Neural networks : Empirical risk minimization

predictors : $f_\theta : X \to \mathbb{R}$

$$\min_\theta \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f_\theta(x_i))$$

Linear models : $f_\theta(x) = \theta^\top \varphi(x)$ with $\varphi(x) \in \mathbb{R}^d$

$= \langle \theta, \varphi(x) \rangle$ with $\varphi(x) \in \mathcal{H}$ Hilbert space

+'s : - convex optimization
- generalization guarantees
- kernels : escape curse of dimensionality if $f^*$ smooth

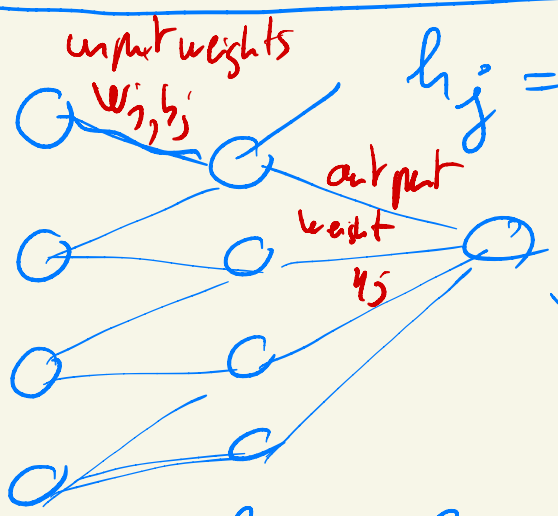
-'s : still exponential in dimension if $f^*$ non-smooth

excess risk $= O(n^{-1/d+2})$

Neural networks : "feature learning" $\Rightarrow$ learn $\varphi(x)$

Today : 3 types of errors $\to$ optimization
$\to$ estimation
$\to$ approximation

# Fully connected single hidden layer NN




$h_j = \sigma(w_j^T x + b_j)$

~~"bias"~~ offset / constant term

affine function of $x$

output layer: $f(x) = \sum_{j=1}^{m} \eta_j \sigma(w_j^T x + b_j)$

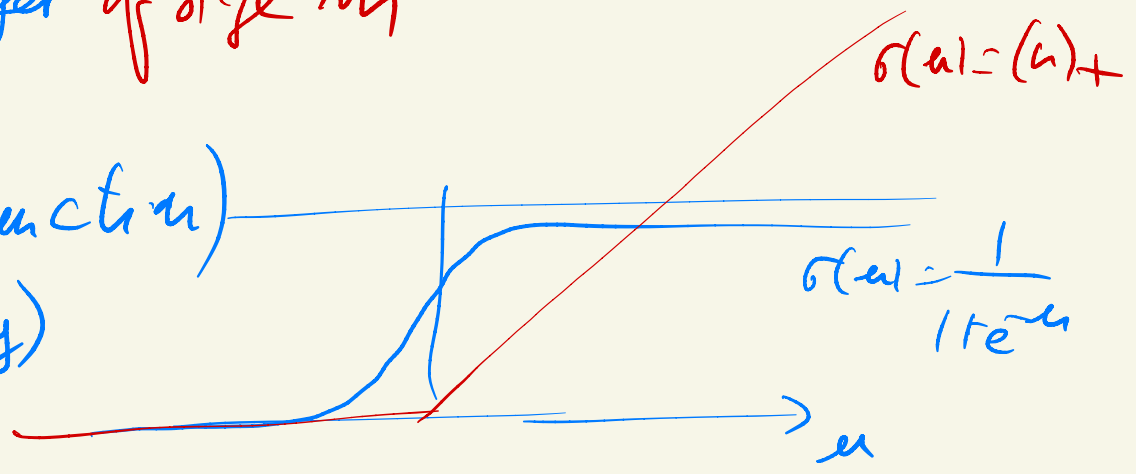
hidden layer of size $m$

Choice of $\sigma$ (activation function)

1. Sigmoid (20th century)
2. ReLU (21st —)

Rectified linear unit

$\sigma(u) = \max(0, u) = (u)_+$




$\Rightarrow$ $\sigma$ can be anything except a polynomial

optimization = $R(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \sum_{j=1}^{m} v_j \sigma(w_j^T x + b_j))$

NON-CONVEX

$\Rightarrow$ (S)GD may get trapped in local minima

non linearity in param.

