ETH zürich

In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning

Venue: Accepted as a workshop contribution at ICLR 2015

Authors: Behnam Neyshabur[1], Ryota Tomioka[1], Nathan Srebro[1]

[1]Toyota Technological Institute at Chicago

Seminar in Advanced Topics in Machine Learning and Data Science

Presented by: Steve Rhyner

March 20, 2024

Executive Summary

Problem:

- Deep Learning is a blackbox

Goal:

- Shed light on deep learning by suggesting existence of implicit form of capacity control
- Understand why optimization directs us to a “simple” minimum

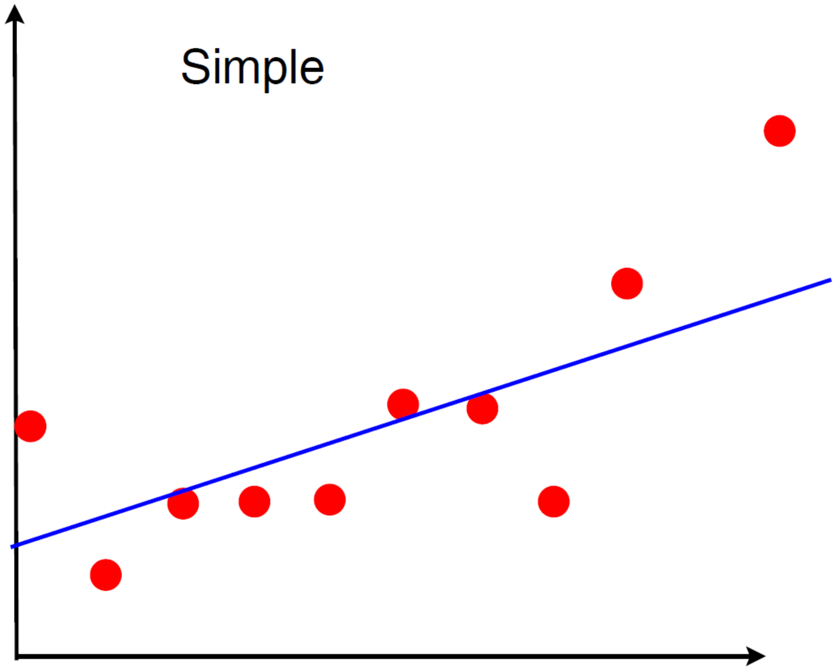
Key Idea:

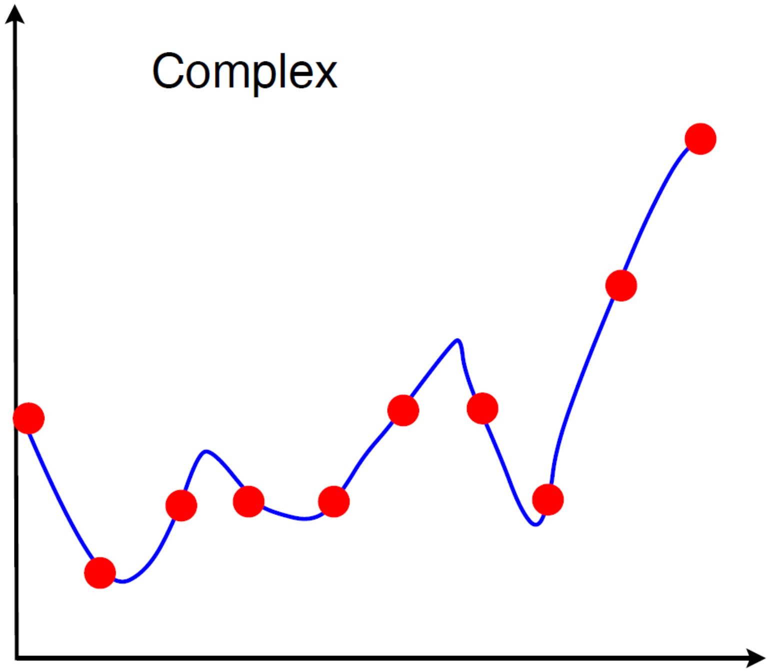
- First paper to explain double-descent phenomenon
- Implicit regularization of optimization as an explanation for phenomenon that increase in network size decreases approximation and estimation error
- Learning and selecting is equivalent if we have sufficiently many hidden units

Mechanism:

- Compare generalization capabilities for increasing number of hidden units
- Draw a matrix factorization analogy to feed-forward networks

