



école —  
normale —  
supérieure —  
paris-saclay —

université  
PARIS-SACLAY

# Introduction to Statistical Learning

Nicolas Vayatis

Lecture # 6 - Statistical analysis of mainstream ML algorithms

Part III - Feedforward Neural Networks and Mirror Descent

## Main theoretical objectives of the course

- Take a well-known ML algorithm which operates in  $\mathcal{F}$  : it produces a (random) sequence of decision rules  $(\hat{f}_n)_{n \geq 1}$  in  $\mathcal{F}$ . Then show :
- **Convergence of estimation error :**

$$L(\hat{f}_n) \rightarrow \inf_{\mathcal{F}} L \text{ almost surely as } n \rightarrow \infty ,$$

- **Upper bounds :** with probability at least  $1 - \delta$ , there exists some constant  $c$  such that :

$$L(\hat{f}_n) - \inf_{\mathcal{F}} L \leq C(\mathcal{F}, n) + c \sqrt{\frac{\log(1/\delta)}{n}} ,$$

where  $C(\mathcal{F}, n) = O(1/\sqrt{n})$  after processing some complexity/stability measure

# Consistency of global methods

## Neural networks

# Historical perspective on neural networks

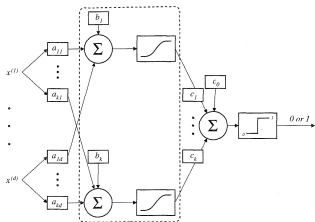
- Cybernetics (1940s-1960s)
  - Achievement : modeling and training one neuron
  - Key algorithm : Linear Perceptron
  - Paper : Rosenblatt (1958)
- Connectionism (1980s)
  - Achievement : training one or two hidden layers
  - Key algorithm : Backpropagation
  - Paper : Rumelhart-Hinton-Williams (1986)
- Deep Learning (2007-....)
  - Achievement : training multiple layers of representation
  - Key algorithm : Stochastic gradient
  - Papers : Hinton (2006), Bengio-LeCun (2007)

# Principle of feedforward neural networks

## Single-layer

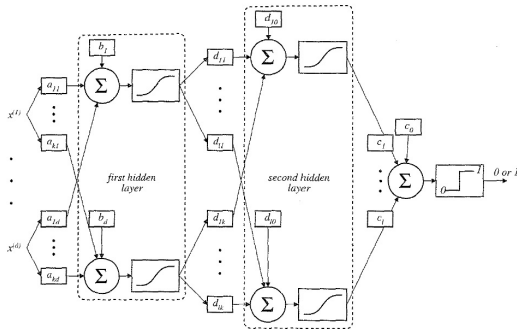
- Form of classifier implemented by a one-hidden layer perceptron :  $g = \text{sgn}(f - 1/2)$ , where :

- $f(x) = c_0 + \sum_{i=1}^p c_i \cdot \sigma \circ \psi_i(x)$  ,  $\forall x \in \mathbb{R}^d$
- $\sigma$  is a sigmoid,
- the  $\psi_i$ 's are linear :  $\psi_i(x) = b_i + \sum_{j=1}^p a_{i,j}x^{(j)}$



# Principle of feedforward neural networks

## Multiple-layer

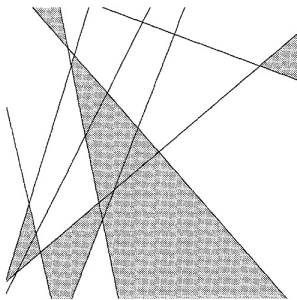
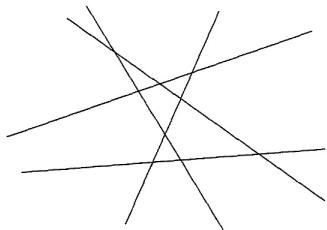


- Hypothesis space : functions of the form

$$f(x, \theta) = \sigma_m \circ A_m \circ \sigma_{m-1} \circ \dots \circ A_2 \circ \sigma_1 \circ A_1 x$$

where  $\theta = (A_1, \dots, A_m)$  and  $A_1, \dots, A_m$  are matrices

## Intuition for complexity analysis : linear arrangements



## Arrangements : definition and key property

### Definition.

A simple arrangement is a collection  $\mathcal{A}$  of hyperplanes in dimension  $d$  such that :

- (i) any  $d$  hyperplanes of  $\mathcal{A}$  have a unique point in common, and
- (ii) any  $d + 1$  hyperplanes of  $\mathcal{A}$  have no point in common.