① No guarantees

② Power of overparametrization

m is large

"Proposition" (see blog post)

if $m \to +\infty$, and weights are entriesed randomly

then GD converges to the global minimum

m large { slow and sad for the planet
overfitting //

## Estimation error :

$R(g) = \mathbb{E}\, \ell(y, g(x))$ expected risk

$\hat{R}(g) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, g(x_i))$ empirical risk

$\mathcal{F}$ space of functions

$$\underbrace{R(\hat{g}) - \inf_{g \text{ measurable}} R(g)}_{\text{excess risk}} = R(\hat{g}) - R(g_{\mathcal{F}}^*) + R(g_{\mathcal{F}}^*) - \inf_{g \text{ measurable}} R(g)$$

$\underbrace{\qquad\qquad}_{\text{estimation error}}$ $\underbrace{\qquad\qquad}_{\text{approximation error}}$

$\forall g \in \mathcal{F}$

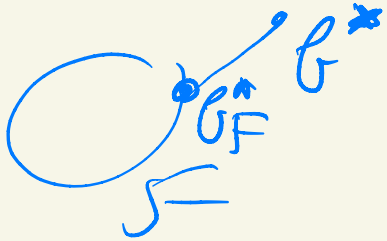
$g_{\mathcal{F}}^* = \arg\min_{g \in \mathcal{F}} R(g)$

$$R(\hat{g}) - \hat{R}(\hat{g}) + \hat{R}(\hat{g}) - \hat{R}(g_{\mathcal{F}}^*) + \hat{R}(g_{\mathcal{F}}^*) - R(g_{\mathcal{F}}^*)$$

$\leq C$ (estimation error) $\leq 0$ if $\hat{g}$ = ERM (minimizer) $\leq C$ (estimation error)

$$\leq \sup_{g \in \mathcal{F}} \left( R(g) - \hat{R}(g) \right) + \sup_{g \in \mathcal{F}} \left( \hat{R}(g) - R(g) \right)$$

estimation error



Rademacher complexity

$$R_n(\mathcal{F}) = \mathbb{E}_{\text{data},\, \varepsilon \in \{-1,1\}^n} \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i \ell(y_i, f(x_i))$$

↳ "Rademacher variables"

$$\mathbb{E} \sup_{f \in \mathcal{F}} \left( R(f) - \hat{R}(f) \right) \leq 2 R_n(\mathcal{F}) \Rightarrow \text{see class 3}$$

Contraction principle: if $\varphi_i : \mathbb{R} \to \mathbb{R}$ are 1-Lipschitz continuous

$$|\varphi_i(u) - \varphi_i(v)| \leq |u - v|$$

$$\mathbb{E} \sup_f \frac{1}{n} \sum_i \varepsilon_i \varphi_i(f(x_i))$$

$$\leq \mathbb{E} \sup_f \frac{1}{n} \sum_i \varepsilon_i f(x_i)$$

For neural nets:

Neural networks: Assumptions: $\|x\|_2 \le R$ almost surely

$$\mathcal{F} = \left\{ \sum_{j=1}^{m} \eta_j \sigma(w_j^T x + b_j),\ \|\eta\|_1 \le D_\eta,\ \Omega\begin{pmatrix} w_j \\ b_j \end{pmatrix} \le D_{w,b} \right\}$$

$L_j$ norm to be chosen later

$\mathcal{F}_+ :=$

$\eta \ge 0$

$\mathcal{F} \subset \mathcal{F}_+ - \{-\mathcal{F}_+\}$ because

$R_n(\mathcal{F}) \le R_n(\mathcal{F}_+) + R_n(-\mathcal{F}_+)$.