Capacity control: Bias-Variance Tradeoff





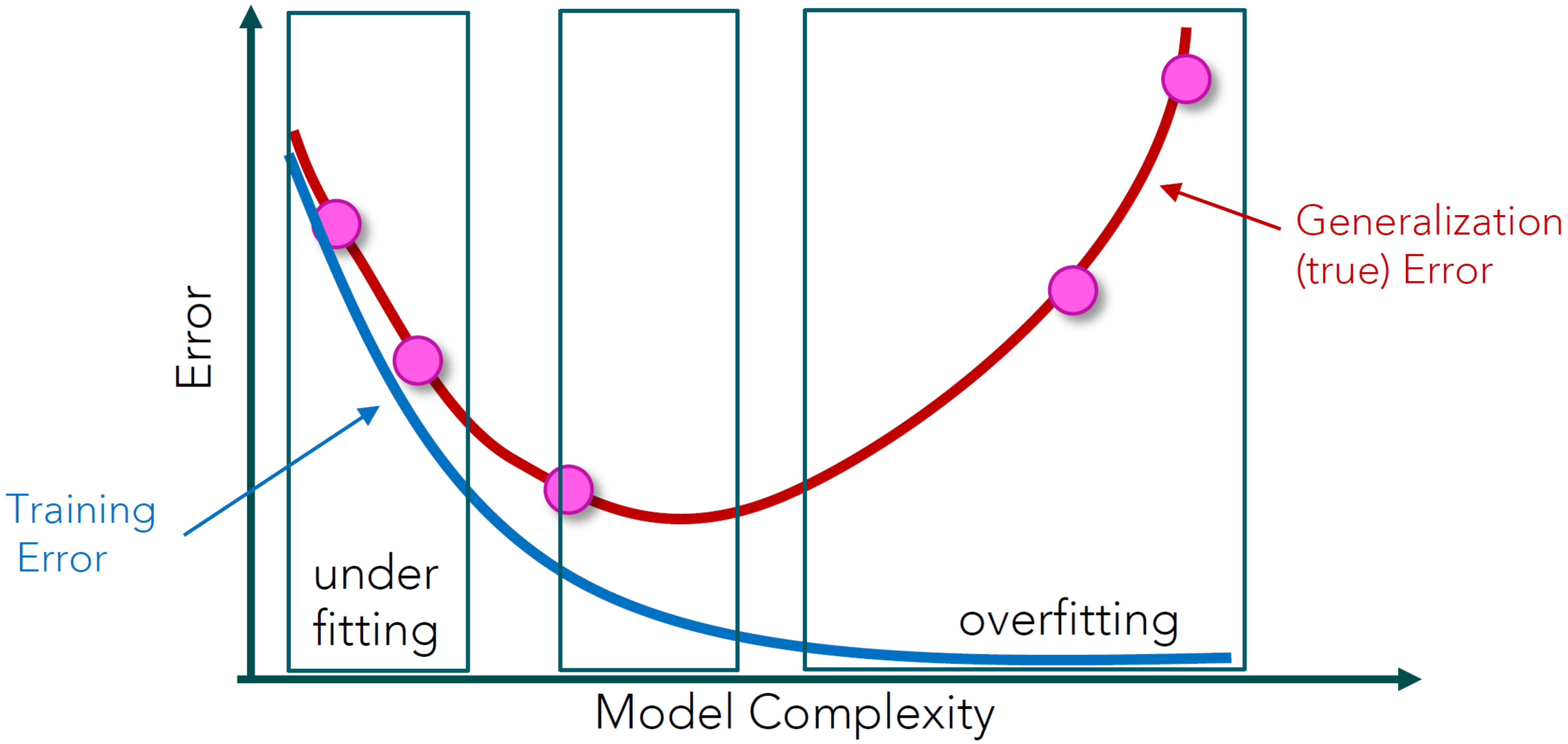
- Simple $\mathcal{C}$ leads to underfitting

- Complex $\mathcal{C}$ leads to overfitting

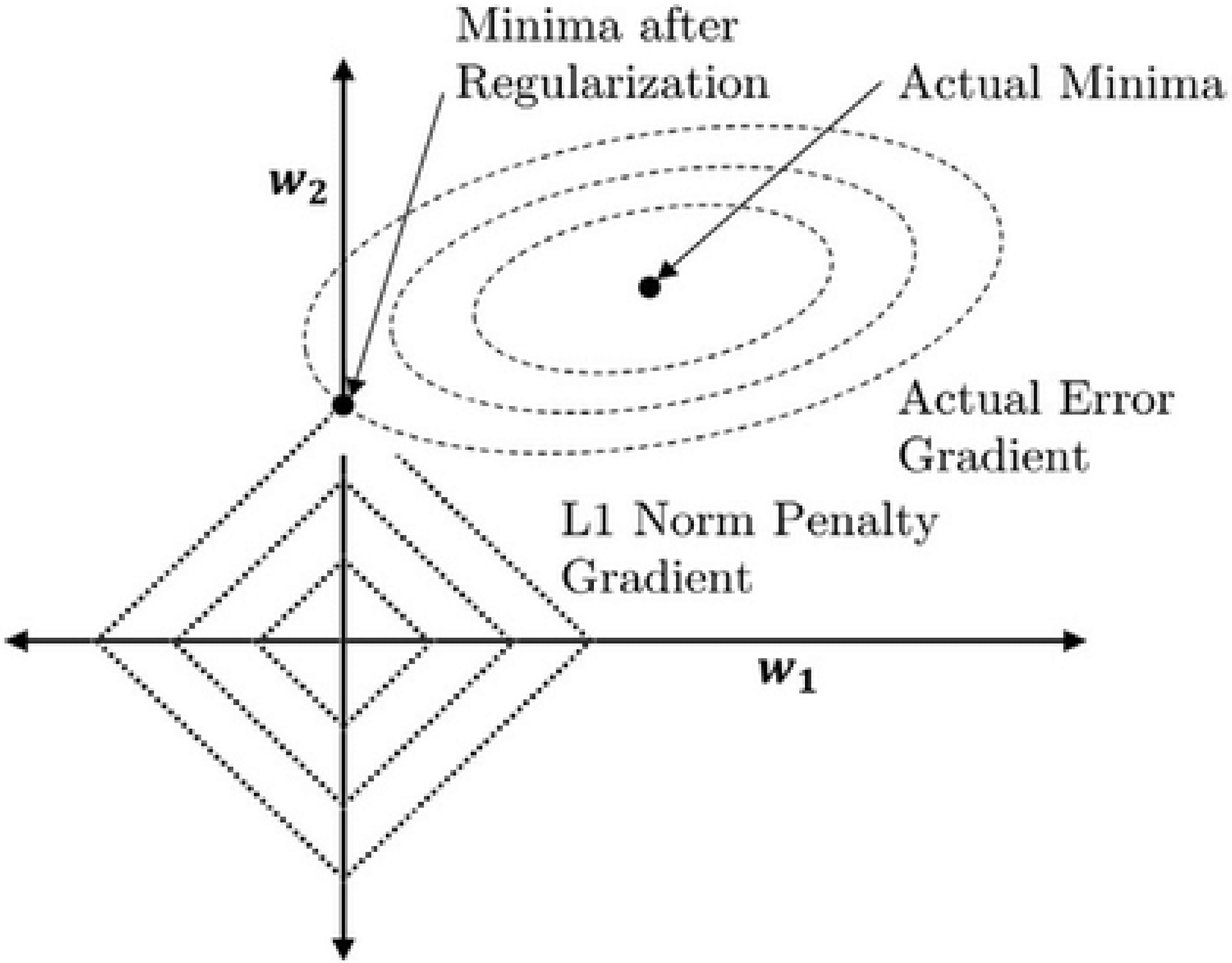
Objective: Find best balance between the two

