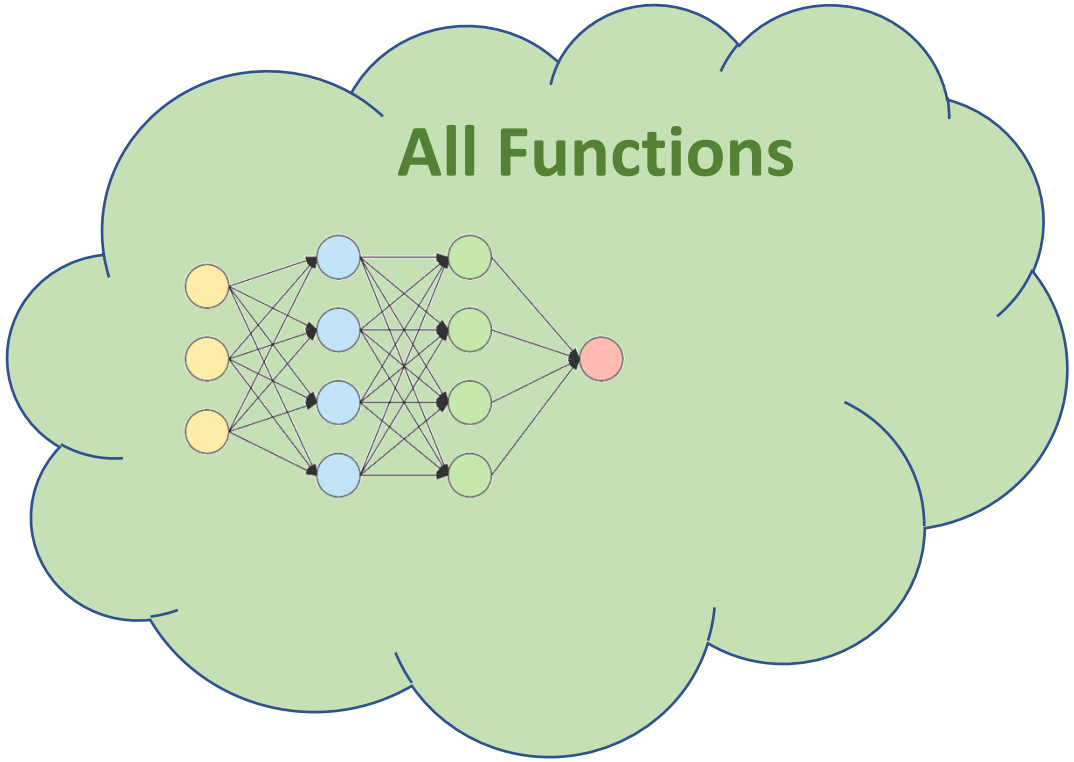
Different optimization algorithm

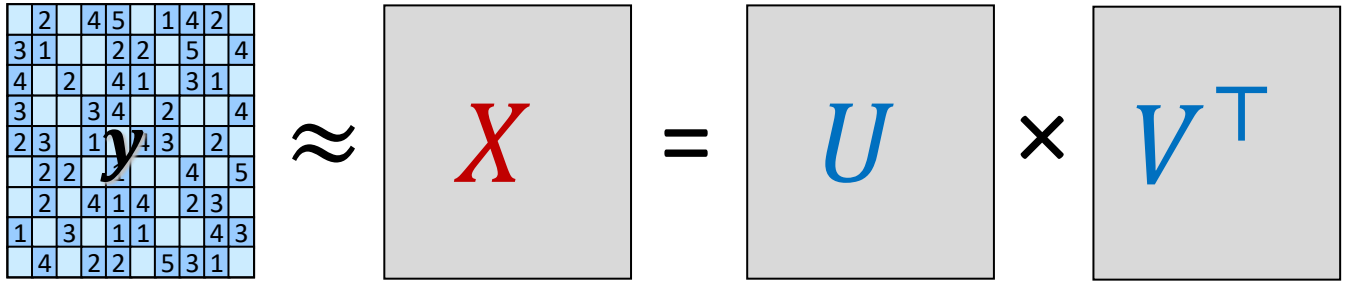
➔ Different bias in optimum reached

➔ Different Inductive bias

➔ Different generalization properties

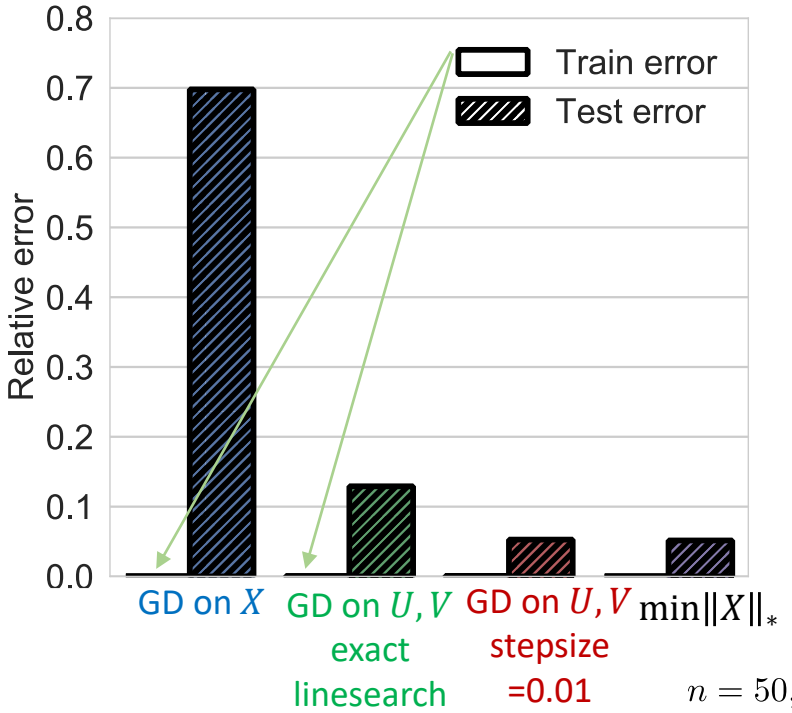


Need to understand optimization alg. not just as reaching ***some*** (global) optimum, but as reaching a ***specific*** optimum



$$\min_{X \in \mathbb{R}^{n \times n}} \|observed(X) - y\|_2^2$$

- Underdetermined non-sensical problem, lots of useless global min
- Since U, V full dim, no constraint on X, all the same non-sense global min



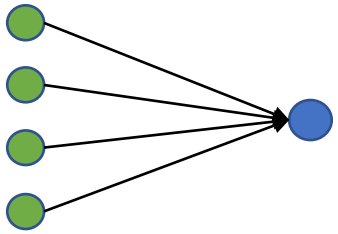
Grad Descent on $U, V \overset{???}{\rightarrow}$ **min $\|X\|_*$ solution**
(with inf. small stepsize and initialization)
→ good generalization if Y (aprox) low rank
[Gunasekar Woodworth Bhojanapalli Neyshabur S 2017]

When $y = \langle A_i, W^* \rangle$, W^* low rank, A_i RIP
[Yuanzhi Li, Hongyang Zhang and Tengyu Ma 2018]

Not always **min $\|X\|_*$** !
[Zhiyuan Li, Yuping Luo, Kaifeng Lyu ICLR 2021]

$n = 50,\ m = 300,\ A_i$ iid Gaussian, X^* rank-2 ground truth
$y = \mathcal{A}(X^*) + \mathcal{N}(0, 10^{-3}),\ y_{\text{test}} = \mathcal{A}_{\text{test}}(X^*) + \mathcal{N}(0, 10^{-3})$

Single Overparametrized Linear Unit



Train single unit with SGD using logistic ("cross entropy") loss

→ **Hard Margin SVM predictor**

$$w(\infty) \propto \arg\min \|w\|_2 \ \ s.t. \forall_i y_i \langle w, x_i \rangle \geq 1$$

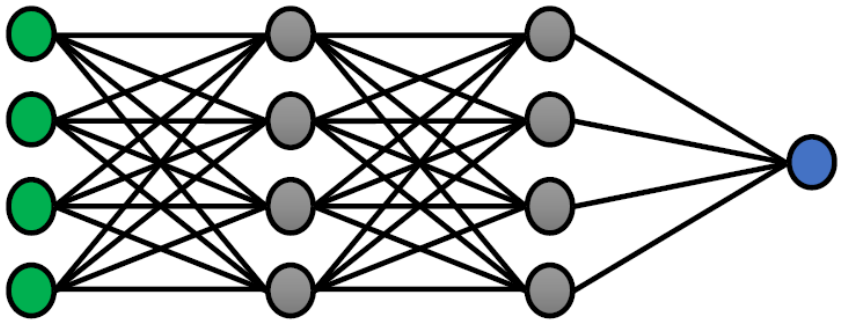
Even More Overparameterization: Deep Linear Networks

Network implements a linear mapping:

$$f_w(x) = \langle \beta_w, x \rangle$$

Training: same opt. problem as logistic regression:

$$\min_w \mathcal{L}(f_w) \equiv \min_\beta \mathcal{L}(x \mapsto \langle \beta, x \rangle)$$

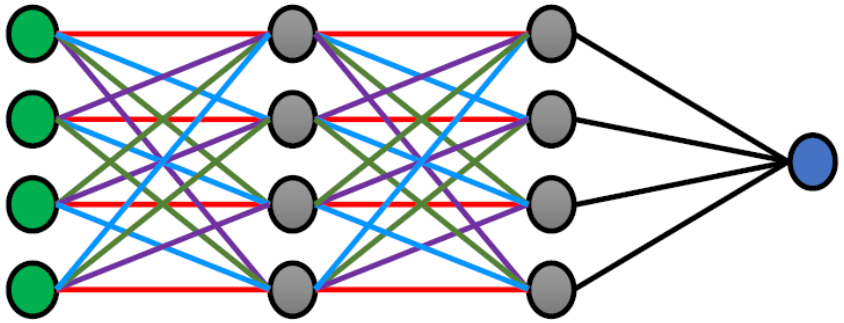


Train w with SGD

→ **Hard Margin SVM predictor**

$$\beta_{w(\infty)} \to \arg\min \|\beta\|_2 \ \ s.t. \forall_i y_i \langle \beta, x_i \rangle \geq 1$$

Linear Conv Nets



L-1 hidden layers, $h_l \in \mathbb{R}^n$, each with (one channel) full-width cyclic "convolution" $w_\ell \in \mathbb{R}^D$:

$$h_l[d] = \sum_{k=0}^{D-1} w_l[k] h_{l-1}[d + k \bmod D] \qquad h_{out} = \langle w_L, h_{L-1} \rangle$$

With single conv layer (L=2), training weights with SGD

→ $\mathbf{arg\,min} \|\boldsymbol{DFT}(\boldsymbol{\beta})\|_1 \; s.t. \forall_i y_i \langle \beta, x_i \rangle \geq 1$

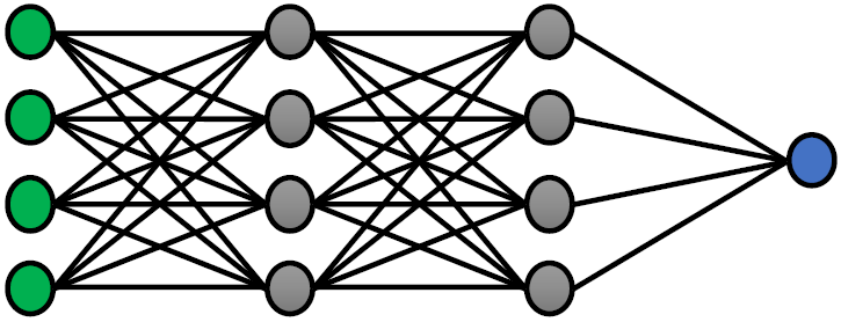
Discrete Fourier Transform

With multiple conv layers

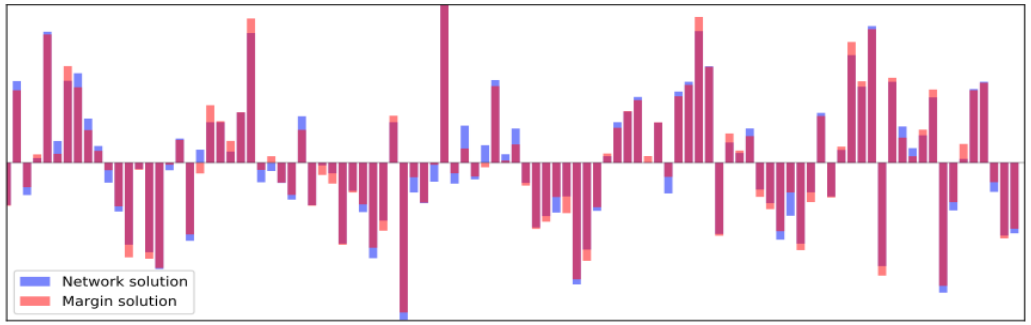
→ critical point of $\mathbf{min} \|\boldsymbol{DFT}(\boldsymbol{\beta})\|_{2/L} \; s.t. \forall_i y_i \langle \beta, x_i \rangle \geq 1$

