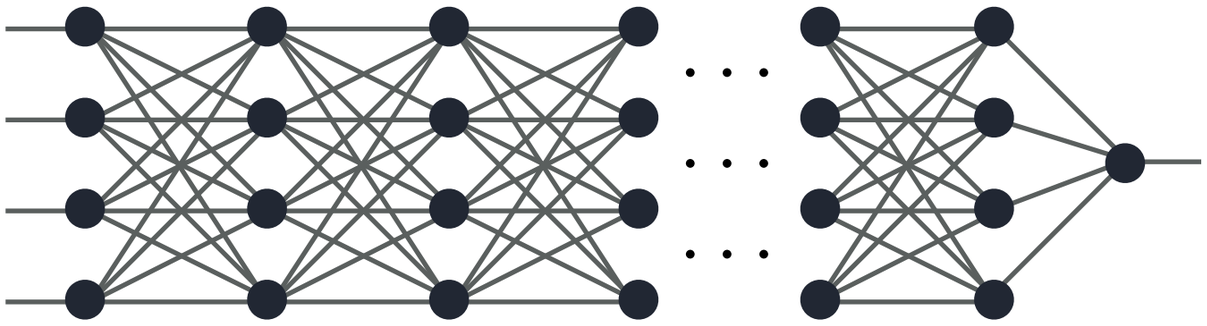
Lower bounds for descent algorithms

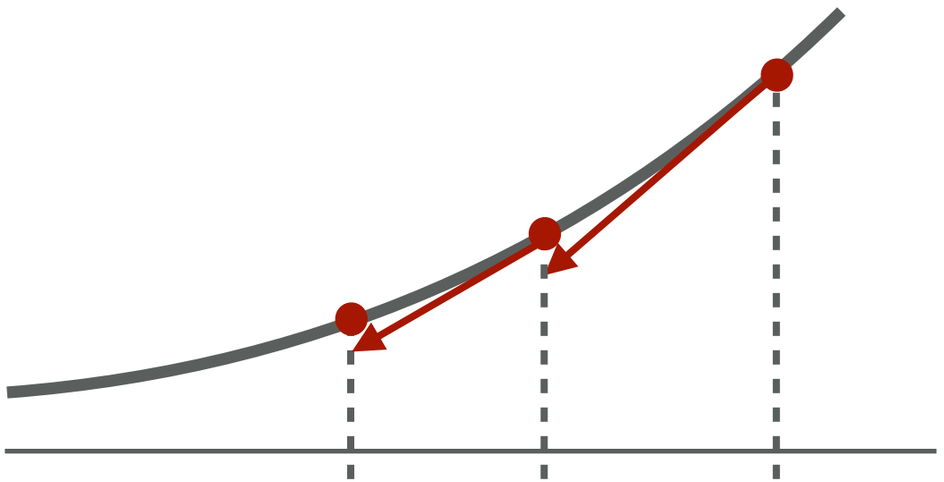
Emmanuel Abbe (EPFL) and Colin Sandon (MIT)

Inference problems: algorithms and lower bounds, 2020

Deep Learning



neural net



initialization

Deep Learning trademark: overparametrized nets and gradient descent

Descent algorithms

Nature picks F

$$W^{(0)} \leftrightarrow F_{W^{(0)}}$$

Choose a net initialization $W^{(0)}$ and a loss function L

For $t = 1, \ldots, T$

Update $W^{(t)} = W^{(t-1)} - \mathbb{E}_{X \sim \hat{P}_{S_m^{(t)}}} \nabla L(\hat{F}_{W^{(t-1)}}(X), F(X)) + Z^{(t)}$

where $S_m^{(t)} = (X_i^{(t)})_{i \in [m]} \overset{\text{iid}}{\sim} P_{\mathcal{X}}$