Split error into Bias and Variance

Capacity control: Bias-Variance Tradeoff



Capacity control: ℓ_1 - regularization



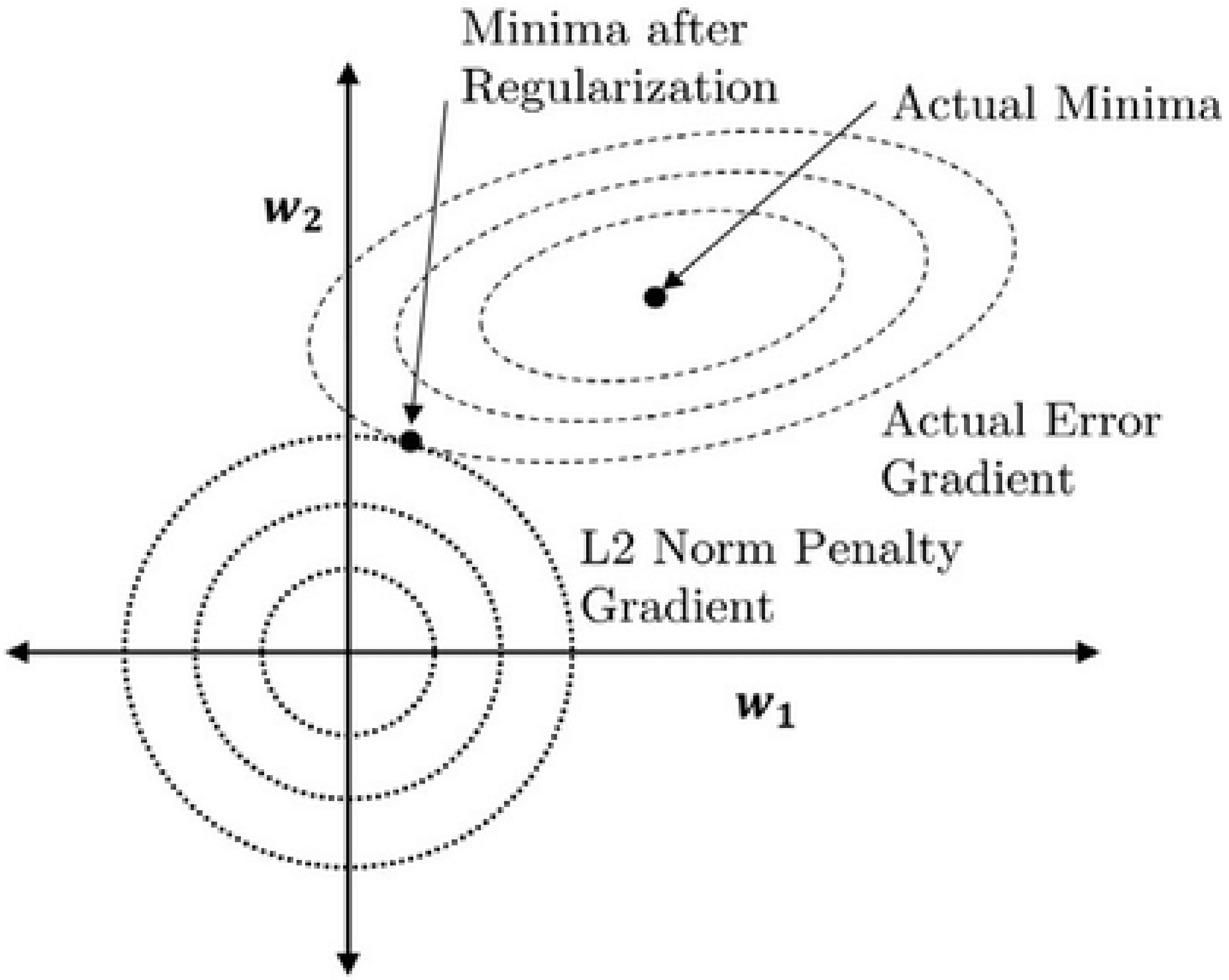
- ℓ_1 -norm

$$\|v\|_1 = \sum_{h=1}^{H} |v_h|$$

- Added as a penalty term to the objective function

- Induces sparsity

Capacity control: ℓ_2 - regularization

