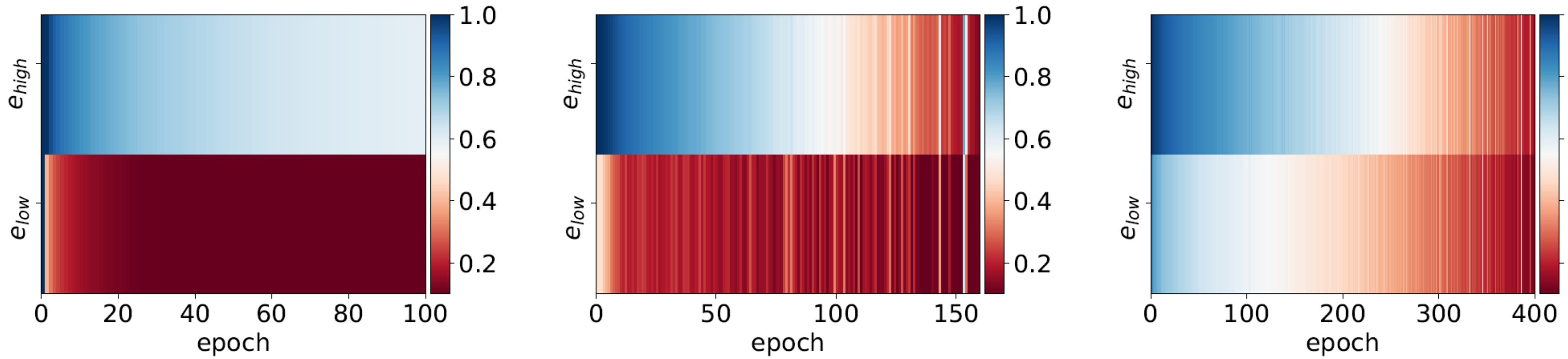
$$e_{\mathrm{low}} = \left(\frac{\sum_i |\mathbf{y}_i^{\mathrm{low},\delta} - \mathbf{h}_i^{\mathrm{low},\delta}|^2}{\sum_i |\mathbf{y}_i^{\mathrm{low},\delta}|^2} \right)^{\frac{1}{2}}$$

$$e_{\mathrm{high}} = \left(\frac{\sum_i |\mathbf{y}_i^{\mathrm{high},\delta} - \mathbf{h}_i^{\mathrm{high},\delta}|^2}{\sum_i |\mathbf{y}_i^{\mathrm{high},\delta}|^2} \right)^{\frac{1}{2}}$$

F-Principle in high-dim space



(a) $\delta = 3$, DNN (b) $\delta = 3$, CNN (c) $\delta = 7$, VGG



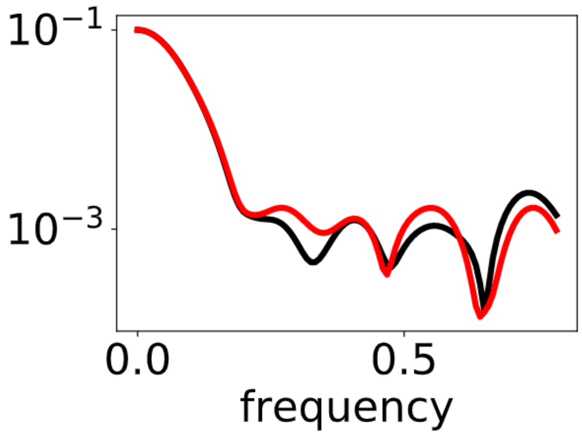
(d) $\delta = 7$, DNN (e) $\delta = 7$, CNN (f) $\delta = 10$, VGG

Implication of F-Principle

Xu, Zhang, Xiao*, Training behavior of deep neural network in frequency domain, 2018*
Xu, Zhang, Luo, Xiao, Ma*, Frequency Principle: Fourier Analysis Sheds Light on Deep Neural Networks, 2019*

