On Mathematical Theories of Deep Learning

Yuan YAO

HKUST

What's wrong with deep learning?

Ali Rahimi NIPS'17: Machine (deep) Learning has become **alchemy**.
https://www.youtube.com/watch?v=ORHFOnaEzPc

Yann LeCun CVPR'15, invited talk: **What's wrong with deep learning?**
One important piece: **missing some theory**!

http://techtalks.tv/talks/whats-wrong-with-deep-learning/61639/

Being alchemy is certainly not a shame, not wanting to work on advancing to chemistry is a shame! -- **by Eric Xing**

Some Theoretical Problems

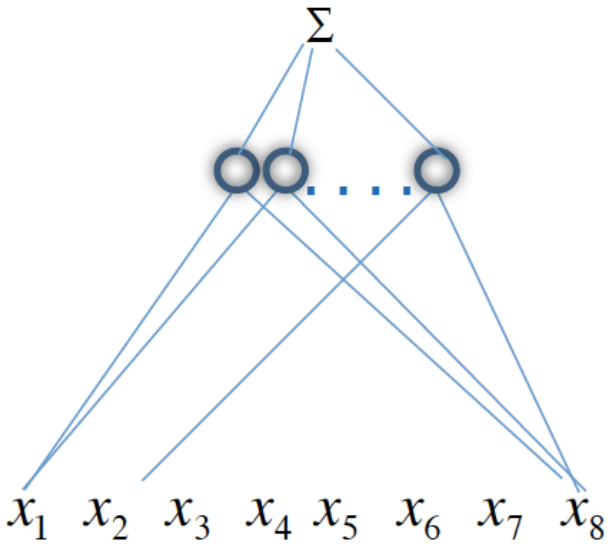
- *Approximation Theory and Harmonic Analysis* : **What functions are represented well by deep neural networks, without suffering the curse of dimensionality and better than shallow networks?**
  - Sparse (local), hierarchical (multiscale), compositional functions avoid the curse dimensionality
  - Group (translation, rotational, scaling, deformation) invariances achieved as depth grows
- *Statistics learning:* **How can deep learning generalize well without overfitting the noise?**
  - Over-parameterized models change nonseparable classification to separable, and maximize margin in gradient descent
- *Optimization:* **What is the landscape of the empirical risk and how to optimize it efficiently?**
  - Over-parameterized models make empirical risk landscapes simple (multilinear or 2-layer NN) with degenerate (flat) equilibria
  - SGD tends to find flat minima, and gradient-free algorithms like block-coordinate-descent

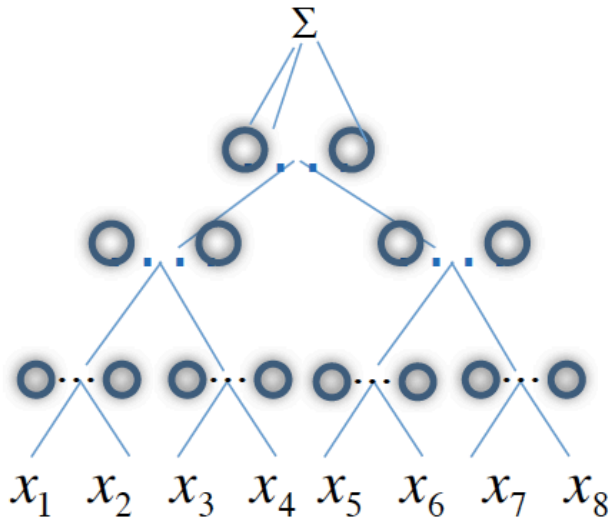
How Deep Learning avoids the Curse of Dimensionality

Locality or Sparsity does the job.

Deep and shallow networks: universality

Theorem *Shallow, one-hidden layer networks with a nonlinear $\phi(x)$ which is not a polynomial are universal. Arbitrarily deep networks with a nonlinear $\phi(x)$ (including polynomials) are universal.*





$$\phi(x) = \sum_{i=1}^{r} c_i \left| < w_i, x > + b_i \right|_+$$

Cybenko, Girosi, ….

Both deep and shallow models can approximate continuous functions, but suffering the curse of dimensionality… [Cybenko (1989), Hornik (1991), Barron (1993), Mhaskar (1994), Micchelli, Pinkus, Chui-Li]

Curse of Dimensionality

$$y = f(x_1, \ldots, x_d)$$

Curse of dimensionality

Both shallow and deep network can approximate a function of d variables equally well. The number of parameters in both cases depends exponentially on d as $O(\varepsilon^{-d})$.

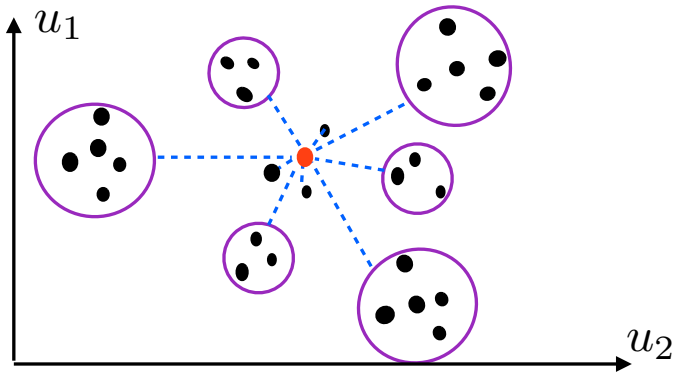
CENTER FOR Brains Minds + Machines

Mhaskar, Poggio, Liao, 2016

A Blessing from Physical world? Multiscale "compositional" sparsity

- Variables $x(u)$ indexed by a low-dimensional u: time/space... pixels in images, particles in physics, words in text...

- Mutliscale interactions of d variables:



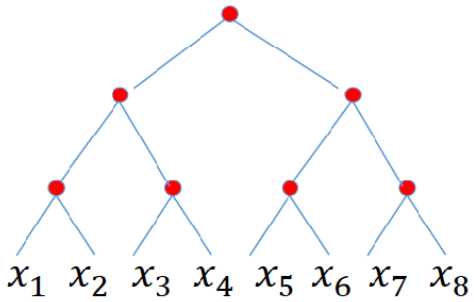
From d^2 interactions to $O(\log^2 d)$ multiscale interactions.

(Or even of constant numbers.)

- Multiscale analysis: wavelets on groups of symmetries. hierarchical architecture.

Hierarchically local compositionality

$$f(x_1, x_2, ..., x_8) = g_3(g_{21}(g_{11}(x_1, x_2), g_{12}(x_3, x_4))g_{22}(g_{11}(x_5, x_6), g_{12}(x_7, x_8)))$$



Theorem (informal statement)

Suppose that a function of d variables is hierarchically, locally, compositional . Both shallow and deep network can approximate f equally well. The number of parameters of the shallow network depends exponentially on d as $O(\varepsilon^{-d})$ with the dimension whereas for the deep network dance is $O(d\varepsilon^{-2})$

CENTER FOR Brains Minds + Machines

Mhaskar, Poggio, Liao, 2016

Convolutional Neural Networks (VGG, ResNet etc.) are of this type.

Important Special Cases in Statistics

Minimax rates of estimations (Stone 1982): if a regression function f is Lipschitz on $\mathbb{R}^d$ α with $0 < \alpha < 1$, then the optimal minimax rate of statistical regression estimators with N samples is $N^{-\frac{2\alpha}{2\alpha+d}}$.

Additive models (Stone 1985): $f(x_1, \ldots, x_d) = f_1(x_1) + \ldots + f_d(x_d)$ with minimax rate $N^{-\frac{2\alpha}{2\alpha+1}}$.
Raskutti-Wainwright-Yu (IEEE TIT, 2011), Yuan-Zhou (AoS, 2016), ...

Interaction models (Stone 1994): $f = \sum_{I \subseteq \{1,\ldots,d\}, |I|=d^*} f_I(x_I)$ with minimax rate $N^{-\frac{2\alpha}{2\alpha+d^*}}$. Here $d^* \in \{1, \ldots, d\}$ and for $I = \{i_1, \ldots, i_{d^*}\} \subseteq \{1, \ldots, d\}$ with $|I| = d^*$, $x_I = (x_{i_1}, \ldots, x_{i_{d^*}})$.

Single index models (Härdle and Stoker 1989): $f = g(a \cdot x)$ for some $a \in \mathbb{R}^d$ and $g : \mathbb{R} \to \mathbb{R}$

Projection pursuit (Friedman and Stuetzle 1981): $f(x_1, \ldots, x_d) = \sum_{k=1}^{K} g_k(a_k \cdot x)$ with $K \in \mathbb{N}$, $a_k \in \mathbb{R}^d$ and univariate functions g_k

Hierarchical interaction models (Kohler1 and Krzyzak 2016)

Simple case: $f = g(f_1(x_{I_1}), f_2(x_{I_2}), \ldots, f_{d^*}(x_{I_{d^*}}))$

Generalized hierarchical model: $f = g(a_1 \cdot x, \ldots, a_{d^*} \cdot x)$

Generalized hierarchical interaction model: $f = \sum_k g_k(f_{1,k}, \ldots, f_{d^*,k})$ with $f_{i,k}(x)$ generalized hierarchical model

Some Historical Results

- A classical **theorem [Sipser, 1986; Hastad, 1987]** shows that deep circuits are more efficient in representing certain Boolean functions than shallow circuits. Hastad proved that highly-variable functions (in the sense of having high frequencies in their Fourier spectrum) in particular the parity function cannot even be decently approximated by small constant depth circuits
- **Chui-Li-Mhaskar (1994)** shows that multilayer networks can do localized approximation while single layer ones can not. Older examples exist: consider a function which is a linear combination of n tensor product Chui–Wang spline wavelets, where each wavelet is a tensor product cubic spline. It was shown by **Chui and Mhaskar** that is impossible to implement such a function using a shallow neural network with a sigmoidal activation function using O(n) neurons, but a deep network with the activation function $(x_+)^2$ do so. In this case, as we mentioned, there is a formal proof of a gap between deep and shallow networks.
- The main result of **[Telgarsky, 2016, Colt]** says that there are functions with many oscillations that cannot be represented by shallow networks with linear complexity but can be represented with low complexity by deep networks.
- **Eldan and Shamir (2016)** show an example of a function expressible by a 3-layer feedforward neural network cannot be approximated by any 2-layer neural network to certain accuracy unless the width is exponential in the dimension.
- **Shaham-Cloningen-Coifman (2018)**: functions on manifolds and order of approximation by fully connected deep neural networks

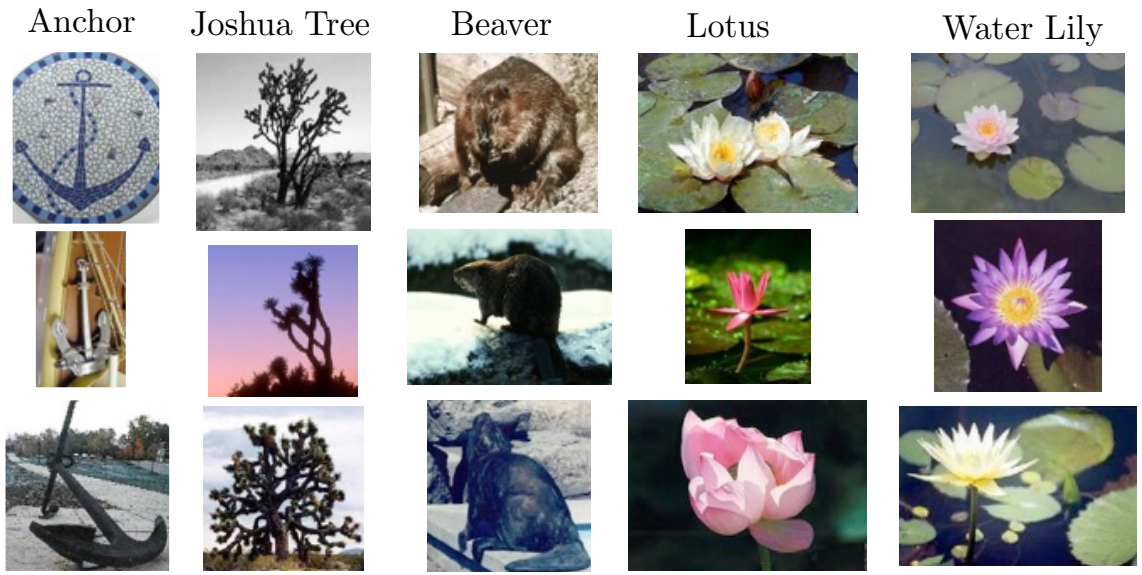
What are the geometric properties of deep networks?