$\rightarrow$ Hope $\hat{F}_{W^{(T)}}$ approximates F well

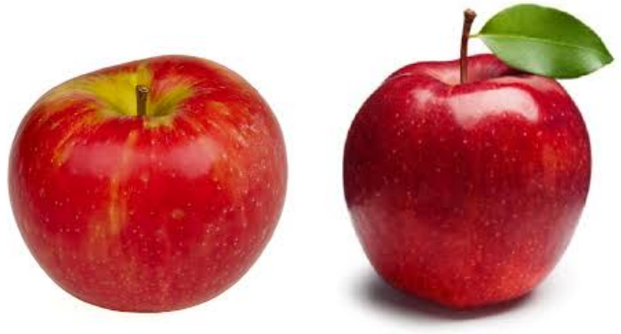
The Deep Learning "miracle"

1. Overparametrization helps producing solutions of low empirical risk (think of random CSPs)

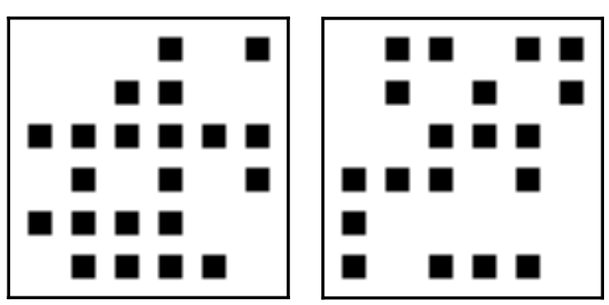
2. Gradient descent seems to reach solutions that are implicitly regularized (without having to add explicitly a regularizer) and that generalize well

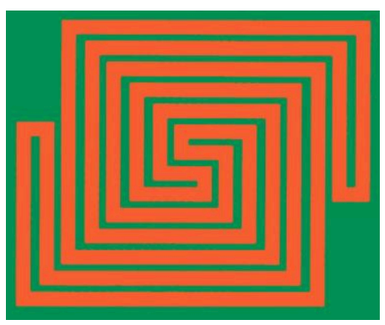
↳ is this specific to certain types of functions/data distributions?