### Theorem. (Edelsbrunner (1987))

The number of cells of a simple arrangement with cardinality  $|\mathcal{A}| = M$  is given by :

$$\begin{cases} 2^M & \text{if } d \geq M \\ \sum_{i=1}^d \binom{M}{i} & \text{if } d < M \end{cases}$$

## Consistency result for ERM on arrangements

Theorem. (Devroye, Györfi, Lugosi (1996))

*The ERM classifier  $\hat{g}_n^M$  on all possible arrangements of size at most  $M$  has expected error which converges to the Bayes error :*

$$\mathbb{E}(L(\hat{g}_n^M)) \rightarrow L^*$$

*for all distributions, as soon as  $M \rightarrow \infty$  and  $M = o(n/\log n)$ .*

The key argument of the proof relies on exact computation of shattering coefficient :

$$\gamma(\mathcal{G}, n) = (2(n^d + 1))^M$$

# Consistency result for $L^1$ -error minimization

**Theorem 30.9.** (LUGOSI AND ZEGER (1995)). *Let  $\sigma$  be an arbitrary sigmoid. Define the class  $\mathcal{F}_n$  of neural networks by*

$$\mathcal{F}_n = \left\{ \sum_{i=1}^{k_n} c_i \sigma(a_i^T x + b_i) + c_0 : a_i \in \mathcal{R}^d, b_i \in \mathcal{R}, \sum_{i=0}^{k_n} |c_i| \leq \beta_n \right\},$$

and let  $\psi_n$  be a function that minimizes the empirical  $L_1$  error

$$J_n(\psi) = \frac{1}{n} \sum_{i=1}^n |\psi(X_i) - Y_i|$$

over  $\psi \in \mathcal{F}_n$ . If  $k_n$  and  $\beta_n$  satisfy

$$\lim_{n \rightarrow \infty} k_n = \infty, \quad \lim_{n \rightarrow \infty} \beta_n = \infty, \quad \text{and} \quad \lim_{n \rightarrow \infty} \frac{k_n \beta_n^2 \log(k_n \beta_n)}{n} = 0,$$

then the classification rule

$$g_n(x) = \begin{cases} 0 & \text{if } \psi_n(x) \leq 1/2 \\ 1 & \text{otherwise} \end{cases}$$

is universally consistent.

## Comments

- More work needed to deal with data-driven arrangements (with or without optimization)
- One hidden layer neural nets are universal approximators (denseness results)
- Upper bounds on VC dimension available
- More theory in [Devroye, Lugosi, and Györfi, 1996]!
- Modern theory involves : margin bounds/concentration inequalities/Rademacher complexities [Bartlett, Foster, Telgarsky, 2019]

# Rademacher complexity of neural networks

## (Setup)

Consider an IID sample  $X_1, \dots, X_n$  of observations over some space  $\mathcal{X}$  and  $\mathcal{F}_0$  is a set of real-valued functions over  $\mathcal{X}$  that includes the zero function.

Assume  $\psi : \mathbb{R} \rightarrow \mathbb{R}$  is  $k$ -Lispchitz and define, for fixed positive real numbers  $V$  and  $B$  :

- a one-layer network as :

$$\mathcal{F}_1 = \left\{ x \mapsto \psi \left( v + \sum_{j=1}^m w_j f_j(x) \right) : |v| \leq V, \|w\|_1 \leq B, f_j \in \mathcal{F}_0 \right\}$$

- a  $p$ -layer network as (iterative definition with fixed layer size) :

$$\mathcal{F}_p = \left\{ x \mapsto \psi \left( v + \sum_{j=1}^m w_j f_j(x) \right) : |v| \leq V, \|w\|_1 \leq B, f_j \in \mathcal{F}_{p-1} \right\}$$

# Rademacher complexity of neural networks

(Exercise)

Prove the following upper bounds on the empirical Rademacher average :

①  $\hat{R}_n(\mathcal{F}_1) \leq k \left( \frac{V}{\sqrt{n}} + 2B\hat{R}_n(\mathcal{F}_0) \right) .$

② Assume in addition that  $\psi(-u) = -\psi(u)$  and  $k = 1$  then show that on  $\mathcal{X} = \{x \in \mathbb{R}^d, \|x\|_\infty \leq 1\}$  :

$$\hat{R}_n(\mathcal{F}_p) \leq \frac{1}{\sqrt{n}} \left( B^p \sqrt{2 \ln(2d)} + V \sum_{l=1}^{p-1} B^l \right) .$$

Cherry on the cake

A mirror descent algorithm

# Motivations

- Ensembles are defined by a set of predictors and their weights
- For bagging and random forests, weights are uniform
- In boosting weights reflect individual performance but are determined iteratively
- Knowing the predictors, what are the optimal weights?