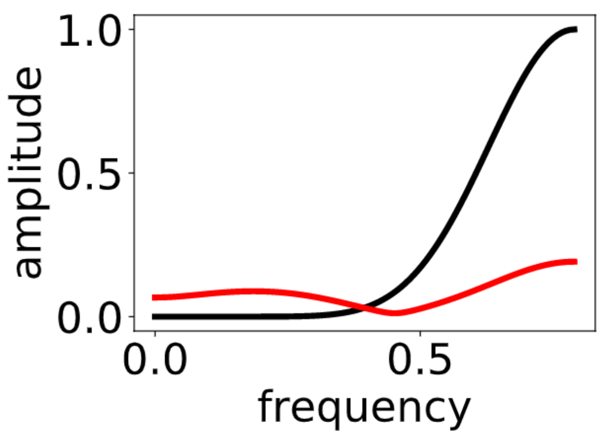
Why don't heavily parameterized neural networks overfit the data?

F-Principe: DNN prefers low frequencies



CIFAR10

Test accuracy: 72% %>>10%



parity

For $\vec{x} \in \{-1,1\}^n$

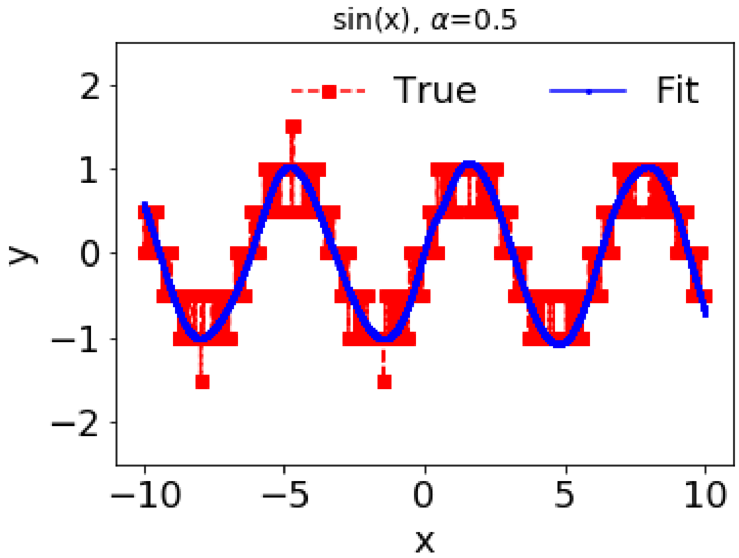
$\boldsymbol{f(\vec{x}) = \prod_{j=1}^{n} x_j}$,

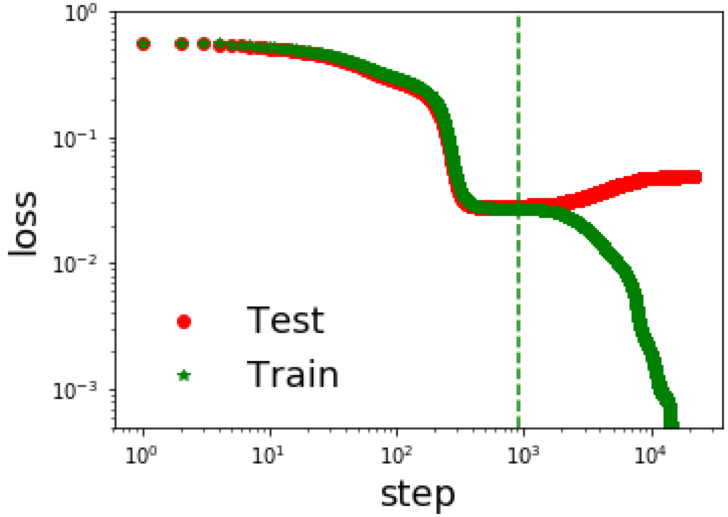
Even #'-1' → 1;

Odd #'-1' → -1.

Test accuracy: ~50%, random guess

When should one stop the backpropagation and use the current parameters?