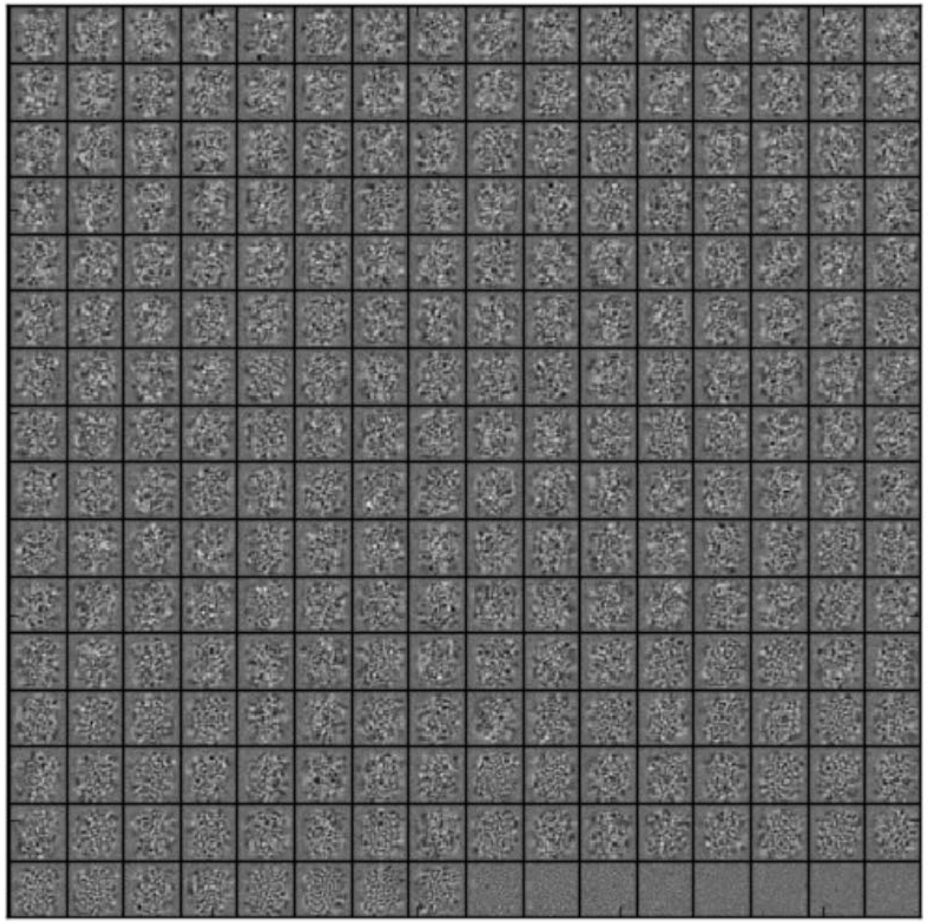


- ℓ_2 -norm

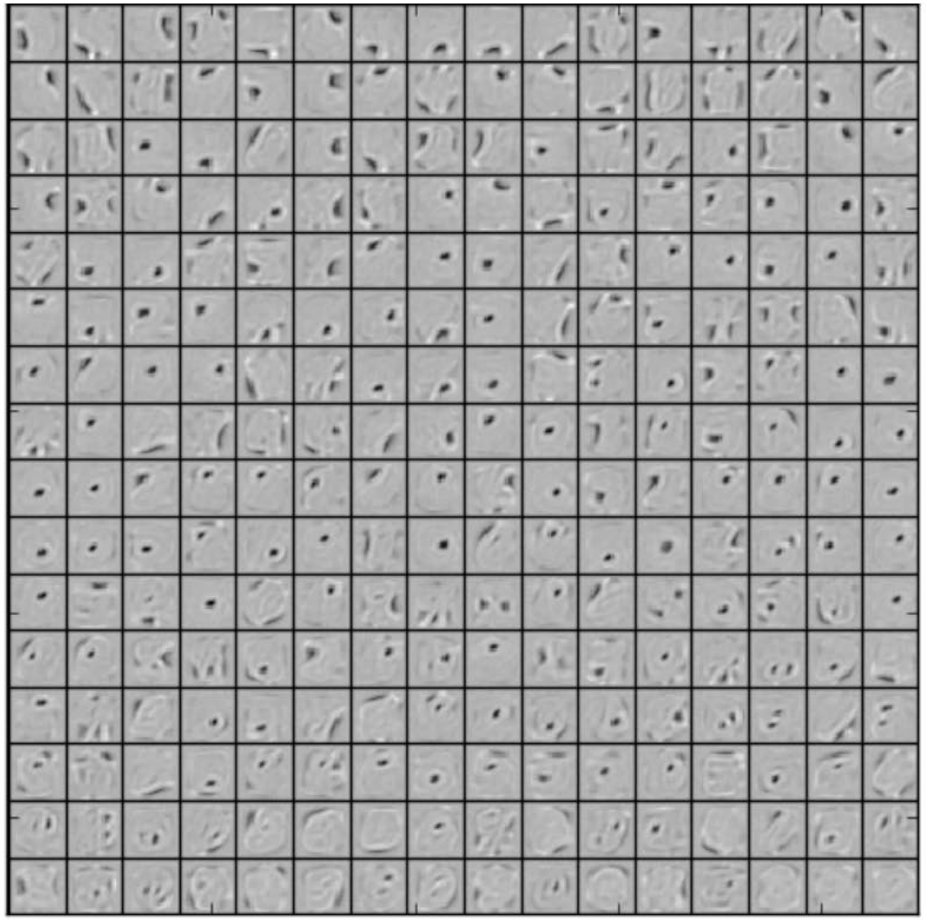
$$\|v\|_2 = \sqrt{\sum_{h=1}^{H} v_h^2}$$

- Added as a penalty term to the objective function
- Encourages "simpler" solutions with smaller Euclidean norm

Capacity Control: Dropout [Srivastava et al., 2014]



