PERSONAL ENCOUNTERS WITH MACHINE LEARNING

– Charles Fefferman 4/2026

This talk will recount two old stories on results related to machine learning.

The two results I'll discuss are unrelated, but the human element exhibits a common theme.

I hope you'll enjoy the stories, and not be put off by the fact that the results aren't new.

The two stories concern

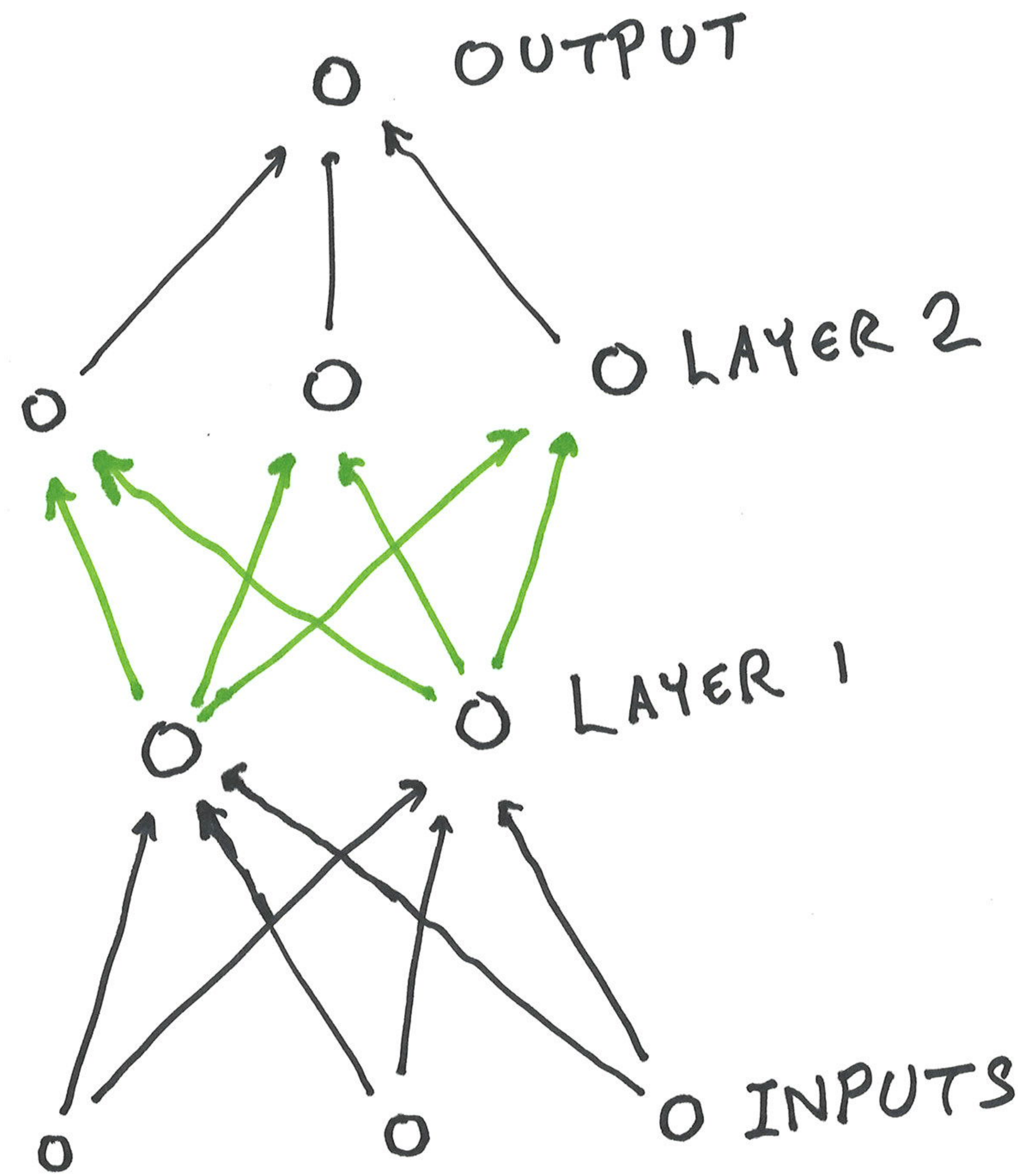
THE UNIQUENESS PROBLEM FOR NEURAL NETS

and

FITTING SMOOTH FUNCTIONS TO DATA.

The Uniqueness Problem for Neural Nets

A NEURAL NET



EACH WIRE IS MARKED WITH A REAL NUMBER (a "WEIGHT").

EACH NODE IS MARKED WITH A REAL NUMBER (a "THRESHOLD")

WHAT A NODE DOES

Suppose a given node, marked with threshold θ, receives inputs

$x_1, \cdots, x_k$

from wires marked with weights

$w_1, \cdots, w_k$, respectively.