Class 1

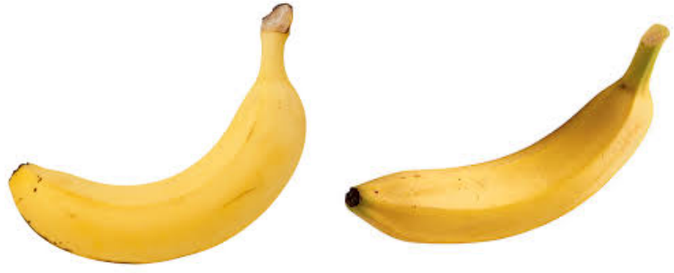


odd

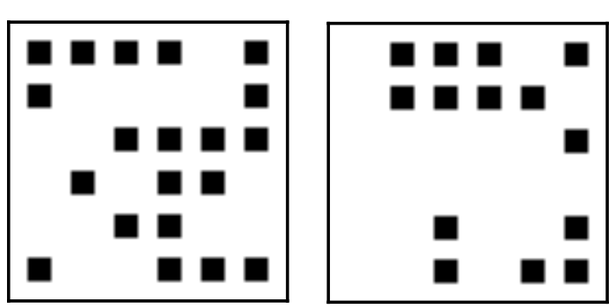




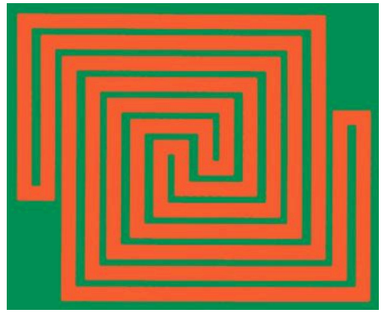
Class 2

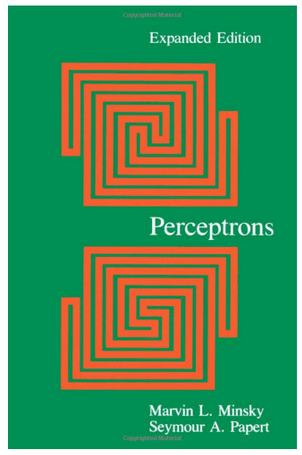


even

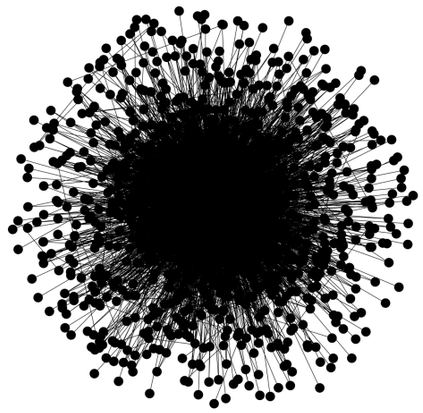


M. Minsky and S. Papert, Perceptrons: an introduction to computational geometry, MIT Press, 1969.

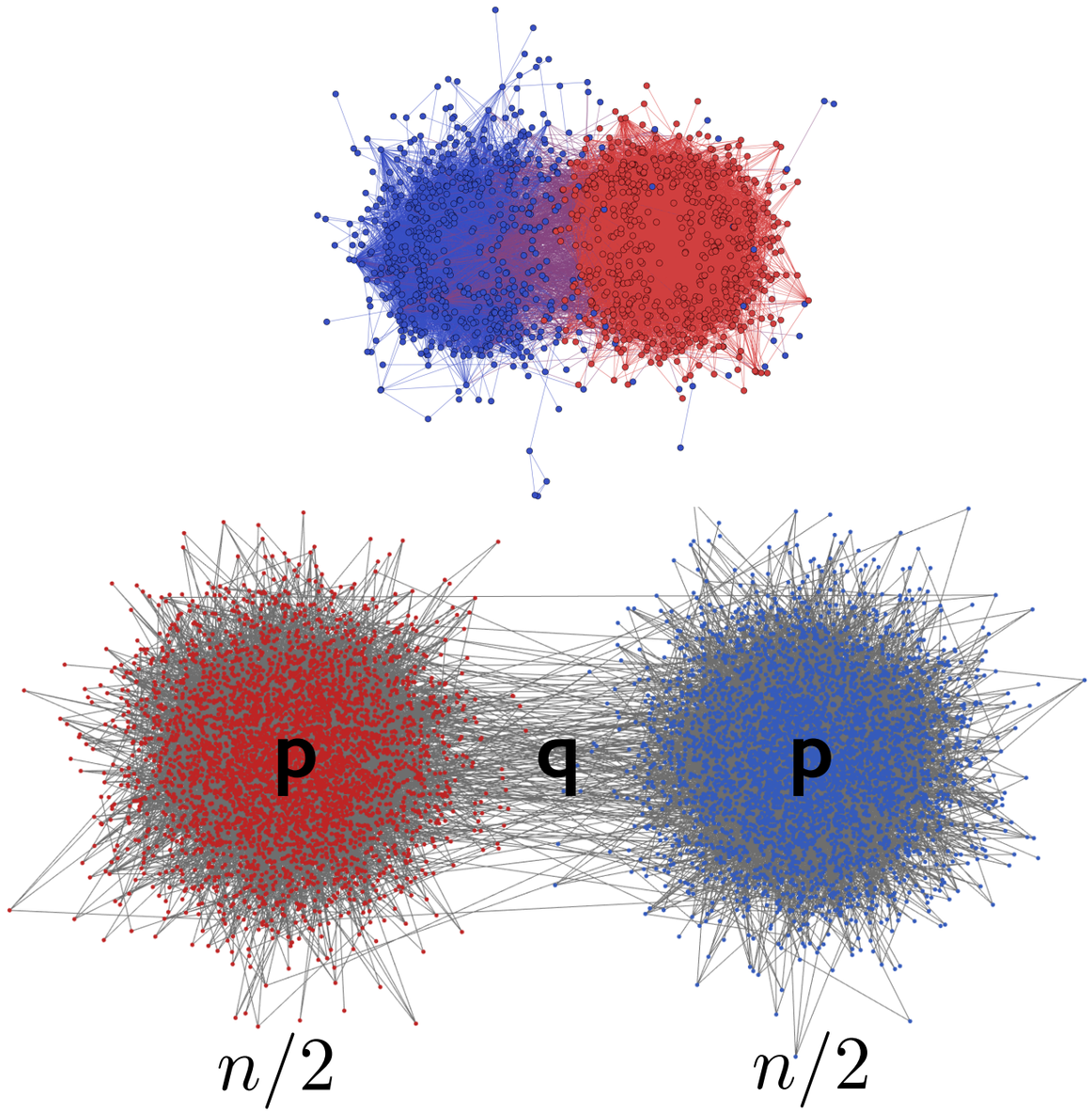




Class 1



Class 2



The Pruned-SBM(n,p,q,r)

Randomly select a cycle of length <r,
delete one of its edges,
repeat until no cycles of length <r.

More serious concerns:

networks, genomics, non-image-based medicine?

Formalizing the problem

Expressibility. Any Boolean function on n variables that can be implemented in poly(n)-time can be expressed by a poly(n)-size NN [Parberry 94, Sipser 06]

-> we cannot allow knowledge on the function **and** freedom on the initialization

Question. Given a **prior** on the set of possible functions, can we learn the function that nature selects with **poly-size NN + poly-time descent**?

$\mathcal{X}$: the data domain $(\{+1,-1\}^n)$ $\quad$ $P_{\mathcal{X}}$: prob. dist. on $\mathcal{X}$

$\mathcal{Y}$: the label domain $(\{+1,-1\})$ $\quad$ $P_{\mathcal{F}}$: prob. dist. on $\mathcal{F} = \mathcal{Y}^{\mathcal{X}}$

Example. Parities: $\quad f_S(x) = \prod_{i \in S} x_i, \quad S \sim_U 2^{[n]}$