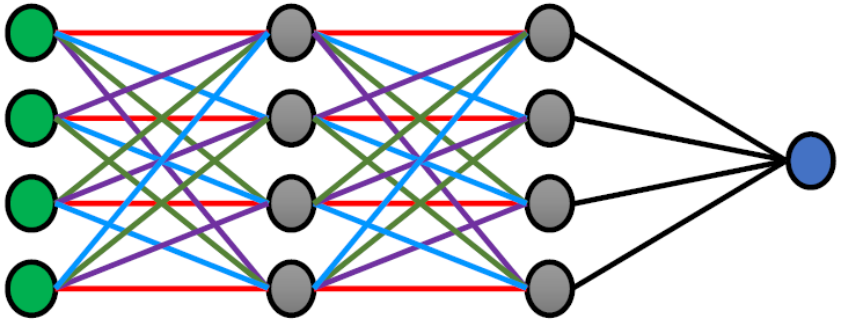
for $\ell(z) = \exp(-z)$, almost all linearly separable data sets and initializations $w(0)$ and any bounded stepsizes s.t. $\mathcal{L} \to 0$, and $\Delta w(t)$ converge in direction

[Gunasekar Lee Soudry S 2018]

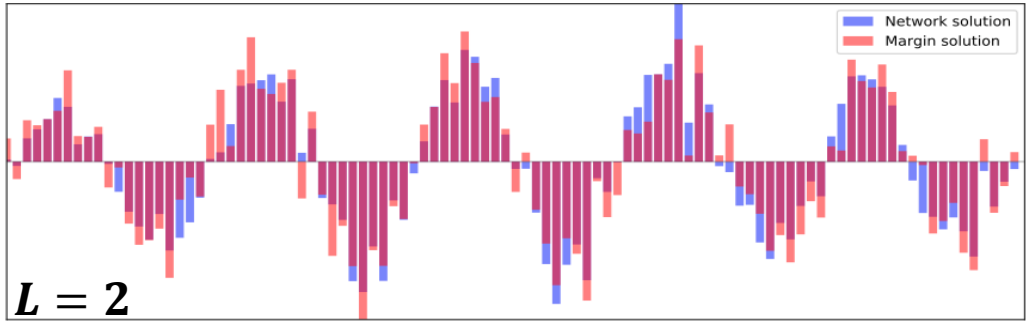


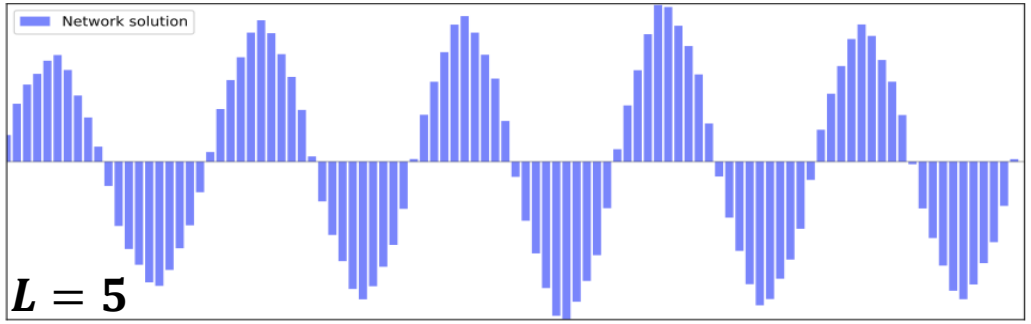
$$\min \|\boldsymbol{\beta}\|_2 \; s.t. \; \forall_i y_i \langle \beta, x_i \rangle \geq 1$$

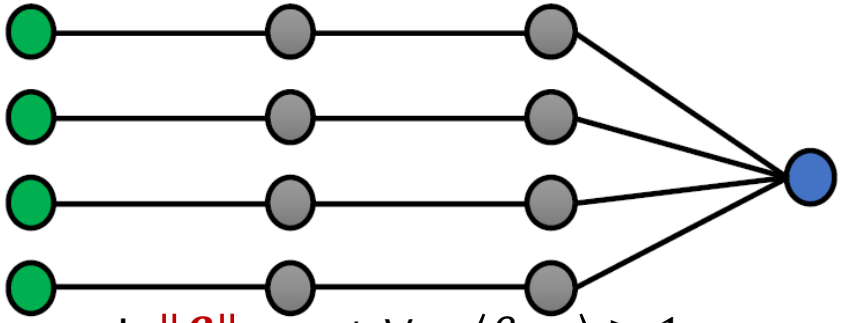




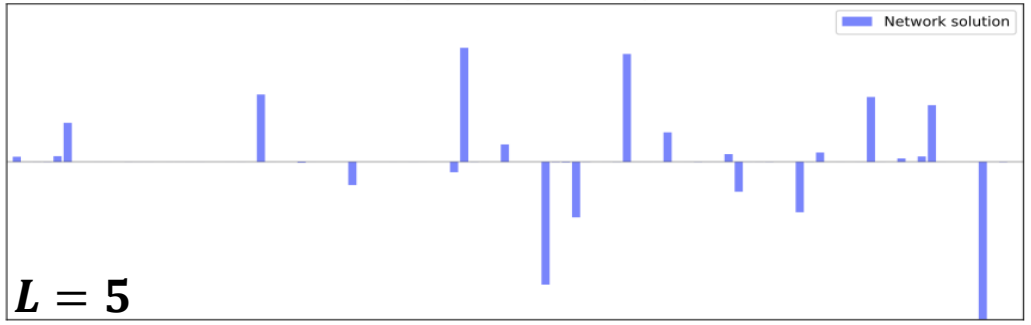
$$\min \|\boldsymbol{DFT}(\boldsymbol{\beta})\|_{2/L} \; s.t. \; \forall_i y_i \langle \beta, x_i \rangle \geq 1$$







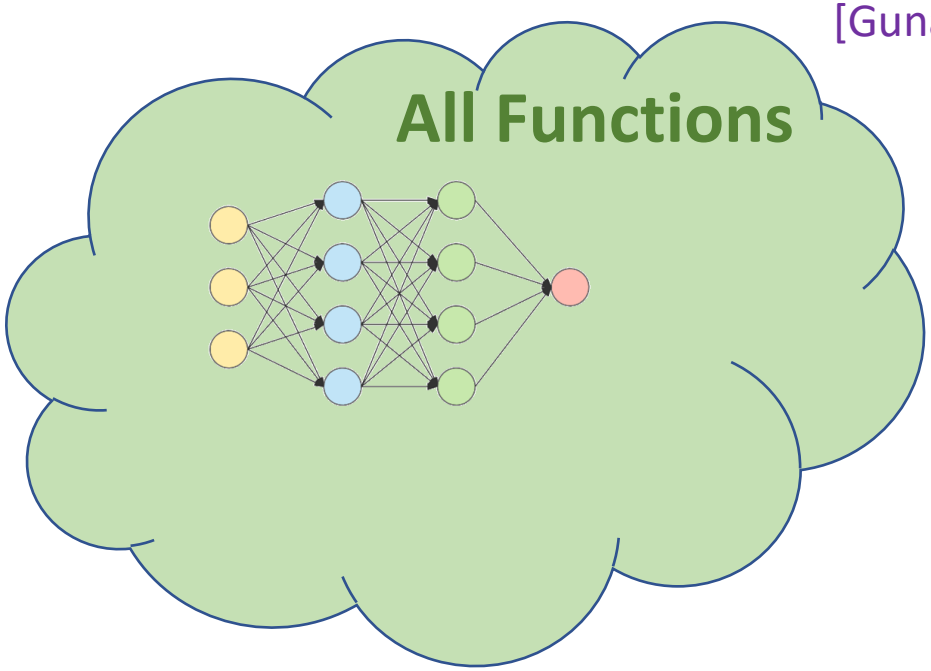
$$\min \|\boldsymbol{\beta}\|_{2/L} \; s.t. \; \forall_i y_i \langle \beta, x_i \rangle \geq 1$$



- **Binary matrix completion** (also: reconstruction from linear measurements)
  - $\boldsymbol{X} = UV$ is over-parametrization of **all matrices** $\boldsymbol{X} \in \mathbb{R}^{n \times m}$
  - GD on U, V
    - ➔ implicitly minimize $\|\boldsymbol{X}\|_*$ [Gunasekar Lee Soudry S 2018a]

- **Linear Convolutional Network**:
  - Complex over-parametrization of **all linear predictors** β
  - GD on weights
    - ➔ implicitly min $\|\boldsymbol{DFT}(\boldsymbol{\beta})\|_{\boldsymbol{p}}$ for $p = \frac{2}{depth}$ (sparsity in freq domain)

[Gunasekar Lee Soudry S 2018b]



- **Binary matrix completion** (also: reconstruction from linear measurements)
  - $\boldsymbol{X} = UV$ is over-parametrization of **all matrices** $\boldsymbol{X} \in \mathbb{R}^{n \times m}$
  - GD on U, V
    - ➔ implicitly minimize $\|\boldsymbol{X}\|_*$ [Gunasekar Lee Soudry S 2018a]

- **Linear Convolutional Network**:
  - Complex over-parametrization of **all linear predictors** β
  - GD on weights
    - ➔ implicitly min $\|\boldsymbol{DFT}(\boldsymbol{\beta})\|_{\boldsymbol{p}}$ for $p = \frac{2}{depth}$ (sparsity in freq domain)

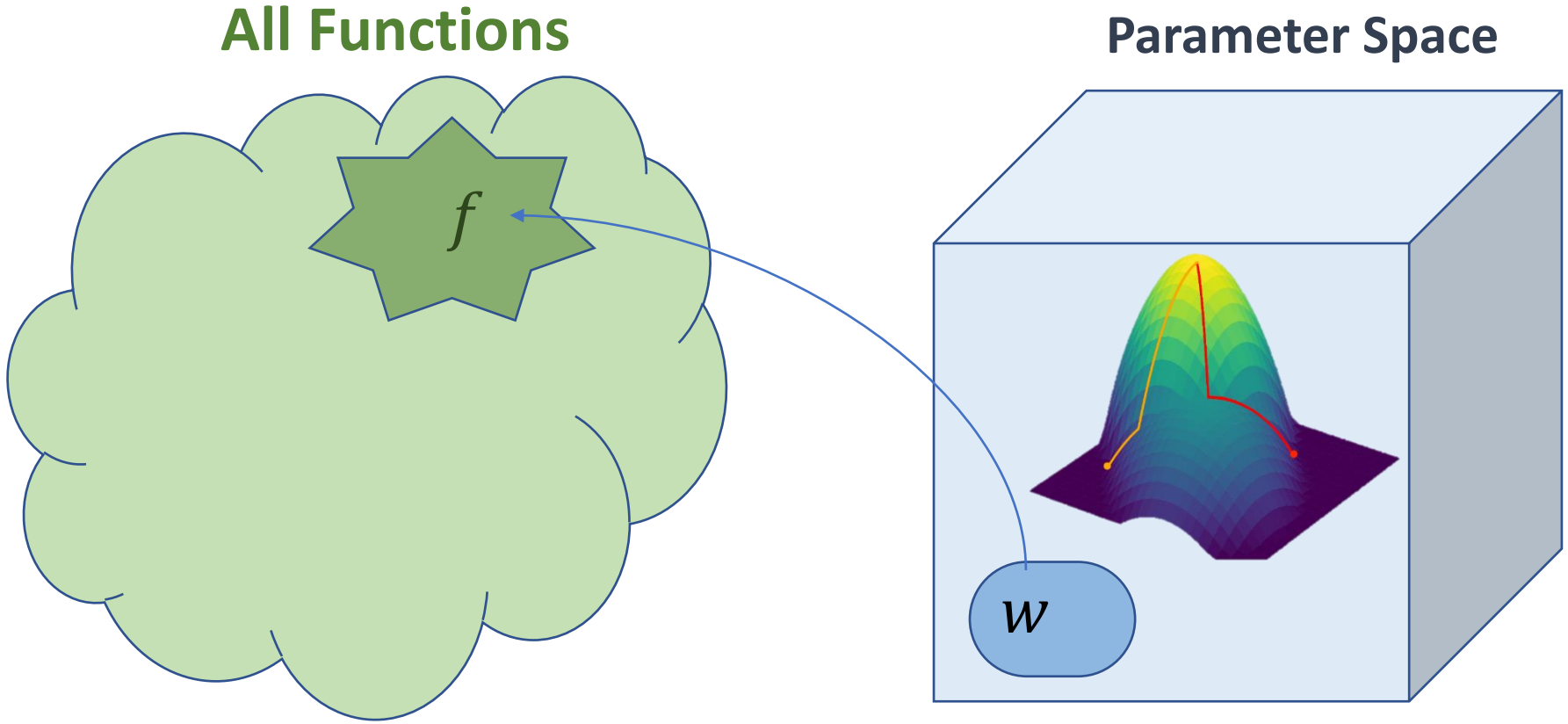
[Gunasekar Lee Soudry S 2018b]