Contraction within-class level sets toward invariants when depth grows, while keeping the separation between classes

High Dimensional Natural Image Classification

- High-dimensional $x = (x(1), ..., x(d)) \in \mathbb{R}^d$:
- **Classification:** estimate a class label $f(x)$ given n sample values $\{x_i\,,\, y_i = f(x_i)\}_{i \leq n}$

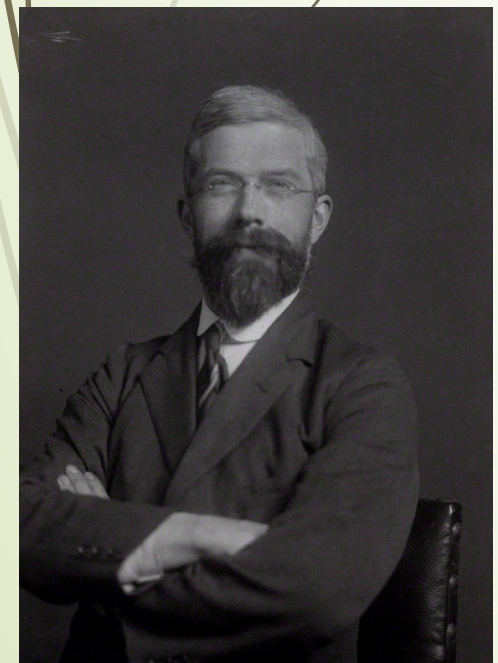
Image Classification $d = 10^6$



Huge variability inside classes

Find invariants

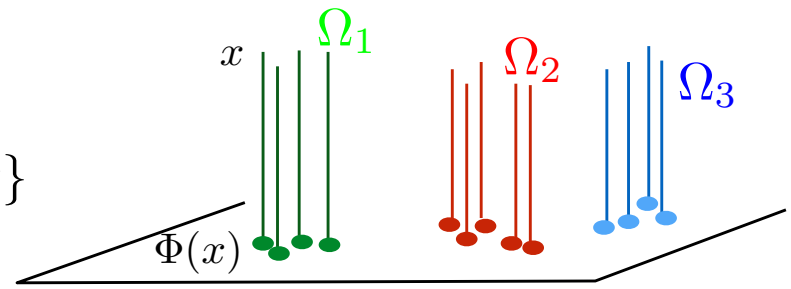
Fisher's Linear Discriminant (1936) (Linear Dimensionality Reduction)



Classes

Level sets of $f(x)$

$$\Omega_t = \{x \ : \ f(x) = t\}$$



If level sets (classes) are parallel to a linear space then variables are eliminated by linear projections: *invariants*.

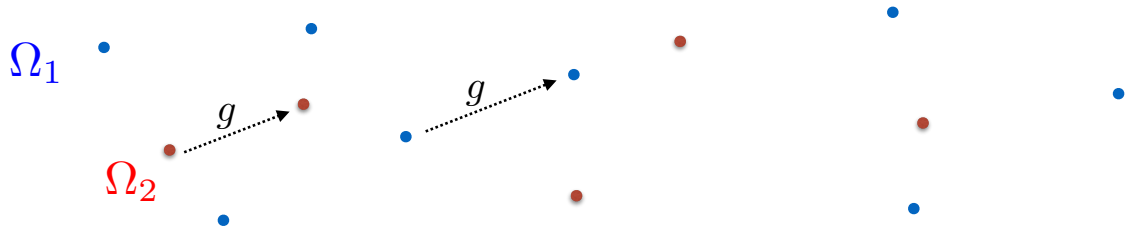
$$\Phi(x) = \alpha \hat{\Sigma}_W^{-1}(\hat{\mu}_1 - \hat{\mu}_0)$$

$$\hat{\mu}_k = \frac{1}{|C_k|} \sum_{i \in C_k} x_i \qquad \hat{\Sigma}_W = \sum_k \sum_{i \in C_k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$$

Nonlinear Level Set Group Symmetries

Level Set Geometry: Symmetries

- Curse of dimensionality $\Rightarrow$ not local but global geometry
  Level sets: classes, characterised by their global symmetries.



- A symmetry is an operator g which preserves level sets:

$$\forall x \quad , \quad f(g.x) = f(x) \; : \; \text{global}$$

If g_1 and g_2 are symmetries then $g_1.g_2$ is also a symmetry

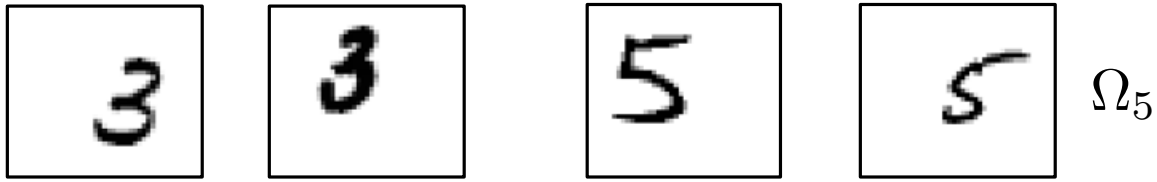
$$f(g_1.g_2.x) = f(g_2.x) = f(x)$$

Level set symmetries lead to groups…

Translation and Deformations

- Digit classification:

$x(u)$ $\quad x'(u) = x(u - \tau(u))$

Ω_3  Ω_5

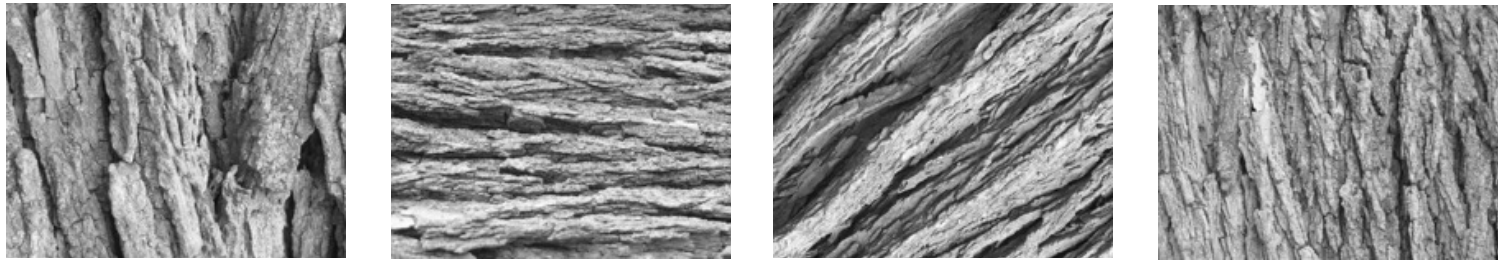
- Globally invariant to the translation group: small
- Locally invariant to small diffeomorphisms: huge group

Video of Philipp Scott Johnson

https://www.youtube.com/watch?v=nUDIoN-_Hxs

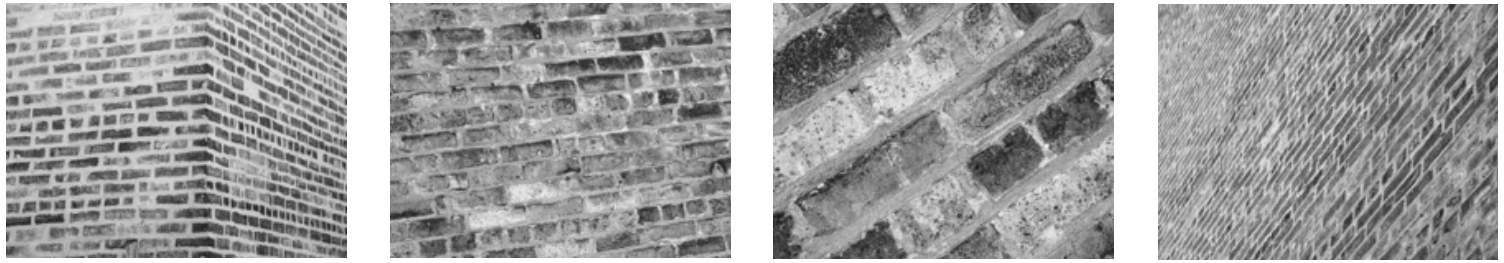
Rotation and Scaling Variability

- Rotation and deformations



Group: $SO(2) \times \mathrm{Diff}(SO(2))$

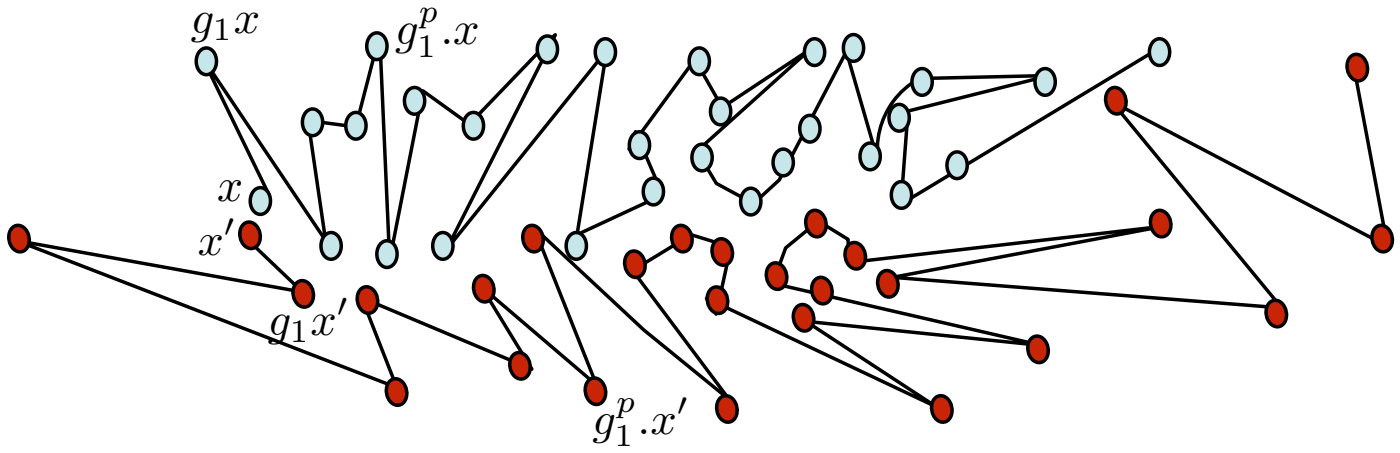
- Scaling and deformations



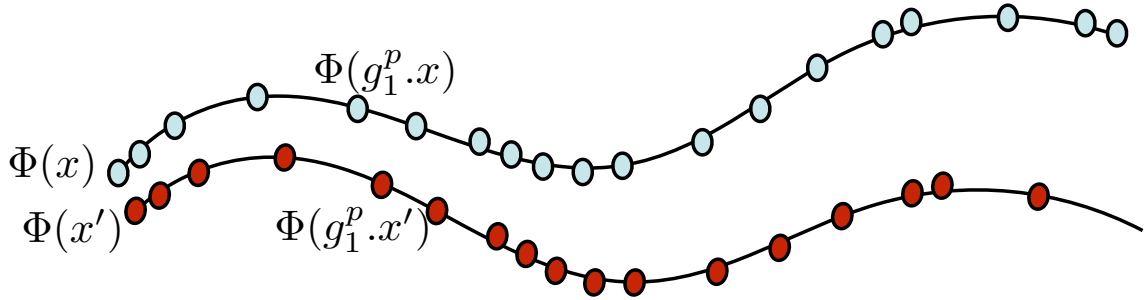
Group: $\mathbb{R} \times \mathrm{Diff}(\mathbb{R})$

Linearize Symmetries

- A change of variable $\Phi(x)$ must linearize the orbits $\{g.x\}_{g \in G}$



- Linearise symmetries with a change of variable $\Phi(x)$



- Lipschitz: $\forall x, g \ : \ \|\Phi(x) - \Phi(g.x)\| \leq C \|g\|$

Wavelet Scattering Net

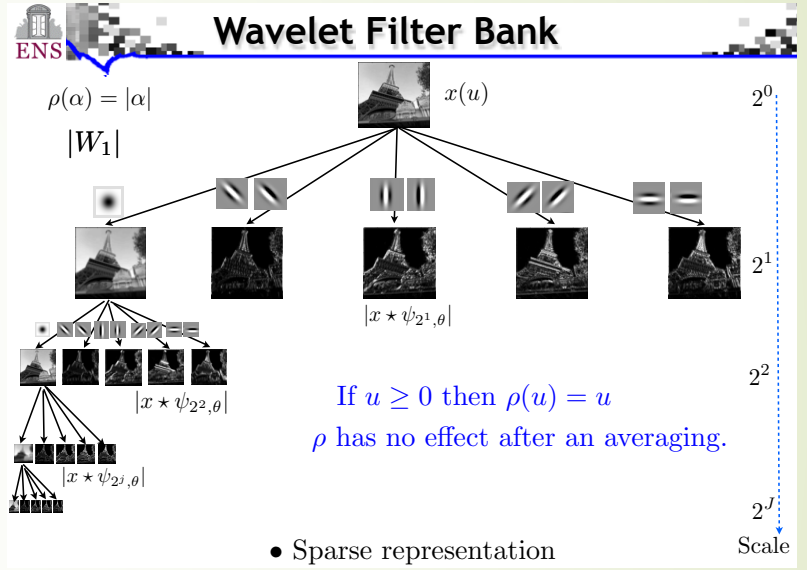
Stephane Mallat et al. 2012

- Architechture:
  - Convolutional filters: band-limited complex wavelets
  - Nonlinear activation: modulus (Lipschitz)
  - Pooling: averaging (L1)
- Properties:
  - A Multiscale Sparse Representation
  - Norm Preservation (Parseval's identity):

$$\|Sx\| = \|x\|$$

  - Contraction:

$$\|Sx - Sy\| \leq \|x - y\|$$



$$Sx = \begin{pmatrix} x \star \phi\,(u) \\ |x \star \psi_{\lambda_1}| \star \phi(u) \\ ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(u) \\ |||x \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}| \star \phi(u) \\ \cdots \end{pmatrix}_{u,\lambda_1,\lambda_2,\lambda_3,\ldots}$$

Invariants/Stability of Scattering Net

- **Translation Invariance** (generalized to **rotation** and **scaling**):

  - The average $|x \star \psi_{\lambda_1}| \star \phi(t)$ is invariant to small translations relatively to the support of ϕ.

