Announcements

1. Let's all take a break!
   On 4/21 I will cover no new material.
   we can just chat about the material
   so far, or your projects etc

2. No office hours on 4/20 though

3. Many of you have talked to me
   about your project in office hours
   or made an appointment to chat
   some other time. Keep doing that!

Lower Bounds for SGD

Today we will delve more into SQs and lower bounds

First, consider an idealized version of SGD for fitting the parameters of a model

Noisy SGD

```
For $t = 1$ to $T$
    Set $\theta_{t+1} = \theta_t - \eta \nabla_\theta R(\theta) + \overset{N(0,\sigma^2 I)}{\overset{\downarrow}{z}}$
```

where $R_\theta = \mathbb{E}_{(x,y)}\left[(h_\theta(x) - y)^2\right]$

__Proposition__: when the marginal distribution on x is known, Noisy GD can be approximated by a CSQ algorithm

Recall that a CSQ is of the form

$$\mathbb{E}[y\, q(x)] \pm \tau$$

To formalize what we mean by "approximated" we need the notion of a coupling:

def: A coupling of two r.v.s X and Y with distributions μ_X and μ_Y respectively is a joint distribution $\mu_{X,Y}$ on (X,Y) pairs

1. the marginal on X is μ_X
2. the marginal on Y is μ_Y

Additionally, when X and Y have the same domain, we call

$$\mathbb{P}_{\mu_{X,Y}}[X = Y]$$

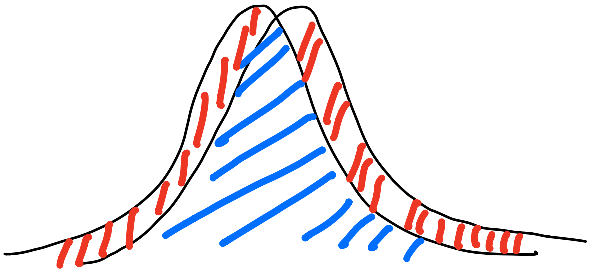
the coupling probability

We will show:

> "If we replace the gradient computation with CSQs, we can almost always couple each step"

In particular, we will be interested in optimal couplings that maximize the coupling probability

These are easy to visualize pictorially:



Fact 1: The optimal coupling probability between two r.v.s. with p.d.f.s. f and g is

$$\int \min(f(x), g(x))\, dx$$
$$= 1 - \frac{1}{2} \underbrace{\int |f(x) - g(x)|\, dx}_{\text{total variation distance}}$$

Fact 2: For two spherical Gaussians

$$d_{TV}\left(\mathcal{N}(\mu, \sigma^2 I), \mathcal{N}(\mu', \sigma^2 I)\right) \leq \frac{\|\mu - \mu'\|_2}{2\sigma^2}$$

Now let's return to the proposition, and see how to set τ along the way

Proof: Let's compute the gradient:

$$\nabla_\theta \mathbb{E}[(h_\theta(x) - y)^2]$$

$$= \mathbb{E}[2(h_\theta(x) - y) \nabla_\theta h_\theta(x)]$$

$$= \underbrace{2\,\mathbb{E}\left[h_\theta(x)\,\nabla_\theta h_\theta(x)\right]}_{(\mathrm{I})} - \underbrace{2\,\mathbb{E}\left[y\,\nabla_\theta h_\theta(x)\right]}_{(\mathrm{II})}$$

Observe that (I) can be computed just from knowledge of the marginal on X — i.e. <u>without CSQs</u>

Now (II) is a p-dimensional vector and each coordinate can be estimated up to $\pm\tau$ with a CSQ

We can usually couple:

$$\begin{matrix}\text{gradient}\\ + \\ \text{noise}\end{matrix} \quad \longleftrightarrow \quad \begin{matrix}\text{vector of CSQs}\\ + \\ \text{noise}\end{matrix}$$

and using Facts 1+2 the failure probability is at most

$$O\left(\frac{\sqrt{p}\,\tau}{\sigma^2}\right)$$

because this is the TV distance of two p-dimensional Gaussians with covariance $\sigma^2 I$ and whose means are at most $\sqrt{p}\,\tau$ in Euclidean distance