$X \sim_U \{\pm 1\}^n \quad \text{label} = f_S(X)$

Objective: generalizing better than guessing

Given $(P_{\mathcal{X}}, P_{\mathcal{F}})$, choose $W^{(0)}, L$ and run SGD for T time steps

$$\rightarrow \text{produce } \hat{F}_{W^{(T)}}$$

Objective: $\mathbb{P}(\hat{F}_{W^{(T)}}(X^{(T+1)}) \neq F(X^{(T+1)})) = 1/2 - \Omega_n(1)$

$X^{(T+1)}$ is a new fresh sample

Failure: $\mathbb{P}(\hat{F}_{W^{(T)}}(X^{(T+1)}) \neq F(X^{(T+1)})) \geq 1/2 - o_n(1)$

Statistical Query (SQ) algorithms

Nature picks $F \in \mathcal{F}$

~~Choose a net initialization $W^{(0)}$ and a loss function L~~

For $t = 1, \ldots, T$

Update ~~$W^{(t)} = W^{(t-1)} - \mathbb{E}_{X \sim \hat{P}_{S_m^{(t)}}} \nabla L(\hat{F}_{W^{(t-1)}}(X), F(X)) + Z^{(t)}$~~

~~where $S_m^{(t)} = (X_i^{(t)})_{i \in [m]} \overset{\text{iid}}{\sim} P_{\mathcal{X}}$~~

$$Y^{(t)} = \mathbb{E}_{X \sim P_{\mathcal{X}}} G_t(X, F(X)) + Z^{(t)}$$

$\text{range}(G_t) = C = 1$
$\text{range}(Z^{(t)}) = \tau$

Kearns '98: Queries can be adaptive, each stored, but are deterministic

Blum et al. 00: If $\mathcal{F}$ has a superpolynomial statistical dimension (number of nearly uncorrelated functions with respect to $P_{\mathcal{X}}$ in $\mathcal{F}$) and the precision is polynomial, then a superpolynomial number of statistical queries is needed.

see also [Boix '20]