  - Full translation invariance at the limit:

$$\lim_{\phi \to 1} |x \star \psi_{\lambda_1}| \star \phi(t) = \int |x \star \psi_{\lambda_1}(u)| \, du = \|x \star \psi_{\lambda_1}\|_1$$

- Stable Small Deformations:

$$stable\ to\ deformations\ \ x_\tau(t) = x(t - \tau(t))$$

$$\|Sx - Sx_\tau\| \leq C \sup_t |\nabla \tau(t)| \, \|x\|$$

Wiatowski-Bolcskei'15

- Scattering Net by Mallat et al. so far
  - Wavelet Linear filter
  - Nonlinear activation by modulus
  - Average pooling
- Generalization by Wiatowski-Bolcskei'15
  - Filters as frames
  - Lipschitz continuous Nonlinearities
  - General Pooling: Max/Average/Nonlinear, etc.
  - As depth grows, the multiplicative pooling factors leads to full invariances.

Filters: Semi-discrete frame $\Psi_n := \{\chi_n\} \cup \{g_{\lambda_n}\}_{\lambda_n \in \Lambda_n}$

$$A_n \|f\|_2^2 \leq \|f * \chi_n\|_2^2 + \sum_{\lambda_n \in \Lambda_n} \|f * g_{\lambda_n}\|^2 \leq B_n \|f\|_2^2, \quad \forall f \in L^2(\mathbb{R}^d)$$

Pooling: In continuous-time according to

$$f \mapsto S_n^{d/2} P_n(f)(S_n \cdot),$$

where $S_n \geq 1$ is the **pooling factor** and $P_n : L^2(\mathbb{R}^d) \to L^2(\mathbb{R}^d)$ is R_n-Lipschitz-continuous

Assume that the filters, non-linearities, and poolings satisfy

$$B_n \leq \min\{1, L_n^{-2} R_n^{-2}\}, \quad \forall n \in \mathbb{N}.$$

Let the pooling factors be $S_n \geq 1$, $n \in \mathbb{N}$. Then,

$$|||\Phi^n(T_t f) - \Phi^n(f)||| = \mathcal{O}\left(\frac{\|t\|}{S_1 \dots S_n}\right),$$

for all $f \in L^2(\mathbb{R}^d)$, $t \in \mathbb{R}^d$, $n \in \mathbb{N}$.

Summary

- All these works partially explains the success of CNNs
  - Contraction within level set symmetries toward invariance when depth grows (invariants)
  - Separation kept between different levels (discriminant)
- Other questions?
  - How deep networks generalize well without overfitting?
  - What's the landscape of empirical risks and how to efficiently optimize?

Generalization Ability

Over-parameterized models generalize well without overfitting by maximizing margins

Generalization Error

- Consider the empirical risk minimization under i.i.d. samples

$$\hat{R}_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i;\theta)) + \mathcal{R}(\theta)$$

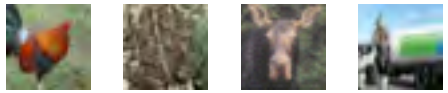
- The population risk with respect to unknown distribution

$$R(\theta) = \mathbf{E}_{x,y\sim P}\ell(y, f(x;\theta))$$

- Fundamental Theorem of Machine Learning (for 0-1 misclassification loss, called 'errors' below)

$$R(\theta) = \underbrace{\hat{R}_n(\theta)}_{\text{training loss/error}} + \underbrace{R(\theta) - \hat{R}_n(\theta)}_{\text{generalization loss/error}}$$

Why big models generalize well?



CIFAR10

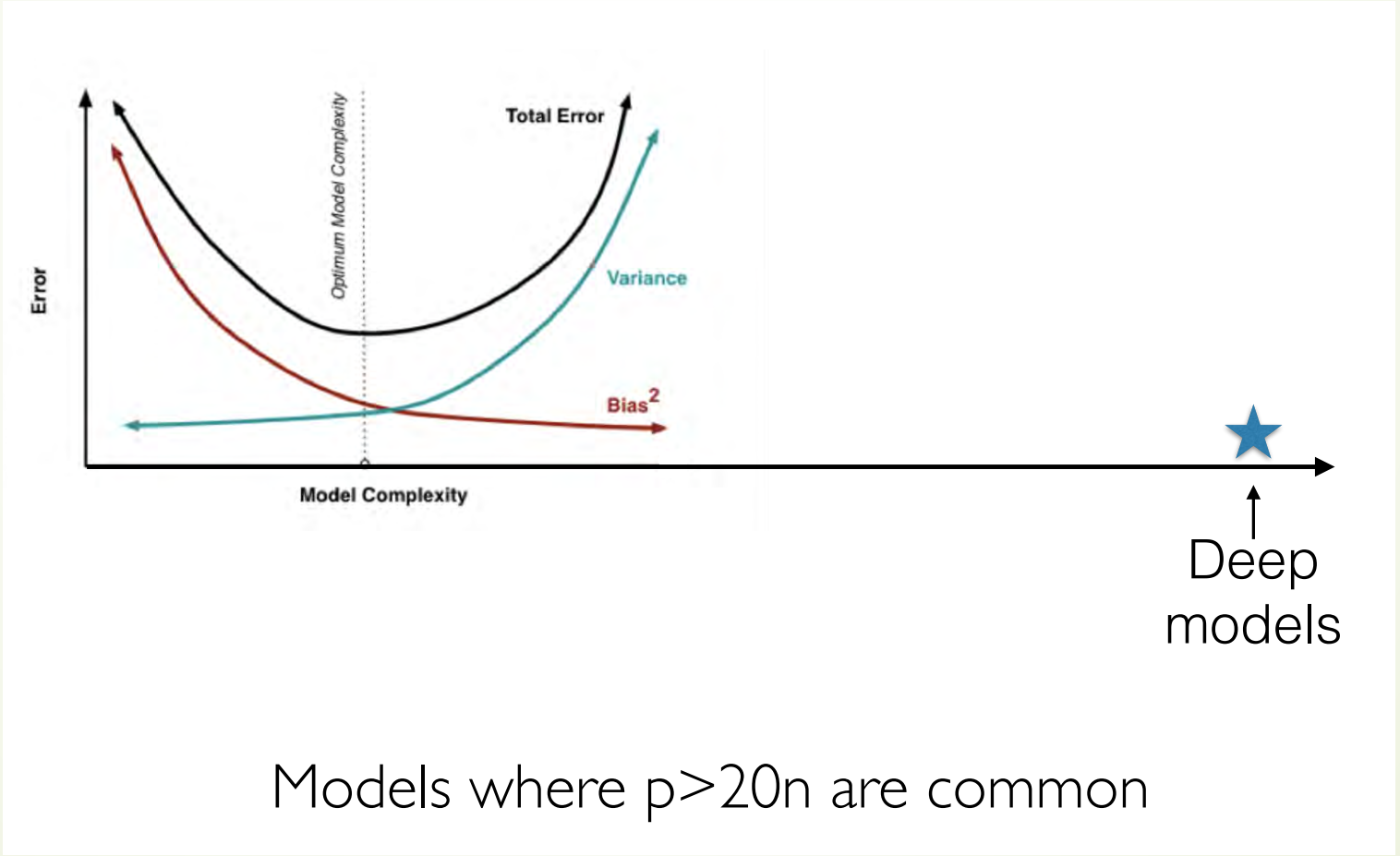
n=50,000
d=3,072
k=10

What happens when I turn off the regularizers?

Model	parameters	p/n	Train ***loss***	Test ***error***
CudaConvNet	145,578	2.9	0	23%
CudaConvNet (with regularization)	145,578	2.9	0.34	18%
MicroInception	1,649,402	33	0	14%
ResNet	2,401,440	48	0	13%

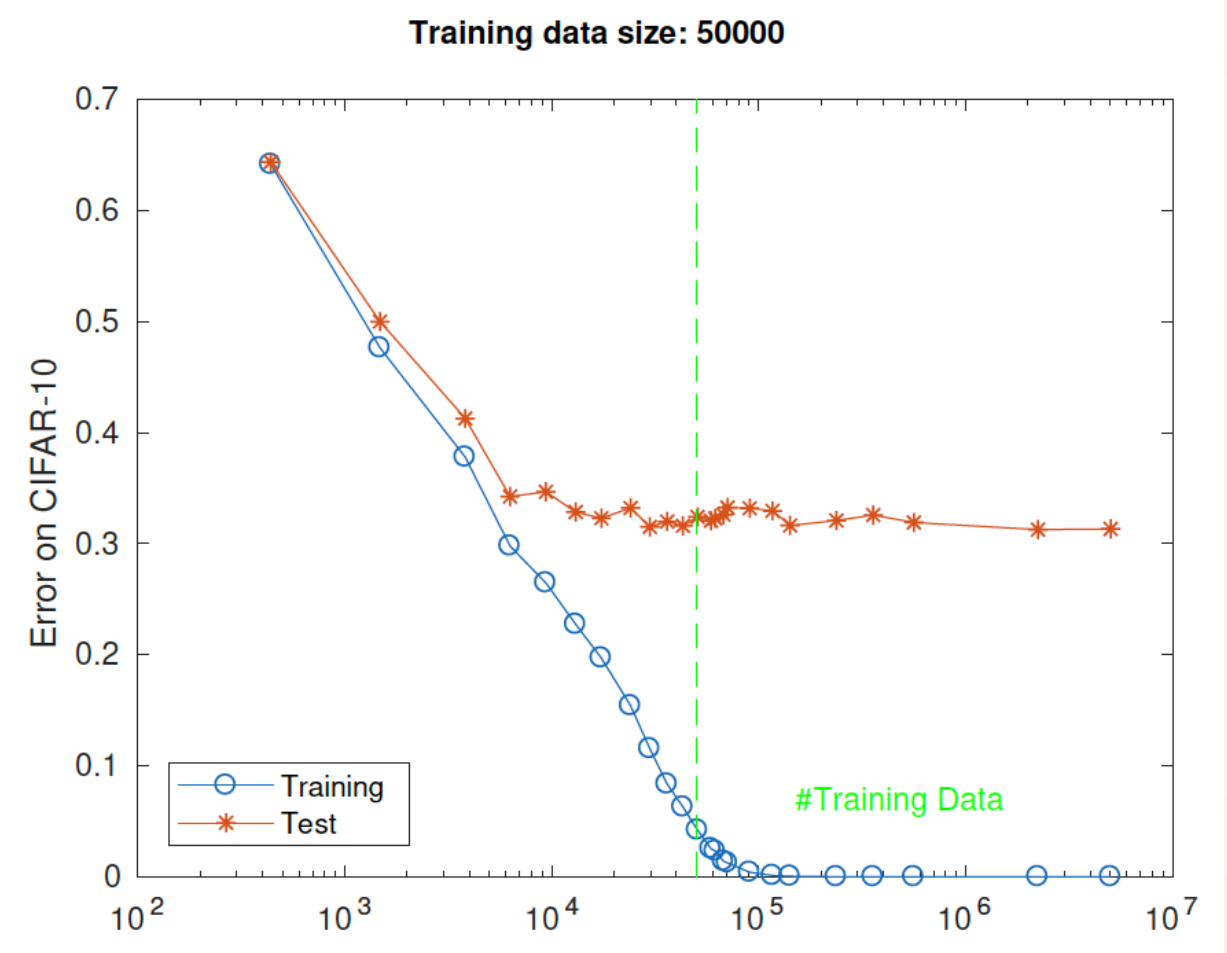
Ben Recht et al. 2016

The Bias-Variance Tradeoff?



Models where p>20n are common

Over-parameterized models

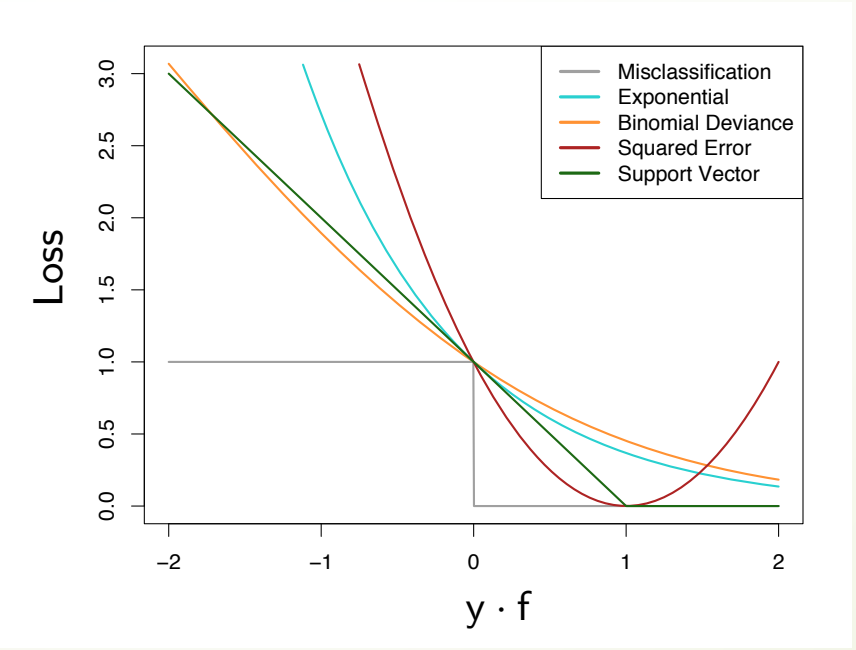


As model complexity grows (p>n), training error goes down to zero, but test error does not increase. Why overparameterized models do not overfit here? -- Tommy Poggio, 2018