Finally if we choose

$$\tau \sim \frac{\delta \sigma^2}{\sqrt{p}\, T}$$

with probability $\geq 1-\delta$ we will be able to couple at every step $\Rightarrow$ we reach the same model parameters. ∎

Thus we have our first lower bound for deep learning:

<u>Corollary</u>: Noisy GD on a polynomial sized deep network must take either exponentially many step or set the noise to be exponentially small to learn parities w.r.t. uniform distribution

These and other lower bounds were shown by Feldman, Guzman, Vempala and Shalev-Schwartz, Shamir, Shamman

What if you don't add your own stochastic noise?

Not only do these proofs break down

<u>Theorem</u> [Abbe, Sandon] SGD is P-complete

i.e. by choosing the appropriate initialization, you can get it to implement any polynomial time algorithm

Proving unconditional lower bounds against SGD $\Rightarrow$ $P \neq NP$

Learning Shallow Deep Nets

we just showed a lower bound against learning parities

Main Question: Are there algorithms for learning shallow deep nets under simple distributions?

we will work with deep nets of the following form

$$f(x) = \sum_{i=1}^{m} a_i \sigma(w_i^T x) \quad (*)$$

where σ is a sigmoid & x is d-dimensional

we will show superpolynomial lower bounds, following Goel, Gollakota, Jin, Karmalkar, Klivans and Diakonikolas, Kane, Kontonis, Zarifis ↖ stronger exponential lower bounds

Theorem 1: Any CSQ algorithm for learning nets of the form (*) under the Gaussian distribution must make at least $d^{\Omega(\log m)}$ queries or set $\tau = d^{-\Omega(\log m)}$

Recall the intuition for parities was

large orthogonal family $\Longrightarrow$ CSQ lower bounds

And we asserted that for Boolean-valued functions where you know the distribution, you can assume all queries are CSQs wlog

Key Question(s): But what about for real-valued labels? Is $SQ >> CSQ$?

For real-valued labels, Szorenyi showed

Theorem 2 (informal) If $\mathcal{H}$ contains a set of ℓ functions that are pairwise orth. under D and have squared norm $\Omega(1) \Rightarrow$ any CSQ algorithm must make either $\Omega(\ell)$ queries or have tolerance $\tau = \ell^{-\Omega(1)}$ to get accuracy $\varepsilon > 0$

Main Question: How do we contruct a large orthogonal family of functions of the form $(*)$?

def: For any set $S \subseteq [d]$ with $|S| = \log m$, let

$$g_S(x) = \sum_{w \in \{\pm 1\}^{\log m}} \chi(w)\, \sigma\left(\frac{w^T x_S}{\sqrt{\log m}}\right)$$

coordinates of x restr. to S

where $\chi(w) = \prod_{i=1}^{\log m} w_i$

Let's show these functions are orthogonal:

Lemma 1: For any $S \neq T$, $\underset{x \sim N(0,I)}{\mathbb{E}}\left[g_S(x)\, g_T(x)\right] = 0$

The key fact we need is

Fact 3: For any $z \in \{\pm 1\}^d$ and x

$$g_S(x \underset{\substack{\uparrow \\ \text{pointwise product}}}{\circ} z) = \chi_S(z)\, g_S(x)$$

Proof: By definition, we have

$$g_S(x \circ z) = \sum_{w \in \{\pm 1\}^{\log m}} \chi(w)\, \sigma\left(\frac{w^T (x \circ z)_S}{\sqrt{\log m}}\right)$$

$$= \sum_{w \in \{\pm 1\}^{\log m}} \chi(w)\, \sigma\left(\frac{(w \circ z_S)^T x_S}{\sqrt{\log m}}\right)$$

$$= \sum_{w \in \{\pm 1\}^{\log m}} \chi(w \circ z_S)\, \chi(z_S)\, \sigma\left(\frac{(w \circ z_S)^T x_S}{\sqrt{\log m}}\right)$$