Descent algorithms

Nature picks $F \sim P_{\mathcal{F}}$

Choose a net initialization $W^{(0)}$ and a loss function L

For $t = 1, \dots, T$

Update $\boxed{W^{(t)} = W^{(t-1)} - \mathbb{E}_{X \sim \hat{P}_{S_m^{(t)}}} \nabla L(\hat{F}_{W^{(t-1)}}(X), F(X)) + Z^{(t)}}$

where $S_m^{(t)} = (X_i^{(t)})_{i \in [m]} \overset{\text{iid}}{\sim} P_{\mathcal{X}}$

We still make a query - the gradient $G = \nabla L$ - but
- the query can be stochastic
- it must update the ‘specific memory’ with a descent step

-> When do descent algorithms fail to learn?

results

Main result

Theorem (Generalization lower-bound).

$$\mathbb{P}(\hat{F}_{W(T)}(X^{(T+1)}) \neq F(X^{(T+1)})) \geq 1/2 - \frac{1}{\sigma} \cdot \mathrm{JF}_T \cdot \mathrm{CP}_m{}^{1/4}$$

Polynomial if all the algorithm parameters are polynomial

T

$$\mathrm{JF}_T(P_{\mathcal{X}}) := \sum_{t=1} (\mathbb{E}\|G_{t-1}(\hat{F}_{W(t-1)}(X(t)), \mathrm{Rand}(t))\|_2^2)^{1/2} \leq CT\sqrt{|\mathrm{Net}|}$$

Junk-Flow

$$\mathrm{CP}_m(P_{\mathcal{X}}, P_{\mathcal{F}}) := \mathbb{E}_{(X^m, F, F')}(\mathbb{E}_{X \sim P_{X^m}} F(X)F'(X))^2$$

Cross-Predict ility

Can be super-polynomially small if m is super-polynomial

GD-based deep learning is not efficiently universal. SGD?

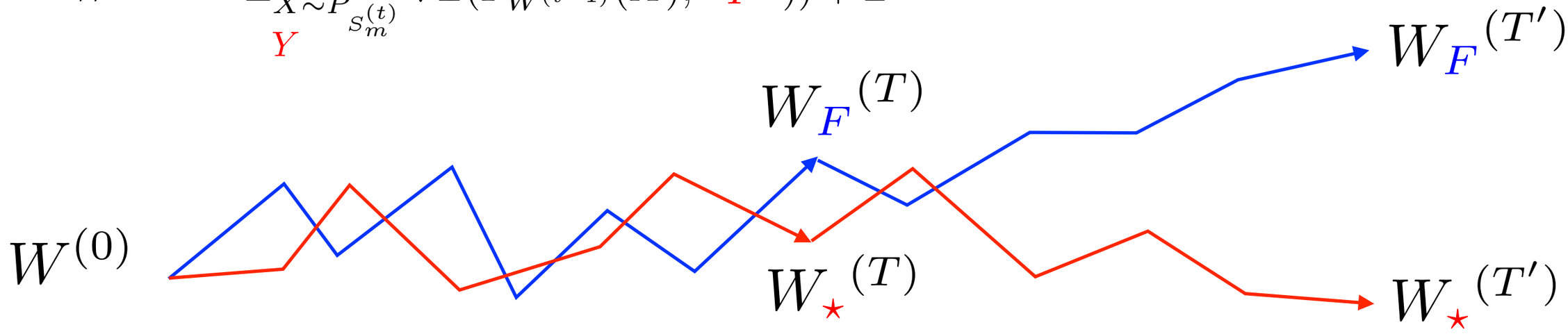
Proof technique

Plan. Show that for some function classes and a small enough horizon, training the net on random data or true data is statistically indistinguishable

Proof technique

Plan. Show that for some function classes and a small enough horizon, training the net on random data or true data is statistically indistinguishable

$$W^{(t)} = W^{(t-1)} - \mathbb{E}_{\substack{X \sim \hat{P}_{S_m^{(t)}} \\ Y}} \nabla L(\hat{F}_{W^{(t-1)}}(X), \quad Y \quad)) + Z^{(t)}$$