Studies elicited by F-Principle

- **Theoretical study**
  - Rahaman, N., Arpit, D., Baratin, A., Draxler, F., Lin, M., Hamprecht, F. A., Bengio, Y. & Courville, A. (2018), 'On the spectral bias of deep neural networks'.
  - Jin, P., Lu, L., Tang, Y. & Karniadakis, G. E. (2019), 'Quantifying the generalization error in deep learning in terms of data distribution and neural network smoothness'
  - Basri, R., Jacobs, D., Kasten, Y. & Kritchman, S. (2019), 'The convergence rate of neural networks for learned functions of different frequencies'
  - Zhen, H.-L., Lin, X., Tang, A. Z., Li, Z., Zhang, Q. & Kwong, S. (2018), 'Nonlinear collaborative scheme for deep neural networks'
  - Wang, H., Wu, X., Yin, P. & Xing, E. P. (2019), 'High frequency component helps explain the generalization of convolutional neural networks'
- **Empirical study**
  - Jagtap, A. D. & Karniadakis, G. E. (2019), 'Adaptive activation functions accelerate convergence in deep and physics-informed neural networks'
  - Stamatescu, V. & McDonnell, M. D. (2018), 'Diagnosing convolutional neural networks using their spectral response'
  - Rabinowitz, N. C. (2019), 'Meta-learners' learning dynamics are unlike learners'',
- **Application**
  - Wang, F., Müller, J., Eljarrat, A., Henninen, T., Rolf, E. & Koch, C. (2018), 'Solving inverse problems with multi-scale deep convolutional neural networks'
  - Cai, W., Li, X. & Liu, L. (2019), 'Phasednn-a parallel phase shift deep neural network for adaptive wideband learning'

Quantitative theory for F-Principle

Zhang, Xu, Luo, Ma, *Explicitizing an Implicit Bias of the Frequency Principle in Two-layer Neural Networks*, 2019

The NTK regime

$$L(\Theta) = \sum_{i=1}^{n} (h(x_i; \Theta) - y_i)^2$$

$$\dot{\Theta} = -\nabla_{\Theta} L(\Theta)$$

- $\partial_t h(x; \Theta) = -\sum_{i=1}^{n} K_{\Theta}(x, x_i)(h(x_i; \Theta) - y_i)$

Where $K_{\Theta}(x, x') = \nabla_{\Theta} h(x; \Theta) \cdot \nabla_{\Theta} h(x'; \Theta)$

- Neural Tangent Kernel (NTK) regime:

$K_{\Theta(\mathrm{t})}(x, x') \approx K_{\Theta(0)}(x, x')$ for any t.

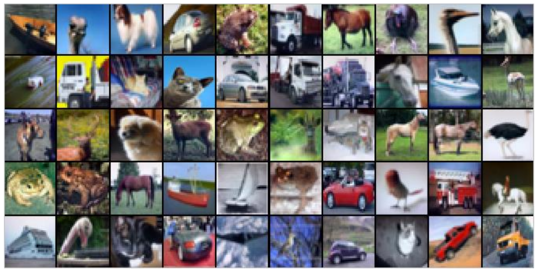
Theorem 1. *For a network of depth L at initialization, with a Lipschitz nonlinearity σ, and in the limit as the layers width $n_1, \ldots, n_{L-1} \to \infty$ sequentially, the NTK $\Theta^{(L)}$ converges in probability to a deterministic limiting kernel:*

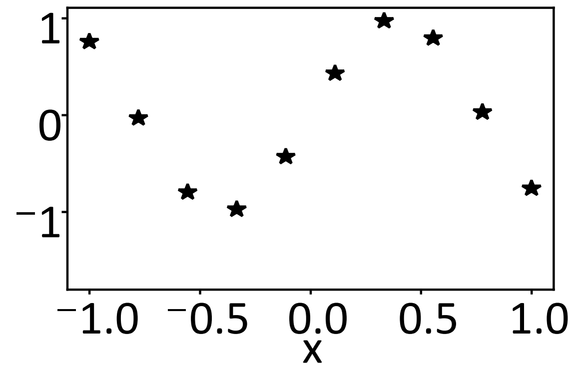
$$\Theta^{(L)} \to \Theta_{\infty}^{(L)} \otimes Id_{n_L}.$$