Now let $w' \triangleq w \circ z_S$. Then

$$= \chi(z_S) \sum_{w' \in \{\pm 1\}^{\log m}} \chi(w') \, \sigma\left(\frac{w' \circ z_S}{\sqrt{\log m}}\right)$$

$$= \chi(z_S)\, g_S(x)$$ ■

Now returning to the lemma

<u>Proof of Lemma 1</u>: Because $\mathcal{N}(0,I)$ is sign symmetric, we have

$$\mathop{\mathbb{E}}_{x\sim \mathcal{N}(0,I)}\left[g_S(x)\, g_T(x)\right] = \mathop{\mathbb{E}}_{z\sim_u \{\pm 1\}^d}\left[\mathop{\mathbb{E}}_{x\sim \mathcal{N}(0,I)}\left[g_S(x\circ z)\, g_T(x\circ z)\right]\right]$$

And from Fact 3, we have

$$= \mathop{\mathbb{E}}_{z\sim_u \{\pm 1\}^d}\left[\mathop{\mathbb{E}}_{x\sim \mathcal{N}(0,I)}\left[\chi_S(z)\,\chi_T(z)\, g_S(x)\, g_T(x)\right]\right]$$

$$= \mathop{\mathbb{E}}_{z\sim_u \{\pm 1\}^d}\left[\cancelto{0}{\chi_S(z)\,\chi_T(z)} \mathop{\mathbb{E}}_{x\sim \mathcal{N}(0,I)}\left[g_S(x)\, g_T(x)\right]\right]$$

■

What remains is to show that each of the functions in the family is non-trivial

Lemma 2: For any S with $|S| \leq \log m$, we have

$$\mathbb{E}_{x \sim N(0,I)} [g_S(x)^2] \geq c^{-m}$$

We will omit the proof. But we can now prove Theorem 1

Proof of Theorem 1: There are $\ell = d^{\Omega(\log m)}$ functions in the orthogonal family

The key point is if we want to learn within error $\varepsilon \triangleq c^{-m}$ then Lemmas 1+2 imply must identify the correct g_S

Now invoking Theorem 2 completes the proof. ☒

Note: There are stronger lower bounds in the probabilistic concept model where

$$x \sim \mathcal{N}(0, I) \text{ and } y \in \{\pm 1\}$$

$$\text{with } \mathbb{E}[y \mid x] = f(x)$$

In this case we get SQ, rather than CSQ, lower bounds

Landscape Design

Next we will study upper bounds for learning

$$y = f(x) = \sum_{i=1}^{m} a_i \sigma(w_i^T x)$$

from Gaussian inputs

Ge, Lee, Ma studied ERM

$$L(\hat{a}, \hat{w}) \triangleq \mathbb{E}[\|y-\hat{y}\|^2] \quad (*)$$

where $\hat{y} = \hat{a}^T \sigma(\hat{w}^T x)$

<u>Theorem</u>: Assuming the w_i's and $\hat{w}_i$'s are unit vectors, the population risk defined in $(*)$ satisfies

$$L(\hat{a}, \hat{w}) = \sum_{k=1}^{\infty} \hat{\sigma}_k^2 \underbrace{\left\| \sum_{i=1}^{m} \hat{a}_i \hat{w}_i^{\otimes k} - \sum_{i=1}^{m} a_i w_i^{\otimes k} \right\|_F^2}_{f_k} + C$$