- **Infinite Width ReLU Net**:
  - Parametrization of essentially **all functions** $h: \mathbb{R}^d \to \mathbb{R}$
  - GD on weights
    - ➔ implicitly minimize $\max\left(\int |\boldsymbol{h''}| \boldsymbol{dx} \ , |h'(-\infty) + h'(+\infty)| \right)$ (d=1)

$$\int \left| \boldsymbol{\partial_b^{d+1} Radon(h)} \right| \quad \text{(d>1)}$$

(need to define more carefully to handle non-smoothness; correction term for linear part)

[Savarese Evron Soudry S 2019][Ongie Willett Soudry S 2020][Chizat Bach 2020]



Optimization Geometry and hence Inductive Bias effected by:

- Choice of parameterization (**architecture**)
- Geometry of local search in parameter space
- Optimization choices: Initialization, Batch Size, Step Size, etc

- **Binary matrix completion** (also: reconstruction from linear measurements)
  - $\boldsymbol{X} = UV$ is over-parametrization of **all matrices** $\boldsymbol{X} \in \mathbb{R}^{n \times m}$
  - GD on U, V **(or explicitly minimize $\|\boldsymbol{U}\|_{\boldsymbol{F}}^2 + \|\boldsymbol{V}\|_{\boldsymbol{F}}^2$)**

    ➔ implicitly minimize $\|\boldsymbol{X}\|_*$ [Gunasekar Lee Soudry S 2018a]

- **Linear Convolutional Network**:
  - Complex over-parametrization of **all linear predictors** β
  - GD on weights **(or explicitly minimize $\|\boldsymbol{weights}\|_2^2$)**

    ➔ implicitly min $\|\boldsymbol{DFT}(\boldsymbol{\beta})\|_{\boldsymbol{p}}$ for $p = \frac{2}{depth}$ (sparsity in freq domain)

    [Gunasekar Lee Soudry S 2018b]

- **Infinite Width ReLU Net**:
  - Parametrization of essentially **all functions** $h: \mathbb{R}^d \to \mathbb{R}$
  - GD on weights **(or explicitly min $\|\boldsymbol{weights}\|_2^2$)**

    ➔ implicitly minimize $\max\left(\int |\boldsymbol{h''}| \boldsymbol{dx} \ , |h'(-\infty) + h'(+\infty)| \right)$ (d=1)

    $\int \left| \boldsymbol{\partial_b^{d+1} Radon(h)} \right|$ (d>1)

    (need to define more carefully to handle non-smoothness; correction term for linear part)

    [Savarese Evron Soudry S 2019][Ongie Willett Soudry S 2020][Chizat Bach 2020]

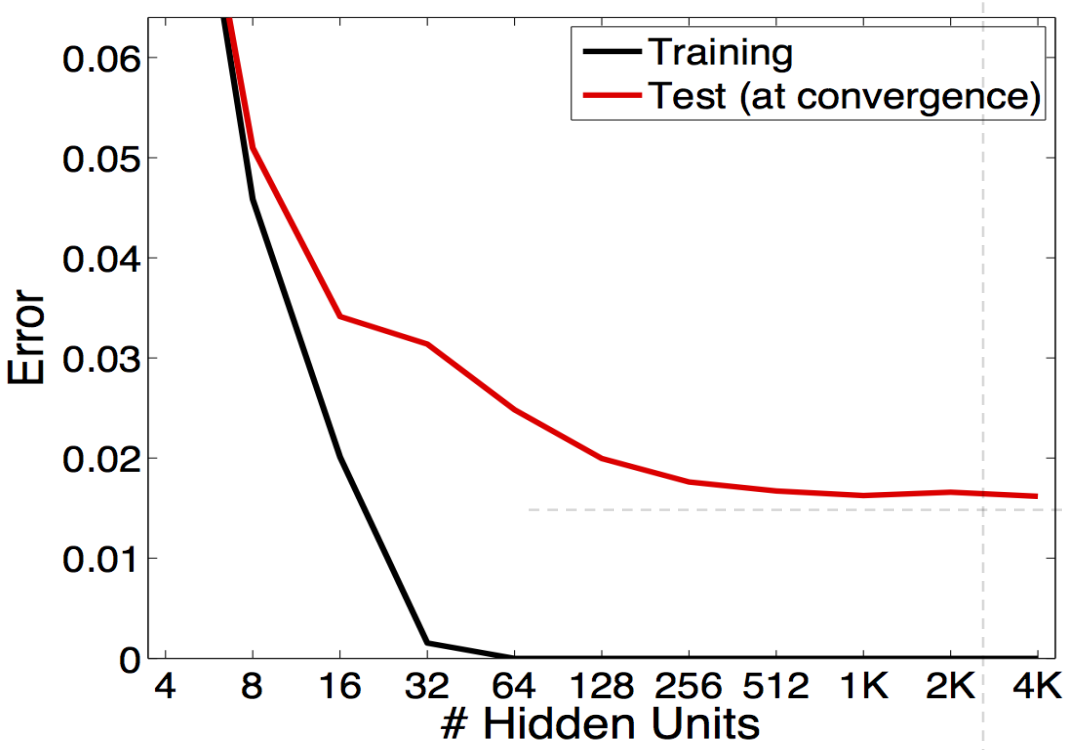
- Does Implicit Bias of Gradient Descent just boil down to regularizing $\|weights\|_2$?

- Answer: sort of, at least asymptotically with logistic/exp loss, for D-homogenous models

…but we'll see that not quite, and in general can be very different

Deep Learning

- Expressive Power
  - We are searching over the space of all functions…
    … but with what bias? What (implicit) assumptions?
  - How does this bias look? Is it reasonable/sensible?
- Capacity / Generalization ability / Sample Complexity
  - What's the true complexity measure (inductive bias)?
  - How does it control generalization?
- Computation / Optimization
  - How and where does optimization bias us?
    Under what conditions?



What fits our understanding:

- Can get generalization even if can fit random labels [we're controlling some other complexity measure]
- Can get implicit regularization (seek small "norm") from optimization algorithm, even if not explicit
- **Generalization becomes better as size increases**

$y = \langle w^*, \phi_\infty(x) \rangle$ ($\|\phi_\infty(x)\|$ bounded)

$\phi_d(x)$ = random projection of $\phi_\infty(x)$

e.g. $\langle \phi_\infty(x), \phi_\infty(x') \rangle = e^{-\|x-x'\|^2}$

and $\phi_d(x)[i] = \frac{1}{\sqrt{d}} \cos(\langle \omega_i, x \rangle + \theta_i)$

$A(S) = \arg\min \|w\| \ \ s.t. L_S(x \mapsto \langle w, \phi_d(x) \rangle) = 0$

i.e. $\forall_{(x_i,y_i) \in S} y_i = \langle w, \phi_d(x_i) \rangle$

A similar example:

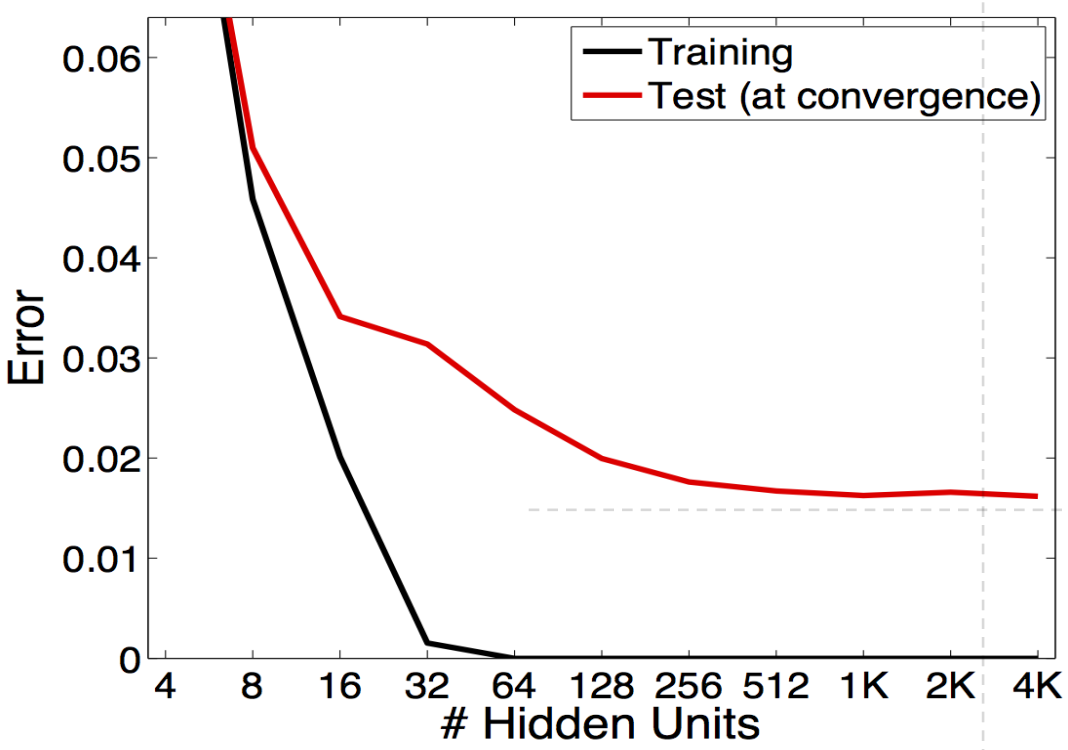
Matrix completion using a rank-*d* factorization:

$L(X) = \|X - A\|_2^2$, $\hat{L}$ based on nk observed entries

$X = UV^\top$, $U, V \in \mathbb{R}^{n \times d}$ ➔ $rank(X) \leq d$

If $d < k$: $\arg\min \hat{L}(X)$ s.t. $rank(X) \leq d$

If $d > k$: $\arg\min \|X\|_*$ $s.t.$ $\hat{L}(X) = 0, rank(X) \leq d$



What fits our understanding:

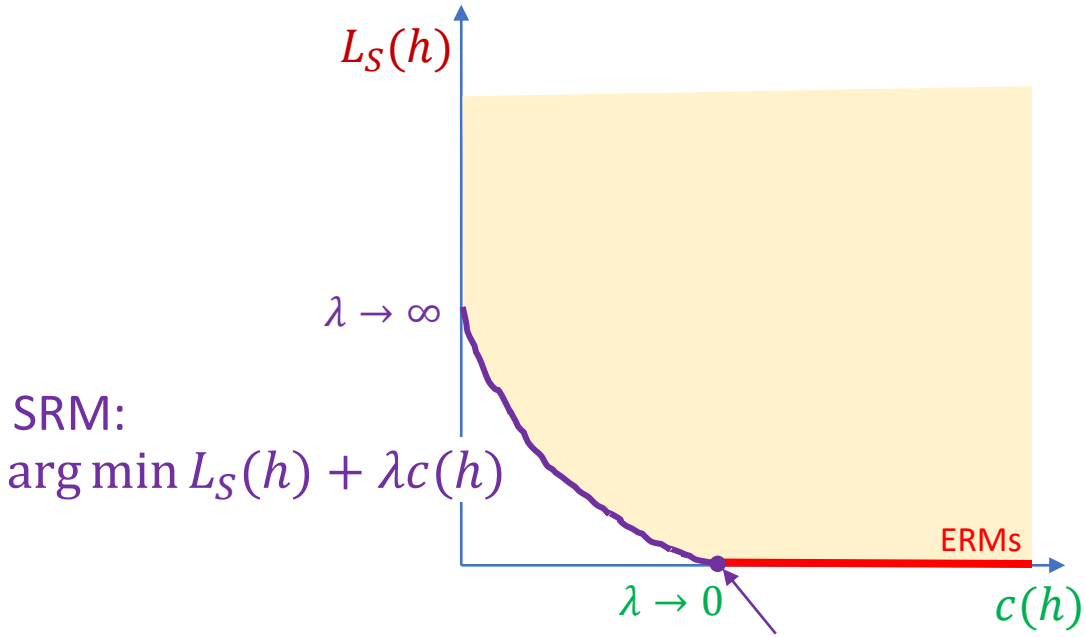
- Can get generalization even if can fit random labels [we're controlling some other complexity measure]
- Can get implicit regularization (seek small "norm") from optimization algorithm, even if not explicit
- **Generalization becomes better as size increases**

What doesn't fit:

- Even when the approximation error>0 (with noise), we get good generalization with $L_S(h) = 0$

Table 1: The training and test accuracy (in percentage) of various models on the CIFAR10 dataset.