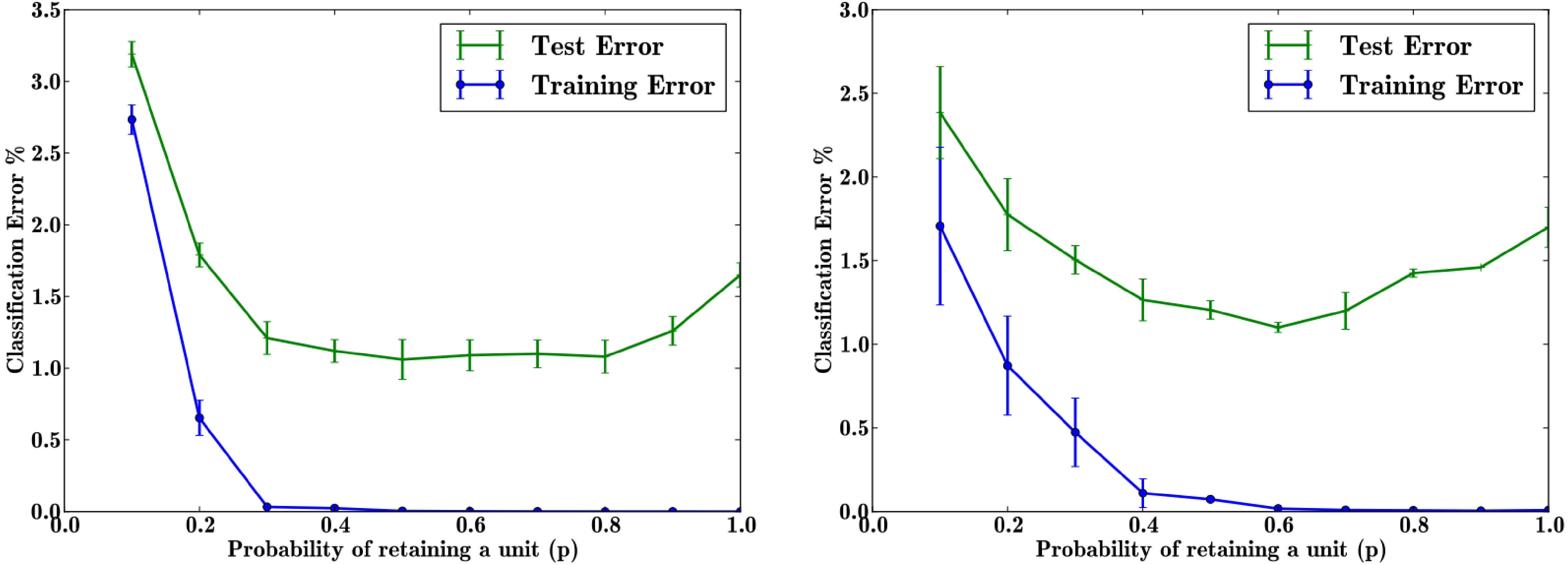
(a) Without dropout



(b) Dropout with $p = 0.5$.

Figure 7: Features learned on MNIST with one hidden layer autoencoders having 256 rectified linear units.

Capacity Control: Dropout [Srivastava et al., 2014]



(a) Keeping n fixed.

(b) Keeping pn fixed.

Figure 9: Effect of changing dropout rates on MNIST.

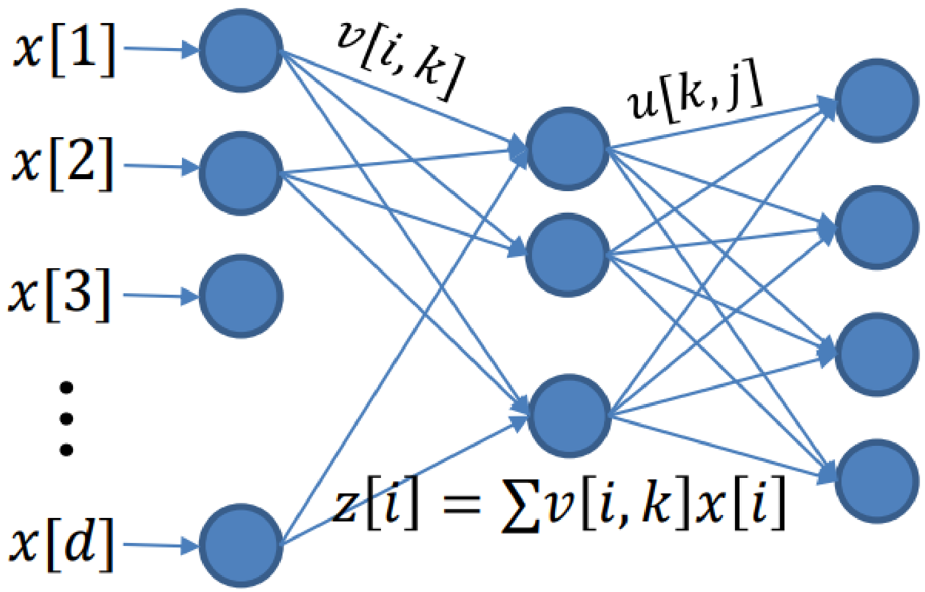
Motivation

- Why is it that we succeed in learning using multilayer feed-forward networks?

- Can we identify a property that makes them possible to learn?

- Is there some alternative inductive bias?

Network Size and Generalization: Methodology



Feed-forward network for **classification** task:

$$y[j] = \sum_{h=1}^{H} v_{hj} [\langle \boldsymbol{u}_h, \boldsymbol{x} \rangle]_+$$

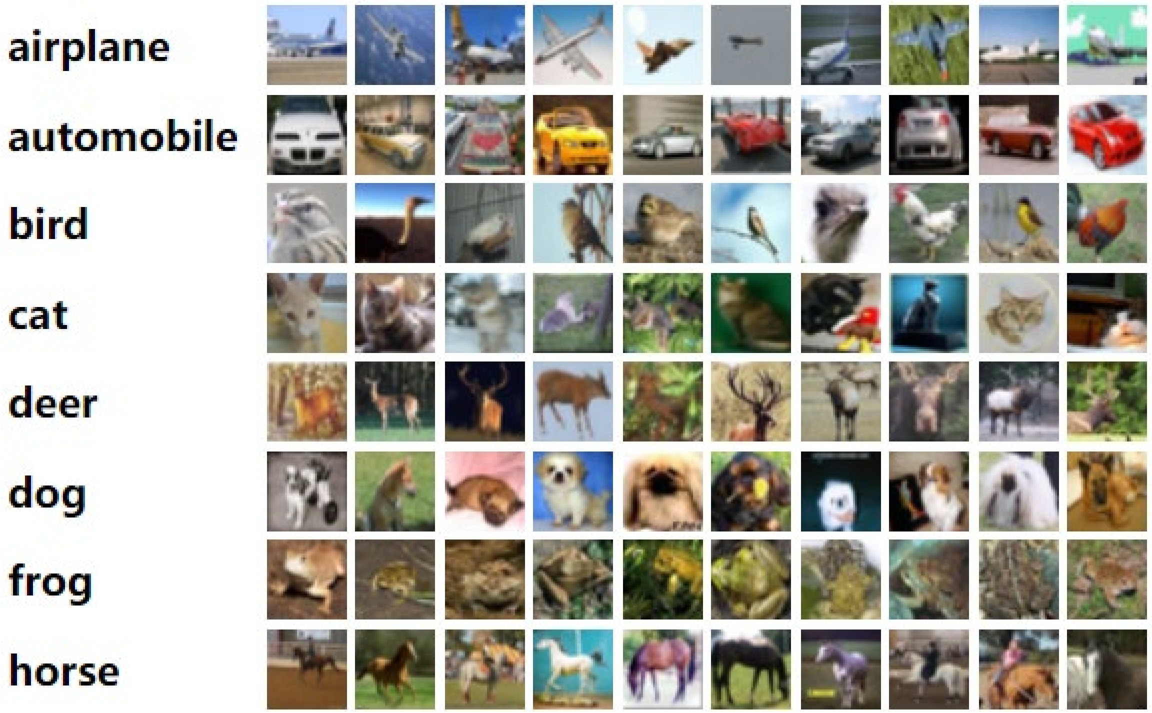
- d real-valued inputs $\boldsymbol{x} = (x[1], \ldots, x[d])$
- k outputs $y[1], \ldots, y[k]$
- a single hidden layer with H rectified linear units
  - $[z]_+ := \max(z, 0)$ as activation function
  - $\boldsymbol{u}_h \in \mathbb{R}^d$, $v_{hj} \in \mathbb{R}$, weights learned by minimizing soft-max cross entropy loss
  - #weights given by $H(d + k)$