$$\eta_j = \underbrace{(\eta_j)_+}_{\ge 0} - \underbrace{(-\eta_j)_+}_{\ge 0}$$

$$|\eta_j| = |(\eta_j)_+| + |(-\eta_j)_+|$$

$$\mathcal{F}_T = \left\{ \sum_{j=1}^{m} \eta_j \sigma(w_j^T x + b_j),\ \|\eta\|_1 \le D_\eta,\ \Omega\begin{pmatrix} w_j \\ b_j \end{pmatrix} \le D_{w,b} \right\}$$

$$= \max\left\{ \|w_j\|_2, \frac{b_j}{R} \right\}$$

$\eta \ge 0$

$$\sup_j \sum_j \eta_j \alpha_j = D_\eta \sup_j \alpha_j$$

$$R_n(\mathcal{F}_T) \le \mathbb{E}_{\varepsilon, data} \sup_{f \in \mathcal{F}_T} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i f(x_i)$$

$$= \mathbb{E}_{\varepsilon, data} \sup_{\eta_j, w_j, b_j,\ j \in \{1,\dots,m\}} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i \sum_{j=1}^{m} \eta_j \sigma(w_j^T x_i + b_j)$$

$$\le \mathbb{E}_{\varepsilon, data} \sup_{j} \sup_{w_j, b_j} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i \sigma(w_j^T x_i + b_j) \cdot D_\eta$$

$$\sup_{x \in D} = \sup_{x \in D} \sup_{y \in E}$$

if $\sigma$ is 1-Lipschitz $\le$ Contractions

$$\mathbb{E} \sup_j \sup_{w_j, b_j} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i \left[ w_j^T x_i + b_j \right] \cdot D_\eta$$

$\|w_0\|_2 \le D_{w,b}$

$|b_j| \le R D_{w,b}$

$\le$

$$\leq D_y \mathbb{E} \sup_{i} \sup_{w,b} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i \left[ w^T x_i + b \right] = b \left[ \frac{1}{n} \sum_i \varepsilon_i \right] + w^T \left[ \frac{1}{n} \sum_i \varepsilon_i x_i \right]$$

$\|w\|_2 \leq D_{w,b}$

$|b| \leq R D_{w,b}$

$$\leq D_y \mathbb{E} \sup RD_{w,b} \left| \frac{1}{n} \sum_i \varepsilon_i \right| + D_{w,b} \left\| \frac{1}{n} \sum_i \varepsilon_i x_i \right\|_2$$

$$\leq D_y D_{w,b} \left[ R \sqrt{\mathbb{E}\left( \frac{1}{n} \sum_i \varepsilon_i \right)^2} + \sqrt{\mathbb{E} \left\| \frac{1}{n} \sum_i \varepsilon_i x_i \right\|_2^2} \right]$$

by Jensen's inequality

$|c^k x - c^k c| \to |c^k| \, |x - c|$

$\varepsilon_i$ are independent and zero mean

$$= \frac{1}{n} \mathrm{Var}(\varepsilon_i) = \frac{1}{n}$$

$$V = \frac{1}{n} \mathbb{E}\, \varepsilon_i^2 \|x_i\|^2 \qquad \varepsilon_i^2 = 1 \leq R^2$$