Jacot et al., 2018

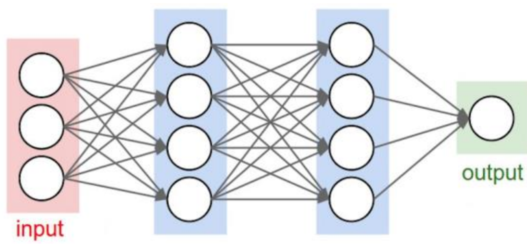
★ Problem simplification

$\mathcal{D}$





$\mathcal{H}$



$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{W}^{[L]}\sigma \circ (\cdots \boldsymbol{W}^{[2]}\sigma \circ (\boldsymbol{W}^{[1]}\boldsymbol{x} + \boldsymbol{b}^{[1]}) + \cdots) + \boldsymbol{b}^{[L]}$$

Two-layer ReLU NN

$$h(x;\Theta) = \sum_{i=1}^{n} w_i \sigma\left(r_i(x + l_i)\right)$$

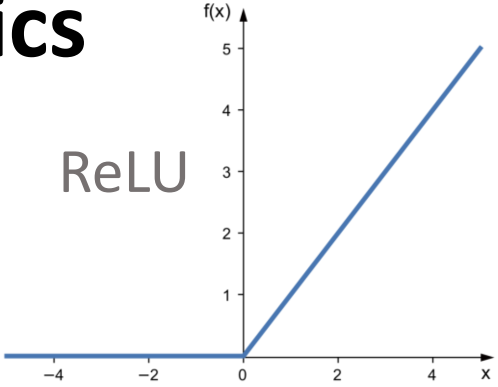
find $\dot{\Theta} = -\nabla_{\Theta} L(\Theta)$

Initialized by special Θ_0

Kernel gradient flow

$$\partial_t f(x,t) = -\Sigma_{i=1}^{n} K_{\Theta_0}(x, x_i)(f(x_i,t) - y_i)$$

Linear F-Principle (LFP) dynamics



2-layer NN: $h(x;\Theta) = \sum_{i=1}^{n} w_i \mathrm{ReLU}(r_i(x + l_i))$

Assumptions:
(i) NTK regime, (ii) sufficiently wide distribution of l_i.

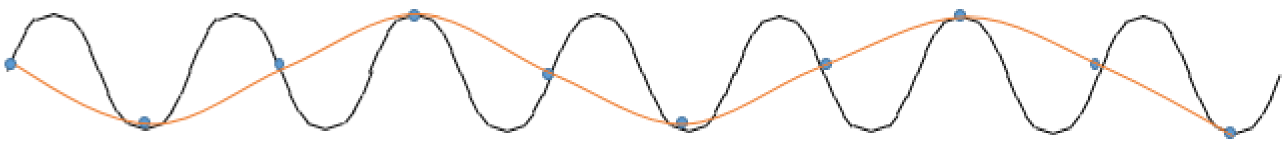
$$\boxed{\partial_t \hat{h}(\xi,t) = -\left[\frac{4\pi^2\langle r^2 w^2\rangle}{\xi^2} + \frac{\langle r^2\rangle + \langle w^2\rangle}{\xi^4}\right]\left(\widehat{h_p}(\xi,t) - \widehat{f_p}(\xi,t)\right)}$$

$\langle\cdot\rangle$: mean over all neurons at initialization

f: target function; $(\cdot)_p = (\cdot)p$,where $p(x) = \frac{1}{n}\sum_{i=1}^{n}\delta(x - x_i)$;

$\hat{\cdot}$: Fourier transform; ξ: frequency

aliasing



Preference induced by LFP dynamics

$$\partial_t \hat{h}(\xi,t) = -\left[\frac{4\pi^2\langle r^2 w^2\rangle}{\xi^2} + \frac{\langle r^2\rangle + \langle w^2\rangle}{\xi^4}\right]\left(\widehat{h_p}(\xi,t) - \hat{f_p}(\xi,t)\right)$$

$$\Updownarrow$$

low frequency preference

$$\boxed{\min_{h\in F_\gamma}\int\left[\frac{4\pi^2\langle r^2 w^2\rangle}{\xi^2} + \frac{\langle r^2\rangle+\langle w^2\rangle}{\xi^4}\right]^{-1}\left|\hat{h}(\xi)\right|^2 \mathrm{d}\xi}$$

s.t. $h(x_i) = y_i$ for $i = 1,\cdots,n$

Case 1: ξ^{-2} dominant

- $\min\int \xi^2\left|\hat{h}(\xi)\right|^2 \mathrm{d}\xi \sim \min\int|h'(x)|^2\,\mathrm{d}\xi \to$ **linear spline**