## Problem formulation

- Convex risk minimization under an  $\ell_1$  constraint
- Set  $w \mapsto A(w)$  to be the convex objective
- $w \in \mathbb{R}_+^T$  is the weight vector s.t.  $\|w\|_1 = C$ , for some  $C > 0$
- Approach : fast iterative algorithm to optimize  $w$  over the  $C$ -simplex

## Setup for learning and optimization

- Vector of predictors  $H(x) \in \mathbb{R}^T$
- Loss function  $\varphi : \mathbb{R} \rightarrow \mathbb{R}_+$  is convex with derivative  $\varphi'$  (monotone version)
- Optimization objective  $A(w) = \mathbb{E}(\varphi(Yw^T H(X)))$  (not necessarily accessible)
- "Observable" : stochastic subgradient at observation  $(X_i, Y_i)$

$$u_i(w) = \varphi'(Y_i w^T H(X_i)) Y_i H(X_i) \in \mathbb{R}^T$$

- Insight : (fast) gradient descent in the dual space

## The "mirror"

- Potential or proxy function  $V$  convex over  $E = (\mathbb{R}^T, \|\cdot\|_1)$
- Dual space :  $E^* = (\mathbb{R}^T, \|\cdot\|_\infty)$
- C-simplex  $\Delta(C) = \{w \in \mathbb{R}_+^T : \|w\|_1 = C\}$
- The  $\beta$ -convex conjugate of  $V$  is defined as : for any  $z \in E^*$ , for any  $\beta > 0$

$$W_\beta(z) = \sup_{w \in \Delta(C)} (-w^T z - \beta V(w))$$

- Here : special case where  $V$  is the entropy proxy function,  $\forall w \in \Delta(C)$

$$V(w) = C \ln(T/C) + \sum_{j=1}^T w^{(j)} \ln w^{(j)}$$

# Mirror descent algorithm for convex risk optimization

For  $i = 1, \dots$

- "Temperature" parameter : Let  $\beta_i = \beta_0 \sqrt{i+1}$
- Gradient updates  $\zeta_i = \zeta_{i-1} + u_i(w_{i-1})$
- Mirror step :  $w_i = -\nabla W_{\beta_i}(\zeta_i)$
- Averaging step :  $\hat{w}_{i+1} = \hat{w}_i - \frac{1}{i+1}(\hat{w}_i - w_i)$

NB : constant stepsize equal to one in this version

## Upper bound on numerical convergence

- Consider a positive and convex loss  $\varphi$  and  $T \geq 2$ , then

$$\mathbb{E}(A(\hat{w}_n)) - \min_{w \in \Delta(C)} A(w) \leq \kappa(\varphi, C) \frac{\sqrt{(n+1) \ln T}}{n}$$

where  $\kappa(\varphi, C)$  is explicit and tight

- Classification case :  $\kappa(\varphi, C) = 2C \sup_{x \in [-C, C]} |\varphi'(x)|$

## Background on mirror descent algorithms

- A.S. Nemirovski and D.B. Yudin. Problem Complexity and Method Efficiency in Optimization. Wiley-Interscience, 1983.
- B.T. Polyak and A.B. Juditsky. Acceleration of stochastic approximation by averaging. SIAM J. Control Optim., 30(4) :838–855, 1992
- A. Beck and M. Teboulle. Mirror descent and nonlinear projected sub-gradient methods for convex optimization. Operations Research Letters,31 :167–175, 2003.
- J. Kivinen and M.K. Warmuth. Additive versus exponentiated gradient updates for linear prediction. Information and computation, 132(1) :1–64, 1997.
- A. Juditsky, A. Nazin, A. Tsybakov, and N. Vayatis. Recursive Aggregation of Estimators via the Mirror Descent Algorithm with averaging. Problems of Information Transmission , 41(4) : 368-384, 2005.

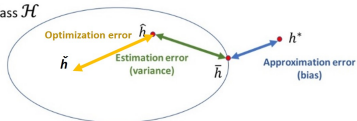
Beyond this course

## Theory of ML : From 2000 to Today

- Consistency/rates/fast rates of convergence of the estimation error for regularized learning methods : SVM (Steinwart, 2005), Boosting (Lugosi-Vayatis, 2004)(Zhang, 2004), general surrogate losses (Bartlett, Jordan, McAuliffe, 2006)
- Theory of ranking and scoring algorithms with advanced concentration inequalities (Cl emen on-Lugosi-Vayatis, 2008)
- Other tracks with theoretical advances : multiclass classification, ranking, multitask learning, structured prediction...
- Other setups : online learning, learning view on game theory, transfer learning, active learning...