$$\leq \frac{R^2}{n}$$

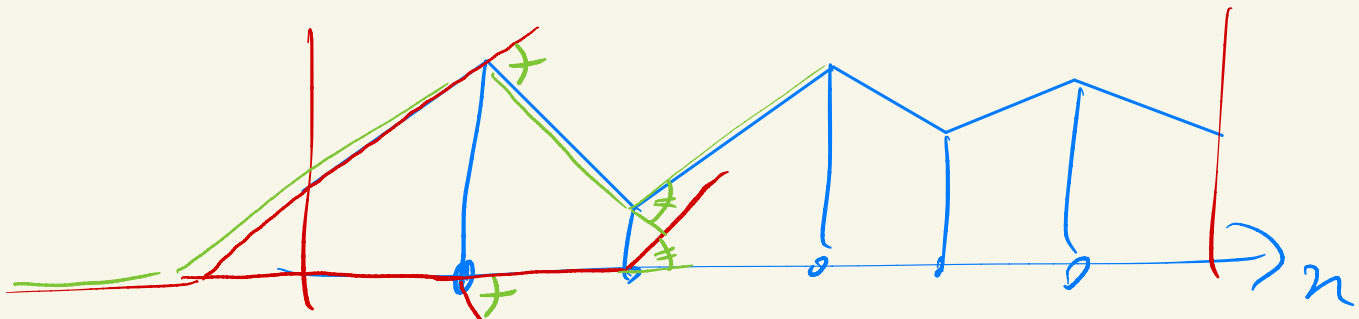
$$\leq D_y D_{w,b} \frac{2R}{\sqrt{n}} \to O\left(\frac{1}{\sqrt{n}}\right)$$

$\to$ independent of $n$ (# of samples)

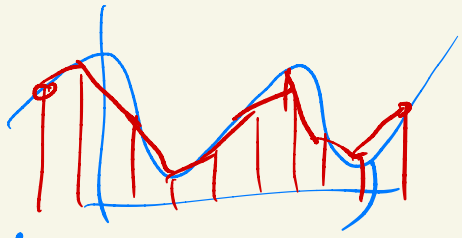
# Approximating a function with neural networks

Universality: 1D : approximate a function $f(x) \approx \sum_{j=1}^{m} v_j (w_j x + b_j)_+ + c$


$f_m$ is piecewise affine with $m$ "kinks"




Proposition: any piecewise affine function in a finite interval and $m+1$ pieces can be written as a NN with $m+1$ neurons (ReLU)



(2) All continuous functions can be approximated by piecewise affine functions

(3) Universality in d dimensions

$$f(x) = \frac{1}{(2\pi)^d} \int \hat{f}(\omega) e^{i\omega^T x} d\omega \quad \text{where } \hat{f}(\omega) = \int f(x) e^{-i\omega^T x} dx$$

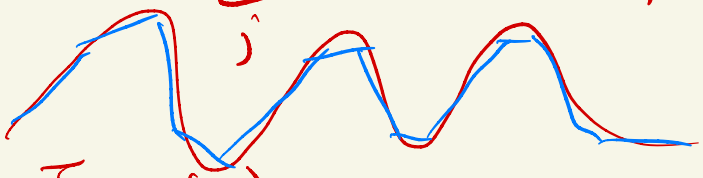
↳ Fourier transform

if $f$ sufficiently regular

$\varphi(\omega^T x)$

cosine $\approx$ piecewise affine function

sine $= \sum_j \eta_j (a_j u + b_j)_+ = \cos u$



$$\cos(\omega^T x) \approx \sum_j \eta_j (a_j \underbrace{\omega^T x}_{u} + b_j)_+$$

$u$ : scalar

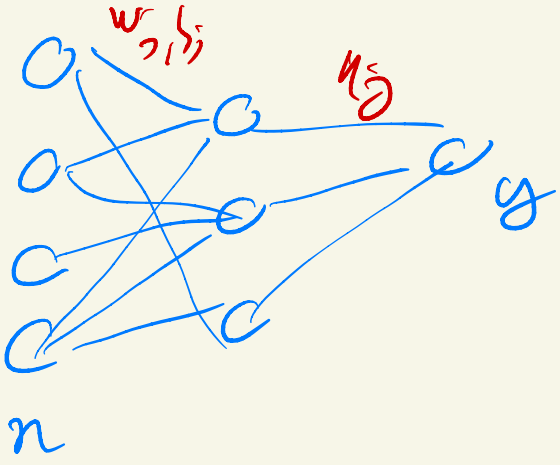
## Neural networks with potentially many neurons