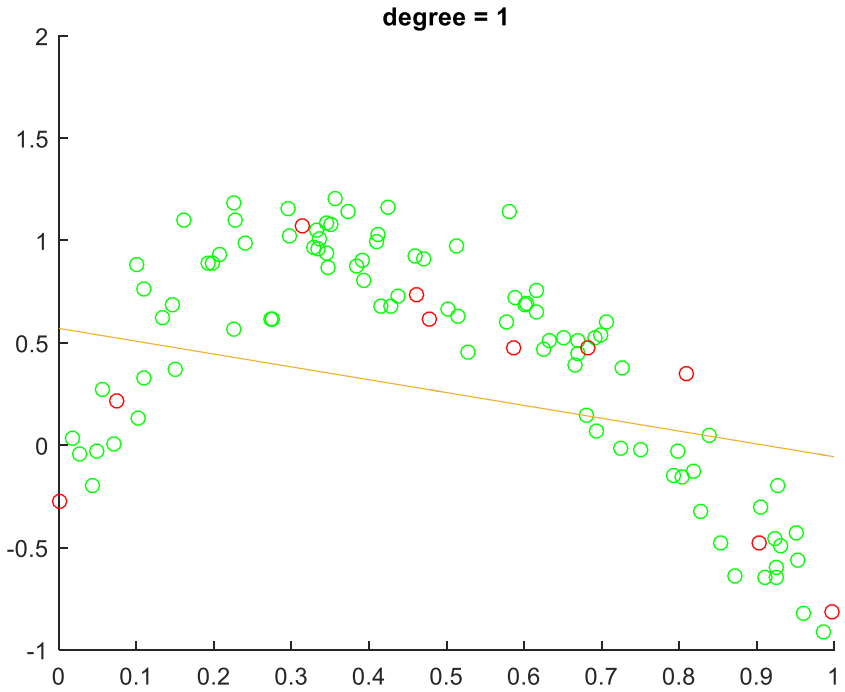
model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
MLP 3x512	1,735,178	no	yes	100.0	53.35
		no	no	100.0	52.39
(fitting random labels)		no	no	100.0	10.48