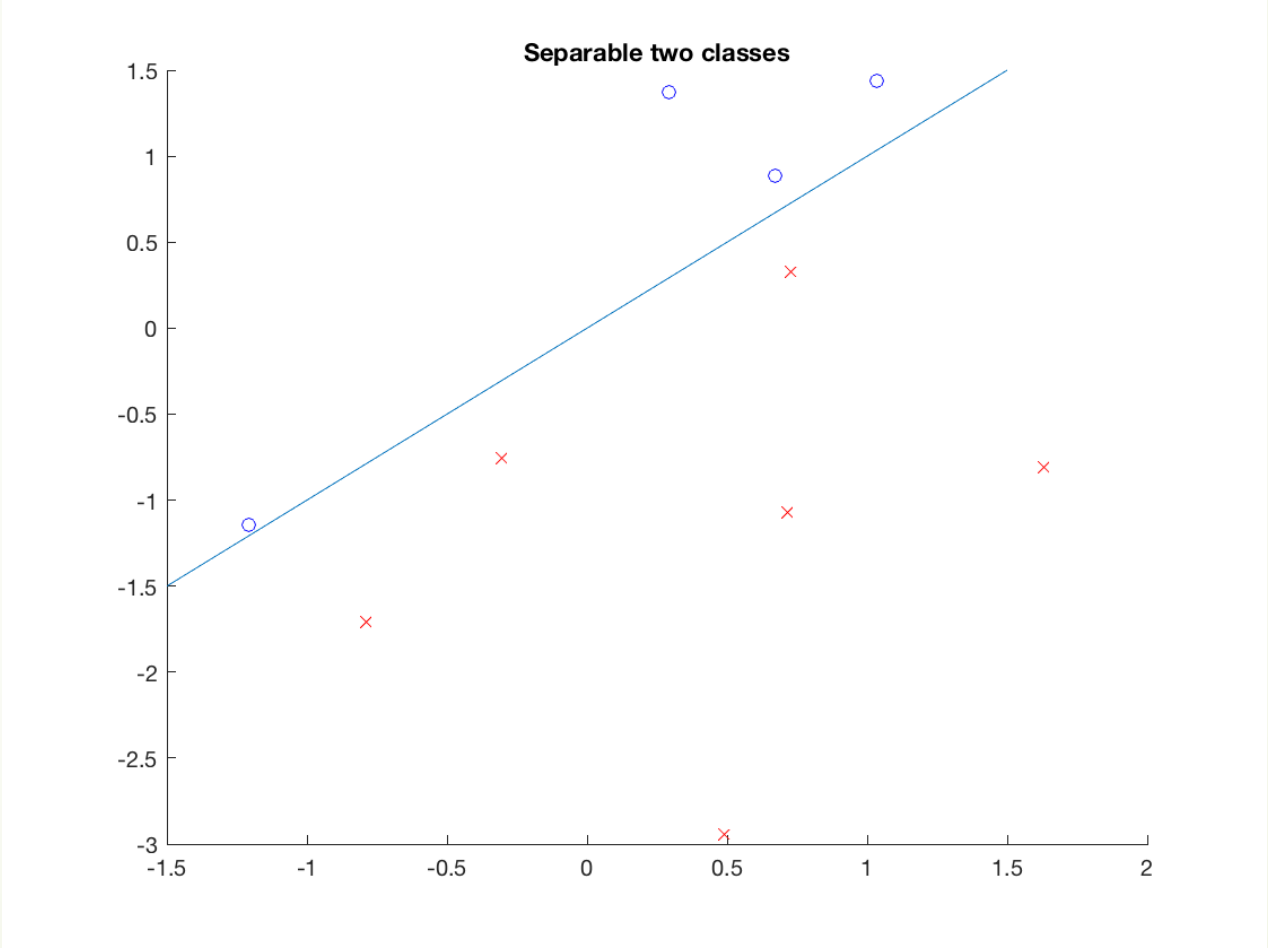
Binary Classification Problem

Consider a dataset $\{\mathbf{x}_n, y_n\}_{n=1}^N$, with $\mathbf{x}_n \in \mathbb{R}^d$ and binary labels $y_n \in \{-1, 1\}$. We analyze learning by minimizing an empirical loss of the form

$$\mathcal{L}(\mathbf{w}) = \sum_{n=1}^{N} \ell\left(y_n \mathbf{w}^\top \mathbf{x}_n\right) . \tag{1}$$



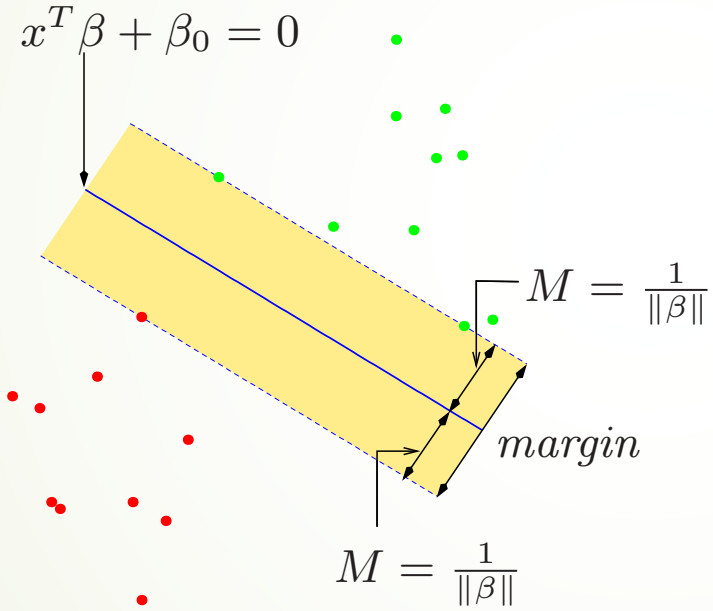
Separable Classification



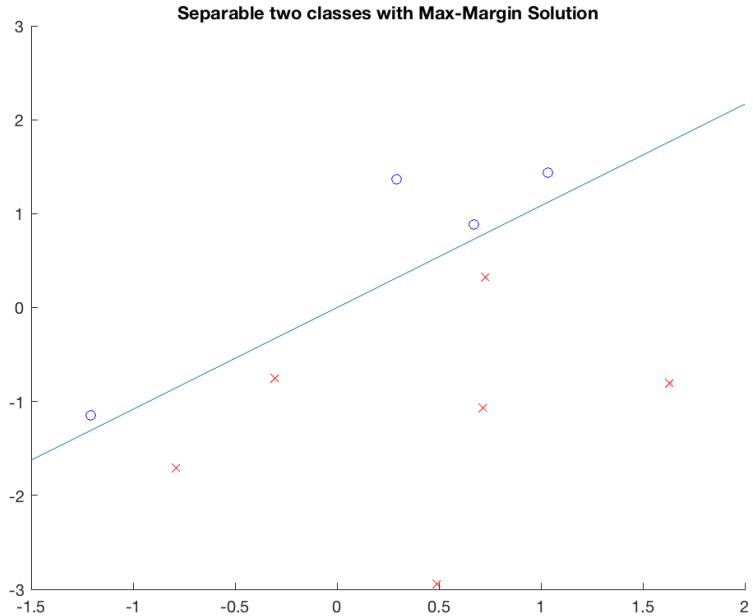
Max-Margin Classifier (SVM)

$$\text{minimize}_{\beta_0,\beta_1,\ldots,\beta_p} \|\beta\|^2 := \sum_j \beta_j^2$$

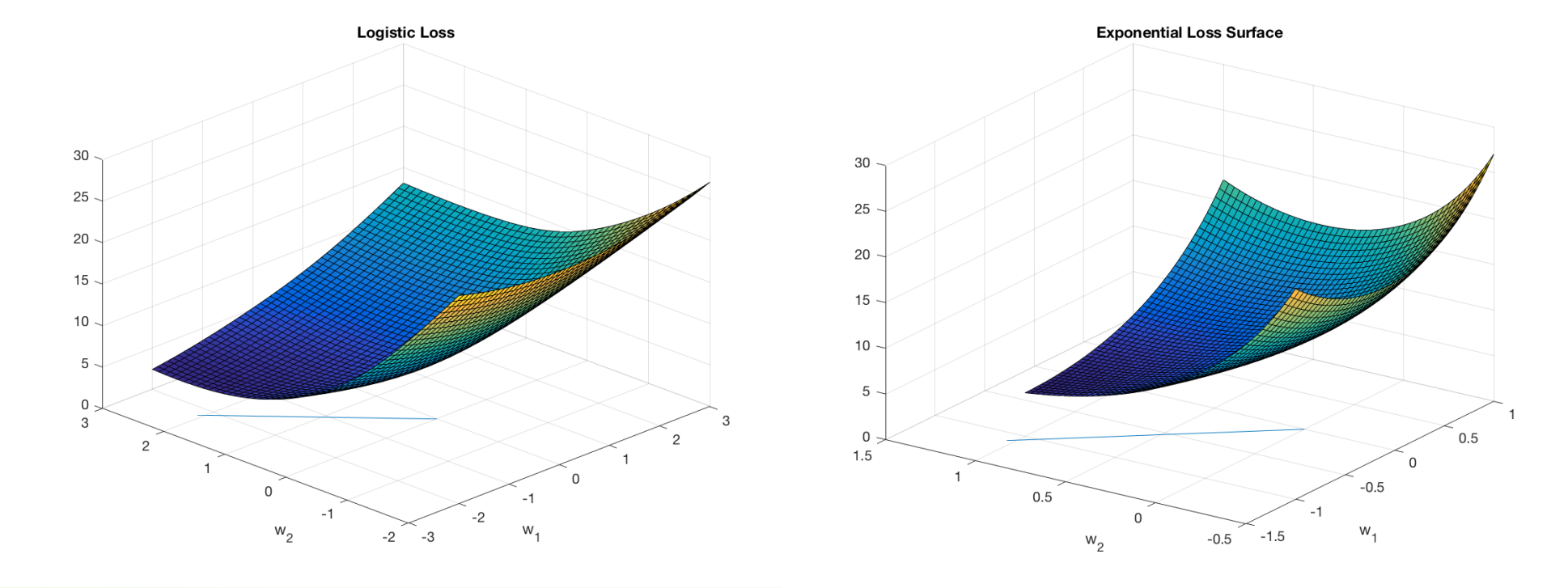
$$\text{subject to } y_i(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip}) \geq 1 \text{ for all } i$$



Vladmir Vapnik, 1994



Landscape of Logistic/Exponential Loss



The minimizers are at infinity, asymptotically in the direction of max-margin classifier

[Soudry, Hoffer, Nacson, Gunasekar, Srebro, 2017]

Theorem 3 *For any dataset which is linearly separable (Assumption 1), any β-smooth decreasing loss function (Assumption 2) with an exponential tail (Assumption 3), any stepsize $\eta < 2\beta^{-1}\sigma_{\max}^{-2}(\mathbf{X})$ and any starting point $\mathbf{w}(0)$, the gradient descent iterates (as in eq. 2) will behave as:*

$$\mathbf{w}(t) = \hat{\mathbf{w}} \log t + \boldsymbol{\rho}(t) , \tag{3}$$

where $\hat{\mathbf{w}}$ is the L_2 max margin vector (the solution to the hard margin SVM):

$$\hat{\mathbf{w}} = \underset{\mathbf{w}\in\mathbb{R}^d}{\operatorname{argmin}} \|\mathbf{w}\|^2 \text{ s.t. } \mathbf{w}^\top \mathbf{x}_n \geq 1, \tag{4}$$

and the residual grows at most as $\|\boldsymbol{\rho}(t)\| = O(\log\log(t))$, and so

$$\lim_{t\to\infty} \frac{\mathbf{w}(t)}{\|\mathbf{w}(t)\|} = \frac{\hat{\mathbf{w}}}{\|\hat{\mathbf{w}}\|}.$$

Furthermore, for almost all data sets (all except measure zero), the residual $\rho(t)$ is bounded.

Assumptions on General Loss Functions

Assumption 1 *The dataset is linearly separable:* $\exists \mathbf{w}_*$ *such that* $\forall n: \mathbf{w}_*^\top \mathbf{x}_n > 0$.

Assumption 2 $\ell(u)$ *is a positive, differentiable, monotonically decreasing to zero*[1], *(so* $\forall u: \ell(u) > 0, \ell'(u) < 0$ *and* $\lim_{u\to\infty} \ell(u) = \lim_{u\to\infty} \ell'(u) = 0$*) and a* β*-smooth function,* i.e. *its derivative is* β*-Lipshitz.*

Assumption 2 includes many common loss functions, including the logistic, exp-loss[2], probit and sigmoidal losses. Assumption 2 implies that $\mathcal{L}(\mathbf{w})$ is a $\beta\sigma_{\max}^2(\mathbf{X})$-smooth function, where $\sigma_{\max}(\mathbf{X})$ is the maximal singular value of the data matrix $\mathbf{X} \in \mathbb{R}^{d\times N}$.