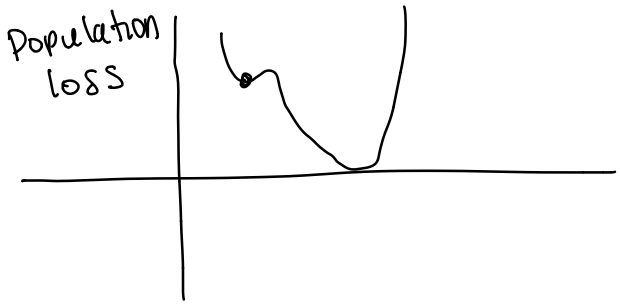
where the inputs come from $\mathcal{N}(0, I)$ and the $\hat{\sigma}_k$'s are coefficients of σ in the Hermite basis

In particular, for $\sigma \equiv \text{ReLU}$ we have

$\hat{\sigma}_1 = \frac{1}{2}$ and for $n \geq 2$

$$\hat{\sigma}_n = \begin{cases} \frac{((n-3)!!)^2}{\sqrt{2\pi n}} & n \text{ even} \\ 0 & n \text{ odd} \end{cases}$$

However, experimentally, SGD fails in the sense that it has spurrious local minima - i.e. it gets stuck



They gave a fix that they called landscape design

Main Question Can we design a new loss function G that has

① no spurrious local minima

② every global minimum of G

$$\Updownarrow$$

ground truth parameters

③ can estimate the gradient of G from samples

Concretel, consider

$$\hat{y}' = \hat{a}^T \gamma(\hat{w}^T x)$$

where $\gamma = \hat{\sigma}_2 \underset{\uparrow}{h_2} + \hat{\sigma}_4 \underset{\uparrow}{h_4}$

(normalized prob.'s) Hermite polynomials

In particular $h_2(t) = \frac{1}{\sqrt{2}}(t^2 - 1)$ and

$$h_4(t) = \frac{1}{\sqrt{24}}(t^4 - 6t^2 + 3t)$$

Now consider

$$G(\hat{a},\hat{w}) = \mathbb{E}[\|\hat{y}' - y\|^2] \quad (\Delta)$$

<u>Theorem</u>: Under the same assumptions as before, the population risk in (Δ) satisfies

$$G(\hat{a},\hat{w}) = \hat{\sigma}_2^2 f_2 + \hat{\sigma}_4^2 f_4 + C$$

Moreover if the a_i's are nonnegative and W is orthogonal then $G(\hat{a},\hat{w})$ satisfies conditions ①, ②, ③ of landscape design

Diakonikolas, Kane, Kontonis, Zarifis gave PCA-based algorithms that just need a_i's to be nonnegative and Gaussian inputs

Key Question: Is nonnegativity of the a_i's actually needed?

The CSQ lower bounds critically need both positive and negative a_i's to get an orthogonal family

Fixed Parameter Tractability

Chen, Klivans, Meka gave a

$$f(m, \underset{\substack{\uparrow \\ \text{Lipschitz constant of deep net}}}{\Delta}, \varepsilon) \text{ poly}(d)$$

time algorcthm for learning deep nets with m hidden units from Gaussian inputs

Key Point: This is much better than the $d^{\Omega(m)}$ CSQ lower bound

Filtered PCA

Consider a matrix

$$M_\psi \triangleq \mathbb{E}\left[\psi(y)\,(xx^T - I)\right]$$

Let $V = \text{span}\{w_i\}$

↑ weight vectors in the first layer

Observe that the output y of the network only depends on $\Pi_V(x)$

Claim 1: For any direction $c \perp V$ we have $c \in \ker(M_\psi)$

The main idea is to show for a suitable choice of τ we have

for any vector c where $|f(c)| \geq \tau$

$$\Downarrow$$

$(\square)$

$$\|\Pi_V(c)\|^2 \geq 2m$$

Then we could pick $\psi(y) \overset{\Delta}{=} \mathbb{1}_{|y| \geq \tau}$

Now let's run PCA on M_ψ to find a direction in V:

Claim 2 Under ($\square$), M_ψ has a positive eigenvalue

Note by Claim 1, we know this direction must be in V

Proof: We will compute the quantity

"sum of the eigenvalues of M_ψ"

This is just the trace, and by Claim 1 we can write this as

$$\mathrm{Tr}(V^T M_\varphi V) \qquad (0)$$

where we have abused notation and V is a $d \times m$ matrix whose columns are an orthonormal basis for the subspace V

Then we have

$$(0) = \mathrm{Tr}(VV^T M_\varphi) = \langle \Pi_V, M_\varphi \rangle$$

where $\langle A, B \rangle$ is the matrix inner-product $\sum_{ij} A_{ij} B_{ij}$

Now plugging in the definition for M_φ, we have

$$(0) = \mathbb{E}\left[\mathbb{1}_{|y| \geq \tau} \left(\langle \Pi_V, xx^T \rangle - \underbrace{\langle \Pi_V, I \rangle}_{m} \right) \right]$$

$$= \mathbb{E}\left[\mathbb{1}_{|y| \geq \tau} \left(\| \Pi_V x \|^2 - m \right) \right]$$

Now using (□) we get

$$\geq \mathbb{P}[|y| \geq \tau] \, m$$

Thus the top eigenvector of M_ψ will be in the subspace V ☒

To put it in words

"If we condition on the response being sufficiently large, the resulting distribution is non-Gaussian"

and we can find such a direction by using PCA

At this point (□) seems optimistic; let's do an illustrative example:

(Assume wlog $\mathbb{E}[f(x)]=0$)

① Let $\sigma(x) = x^2$, quadratic activation

② Let $V = \text{span}\{e_1, \dots, e_k\}$

Now let $\Psi(y) = y^2$. we are interested in

$$\mathbb{E}\left[y^2(x_1^2 + \dots + x_m^2 - m)\right]$$

which we can break up into m terms of the form

$$\mathbb{E}\left[y^2(x_1^2 - 1)\right]$$

Now the key is to use Gaussianity

Lemma [Stein] For any differentiable function g

$$\underset{x\sim \mathcal{N}(0,1)}{\mathbb{E}}\left[g(x)\,x\right] = \underset{x\sim \mathcal{N}(0,1)}{\mathbb{E}}\left[g'(x)\right]$$

Proof: Using integration by parts

$$\mathbb{E}[g'(x)] = \int_{-\infty}^{\infty} g'(x)\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}\right)dx$$

$$= \left.\overset{\to 0}{\cancel{\frac{g(x)\,e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}}}\right|_{-\infty}^{\infty} + \underbrace{\int_{-\infty}^{\infty} x\,g(x)\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}\right)dx}_{\mathbb{E}[x\,g(x)]}$$

∎

Corollary 2: $\mathbb{E}[g(x)(x^2-1)] = \mathbb{E}[g''(x)]$

Proof: we have

$$\mathbb{E}[g(x)(x^2-1)] = \mathbb{E}[g(x)x^2] - \mathbb{E}[g(x)]$$

$$\overset{\text{stein's lemma}}{=} \mathbb{E}\left[\frac{\partial}{\partial x}(g(x)x)\right] - \mathbb{E}[g(x)]$$

$$= \mathbb{E}[g'(x)x] \overset{\text{stein's lemma}}{=} \mathbb{E}[g''(x)]$$

∎

Now applying Corollary 2

$$\mathbb{E}\left[f(x)^2 (x_1^2 - 1) \right] = \mathbb{E}\left[\frac{d^2}{dx_1} f(x)^2 \right]$$

$$= \underbrace{2\, \mathbb{E}\left[\left(\frac{d}{dx_1} f(x) \right)^2 \right]}_{>0} + \underbrace{2\, \mathbb{E}\left[f(x) \frac{d^2}{dx_1^2} f(x) \right]}_{0}$$

which again implies the top eigenvector of M_φ is in V