$$B = \{ w_j^2 + \zeta_j^2 \le 1 \}$$

$$f(x) = \sum_{j=1}^{\infty} \eta_j \sigma(w_j^T x + \zeta_j) = \int_B \sigma(w^T x + b)\, d\mu(w,\zeta)$$

where $d\mu(w,\zeta) = \sum_{j=1}^{m} \eta_j \delta_{w_j,\zeta_j}$

$$\int g(w,\zeta)\, \delta_{w_j,\zeta_j} = g(w_j,\zeta_j)$$




Two function spaces:

$$f(x) = \int \sigma(w^T x + \zeta)\, d\mu(w,\zeta)$$

$\|\cdot\|_1$: $\Omega_1(f) = \int |d\mu(w,\zeta)|$ total variation

$= \|\gamma\|_1$

$\|\cdot\|_2$: $\Omega_2(f)^2 = \int \left| \frac{d\mu(w,\zeta)}{d\tau(w,\zeta)} \right|^2 d\tau(w,\zeta)$

uniform measure on the ball B

Similar to

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

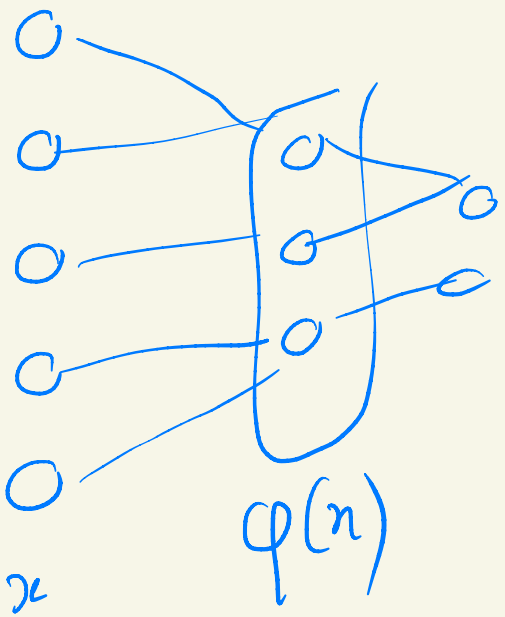
$$\int \ell(y, f(x))\, d\hat{\rho}(x,y)$$

$$d\hat{\rho} = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i y_i} \xrightarrow{n \to \infty} \rho$$

empirical measure

first distribution

# Link between neural network and kernel methods




$$f(x) = \sum_{j=1}^{m} \alpha_j \, \sigma(w_j^T x + b_j)$$

$$= \theta^T \varphi(x) \quad \text{if} \quad \varphi(x) \in \mathbb{R}^m$$

$$\varphi(x)_j = \frac{1}{\sqrt{m}} \sigma(w_j^T x + b_j)$$

$$\theta \in \mathbb{R}^m, \ \theta_j = \alpha_j \sqrt{m}$$

in a kernel method, $k(x, x') = \varphi(x)^T \varphi(x')$ is fixed

in a neural net, $k(x, x')$ is parameterized ($w_j, b_j$) and learned

$\Rightarrow$ kernel learning

$$k(x, x') = \varphi(x)^T \varphi(x') = \frac{1}{m} \sum_{j=1}^{m} \sigma(w_j^T x + b_j) \, \sigma(w_j^T x' + b_j)$$

converge to the expectation

what if $(w_j, b_j)$ random?

converge to the expectation

$$k(x,x') = \varphi(x)^T \varphi(x') = \frac{1}{m} \sum_{j=1}^{m} \sigma(w_j^T x + b_j)\, \sigma(w_j^T x' + b_j)$$

what if $(w_j, b_j)$ random?

$$\text{Expectation} = \mathop{\mathbb{E}}_{w,b}\, \sigma(w^T x + b)\, \sigma(w^T x' + b) = k(x,x')$$

if $m \to +\infty$, and input weights are not optimized

$\Rightarrow$ NN $\Rightarrow$ kernel method

$\Rightarrow$ see blog posts (june/july 2020).