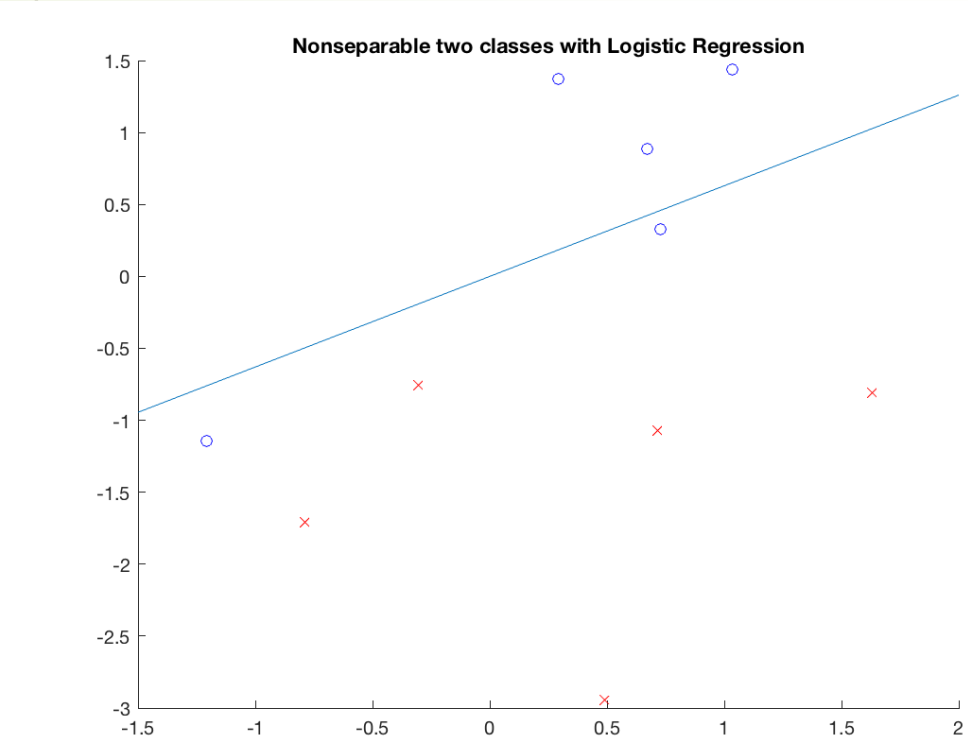
Definition 2 *A function* $f(u)$ *has a "tight exponential tail", if there exist positive constants* c, a, μ_+, μ_-, u_+ *and* u_- *such that*

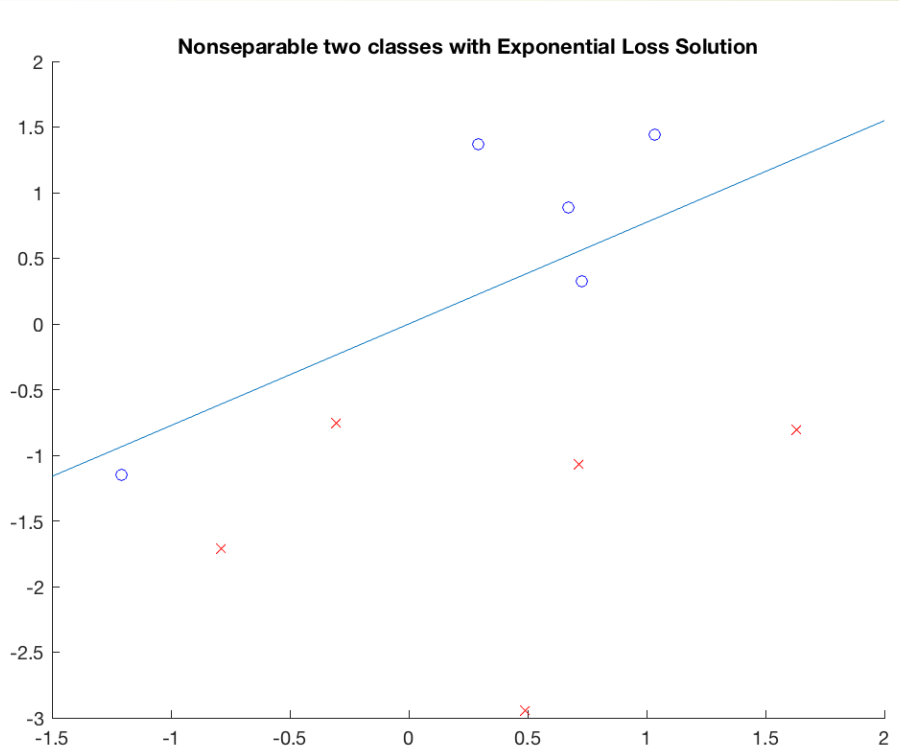
$$\forall u > u_+ : f(u) \le c\left(1 + \exp\left(-\mu_+ u\right)\right) e^{-au}$$
$$\forall u > u_- : f(u) \ge c\left(1 - \exp\left(-\mu_- u\right)\right) e^{-au}.$$

Assumption 3 *The negative loss derivative* $-\ell'(u)$ *has a tight exponential tail (Definition 2).*

For example, the exponential loss $\ell(u) = e^{-u}$ and the commonly used logistic loss $\ell(u) = \log\left(1 + e^{-u}\right)$ both follow this assumption with $a = c = 1$. We will assume $a = c = 1$ — without loss of generality, since these constants can be always absorbed by re-scaling $\mathbf{x}_n$ and η.

Nonseparable classification?

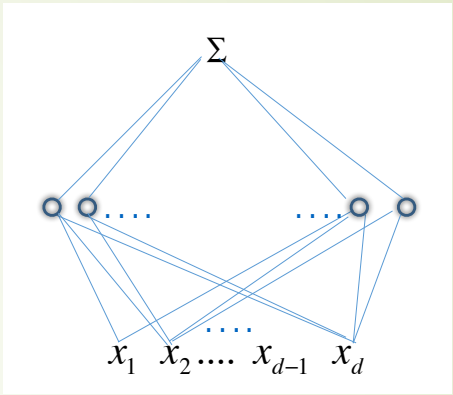




Left: GD solution for logistic loss

Right: GD solution for exponential loss

Deep Networks makes it separable

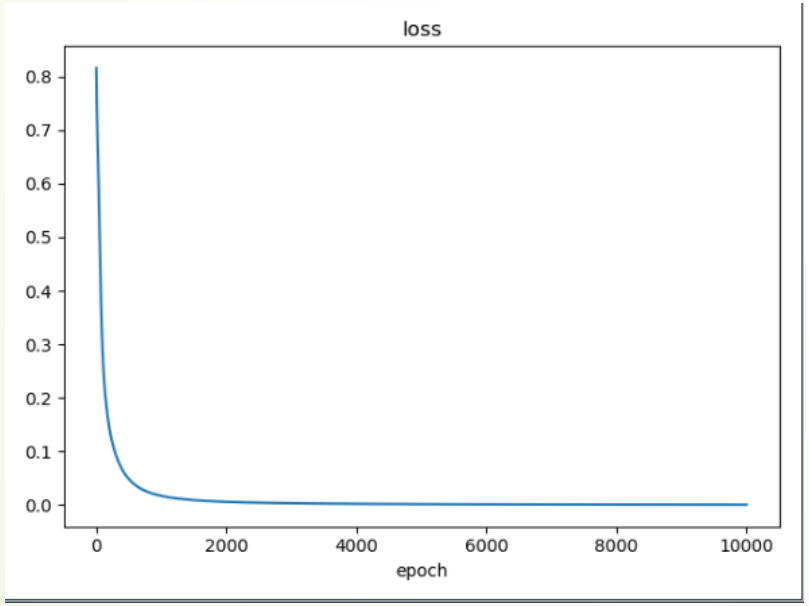


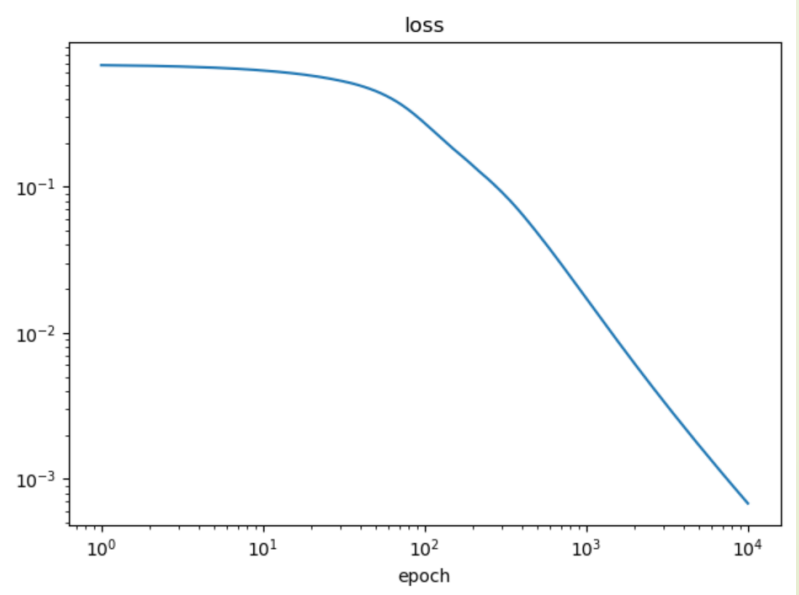
2-layer neural network:

$$f(x) = W_2\sigma(W_1x)$$

where $\sigma(u) = \max(0, u)$ is ReLU, $W_1 \in R^{d\times q}$, and $W_2 \in R^{q\times 1}$

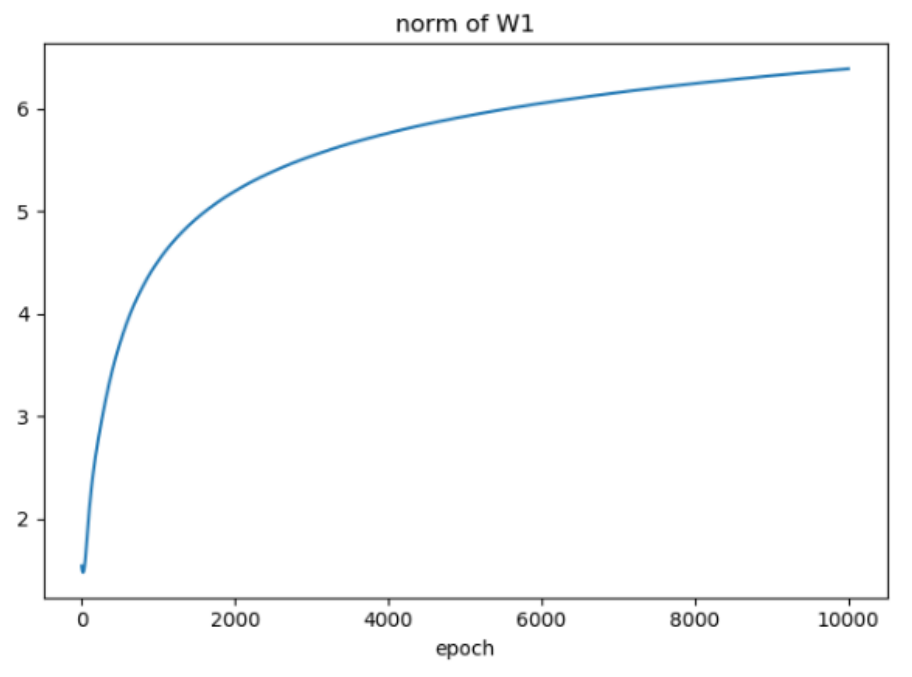
For large *q*, e.g. *q*=5, it becomes separable: logistic loss drops down at ~1/k

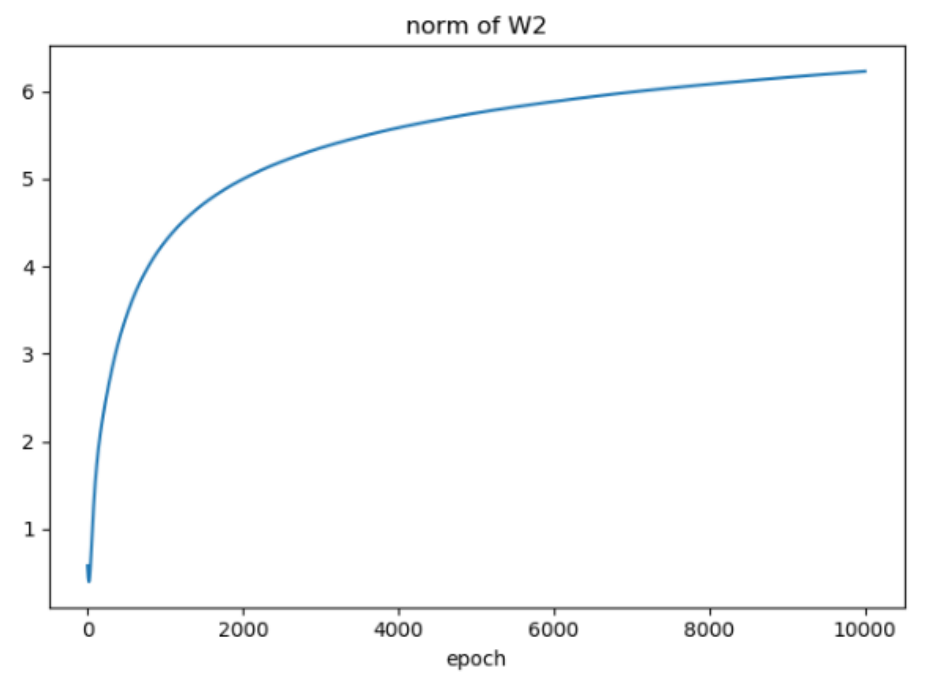




By *Yifei HUANG*, HKUST

Both W_1 and W_2 grows to infinity (log k)!



