18.408: Theoretical Foundations for Deep Learning

Course Goal: ~~Fully explain why deep learning works, with provable guarantees~~

More modestly:

Learn about the foundational tools and frameworks for reasoning about ML algorithms, what they have say about DL, and where they fall short

Depending on your backgrounds, might be wondering:

> "Why do we want to prove theorems about deep learning? Isn't it enough that it just works?"

If you give me well-defined objectives like

(1) predict the label of this image

(2) find the best next move in Go

(3) predict the 3D-structure of this protein from its chemical sequence

Then deep learning is your best bet

But outside of this sandbox, there is a lot more you care about:

Is what I've learned:

Robust?

Interpretable?

Fair?

Transferrable?

These can't be translated into well-defined objective functions

In this class, we will use theory to probe what we think we know about DL

Four main parts

(1) Approximation: what kinds of functions can be represented by deep networks?

How does the expressivity change when you change the architecture (depth)?

(2) Optimization: How do we algorithmically fit a model to data?

We solve an optimization problem, but it's nonconvex and its character depends on the architecture (and data)

(3) Generalization: why do deep networks that interpolate the data still work well on new data?

Baffling empirical phenomena: why does

increasing the size improve generalization?

(4) Representation Learning: what are good models for natural data?

Even for simple learning problem, we need some articulation of why natural data is "nice"

My view: Future progress will come from finding ways to integrate and synthesize these perspectives

Learning Theory Foundations

What is learning theory? we want to make good predictions on future data

The usual setup is:

(1) An unknown distribution μ on $\mathbb{R}^d \times \{\pm 1\}$

(2) A training sample

$x_1, x_2 \dots x_n$

with labels $y_1, y_2 \dots y_n$, $(x_i, y_i) \sim \mu$

(3) A loss function

$\ell(\hat{y}, y)$

($\hat{y}$: our prediction)

Common examples include

square loss: $\ell(y, \hat{y}) \triangleq \frac{(y-\hat{y})^2}{2} = \frac{(y\hat{y}-1)^2}{2}$

logistic loss: $\ell(y, \hat{y}) \triangleq \ln(1+e^{-y\hat{y}})$

With this notation, we care about the population risk, which is

$$R(f) \triangleq \mathbb{E}_{(x,y)\sim\mu}[\ell(f(x), y)]$$

Goal: Give an algorithm that learns from a reasonable (polynomial) amount of data and achieves low risk

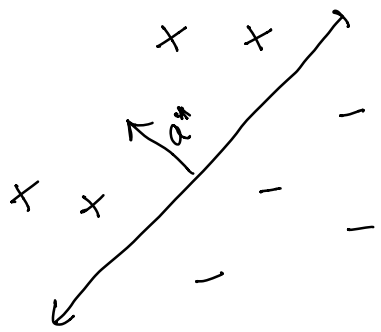
we still need to make an assumption about how the true labels are chosen

Realizable case: There is a known class of functions $\mathcal{H}$ and we have

$$y = f^*(x) \text{ , for some } f^* \in \mathcal{H}$$

Some important examples

① halfspaces $f^*(x) = \text{sgn}(a^{*T}x + b^*)$

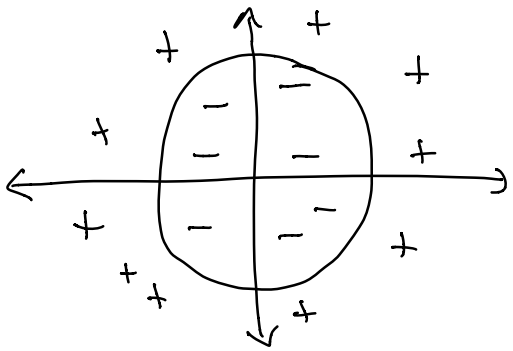


② More generally, halfspaces in an RKHS (reproducing kernel Hilbert space)

specified by a kernel mapping

$$\phi : \mathbb{R}^d \to \mathbb{R}^m$$

Now we can find nonlinear decision rules, like:



If we choose $\phi(x) \triangleq \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \\ x_2^2 \\ x_1 x_2 \end{bmatrix}$ $\uparrow \mathbb{R}^2$

then the decision rule is

$$\text{sgn}\left([0\ 0\ 1\ 1\ 0]\, \phi(x) - 1 \right)$$

$$= \pm 1 \mathbb{1}_{x \text{ is outside the circle}}$$

Another more flexible assumption about the labels is called the probabilistic concept model, of Kearns and Shapire

Now $\mathcal{H}$ is a class of functions $f: \mathbb{R}^d \to \mathbb{R}$

$$\mathbb{E}[y|x] = f(x)$$

Thus the label is no longer a deterministic function of the feature vector, but, crucially the noise has expectation zero

Universal Approximation

We will often work with the following basic model of deep networks:

$$f(x; w) = \sigma_L(W_L \sigma_{L-1}(\dots \sigma_1(W_1 x + b_1) \dots))$$

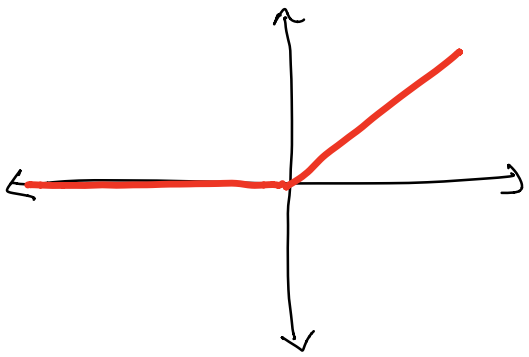
where:

(1) each $W_i \in \mathbb{R}^{n_i \times n_{i-1}}$ and $b_i \in \mathbb{R}^{n_i}$

(2) each σ_i is a coordinate-wise nonlinear activation function

e.g. rectified linear unit (ReLU)

$$\sigma(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}$$



sigmoid

$$\sigma(x) \triangleq \frac{1}{1+e^{-x}} = \frac{e^x}{1+e^x}$$

we will bundle all of these parameters together:

$$W = (W_1, b_1, \ldots W_L, b_L)$$

with "depth" $= L$ and "width" $= \max_i d_i$

Coming back to learning theory, the basic idea of DL is to train a deep network on our samples, and use it to make future predictions

Main Question: what class of functions can we express with deep networks?

Answer: Everything (that is continuous)

def: we say a class of functions $\mathcal{H}$ is a universal approximator if for any continuous fnctn g and compact domain D and any $\varepsilon > 0$, $\exists f \in \mathcal{H}$

$$|f(x) - g(x)| \leq \varepsilon \quad \forall x \in D$$

From analysis you might remember:

Theorem: Polynomials (with unbounded degree) are universal approximators

This follows from the Stone-Weierstrauss Theorem:

Theorem: suppose $\mathcal{H}$ satisfies the following prop:

1. each $f \in \mathcal{H}$ is continuous
2. $\forall x \in D$, $\exists f \in \mathcal{H}$ with $f(x) \neq 0$
3. $\forall x \neq x' \in D$, $\exists f \in \mathcal{H}$ with $f(x) \neq f(x')$
4. $\mathcal{H}$ is closed under $\times$ and vector space ops.

$\Rightarrow$ $\mathcal{H}$ is a universal approximator

Deep networks give us an alternative to polynomials

Meta Theorem: For many activation functions, depth two networks of unbounded width are universal

e.g. we will prove it for ReLUs, and Hornik et al showed it holds for any continuous σ with

$$\lim_{z \to -\infty} \sigma(z) = 0 \quad \text{and} \quad \lim_{z \to +\infty} \sigma(z) = 1$$