Network Size and Generalization: Methodology

- Train networks for classification task
- Datasets

MNIST

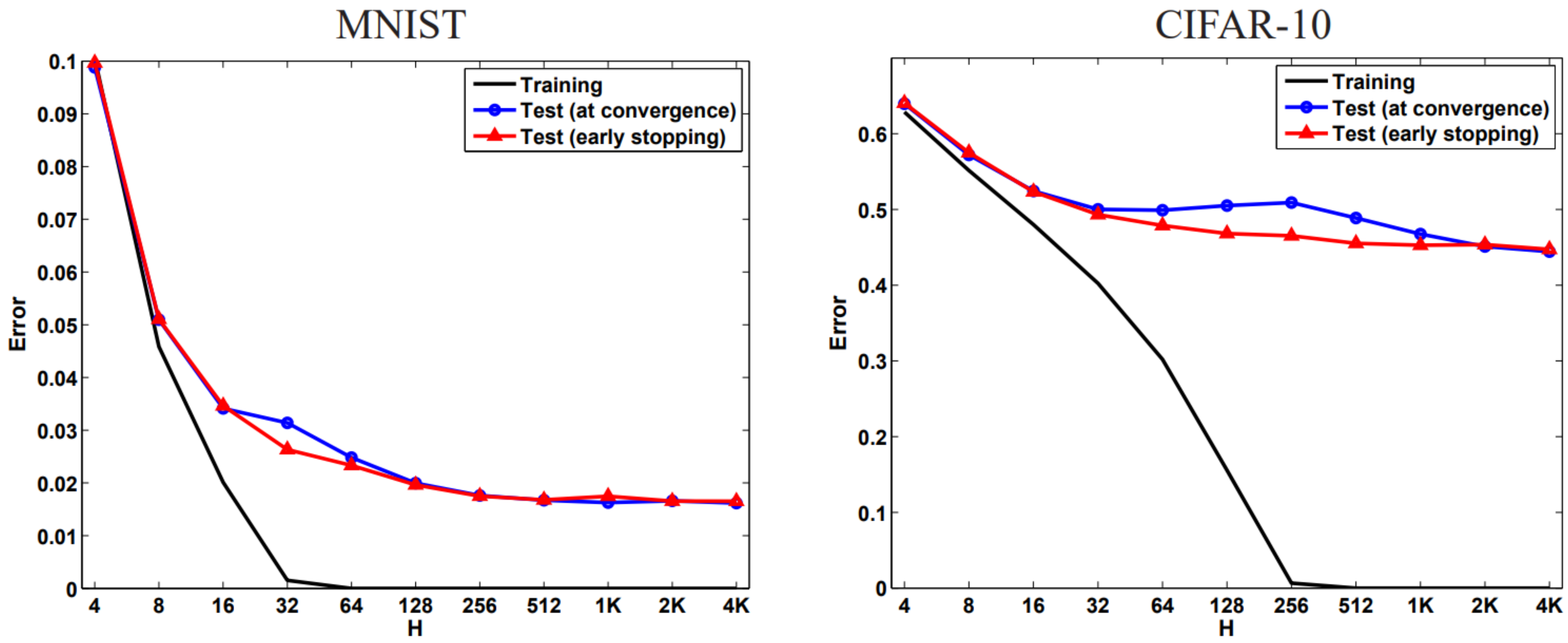
CIFAR-10



Network Size and Generalization: Methodology

- Train and test error based on two stopping criteria
  - Test set at convergence
  - Test set early stopping based on the error on validation set
- Optimization with stochastic gradient descent with momentum
- Standard learning schedule
- No explicit regularization
- Initialization from Gaussian distribution

Network Size and Generalization: Results



- Both training and test error initially decrease
- Without any regularization, even with zero training error, increasing the number of hidden units reduces estimation error