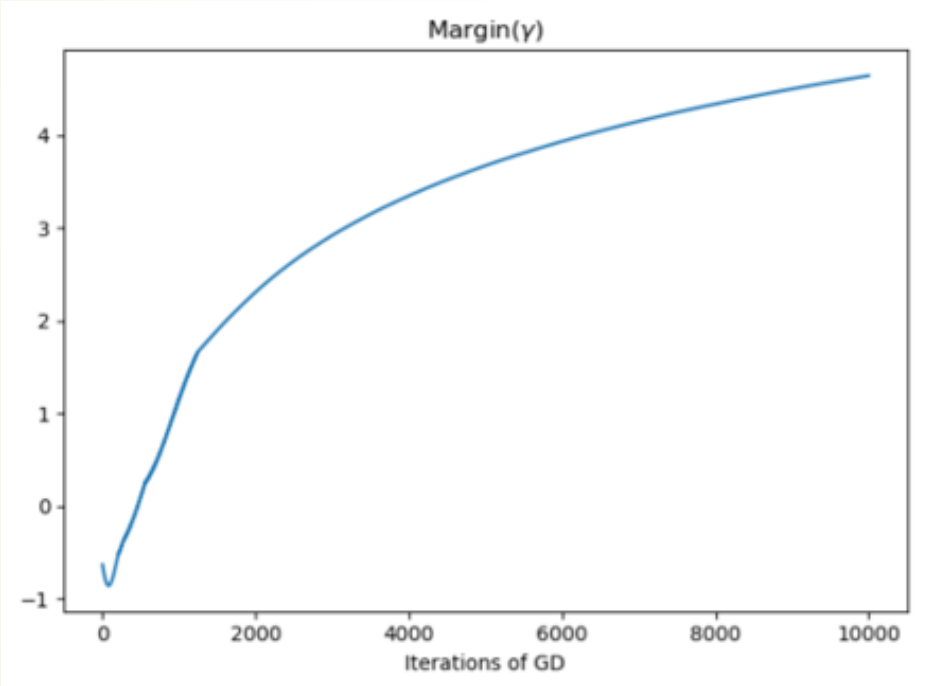


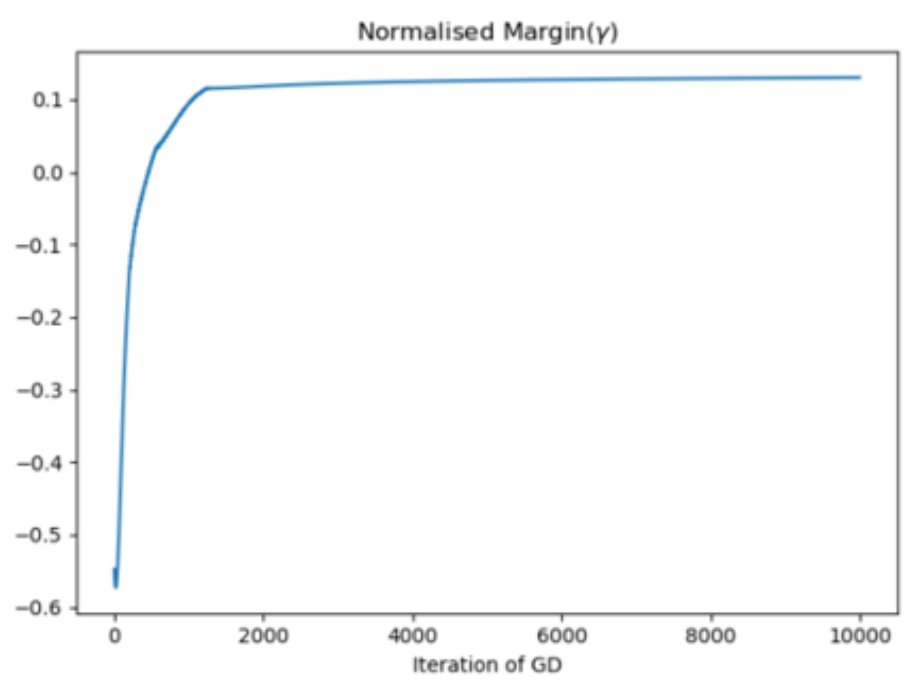
By *Yifei HUANG*

Normalized Margin stabilizes!

$$\gamma := \min_i y_i f(x_i)$$

$$\gamma_n := \frac{\gamma}{\prod_{i=1}^{2} \|W_i\|}$$





After about 1000 epochs, it correctly classifies all training examples and continues to improve the margin.

By *Yifei HUANG*.

Spectrally-Normalized Margin Bounds [Bartlett-Foster-Telgarsky'2017]

$$F_{\mathcal{A}}(x) := \sigma_L(A_L\sigma_{L-1}(A_{L-1}\cdots\sigma_1(A_1x)\cdots)). \tag{1.1}$$

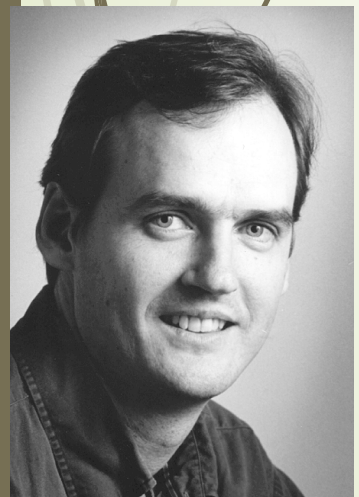
$\mathcal{A} = (A_1, \ldots, A_L)$ *reference matrices* $(M_1, \ldots, M_L)$ with the same dimensions as $A_1, \ldots, A_L$

$$R_{\mathcal{A}} := \left(\prod_{i=1}^{L} \rho_i \|A_i\|_\sigma\right)\left(\sum_{i=1}^{L} \frac{\|A_i^\top - M_i^\top\|_{2,1}^{2/3}}{\|A_i\|_\sigma^{2/3}}\right)^{3/2}. \tag{1.2}$$

Theorem 1.1. *Let nonlinearities* $(\sigma_1, \ldots, \sigma_L)$ *and reference matrices* $(M_1, \ldots, M_L)$ *be given as above (i.e.,* σ_i *is* ρ_i*-Lipschitz and* $\sigma_i(0) = 0$*). Then for* $(x, y), (x_1, y_1), \ldots, (x_n, y_n)$ *drawn iid from any probability distribution over* $\mathbb{R}^d \times \{1, \ldots, k\}$*, with probability at least* $1 - \delta$ *over* $((x_i, y_i))_{i=1}^n$*, every margin* $\gamma > 0$ *and network* $F_{\mathcal{A}} : \mathbb{R}^d \to \mathbb{R}^k$ *with weight matrices* $\mathcal{A} = (A_1, \ldots, A_L)$ *satisfy*

$$\Pr\left[\arg\max_j F_{\mathcal{A}}(x)_j \neq y\right] \leq \widehat{\mathcal{R}}_\gamma(F_{\mathcal{A}}) + \widetilde{\mathcal{O}}\left(\frac{\|X\|_2 R_{\mathcal{A}}}{\gamma n}\ln(W) + \sqrt{\frac{\ln(1/\delta)}{n}}\right),$$

where $\widehat{\mathcal{R}}_\gamma(f) \leq n^{-1}\sum_i \mathbb{1}\left[f(x_i)_{y_i} \leq \gamma + \max_{j \neq y_i} f(x_i)_j\right]$ *and* $\|X\|_2 = \sqrt{\sum_i \|x_i\|_2^2}$.



N. Srebro et al. ICLR 2018 has another PAC-Bayes generalization error bound.

Summary

- For separable classification, GD for logistic regression, cross entropy loss, and exponential loss, etc., converges at infinity to the maximal margin solution in direction
- For non-separable classification, over-parameterized deep networks may make it separable and GD converges toward some max-margin solution at infinity
- Spectrally-normalized margin provides a data-dependent measure of generalization error

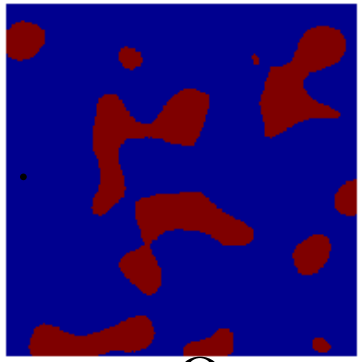
What's the Landscape of Empirical Risks and How to optimize them efficiently?

Over-parameterized models lead to simple landscapes while SGD finds flat minina.

Sublevel sets and topology

- Given loss $E(\theta)$, $\theta \in \mathbb{R}^d$, we consider its representation in terms of level sets:

$$E(\theta) = \int_0^\infty \mathbf{1}(\theta \in \Omega_u) du \ , \ \Omega_u = \{y \in \mathbb{R}^d \ ; \ E(y) \leq u\}$$



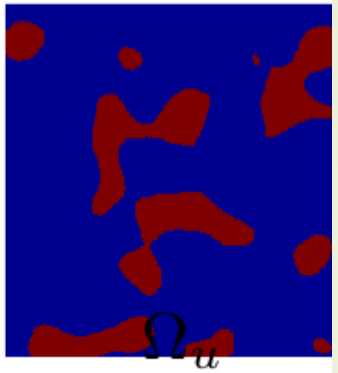
Ω_u

- A first notion we address is about the topology of the level sets .

- In particular, we ask how connected they are, i.e. how many connected components N_u at each energy level u?

Topology of Non-convex Risk Landscape

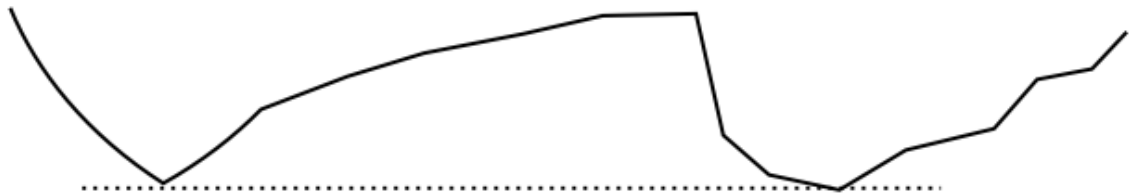
- A first notion we address is about the topology of the level sets .
  - In particular, we ask how connected they are, i.e. how many connected components N_u at each energy level u?
- This is directly related to the question of global minima:



Proposition: If $N_u = 1$ for all u then E has no poor local minima.

$$(\text{i.e. no local minima } y^* \text{ s.t. } E(y^*) > \min_y E(y))$$

- We say *E* is *simple* in that case.
- The converse is clearly not true.



Weaker: P.1, no spurious local valleys

Given a parameter space Θ and a loss function $L(\theta)$ as in (2), for all $c \in \mathbb{R}$ we define the sub-level set of L as

$$\Omega_L(c) = \{\theta \in \Theta : L(\theta) \leq c\}.$$

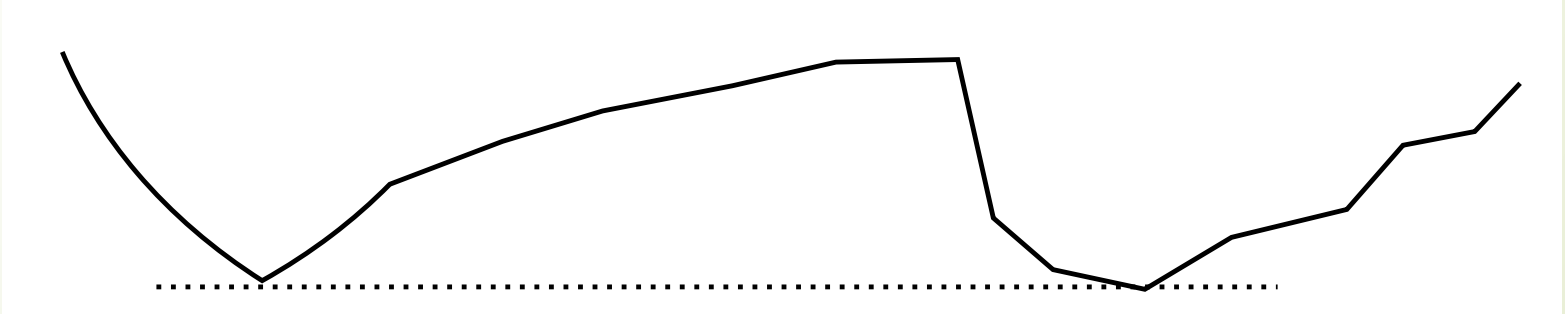
We consider two (related) properties of the optimization landscape. The first one is the following:

P.1 Given any *initial* parameter $\theta_0 \in \Theta$, there exists a continuous path $\theta : t \in [0, 1] \mapsto \theta(t) \in \Theta$ such that:

(a) $\theta(0) = \theta_0$

(b) $\theta(1) \in \arg\min_{\theta \in \Theta} L(\theta)$

(c) The function $t \in [0, 1] \mapsto L(\theta(t))$ is non-increasing.



The landscape has no spurious local valleys.

Overparameterized LN -> Single Basin

$$E(W_1, \ldots, W_K) = \mathbb{E}_{(X,Y)\sim P} \|W_K \ldots W_1 X - Y\|^2 \ .$$

Proposition: [BF'16]

1. If $n_k > \min(n, m)$, $0 < k < K$, then $N_u = 1$ for all u.

2. (2-layer case, ridge regression)
   $E(W_1, W_2) = \mathbb{E}_{(X,Y)\sim P} \|W_2 W_1 X - Y\|^2 + \lambda(\|W_1\|^2 + \|W_2\|^2)$
   satisfies $N_u = 1 \ \forall \ u$ if $n_1 > \min(n, m)$.

- We pay extra redundancy price to get simple topology.

Bruna, Freeman, 2016

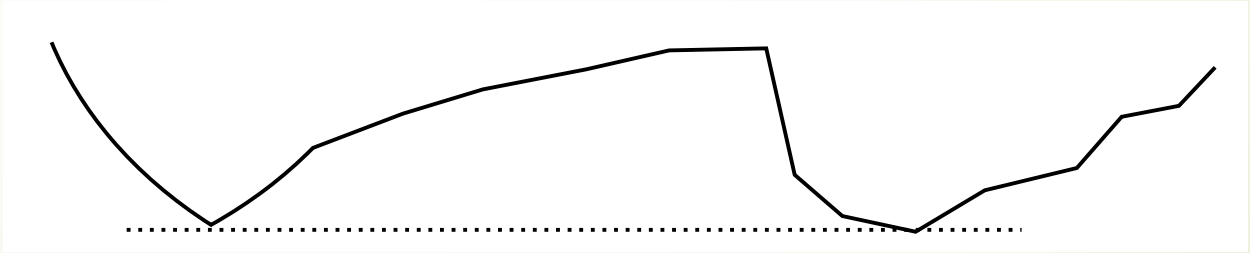
Venturi-Bandeira-Bruna'18

$$\Phi(x;\theta) = W_{K+1}\cdots W_1 x \ , \tag{13}$$

where $\theta = (W_{K+1}, W_K, \ldots, W_2, W_1) \in \mathbb{R}^{n\times p_{K+1}} \times \mathbb{R}^{p_{K+1}\times p_K} \times \cdots \mathbb{R}^{p_2\times p_1} \times \mathbb{R}^{p_1\times n}$.