which covers sigmoids

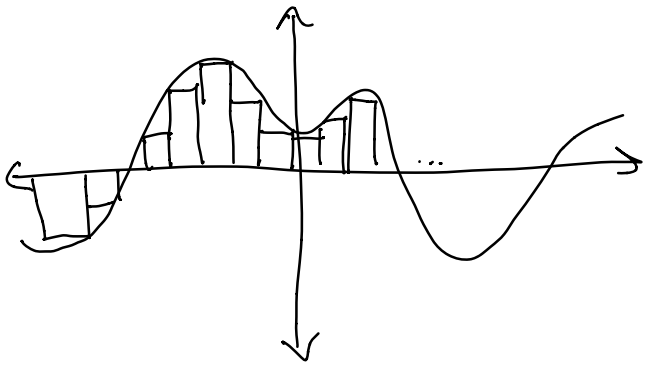
Univariate Approximations

def: A function $g: \mathbb{R} \to \mathbb{R}$ is β-Lipschitz if

$$\forall x, y \quad |g(x) - g(y)| \leq \beta |x - y|$$

Any Lipschitz function can be approximated arbitrarily well by step functions:

Proof by picture:



Proposition: If $g: \mathbb{R} \to \mathbb{R}$ is P-Lipschitz $\exists$ a depth two network with $\lceil \frac{P}{\varepsilon} \rceil$ hidden nodes that ε-approximates g on $[0,1]$, with threshold activations $\sigma(x) = \mathbb{1}_{x \geq 0}$

Note: Similar bounds hold for other activation functions

Proof: Consider a network of the form

$$\hat{g}(x) = \sum_{i=1}^{m} a_i \mathbb{1}_{x - b_i}$$

where $m = \lceil \frac{P}{\varepsilon} \rceil$ and b_i's are evenly spaced, $b_i = \frac{(i-1)\varepsilon}{P}$

set $a_1 = g(x_1) = g(0)$, $a_2 = g(x_2) - g(x_1) \ldots$

Now let's check that this construction works: Consider an arbitrary $x \in [0,1]$ and let x_i be the largest i with $x_i \leq x$

$$|g(x) - \hat{g}(x)| = |g(x) - \hat{g}(x_i)|$$

$$\leq \underbrace{|g(x) - g(x_i)|}_{①} + \underbrace{|g(x_i) - \hat{g}(x_i)|}_{②}$$

$$① \leq P|x - x_i| \leq \frac{P\varepsilon}{P} = \varepsilon$$

$$(2) = \left| g(x_i) - g(x_1) - \sum_{j=2}^{i} \left(g(x_j) - g(x_{j-1}) \right) \mathbb{1}_{x_i \geq x_j} \right|$$

$$= \left| g(x_i) - g(x_1) + g(x_1) - g(x_2) \dots \right| = 0$$

∎

Multivariate Approximations

The approach in higher dimensions is a bit different because we need to create a "bump"

Note: Again, the end results will hold for many activation functions, but let's work with w/e is the cleanest

In this part, let's work with trigonometric functions

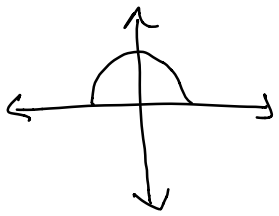
$$\sigma(x) = \cos(x)$$

Why? Recall from Stone-Weierstrauss it is useful to be able to multiply and remain in the class

In particular, the following identity will be key:

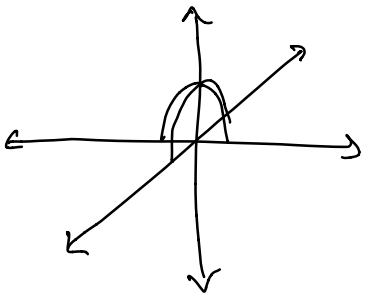
$$\cos(x)\cos(y) = \frac{\cos(x+y) + \cos(y-x)}{2}$$

Thus when we have a "bump" in 1-d like

$\cos(x) \longrightarrow$ 

we can create a "bump" in higher dimensions just by multiplying

$$\prod_{i=1}^{d} \cos(x_i)$$



and we can use the identity above to express this higher-d "bump" as a shallow network

e.g.

$$\cos(x_1)\cos(x_2)\cos(x_3) = \frac{\cos(x_1 \pm x_2 \pm x_3)}{4}$$

depth two network with four hidden units

Now let's give a multivariate approximation theorem

Note: We could just apply Stone-Weierstrauss. However what we really care about are quantitative bounds on how big our network is