Network Size and Generalization: Results

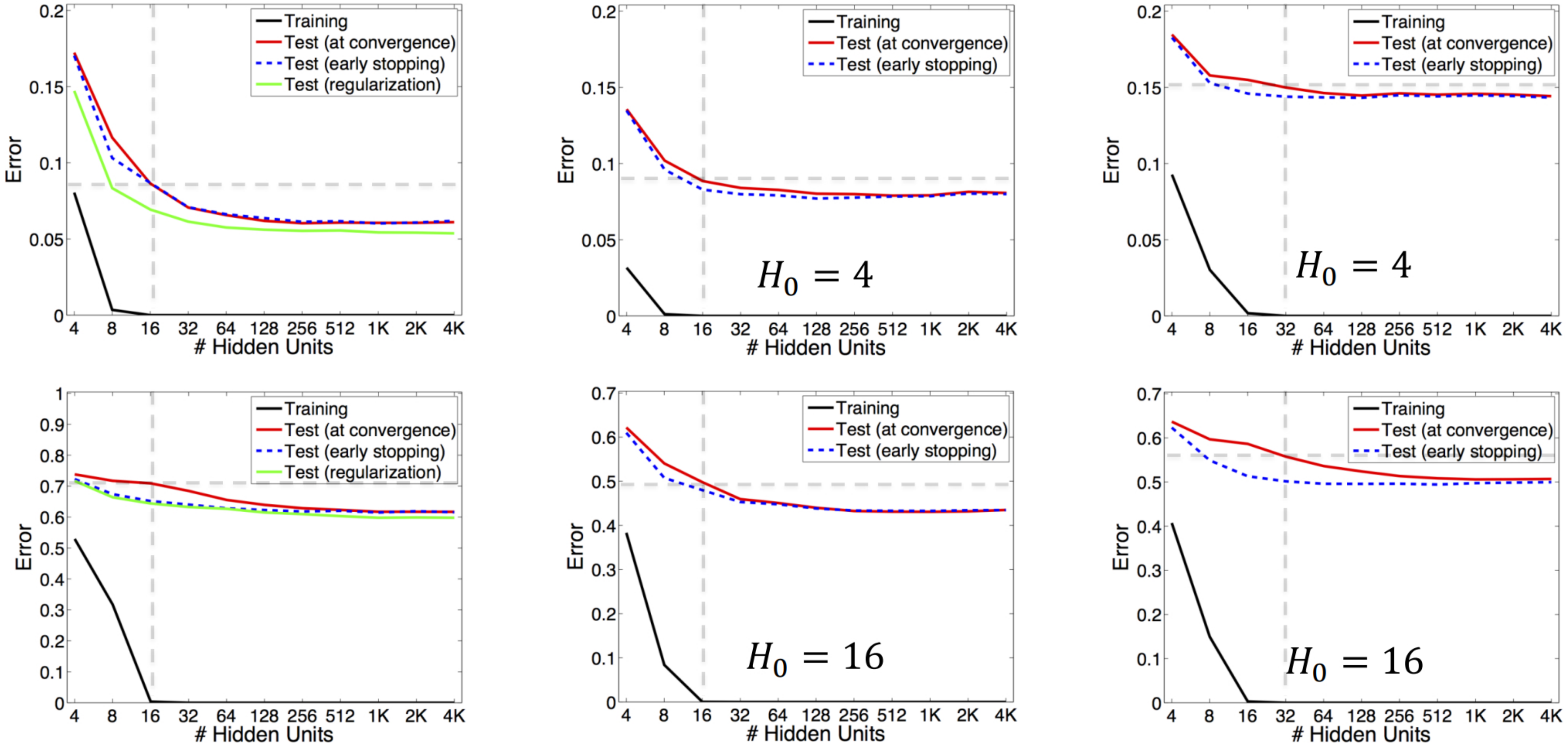
Reducing #Samples

Reducing #Samples + Censoring Labels

Reducing #Samples + Censoring Labels + Label Noise

MNIST

CIFAR-10



A Matrix Factorization Analogy

- Same feed-forward network, but now with linear activation function:

$$y[j] = \sum_{h=1}^{H} v_{hj} \left\langle \boldsymbol{u}_h, \boldsymbol{x} \right\rangle$$

- Matrix-factorization model:

$$\boldsymbol{y} = \boldsymbol{W}\boldsymbol{x} \quad \text{and} \quad \boldsymbol{W} = \boldsymbol{V}\boldsymbol{U}^\top$$

- Capacity control correspondence given by:

$$r \text{ hidden units} \Leftrightarrow rank(\mathbf{W}) \leq r$$

A Matrix Factorization Analogy

$$\|U\|_F := \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |u_{ij}|^2}$$

- Much success for learning with low norm factorizations

- Instead of constraining dimensionality H of $\boldsymbol{U}, \boldsymbol{V},$ we only regularize, their norm

- Trace-norm as inductive bias [Srebro et. al., 2004]:

$$\|\boldsymbol{W}\|_{tr} = \min_{\boldsymbol{W}=\boldsymbol{V}\boldsymbol{U}^\top} \frac{1}{2}(\|\boldsymbol{U}\|_F^2 + \|\boldsymbol{V}\|_F^2)$$

A Matrix Factorization Analogy

- Other norms of the factorization lead to different regularizers

- Unlike the rank, the trace-norm is convex, and leads tractable learning problems

- In summary:

Matrix Factorization	Low r: intractable	Trace-norm	Higher rank ⇒ lower trace-norm⇒better generalization
Feed-forward Networks	Low r: intractable	Some norm?	More hidden units⇒ lower norm⇒better generalization?

Related Works: Benign overfitting in linear regression, Bartlett et. al., 2020

- Benign overfitting: Phenomena where models almost interpolate the train data but still generalize well