Case 2: ξ^{-4} dominant

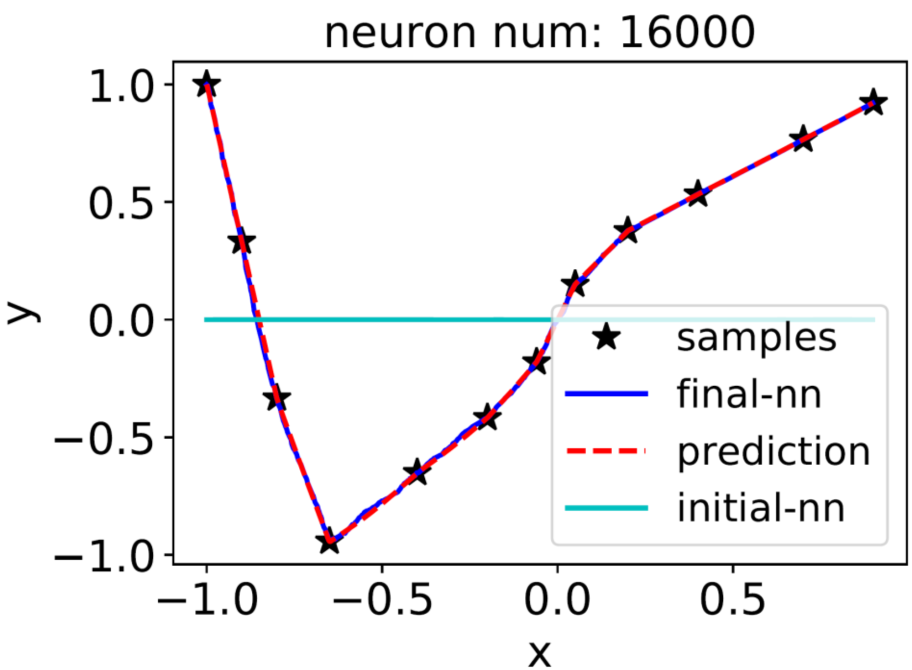
- $\min\int \xi^4\left|\hat{h}(\xi)\right|^2 \mathrm{d}\xi \sim \min\int|h''(x)|^2\,\mathrm{d}\xi \to$ **cubic spline**

Regularity can be changed through initialization

Case 1

$$\langle r^2\rangle + \langle w^2\rangle \gg 4\pi^2\langle r^2 w^2\rangle$$

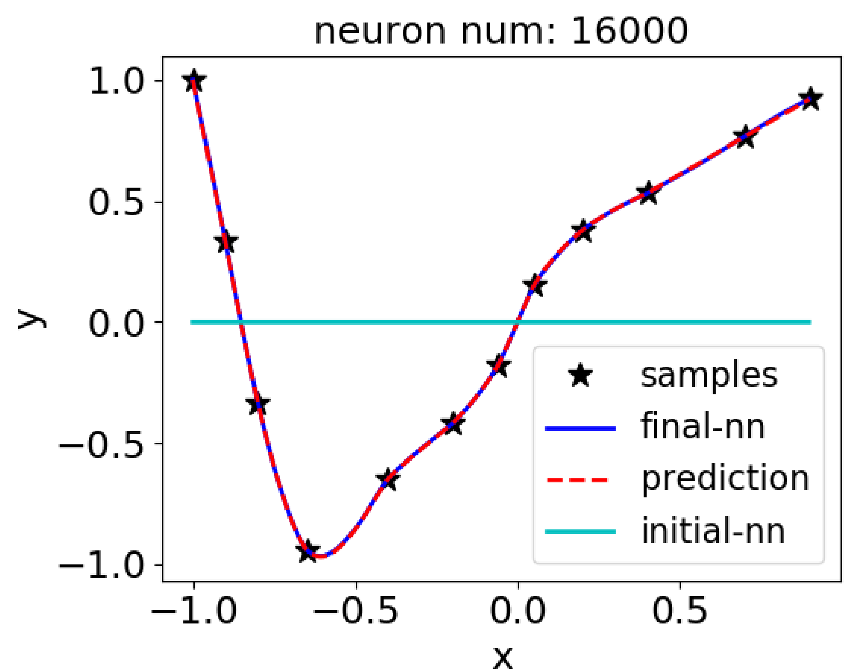
$$\min \int \xi^2 \left|\hat{h}(\xi)\right|^2 \mathrm{d}\xi$$