Proposition: Let $g: \mathbb{R}^d \to \mathbb{R}$ be continuous and suppose $\varepsilon, \delta > 0$ s.t.

$$\|x - y\|_\infty \leq \delta \Rightarrow |g(x) - g(y)| \leq \varepsilon$$

$\exists$ a depth two network (with cosine act.) f with $\frac{C}{\delta^d}$ hidden units and satisfies $\sup_{x \in [0,1]^d} |f(x) - g(x)| \leq \varepsilon$

we won't prove exactly this, because it gets messy Instead:

Simple Proposition: For any partition P

$$P = (R_1, \ldots R_N)$$

of $[0,1]^d$ into rectangles with all side lengths $\leq \delta$, there are scalars $\alpha_1 \ldots \alpha_N$ s.t.

$$\sup_{x \in [0,1]^d} |g(x) - f(x)| \leq \varepsilon \quad \text{where} \quad f(x) = \sum_{i=1}^{N} \alpha_i \mathbb{1}_{x \in R_i}$$

Proof: For each R_i, pick any $x_i \in R_i$ and let

$$\alpha_i = g(x_i)$$

Then we have:

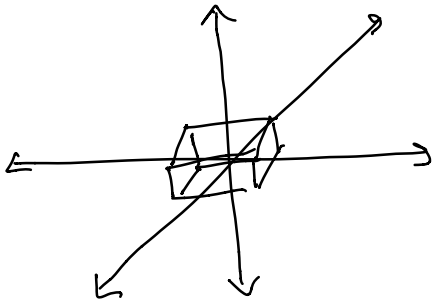
$$\sup_{x \in [0,1]^d} |g(x) - f(x)| = \sup_{i \in [N]} \sup_{x \in R_i} |g(x) - f(x)|$$

$$\leq \sup_{i \in [N]} \sup_{x \in R_i} \left(|g(x) - g(x_i)| + \cancel{|g(x_i) - f(x)|}^{0} \right)$$

$\leq \varepsilon$

Each rectangle has volume $\sim \delta^d \Rightarrow$ can set $N = \frac{1}{\delta^d}$

A rectangle is an idealized "bump"



For the full proposition, need to approximate it with cosines and worry about the spillover

↑ can get a good approximation via Fourier analysis

The important point is:

universal approximation $\Rightarrow$ $\frac{1}{\delta^d}$ **hidden nodes**

Thus it is essentially a lookup table:

To compute $g(x)$:

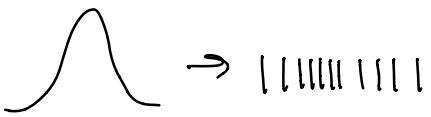
Find x_i with $\|x_i - x\|_\infty \leq \delta$

| output $g(x_i)$

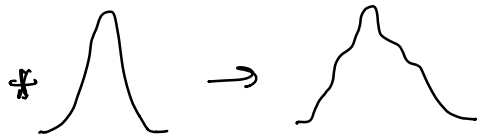
Comment: This type of "curse of dimensionality" comes up in statistical problems too

e.g. How can I approximate the density $f(x)$ given samples from it?

① Form empirical distribution



(2) Convolve with smooth bump, e.g. Gaussian



This is called a kernel density estimate — rich theory of what smoothness assumptions (e.g. Besov spaces) or other structure (function supported on manifold) lead to better bounds

Epilogue: what if you fix the width and send the depth to infinity

Theorem: [Park et al] For any square integrable function $f: \mathbb{R}^d \to \mathbb{R}$ and any $\varepsilon > 0$, $\exists$ is a width $d+1$ network with ReLU activations

$$\int |f(x) - \hat{f}(x)|^2 dx \leq \varepsilon$$

Looking Forward

The answer to the question "what functions can you express with deep nets?" is somewhat vacuous

A better question is:

> what kinds of functions can we succinctly express with deep nets?