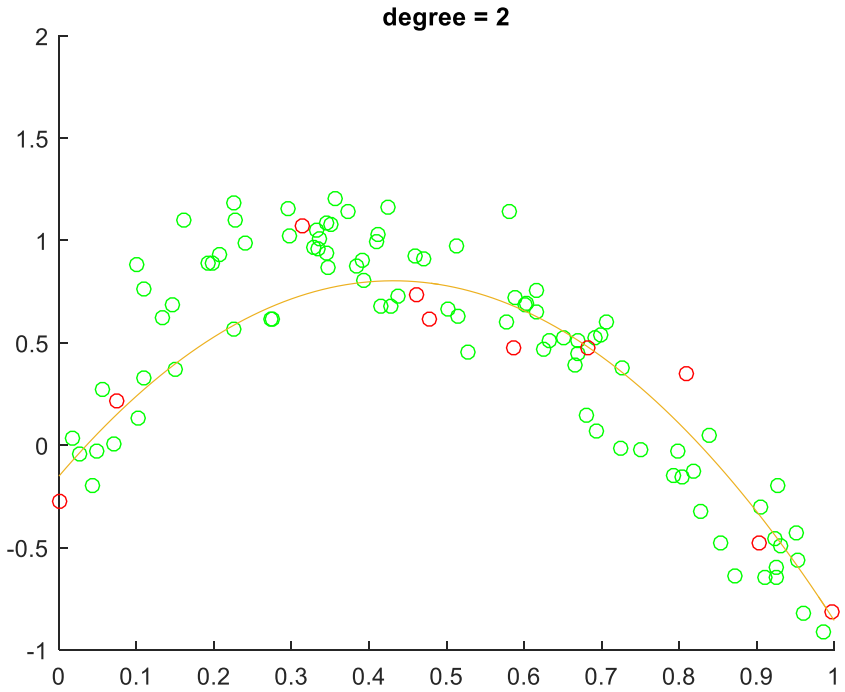
SRM:
$\arg\min L_S(h) + \lambda c(h)$

MDL:
$\arg\min c(h)\ \ s.t. L_S(h) = 0$

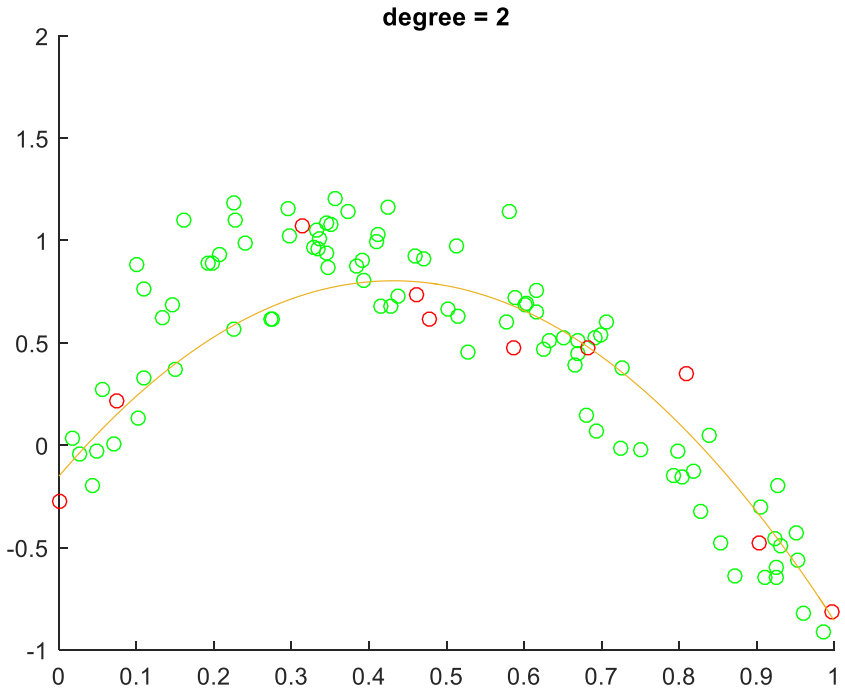
Intro to Machine Learning, Lecture 2



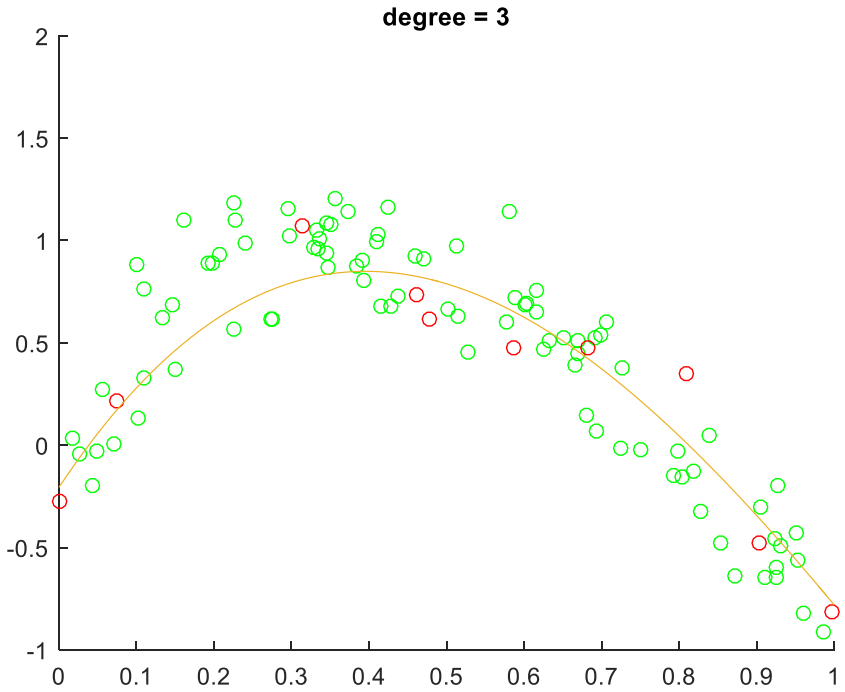
Intro to Machine Learning, Lecture 2



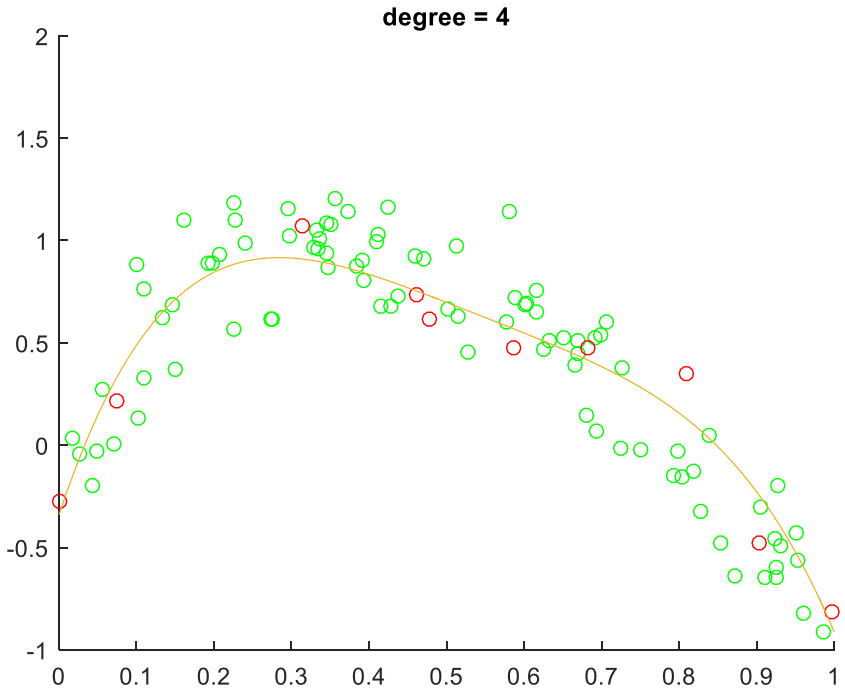
Intro to Machine Learning, Lecture 2



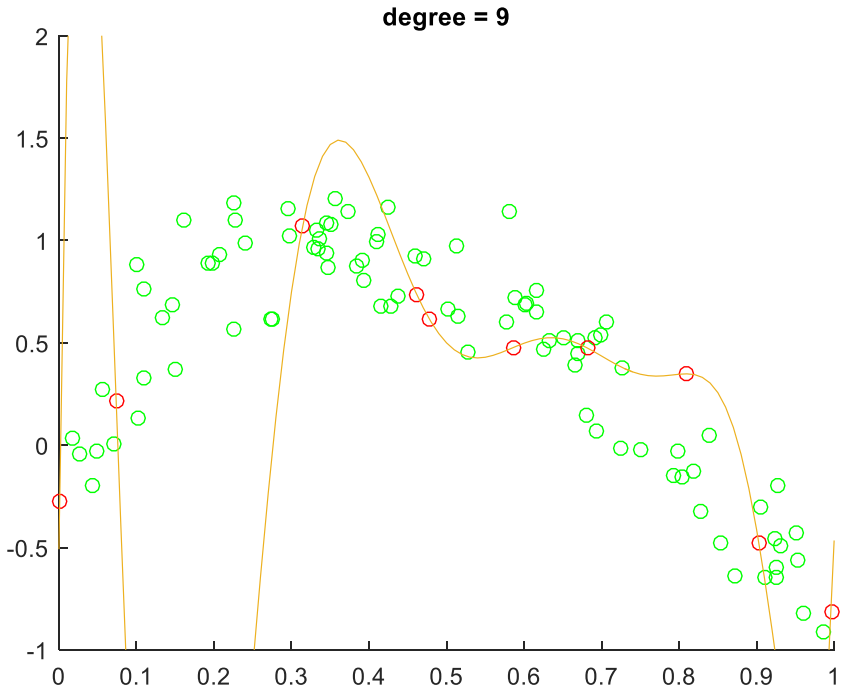
Intro to Machine Learning, Lecture 2

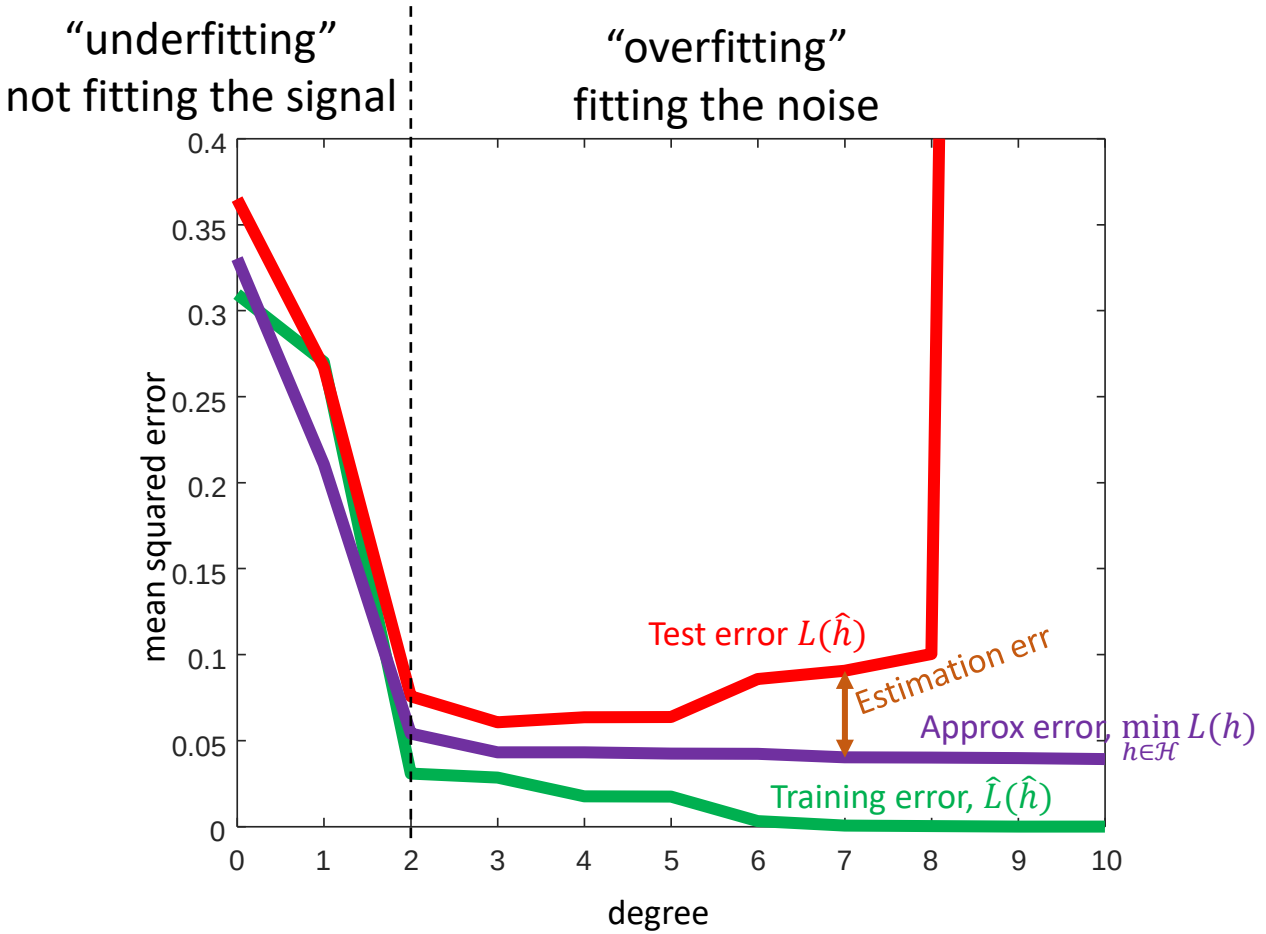


Intro to Machine Learning, Lecture 2



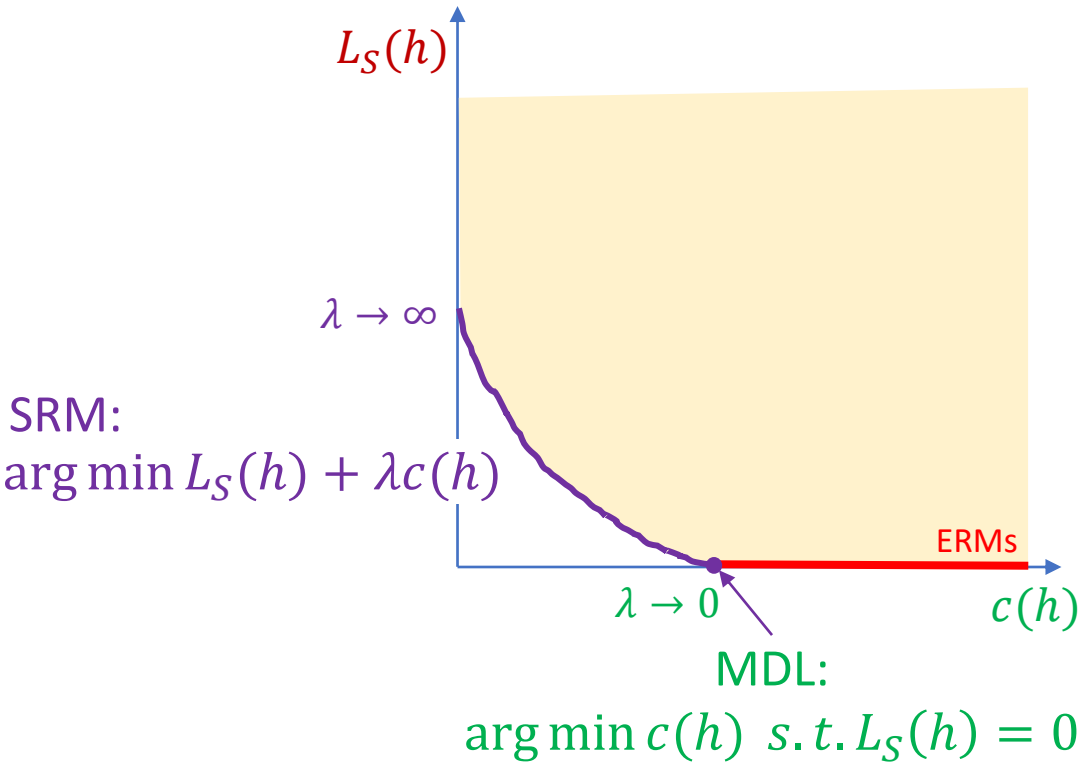
Intro to Machine Learning, Lecture 2

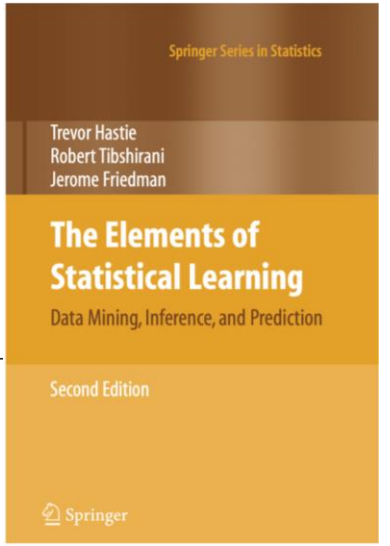


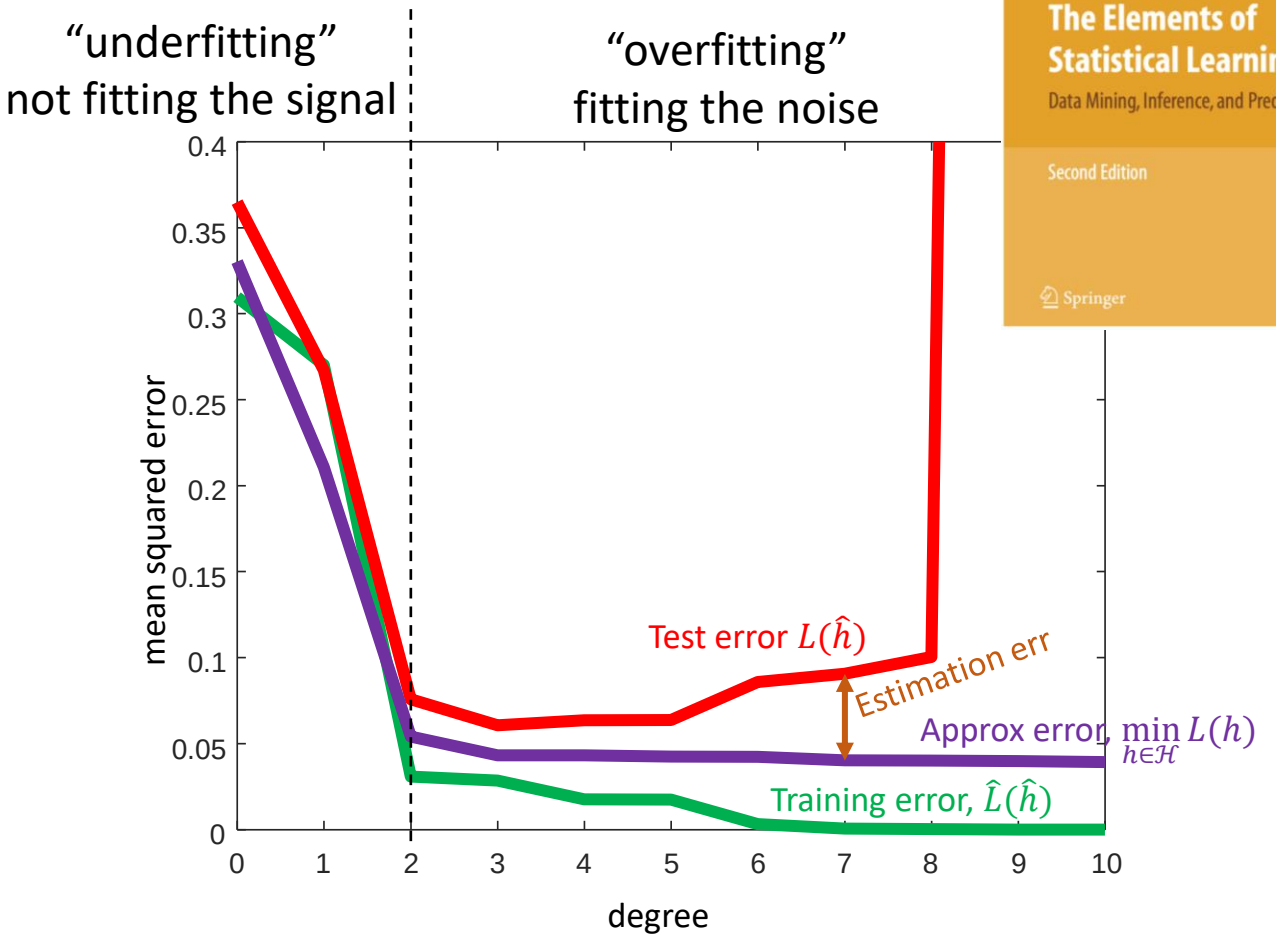


$$L(\hat{h}) \leq \underbrace{\inf_{h\in\mathcal{H}} L(h)}_{\text{approximation error}} + \underbrace{\sqrt{\frac{\log|\mathcal{H}| + 2\log 2/\delta}{n}}}_{\text{estimation error}}$$

“a model with zero training error is overfitting […] and will typically generalize poorly”