Case 2

$$4\pi^2\langle r^2 w^2\rangle \gg \langle r^2\rangle + \langle w^2\rangle$$

$$\min \int \xi^4 \left|\hat{h}(\xi)\right|^2 \mathrm{d}\xi$$



High-dimensional Case

$$\partial_t \hat{h}(\xi,t) = -\left[\frac{\langle |r|^2\rangle + \langle w^2\rangle}{|\xi|^{d+3}} + \frac{4\pi^2\langle |r|^2 w^2\rangle}{|\xi|^{d+1}}\right]\left(\widehat{h_p}(\xi,t) - \widehat{f_p}(\xi,t)\right)$$

where f: target function; $(\cdot)_p = (\cdot)p$,where $p(x) = \frac{1}{n}\sum_{i=1}^{n}\delta(x - x_i)$; $\widehat{(\cdot)}$: Fourier transform; ξ: frequency.

Theorem (informal). Solution of LFP dynamics at $t \to \infty$ with initial value h_{ini} is the same as solution of the following optimization problem

$$\min_{h - h_{\text{ini}} \in F_\gamma} \int \left[\frac{\langle |r|^2\rangle + \langle w^2\rangle}{|\xi|^{d+3}} + \frac{4\pi^2\langle |r|^2 w^2\rangle}{|\xi|^{d+1}}\right]^{-1} \left|\hat{h}(\xi) - \hat{h}_{\text{ini}}(\xi)\right|^2 \mathrm{d}\xi$$

$$\text{s.t. } h(X) = Y.$$

FP-norm and FP-space

We define the FP-norm for all function $h \in L^2(\Omega)$:

$$\|h\|_\gamma := \|\hat{h}\|_{H_\Gamma} = \left(\sum_{k \in \mathbb{Z}^{d*}} \gamma^{-2}(k) \left|\hat{h}(k)\right|^2\right)^{1/2}$$

Next, we define the FP-space:

$$F_\gamma(\Omega) = \{h \in L^2(\Omega) : \|h\|_\gamma < \infty\}$$

A priori generalization error bound

Theorem (informal). Suppose that the real-valued target function $f \in F_\gamma(\Omega)$, h_n is the solution of the regularized model

$$\min_{h \in F_\gamma} \|h\|_\gamma \text{ s.t. } h(X) = Y$$

Then for any $\delta \in (0,1)$ with probability at least $1 - \delta$ over the random training samples, the population risk has the bound

$$L(h_n) \leq \left(\|f\|_\infty + 2\|f\|_\gamma \|\gamma\|_{l^2}\right)\left(\frac{2}{\sqrt{n}} + 4\sqrt{\frac{2\log(4/\delta)}{n}}\right)$$