Barron's Theorem

Barron's seminal work identified a rich class of functions that you can approximate by a small depth two network

For a function $f: \mathbb{R}^d \to \mathbb{R}$ we write its fourier transform as

$$\hat{f}(\omega) \triangleq \int e^{-2\pi i \omega^T x} f(x) dx$$

Note that $\hat{f}: \mathbb{R}^d \to \mathbb{R}$ too. This transform is invertible in the sense that

$$f(x) = \int e^{2\pi i \omega^T x} \hat{f}(\omega) d\omega$$

under technical conditions on $f, \hat{f}$ (e.g. both are in L_1)

Now we can state Barron's Theorem:

Theorem: For a function $f: \mathbb{R}^d \to \mathbb{R}$ $\exists$ is a depth two network with k hidden units that computes g

$$\int |f(x) - g(x)|^2 dx \leq \frac{4 C_f}{k}$$

where $C_f = \|\hat{\nabla} f\|_1^2 = \left(\int \|w\| |\hat{f}(w)| dw \right)^2$

"the square integral of the Fourier tranform of the gradient of f"

A modern framing of the proof is:

① rewrite the fourier inversion formula as a depth two infinite width network

② sample from this representation to find a good succinct approximation

we will start from ②, which is called Maurey's Lemma

Theorem: Suppose $X \in \text{conv}(S)$

X: vector of dimension d

S: set of vectors of dimension d, possibly infinite

then $\exists$ a convex combination

$$\|X - \sum_{i=1}^{k} \alpha_i v_i\|^2 \leq \frac{\sup_{v \in S} \|v\|^2}{k}$$

where $v_i \in S$; $\sum_{i=1}^{k} \alpha_i v_i$: convex comb.

Actually this theorem holds even in infinite dimensions (i.e. X and v_i's are functions)

Proof: If $x \in \text{conv}(S)$ then there is a r.v. V supported in S with

$$X = \mathbb{E}_{\mu}[V]$$

Let $v_1 \dots v_k$ be iid draws of V. Then

$$\mathbb{E}\left[\|X - \frac{1}{k}\sum_{i=1}^{k} v_i\|^2\right] = \frac{\mathbb{E}\left[\|\sum_{i=1}^{k} X - v_i\|^2\right]}{k^2}$$

$$\overset{\text{mean zero}}{=} \frac{\sum_{i=1}^{k} \mathbb{E}\left[\|X - v_i\|^2\right]}{k^2} = \frac{\mathbb{E}\left[\|X - V\|^2\right]}{k} = \frac{\mathbb{E}\left[\|V\|^2 - \|X\|^2\right]}{k}$$

$$\leq \frac{\mathbb{E}\left[\|V\|^2\right]}{k} \leq \frac{\sup_{v \in S} \|v\|^2}{k}$$ ☒

we used the fact that

$$X = \mathbb{E}_{\mu}[V]$$

but we could just as well allow X to be rep.

as an arbitrary linear combination of elements in S

Abusing notation, also call this M. Then the same proof would give:

$$\left\| x - \sum_{i=1}^{k} \alpha_i v_i \right\|^2 \leq \frac{\|M\|_{\ell^1}^2 \sup_{v \in S} \|v\|^2}{k}$$

$\uparrow$ now these are general weights, not necessarily nonnegative

Now let's work towards (1) and massage the fourier inversion formula into a useful form

Proposition: For $\|x\| \leq 1$ we have

$$f(x) - f(0) = -\int_0^{\|w\|} \int \mathbb{1}_{w^T x \geq b} \frac{\sin(2\pi b + 2\pi\theta(w))}{\|w\|} \|\widehat{\nabla f(w)}\| \, db \, dw$$

$$+ \int_{-\|w\|}^{0} \int \mathbb{1}_{w^T x \leq b} \frac{\sin(2\pi b + 2\pi\theta(w))}{\|w\|} \|\widehat{\nabla f(w)}\| \, db \, dw$$

where $\hat{f}(w) = |\hat{f}(w)| e^{2\pi i \theta(w)}$, i.e. $\theta(w)$ is the angle

First, why is this useful?

It is an infinite width, depth two rep.

Next time: finish Barron's theorem, and do depth separations