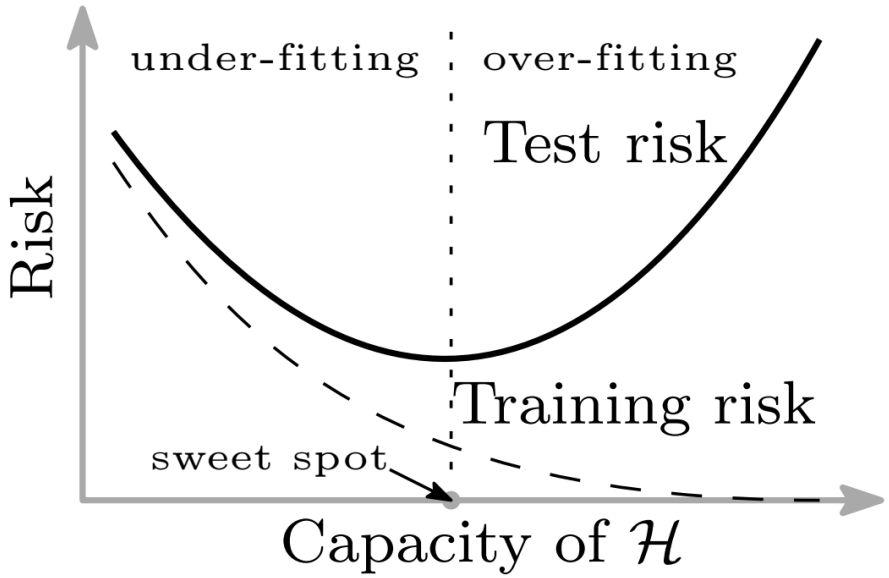


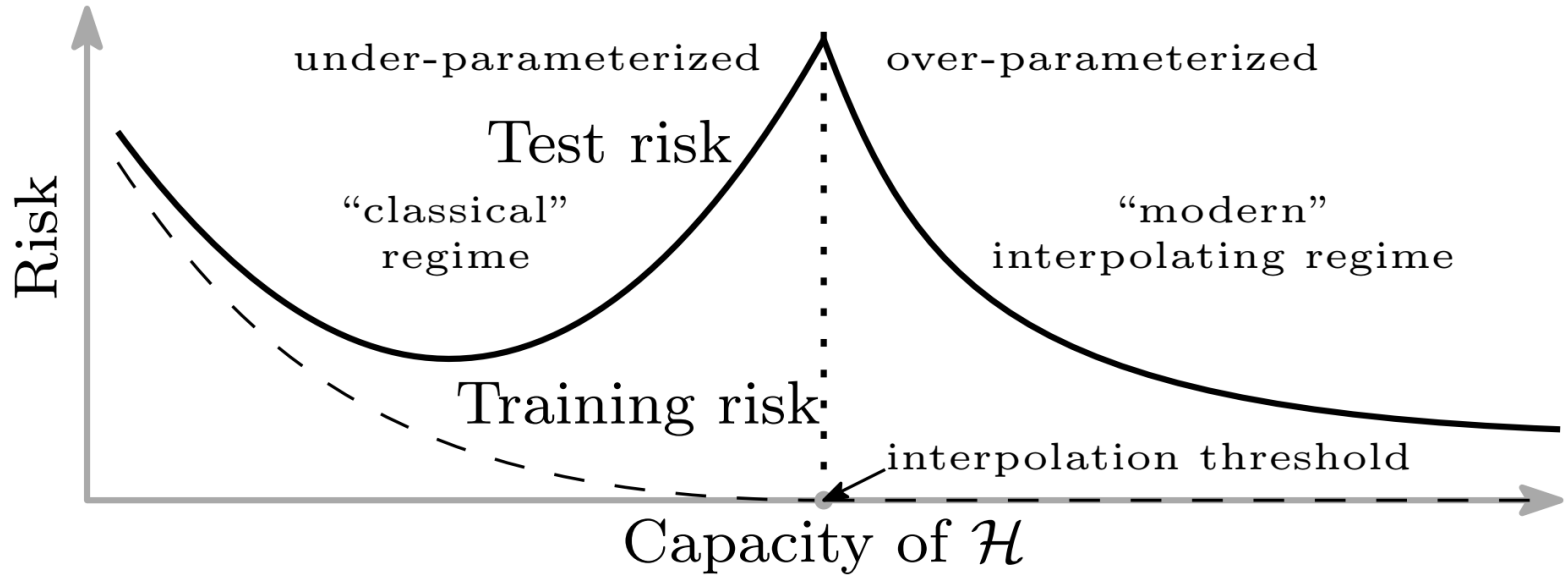


$$L(\hat{h}) \le \underbrace{\inf_{h\in\mathcal{H}} L(h)}_{\text{approximation error}} + \underbrace{\sqrt{\frac{\log|\mathcal{H}| + 2\log^{2}/_{\delta}}{n}}}_{\text{estimation error}}$$

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

Mikhail Belkin[a,b,1], Daniel Hsu[c], Siyuan Ma[a], and Soumik Mandal[a]





$L(w) = \mathbb{E}[(\langle w, \phi_d(x)\rangle - y)^2]$ $\quad \hat{L}(w) = \frac{1}{n}\sum_i(\langle w, \phi_d(x_i)\rangle - y_i)^2$

$\phi_d(x) \in \mathbb{R}^d$

~~$\hat{w} = \arg\min \hat{L}(w)$~~

$\hat{w}$ = GD on $\hat{L}(w)$

```
>> w = PhiX \ y
```

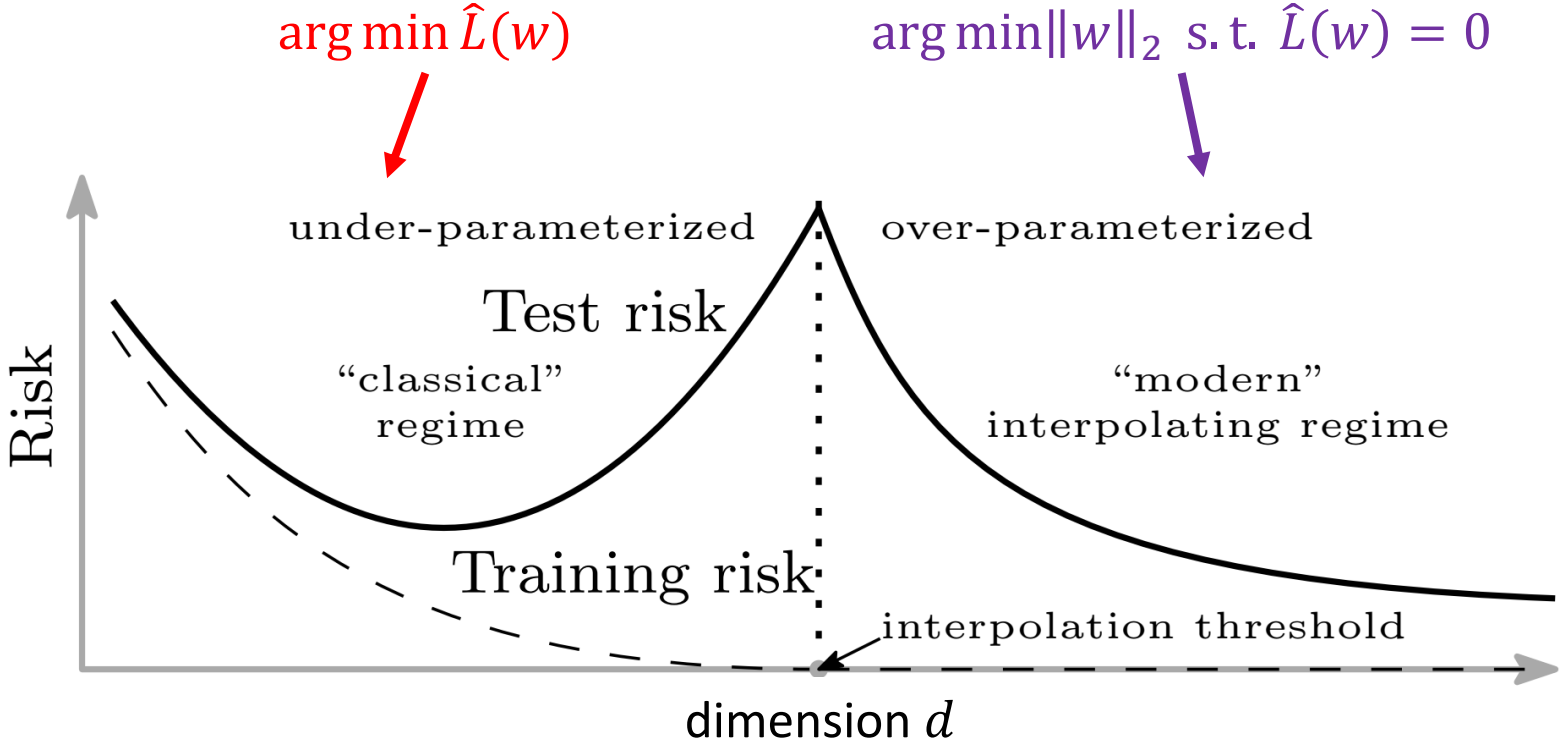
```
>>> w = np.linalg.lstsq(PhiX,y)[0]
```

$x = \langle w^*, \phi_\infty(x)\rangle \quad (\|\phi_\infty(x)\| \le 1)$

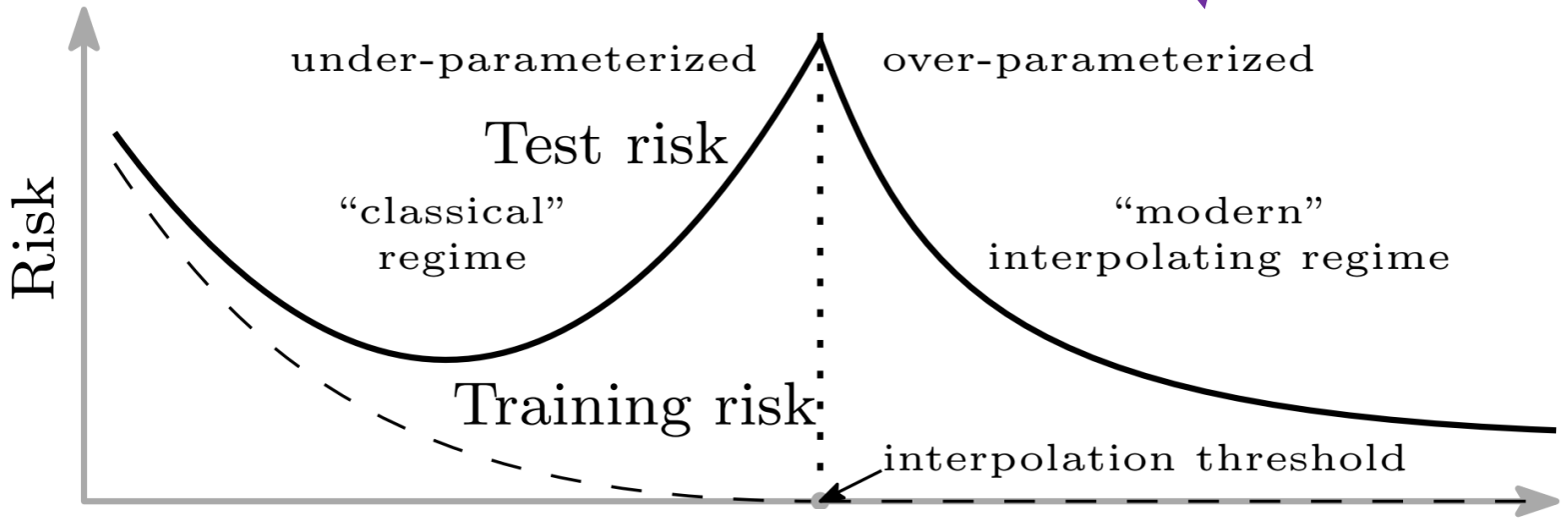
$\phi_d(x)$ = random projection of $\phi_\infty(x)$