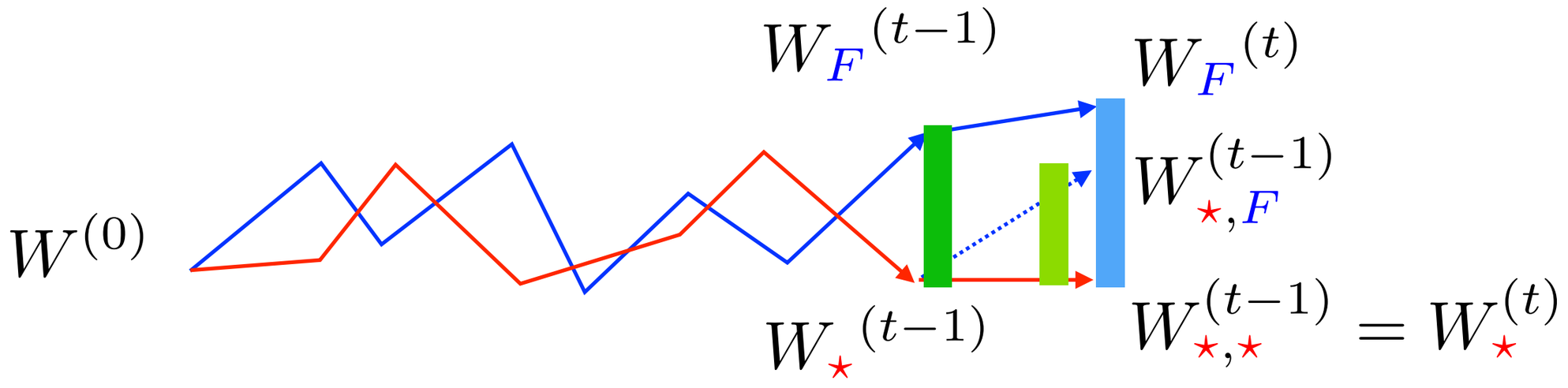
$$\mathbb{P}\{\hat{F}_{W_F{}^{(T)}}(X) = F(X)\} \leq \underbrace{\mathbb{P}\{\hat{F}_{W_\star{}^{(T)}}(X) = F(X)\}}_{1/2} + \underbrace{\mathrm{TV}(W_F{}^{(T)}, W_\star{}^{(T)})}_{\substack{\text{upper-bounded this as} \\ o_n(1) \text{ if } T = n^{O(1)}}}$$

Proof technique

Next. Upper-bound $\mathrm{TV}(W_F{}^{(T)}, W_\star{}^{(T)})$

Lemma 1. Sub-additivity:

$$TV(W_F^{(t)}, W_\star^{(t)}) \leq TV(W_F^{(t-1)}, W_\star^{(t-1)}) + \overbrace{TV(W_{\star,F}^{(t-1)}, W_{\star,\star}^{(t-1)})}^{\text{“one-step” TVs}}$$



Proof. Triangular and Data-Processing inequalities

Proof technique

Next. Upper-bound the **one-step** TV: $\mathrm{TV}(W_F{}^{(1)}, W_\star{}^{(1)})$

Lemma 2. L2-Upper-Bound:

$$TV(W_F{}^{(1)}, W_\star^{(1)}) \leq \frac{1}{2\sigma}(\mathbb{E}_F \|\mathbb{E}_{D_F} \nabla L - \mathbb{E}_{D_\star} \nabla L\|_2^2)^{1/2}$$

Proof.

- **Pinsker's** inequality: Upper-bound TV with **KL**
- This is the KL between two Gaussians:

$$W_{F/\star}^{(1)} = W^{(0)} - \mathbb{E}_{X \sim \hat{P}_{S_m^{(t)}}} \nabla L(\hat{F}_{W^{(0)}}(X), F(X)/Y) + Z^{(0)}$$

  -> KL relies on the **L2-distance** of the means

- **Jensen** to bring the squared-norm: $\mathbb{E}X \leq (\mathbb{E}X^2)^{1/2}$

Proof technique

Next. Upper-bound the **L2-norm**: $\mathbb{E}_F \|\mathbb{E}_{D_F} \nabla L - \mathbb{E}_{D_\star} \nabla L\|_2^2$