Leo Breiman 1995

1. **Why don’t heavily parameterized neural networks overfit the data?**

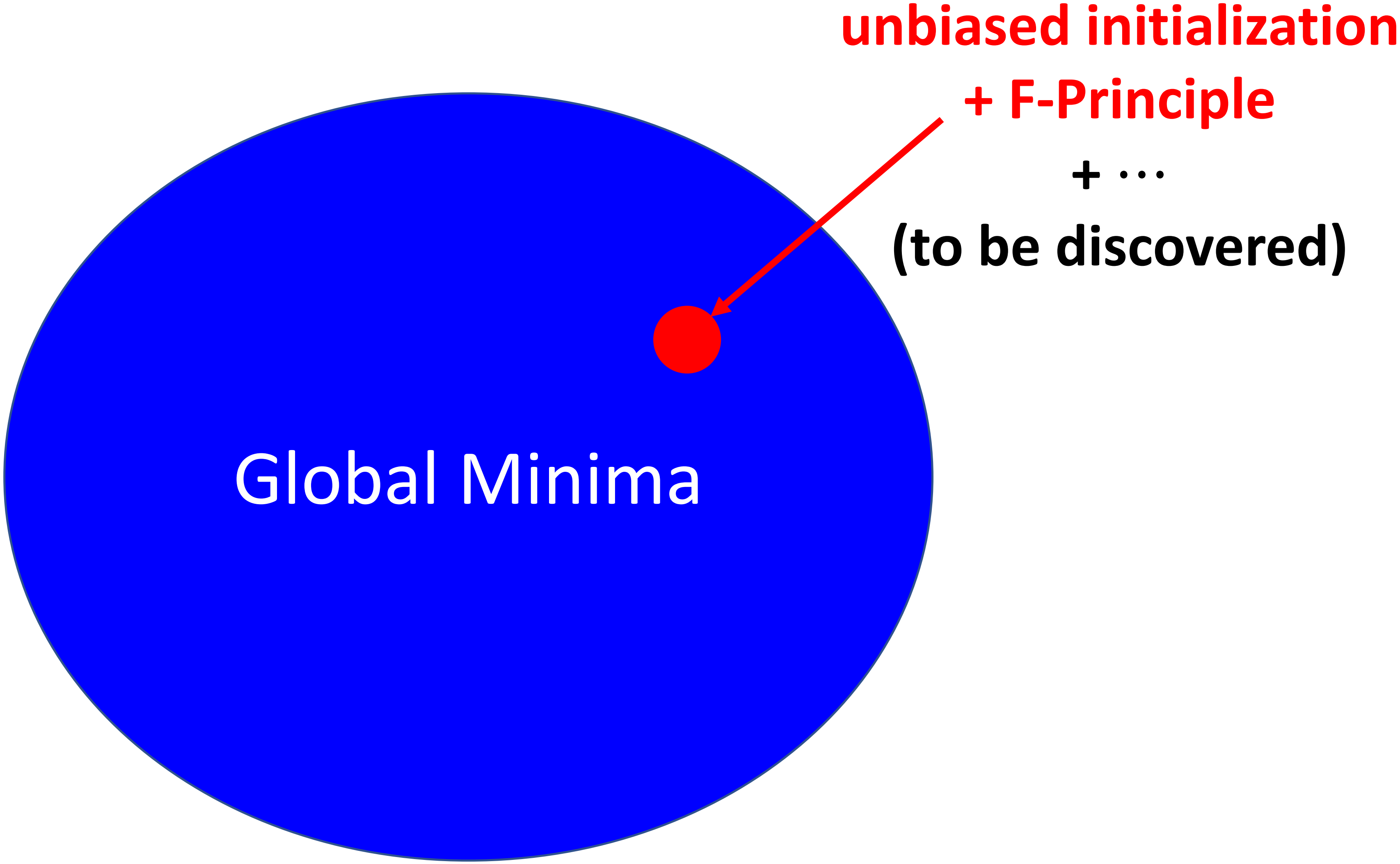
2. What is the effective number of parameters?

3. Why doesn’t backpropagation head for a poor local minima?

4. When should one stop the backpropagation and use the current parameters?

Leo Breiman Reflections After Refereeing Papers for NIPS

A picture for the generalization mystery of DNN



Conclusion

DNNs prefer low frequencies!

References:

- Xu, **Zhang**, Xiao*, **Training behavior of deep neural network in frequency domain**, 2018*
- Xu, **Zhang**, Luo, Xiao, Ma*, **Frequency Principle: Fourier Analysis Sheds Light on Deep Neural Networks**, 2019*
- **Zhang**, Xu, Luo, Ma*, **Explicitizing an Implicit Bias of the Frequency Principle in Two-layer Neural Networks**, 2019*
- **Zhang**, Xu, Luo, Ma*, **A type of generalization error induced by initialization in deep neural networks**, 2019*
- Luo, Ma, Xu, **Zhang***, **Theory on Frequency Principle in General Deep Neural Networks**, 2019.*