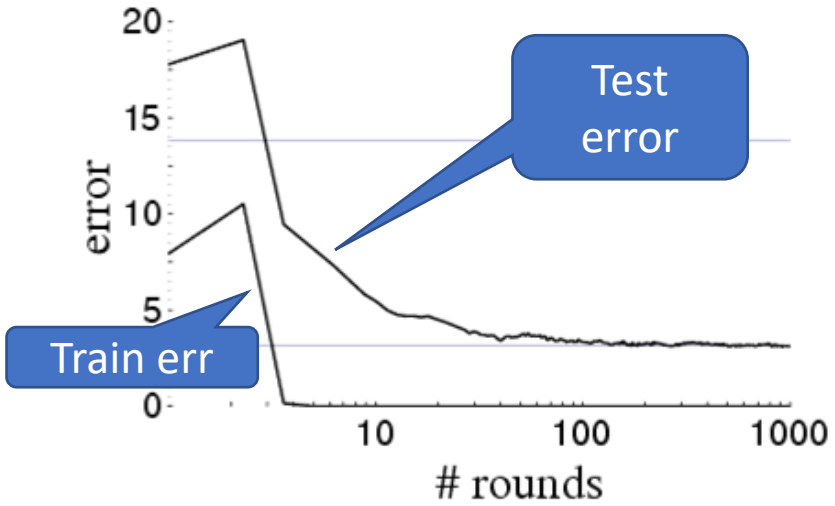
e.g. $\langle \phi_\infty(x), \phi_\infty(x')\rangle = e^{-\|x-x'\|^2}$

and $\phi_d(x)[i] = \frac{1}{\sqrt{d}}\cos(\langle \omega_i, x\rangle + \theta_i)$

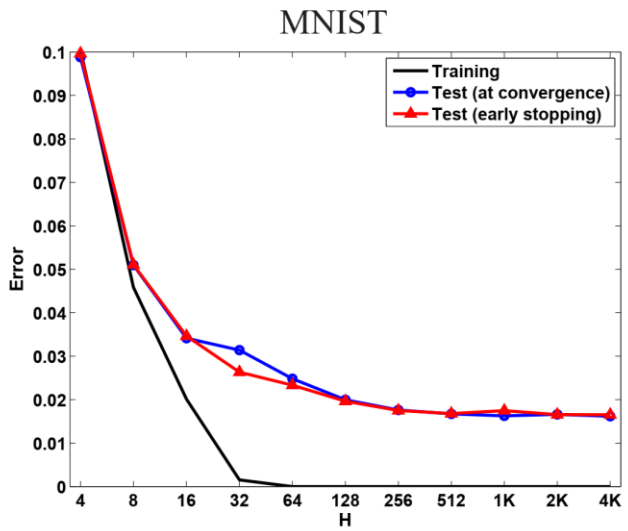


$\arg\min \|w\|_2 \text{ s.t. } \hat{L}(w) = 0$





[Bartlett et al "Boosting the Margin" 1998]



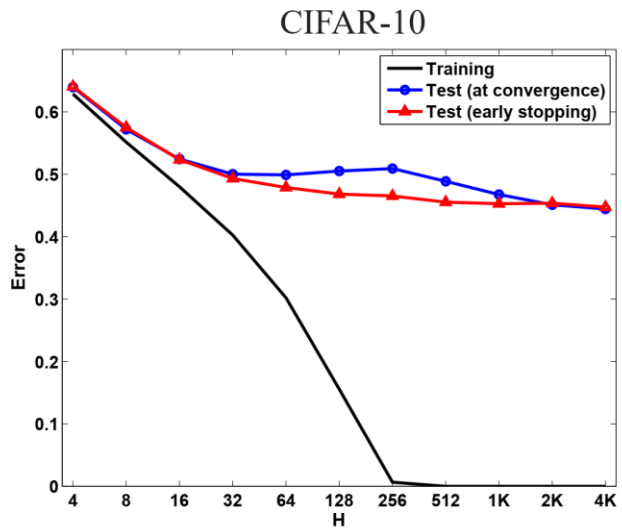
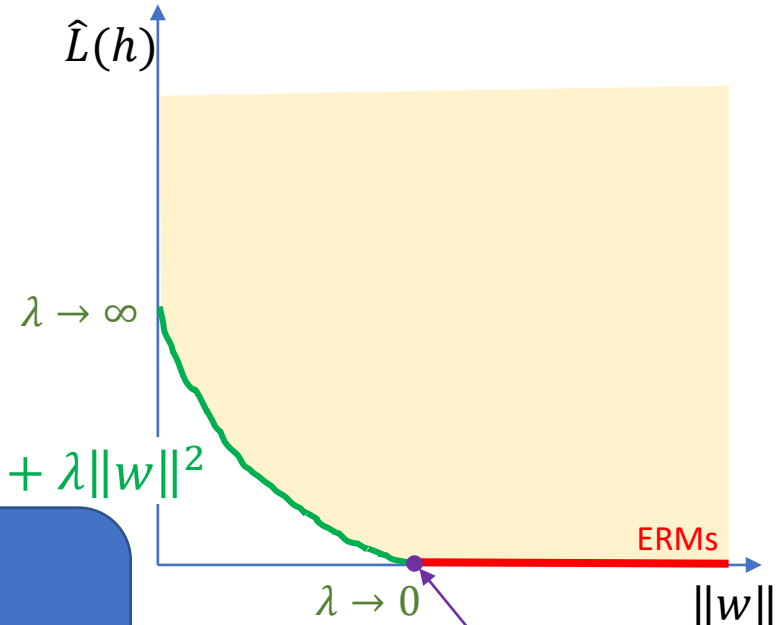


Table 1: The training and test accuracy (in percentage) of various models on the CIFAR10 dataset.

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

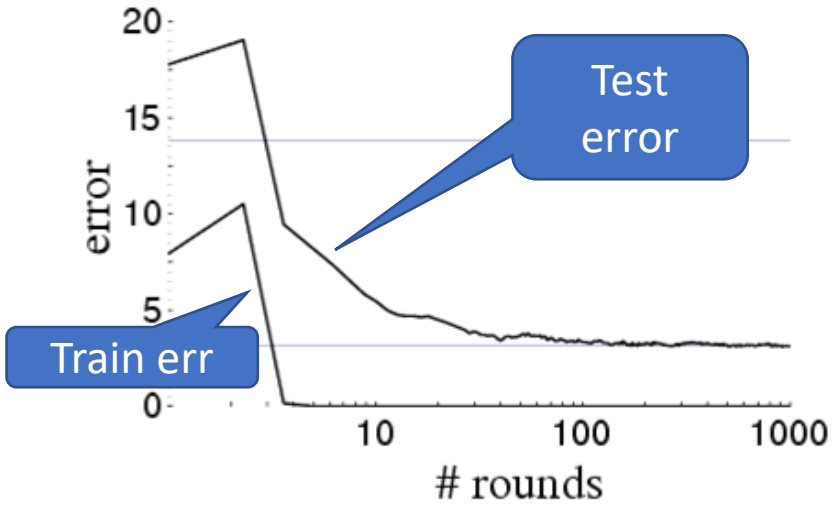
[Zhang et al "Rethinking generalization" ICLR 2017]



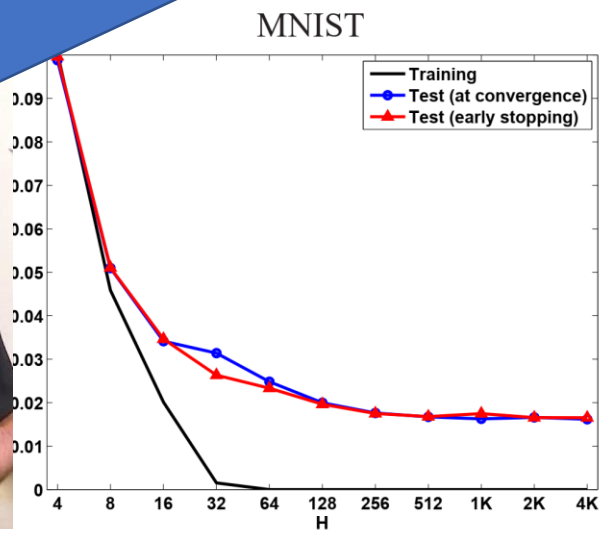
SRM:
$\arg\min \hat{L}(w) + \lambda\|w\|^2$

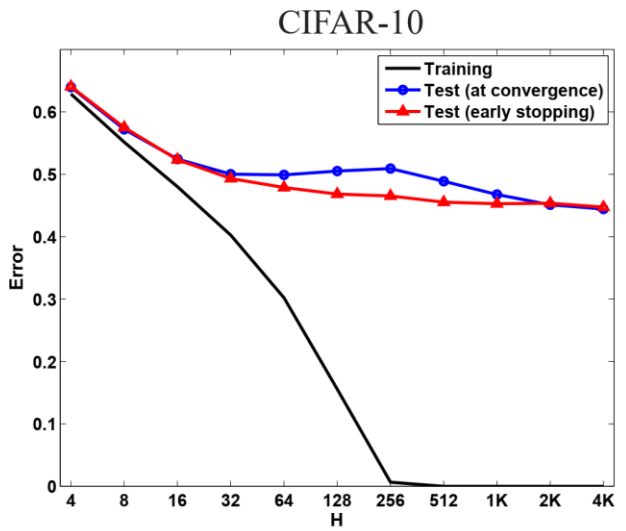
MDL:
$\arg\min \|w\| \ s.t. \hat{L}(w) = 0$

We can learn with MDL ($\hat{L}(w) = 0$, "**interpolation learning**") in many settings where $L(w^*) \gg 0$, eg noisy settings where $y = h_{w^*}(x) + noise$.
Often, **overfitting (fitting the noise) is benign**, and not as harmful as theory tells us.
-Misha Belkin, 2018



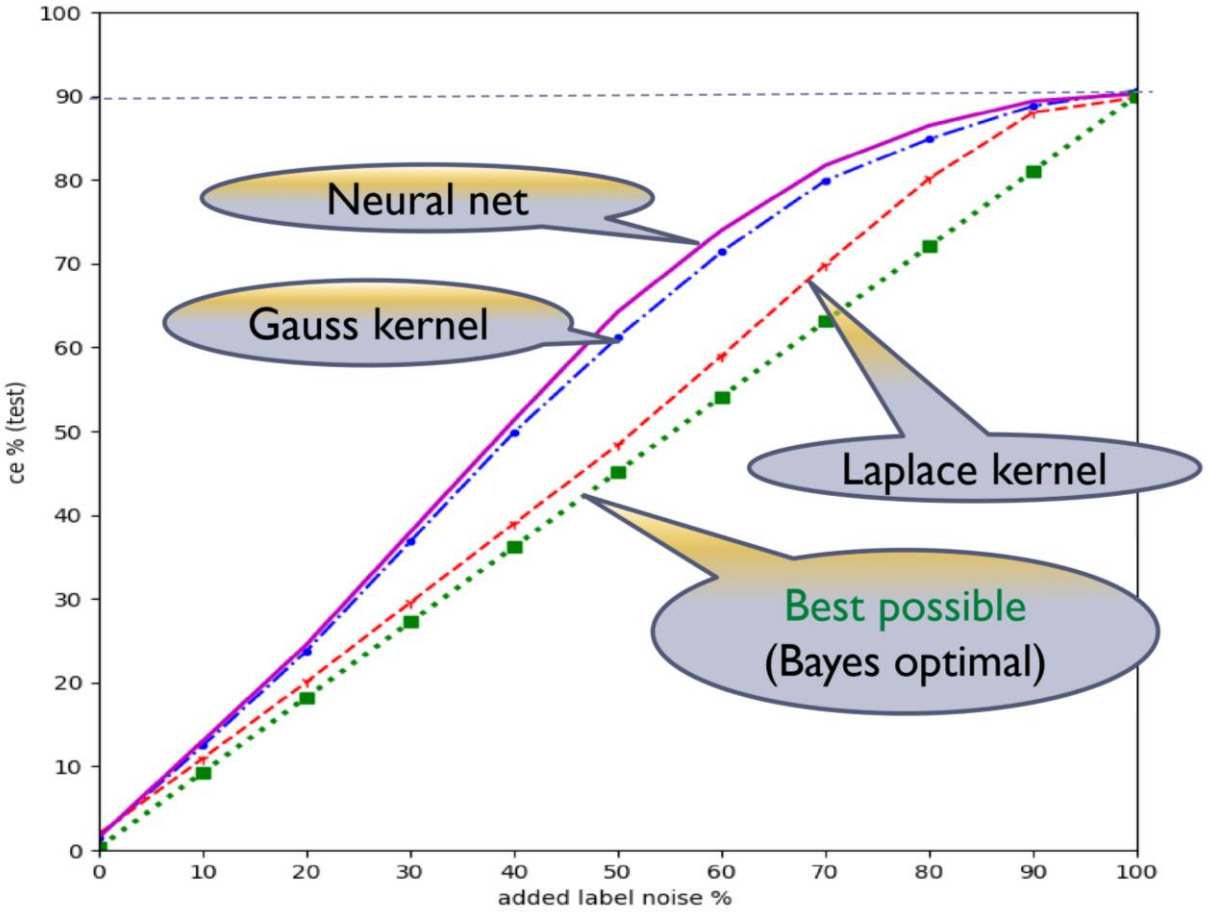
[Bartlett et al "Boosting the Margin" 1998]