# Bias-variance revisited

Hypothesis class  $\mathcal{H}$



$$\begin{aligned}\mathcal{E} &= \mathbb{E}[E(f_{\hat{h}}^*) - E(f^*)] + \mathbb{E}[E(f_n) - E(f_{\hat{h}}^*)] + \mathbb{E}[E(\tilde{f}_n) - E(f_n)] \\ &= \mathcal{E}_{\text{app}} + \mathcal{E}_{\text{est}} + \mathcal{E}_{\text{opt}}.\end{aligned}$$

Trade-off wrt : Search space  $\mathcal{F}$ , sample size  $n$ , numerical tolerance  $\rho$

		$\mathcal{F}$	$n$	$\rho$
$\mathcal{E}_{\text{app}}$	(approximation error)	$\searrow$		
$\mathcal{E}_{\text{est}}$	(estimation error)	$\nearrow$	$\searrow$	
$\mathcal{E}_{\text{opt}}$	(optimization error)	$\dots$	$\dots$	$\nearrow$
$T$	(computation time)	$\nearrow$	$\nearrow$	$\searrow$

[The trade-offs of Large Scale Learning, L. Bottou, O. Bousquet, 2011]

## Shallow learning vs. Deep Learning

# Mysteries about deep learning

- Approximation : deep better than shallow?
- Optimization : nonconvex with millions of dimensions (!)
- Overfitting : huge complexity

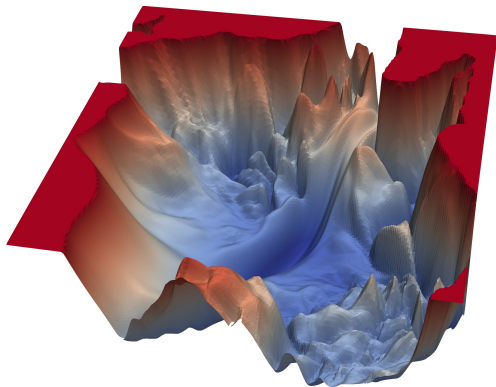
# Facts about approximation theory

## Comparison of Shallow vs. Deep Networks

- Poggio and Liao (2018) : approximation of compositional functions
- Liang and Srikant (2017) : approximation of polynomial functions
- Similar findings :  
*" the number of neurons needed by a shallow network to approximate a function is exponentially larger than the corresponding number of neurons needed by a deep network for a given degree of function approximation. "*

# The loss landscape of Deep Learning

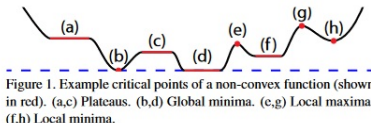
View on a 56-layer neural network without skip-connection



From [Visualizing the Loss Landscape of Neural Nets,  
H. Li, Z. Xu<sup>1</sup>, G. Taylor, C. Studer, T. Goldstein, 2018]

# Facts about optimization in Deep Learning

- Under certain conditions, no poor local minima



- SGD avoids bad critical points
- Larger networks are better behaved (local minima are global)

## References :

Soudry and Carmon (2016), "No bad local minima : Data independent training error guarantees for multilayer neural networks".

Kawaguchi (2016), "Deep learning without poor local minima".

Haeffele and Vidal (2017), "Global optimality in neural network training".

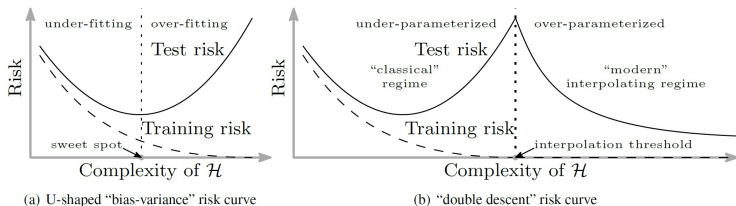
Janzamin, Sedghi, and Anandkumar (2015), "Beating the perils of non-convexity : Guaranteed training of neural networks using tensor methods".

Panageas and Piliouras (2016), "Gradient descent only converges to minimizers : Non-isolated critical points and invariant regions".

Brutzkus, Alon et al. (2017), "SGD Learns Over-parameterized Networks that Provably Generalize on Linearly Separable Data".

# The theory of a double descent risk curve

How Deep Learning (and random forests) avoid overfitting



From [Reconciling modern machine learning and the bias-variance trade-off, M. Belkin, D. Hsu, S. Ma, S. Mandal, 2018]

End of the course

Thank you !