Theorem 8 *For linear networks* (13) *of any depth* $K \geq 1$ *and of any layer widths* $p_k \geq 1$, $k \in [1, K+1]$, *and input-output dimensions* n, m, *the square loss function* (2) *admits no spurious valleys.*

Symmetry $f(W_i) = f(QW_i)$ $(Q \in GL(R^{n_l}))$ helps remove the network width constraint.



2-layer Neural Networks

[Venturi, Bandeira, Bruna, 2018]

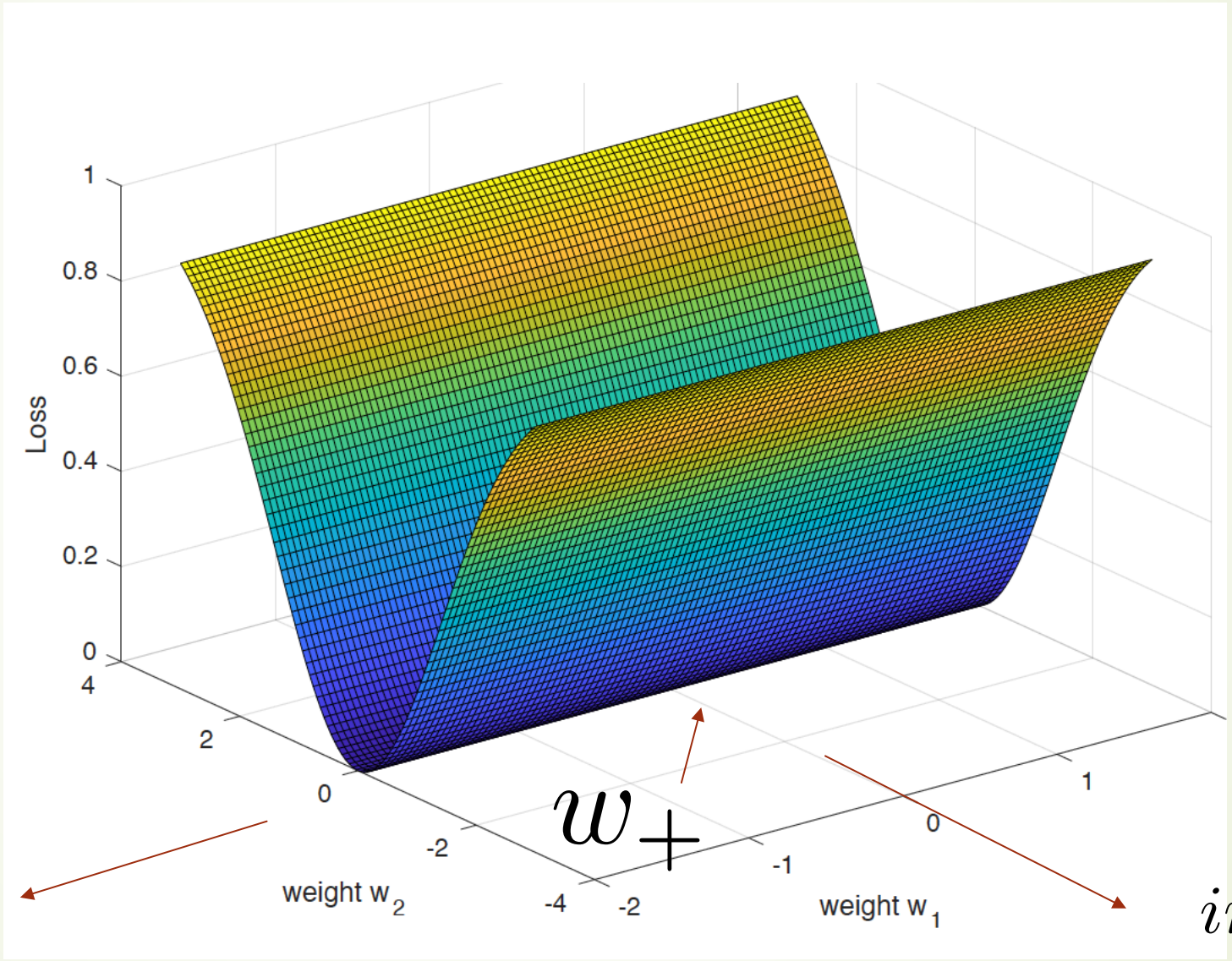
Theorem 5 *The loss function*

$$L(\theta) = \mathbb{E}\|\Phi(X;\theta) - Y\|^2$$

of any network $\Phi(x;\theta) = U\rho Wx$ *with effective intrinsic dimension* $q < \infty$ *admits no spurious valleys, in the over-parametrized regime* $p \geq q$. *Moreover, in the over-parametrized regime* $p \geq 2q$ *there is only one global valley.*

- Reproducing Kernel Hilbert Spaces (RKHS) are exploited in the proof!
- Matrix factorizations are of similar ideas.

Over-parameterized Landscapes: as p>n, equilibria are all degenerate



$ker(X)$

$im(X^T)$

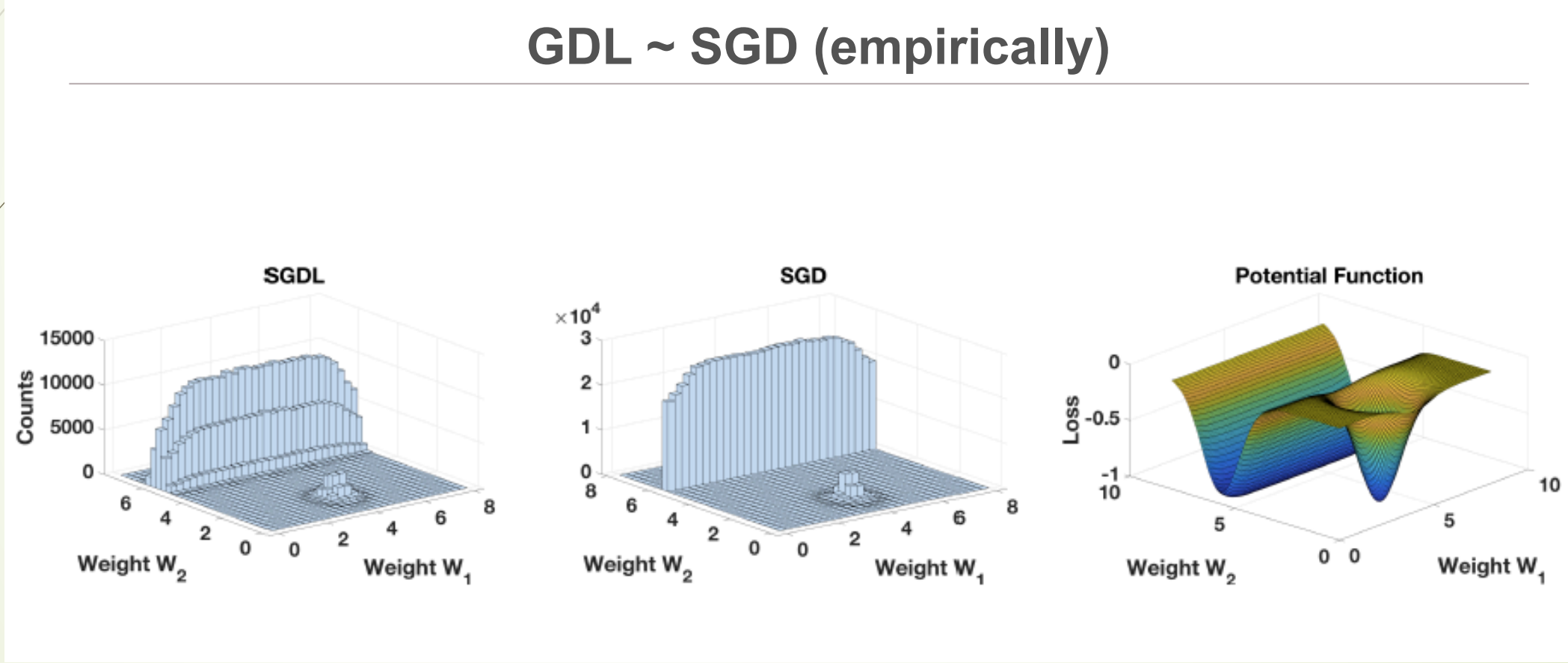
Recall: SGD behaves like Gradient Descent Langevin dynamics (GDL)

$$\frac{dw}{dt} = -\gamma_t \nabla V(w(t), z(t)) + \gamma_t' dB(t)$$

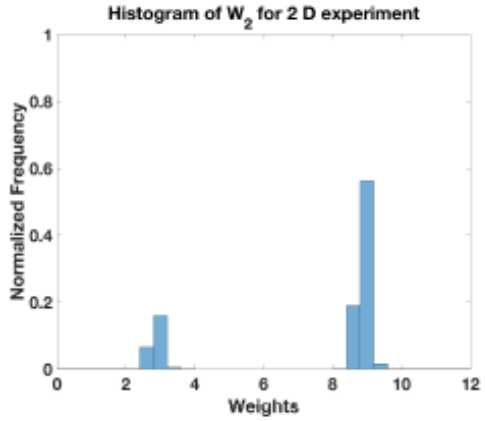
with the Boltzmann equation as asymptotic “solution”

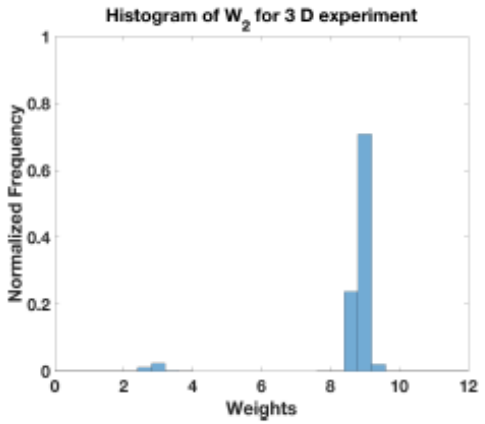
$$p(w) \sim \frac{1}{Z} = e^{-\frac{V(w)}{T}}$$

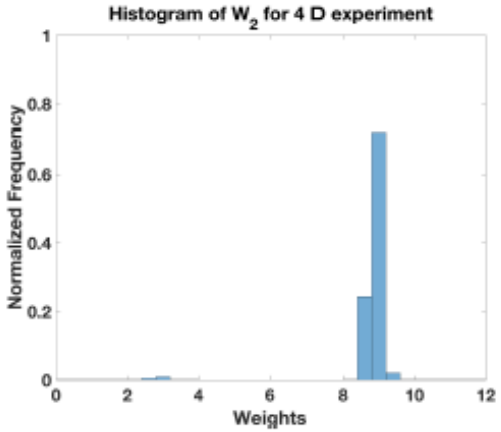
SGD/GDL selects larger volume minima e.g. degenerate

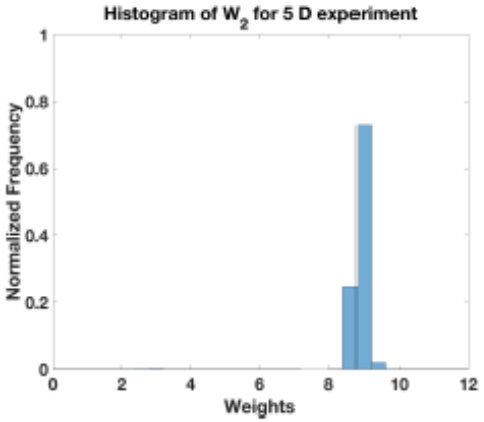


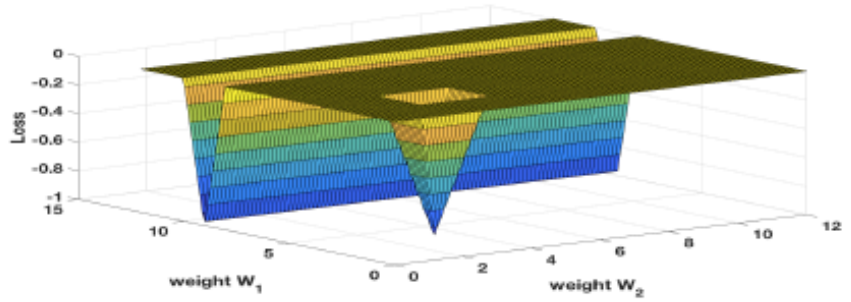
Concentration because of high dimensionality











Poggio, Rakhlin, Golovitc, Zhang, Liao, 2017

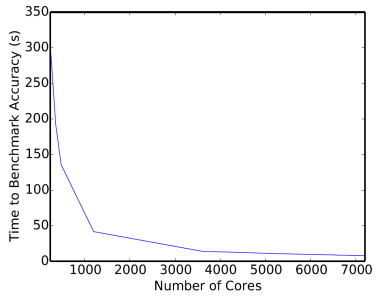
Summary

- Over-parameterization may lead to simple risk landscapes with **flat** (degenerate) global minima
- SGD tends to find **flat** global minima like Langevin Dynamics
- But SGD for multilayer neural networks suffer from vanishing gradient issue (architecture revision: ReLU, ResNet, LSTM)…