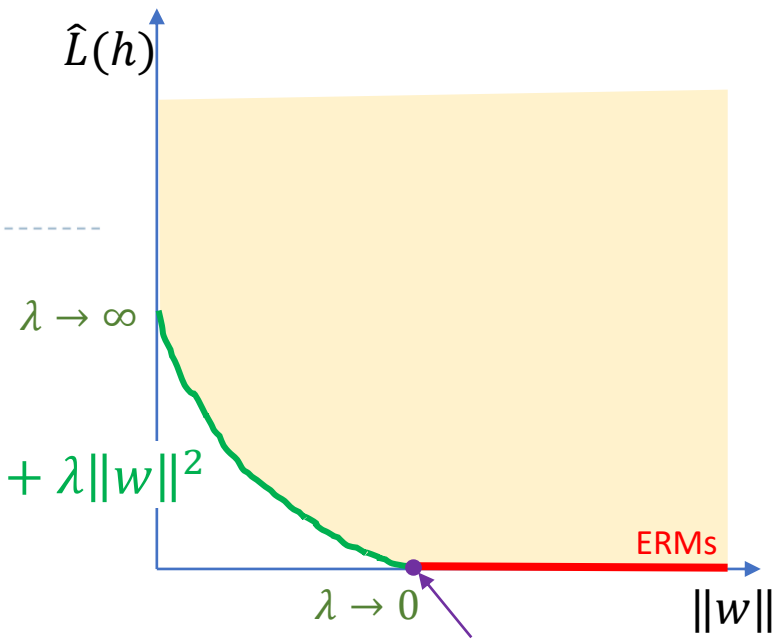


Interpolation does not overfit even for very noisy data

All methods (except Bayes optimal) have zero training *square* loss.



[Belkin Ma Mandal, ICML 18]

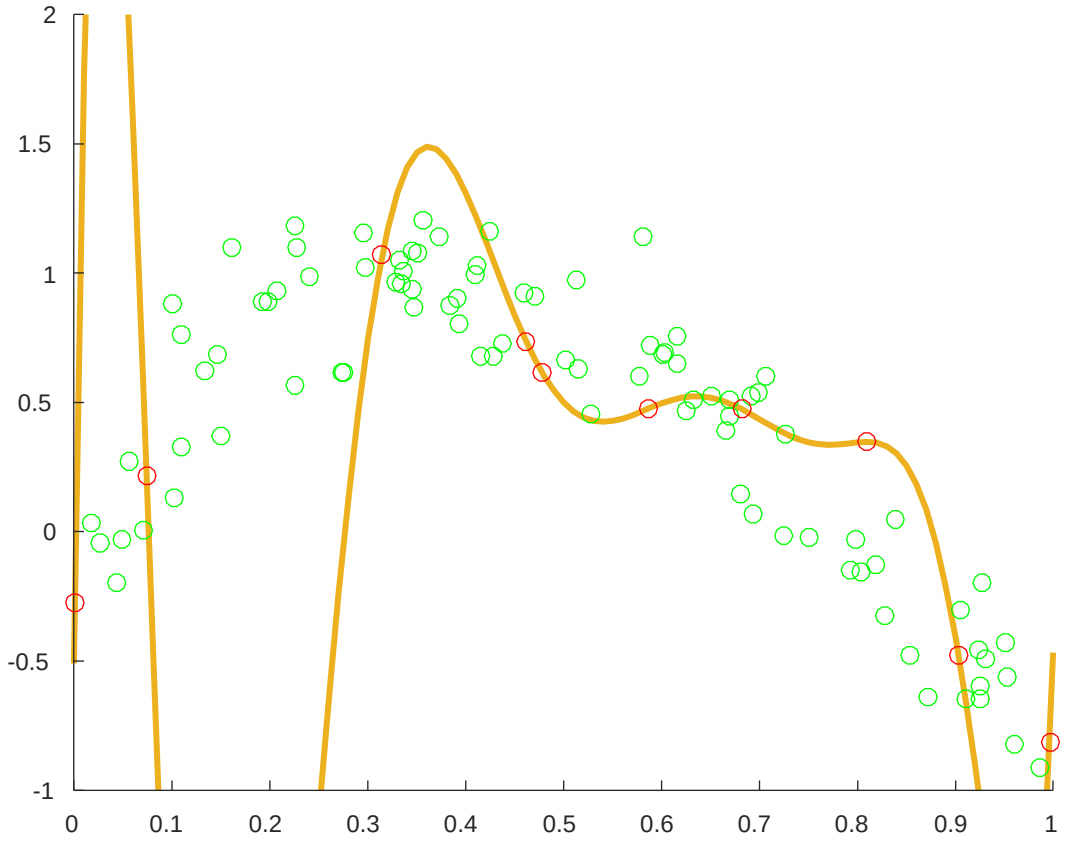


SRM:
$\arg\min \hat{L}(w) + \lambda\|w\|^2$

MDL:
$\arg\min \|w\| \ \ s.t. \hat{L}(w) = 0$

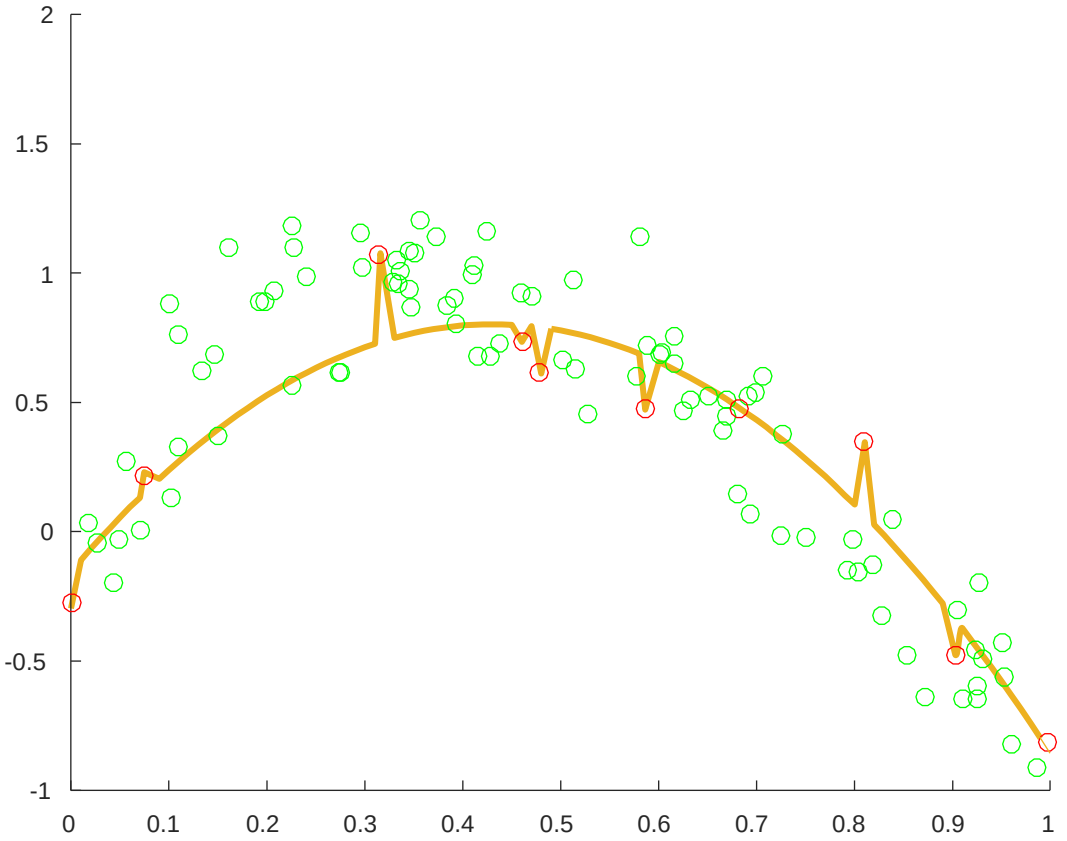
Harmful Overfitting

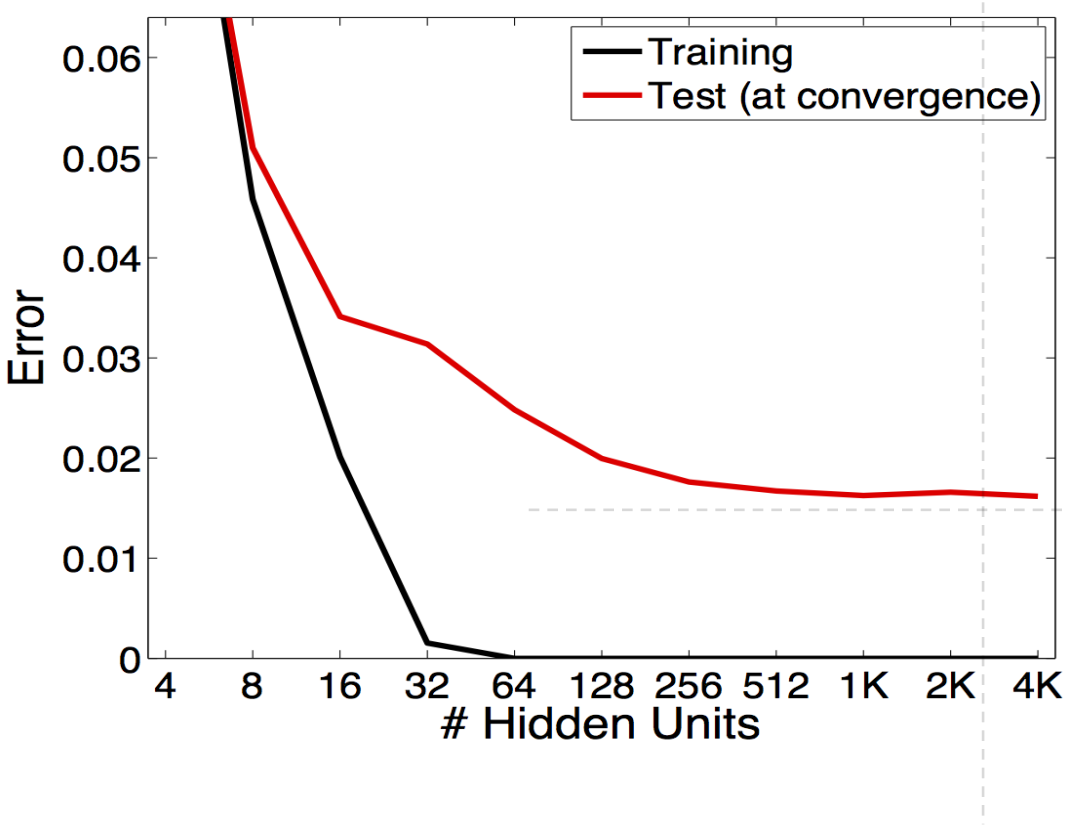
(fitting noise has large effect everywhere, overwhelms signal fit)



Benign Overfitting

(fitting noise has measure ≈0 effect)





What fits our understanding:

- Can get generalization even if can fit random labels
- Can get implicit regularization (seek small norm) from optimization algorithm, even if not explicit
- Generalization becomes better as size increases (because laten complexity is getting smaller)

What we need to ask:

- What's the complexity measure?
- How is it minimized?
- How does it ensure generalization?

Can we explain behavior using complexity measure?

What we need to rethink:

- Even when the approximation error>0 (with noise), we get good generalization with $L_S(h) = 0$

Ultimate Question: What is the true Inductive Bias? What makes reality *efficiently* learnable by fitting a (huge) neural net with a specific algorithm?

The "complexity measure" approach

Identify $c(h)$ s.t.

- Optimization algorithm biases towards low $c(h)$

  …and if there h with low $c(h)$ and $L_S(h) = 0$ (or low $L_S(h)$), opt alg finds it
- $\mathcal{H}_{c(reality)} = \{h | c(h) \leq c(reality)\}$ has low capacity
- Reality is well explained by low $c(h)$

- Mathematical questions:
  - What is the bias of optimization algorithms?
  - What is the capacity (≡sample complexity) of the sublevel sets $\mathcal{H}_c$?
- Question about reality (scientific Q?): does it have low $c(h)$?