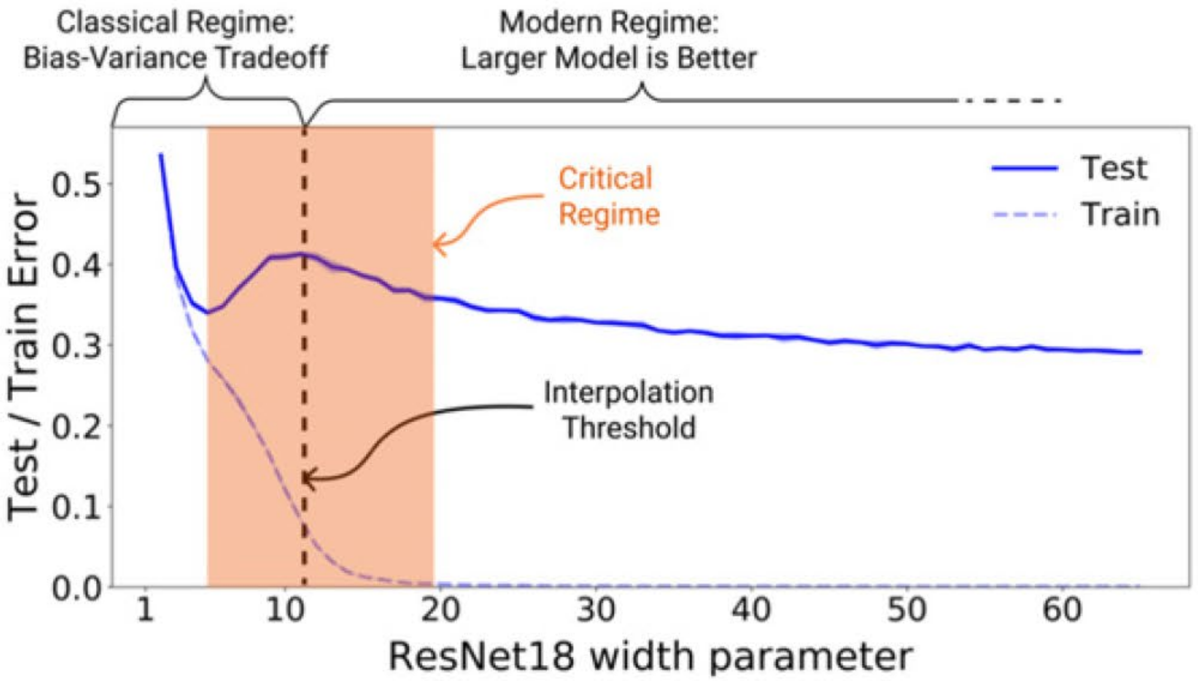
- Still an active area of research

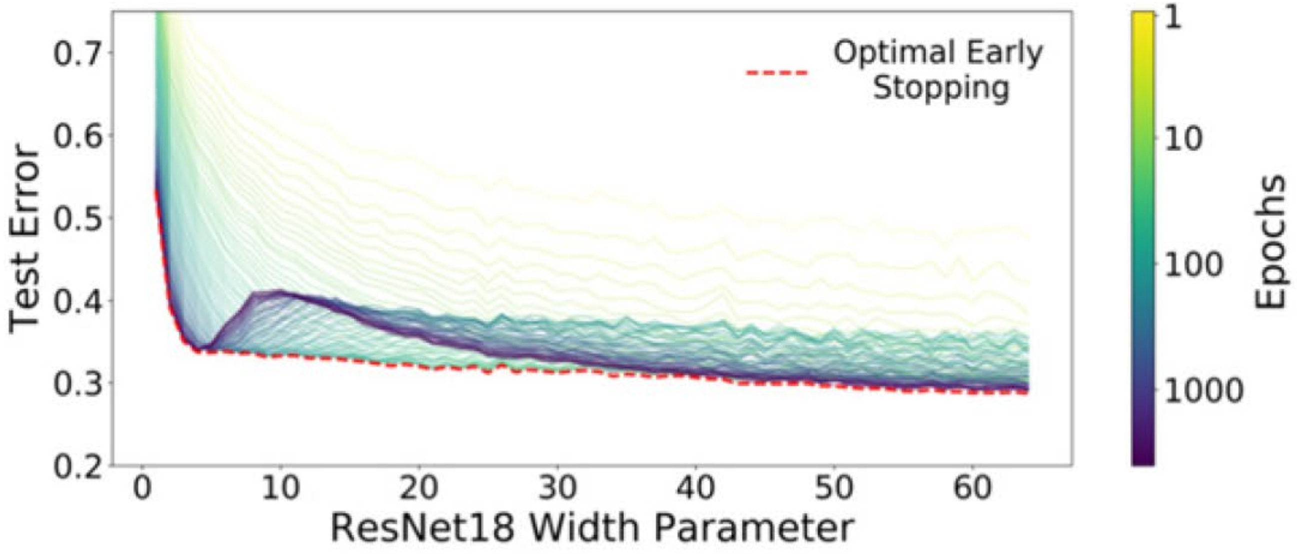
Related Works: Deep double descent: where bigger models and more data hurt, Nakkiran, et al., 2021

- Empirical evaluation of deep learning tasks exhibiting the double-descent phenomenon

- Double-descent not only as a function of model size **and** number of training epochs

Related Works: Deep double descent: where bigger models and more data hurt, Nakkiran, et al., 2021





Related Works: Multiple Descent: Design Your Own Generalization Curve, Chen et al., 2021

- Generalization curve can have an arbitrary number of peaks
- Peaks can be explicitly controlled
- Traditional bias-variance generalization curve and the double-descent phenomenon are not intrinsic properties of the model family

Related Works: Implicit Regularization in Deep Learning May Not Be Explainable by Norms, Razin et al., 2020

- Rather than perceiving the implicit regularization via norms, a potentially more useful interpretation is minimization of rank

- They hypothesize that it may be key to explaining generalization in deep learning

Related Works: Sensitivity and Generalization in Neural Networks: An Empirical Study, Novak et al., 2018

- Builds on top of the paper discussed today

- Similar study, but for Deep Neural Networks

Strengths

- To my knowledge, the first paper to introduce a notion of inductive bias, i.e. implicit regularization in the context of Machine Learning
- Optimization as inductive bias
- It tackles an important problem
- Corroborate their intuition with experiments
- Draw a matrix-factorization analogy which is much better understood

Weaknesses

- Some assumptions are unjustified
  - Do different optimization algorithms have a different inductive bias?
  - Might momentum be introducing some kind of bias?
  - Could subsampling of the dataset be a source of some implicit bias?

- Their methodology is not clearly communicated

- No explanation why we can draw a matrix factorization methodology by dropping non-linearities

Thank you for your attention.

Discussion/Questions?

References

- Nati Srebro, Algorithmic Bias in Underdetermined Optimization and Deep Learning, https://www.youtube.com/watch?v=7uRVR9hsF0g
- Poster of the presented paper: https://www.neyshabur.net/papers/inductive_bias_poster.pdf
- Image l1-regularization & l2-regularization: https://www.researchgate.net/figure/Parameter-norm-penalties-L2-norm-regularization-left-and-L1-norm-regularization_fig2_355020694
- Bias-Variance Tradeoff: IML 2022 & AML 2022 lecture slides
- Dropout: A Simple Way to Prevent Neural Networks from Overfitting, Srivastava et al.