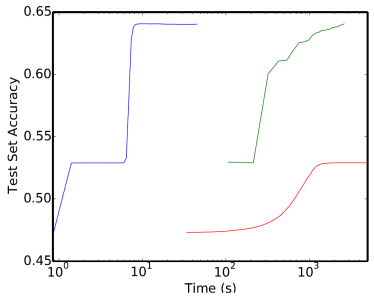
Alternative: Variable Splitting with Block Coordinate Descent

- ADMM-type: **Taylor** et al. ICML 2016
- **Proximal Propagation**, ICLR 2018
- BCD with zero Lagrangian multiplier: **Zhang** et al. NIPS 2017
- Discrete EMSA of PMP: **Qianxiao LI** et al 2017, talk on Monday in IAS workshop
  - No-vanshing gradients and parallelizable
- Global convergence can be established via Kurdyka-Lojasiewicz inequality: with **Jinshan Zeng and Tsz Kit Lau et al.**

Experiment results on Higgs dataset from Taylor et al'16

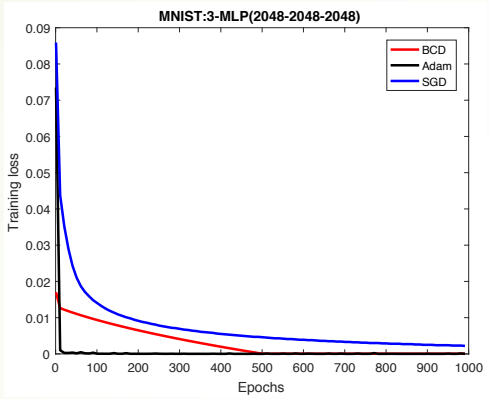


(a) **Time required for ADMM to reach 64% test accuracy when parallelized over varying levels of cores**. L-BFGS on a GPU required 181 seconds, and conjugate gradients required 44 minutes. SGD never reached 64% accuracy.

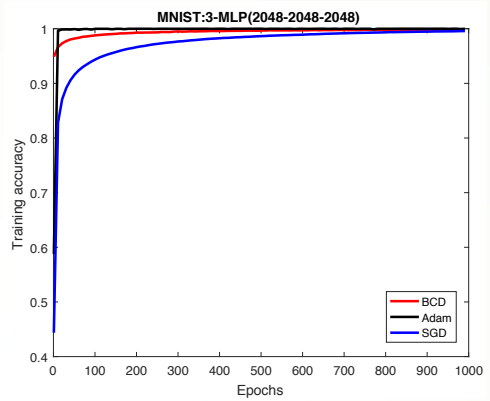


(b) **Test set predictive accuracy as a function of time** for ADMM on 7200 cores (blue), conjugate gradients (green), and SGD (red). Note the x-axis is scaled logarithmically.

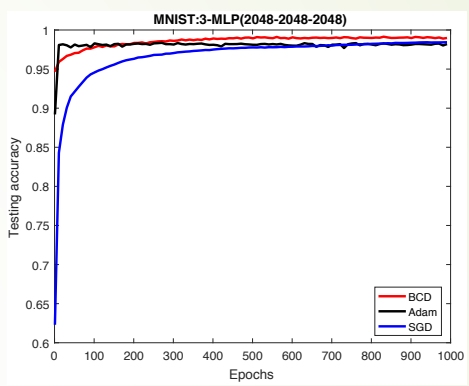
Adam, BCD, vs. SGD



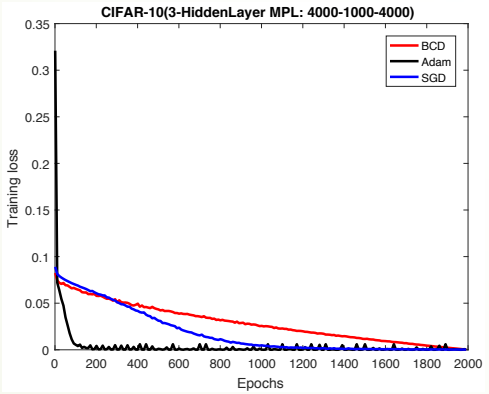
(a) MNIST (Training loss)



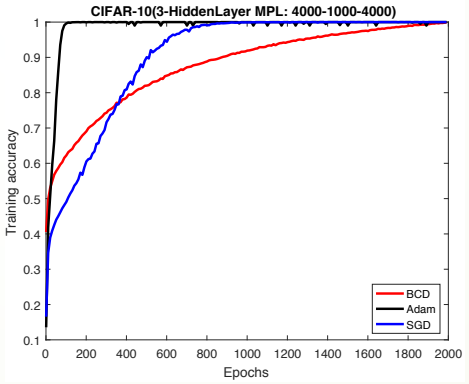
(b) MNIST (Training Acc.)



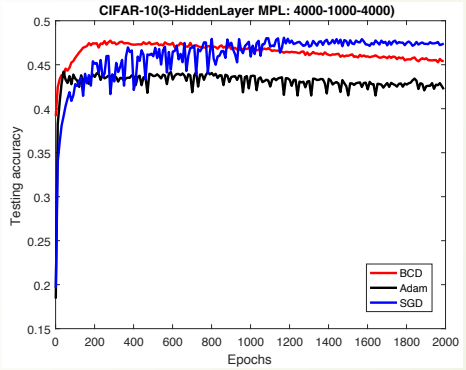
(c) MNIST (Testing Acc.)



(d) CIFAR-10 (Training loss)



(e) CIFAR-10 (Training Acc.)

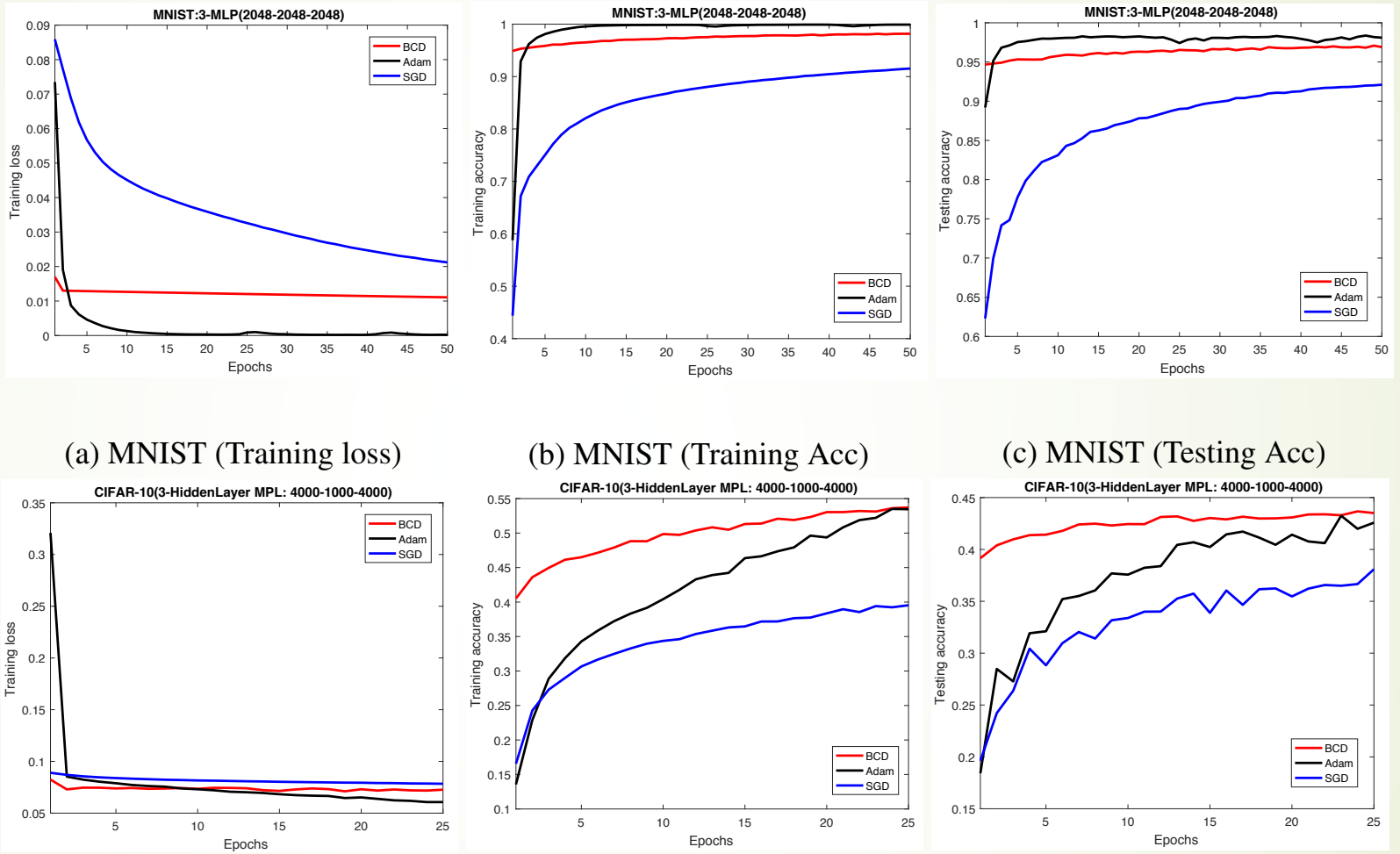


(f) CIFAR-10 (Testing Acc.)

Figure 1: Comparisons on MNIST and CIFAR-10 datasets. The first row consists of the figures including the curves of training loss, training accuracy and testing accuracy on MNIST dataset. The second row gives the associated figures on CIFAR-10 dataset.

by **Jinshan Zeng** et al.

BCD may be good initializers



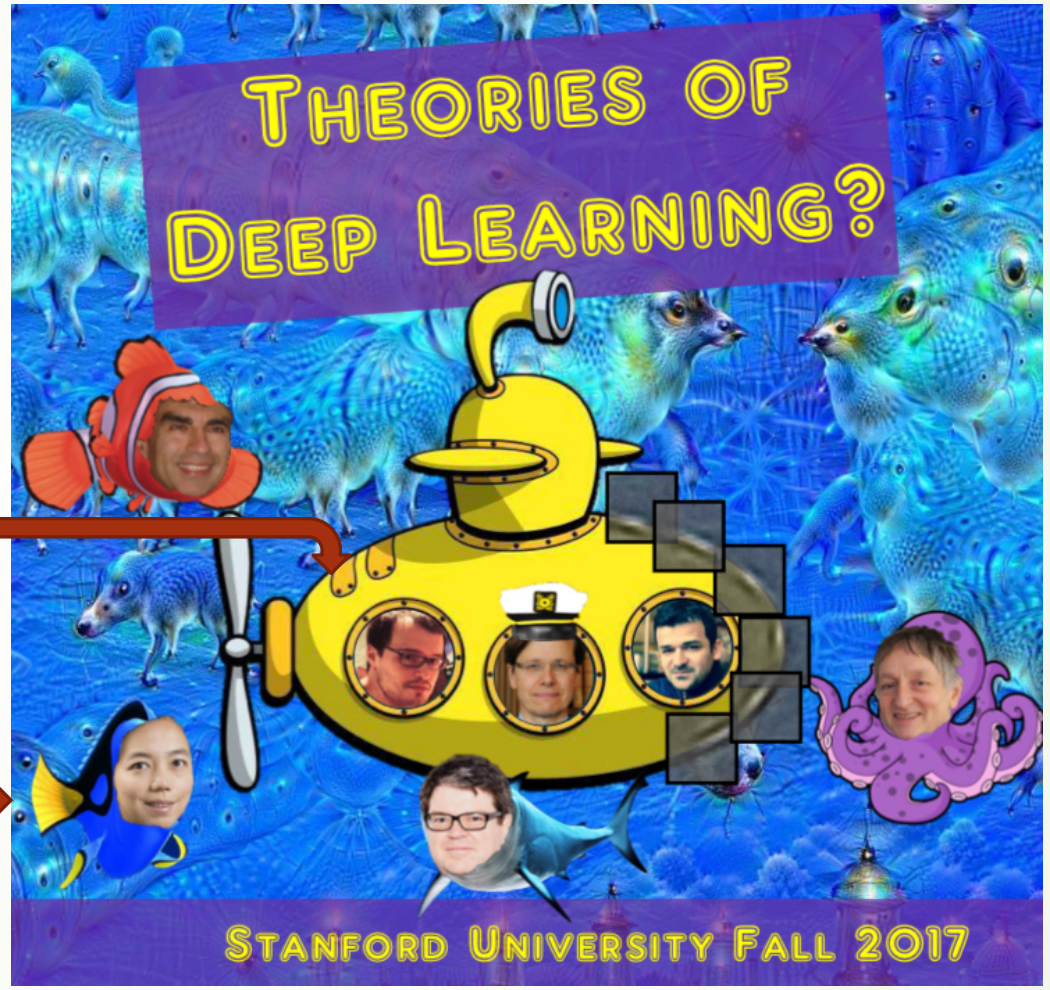
(a) MNIST (Training loss) (b) MNIST (Training Acc) (c) MNIST (Testing Acc)

(d) CIFAR-10 (Training loss) (e) CIFAR-10 (Training Acc) (f) CIFAR-10 (Testing Acc)

Figure 2: Comparisons of the performance of three methods at the early stage.

by **Jinshan Zeng** et al.

Acknowledgement



https://stats385.github.io/

http://cs231n.github.io/

A following-up course at HKUST: https://deeplearning-math.github.io/

Thank you!