Lemma 2. One-edge CP bound:

$$\mathbb{E}_F(\mathbb{E}_{D_F} \nabla_e - \mathbb{E}_{D_\star} \nabla_e)^2 \leq (\mathbb{E}_{D_\star} \nabla_e^2) \cdot \mathrm{CP}_m^{1/2}$$

Proof.

$$\mathbb{E}_F(\mathbb{E}_{D_F} \nabla_e - \mathbb{E}_{D_\star} \nabla_e)^2 = \mathbb{E}_F(\mathbb{E}_{D_\star} \nabla_e (1 - D_F/D_\star))^2 \qquad \text{(Radon-Nikodym)}$$

$$= \mathbb{E}_F \mathbb{E}_{D_\star} \nabla_e^{\otimes 2} (1 - D_F/D_\star)^{\otimes 2} \qquad \text{(Lifting)}$$

$$\leq (\mathbb{E}_{D_\star} \nabla_e^2)(\underbrace{\mathbb{E}_{F,F'} \mathbb{E}_{D_\star} (1 - D_F/D_\star)^{\otimes 2} (1 - D_{F'}/D_\star)^{\otimes 2}}_{\mathbb{E}_{F,F'}(\mathbb{E}_X F(X) F'(X))^2 = \mathrm{CP}_m})^{1/2} \qquad \text{(CS+replica)}$$

$$(\mathbb{E} Z)^2 = \mathbb{E} Z Z'$$

Main result (again)

Theorem (Generalization lower-bound).

$$\mathbb{P}(\hat{F}_{W(T)}(X^{(T+1)}) \neq F(X^{(T+1)})) \geq 1/2 - \frac{1}{\sigma} \cdot \mathrm{JF}_T \cdot {\mathrm{CP}_m}^{1/4}$$

$$\mathrm{JF}_T(P_{\mathcal{X}}) := \sum_{t=1}^{T} (\mathbb{E}\|G_{t-1}(\hat{F}_{W(t-1)}(X(t)), \mathrm{Rand}(t))\|_2^2)^{1/2} \leq CT\sqrt{|\mathrm{Net}|}$$

$$\mathrm{CP}_m(P_{\mathcal{X}}, P_{\mathcal{F}}) := \mathbb{E}_{(X^m,F,F')}(\mathbb{E}_{X\sim P_{X^m}} F(X)F'(X))^2$$

$$= 1/m + (1 - 1/m)\,\mathrm{CP}_\infty$$

$$\mathrm{CP}_\infty := \mathbb{E}_{F,F'}(\mathbb{E}_{X\sim P_{\mathcal{X}}} F(X)F'(X))^2 \overset{\text{parities}}{=} 2^{-n}$$

For **m=∞** -> **GD** cannot learn parities with poly-parameters

For **m=1** the bound gives no failure -> **SGD** could still learn parities

perfect-GD v.s. SGD

- If the initialization is a degree of freedom: **SGD is efficiently universal**

arXiv.org > cs > arXiv:2001.02992

Computer Science > Machine Learning

Poly-time universality and limitations of deep learning

Emmanuel Abbe, Colin Sandon

(Submitted on 7 Jan 2020)

- For **random initializations,** CP strikes even for SGD

- If the initialization is a degree of freedom, **GD** fails when the CP overcomes the JF

> **Corollary.** GD-based deep learning can efficiently learn random monomials of degree k if and only if $k = O(1)$.

$$\mathrm{CP} \asymp n^{-k}$$

- Better bounds on the JF give more failure cases

Random initialization and fixed target functions

If we fail to learn a function no matter what the initialization is,
we fail with a random initialization

If the random initialization is i.i.d. (or exchangeable),
and if the function class is symmetric (permutation invariant),
then we must fail for any chosen function in the class

For example: GD cannot learn $x_1 x_2 \cdots x_{n/2}$ with a random initialization

But this uses a Hammer result for a much more specific case (random init.)
SGD must have stronger limitations with random initializations

thank you