Then the node outputs

$$\sigma(w_1 x_1 + \cdots + w_K x_K + \theta)$$

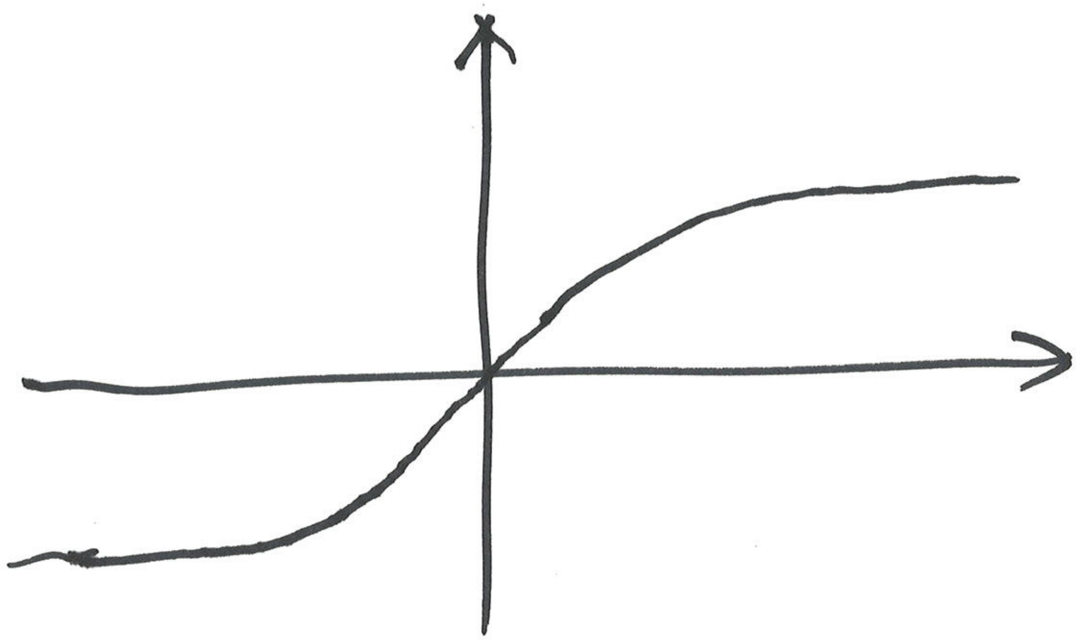
for a fixed standard non-linear function σ.

ONE POPULAR

(ONCE POPULAR ?) σ

is

$$\sigma(t) = \tanh(t)$$



MOTIVATED BY ANALOGY

TO BRAIN

The topology of the wires

HOW MANY LAYERS ?
HOW MANY NODES
IN EACH LAYER ?

is called the

ARCHITECTURE

of the NET.

So a NEURAL NET
is specified by its
architecture,
and its
weights & thresholds

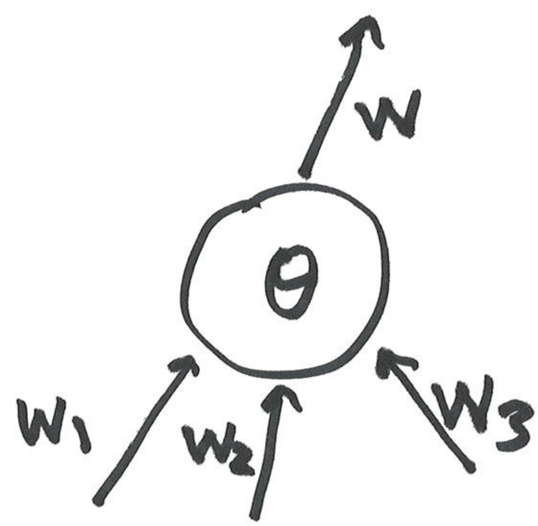
THE UNIQUENESS PROBLEM

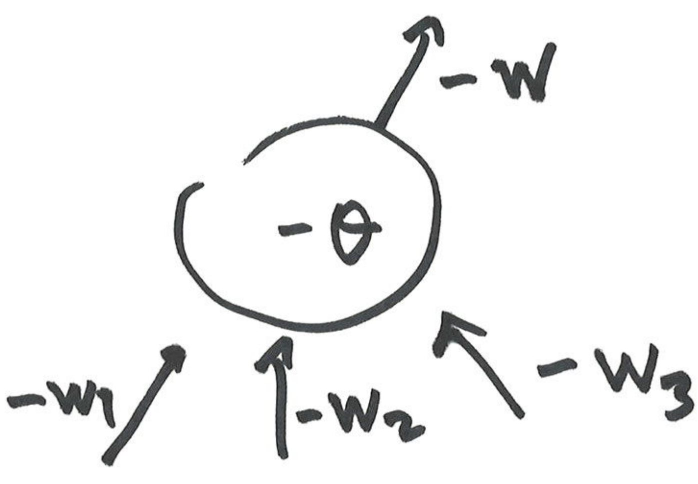
FOR NEURAL NETS:

If two NEURAL NETS DO EXACTLY THE SAME THING, then DO THEY HAVE THE SAME WEIGHTS & THRESHOLDS (up to obvious equivalence)?

Obvious Equivalences

- PERMUTING THE NODES IN A GIVEN LAYER LEADS TO 2 NETS THAT OBVIOUSLY DO THE SAME THING.





ANOTHER OBVIOUS EQUIVALENCE

We should avoid
silly redundancy

e.g., if
two nodes in the same layer
use the exact same
weights & thresholds.

So we will assume that the ratio of two weights is never a rational number with small denominator.

UNIQUENESS THM

Suppose two NEURAL NETS $\mathcal{N}, \mathcal{N}'$ compute the same function, and are free of silly redundancies.

Then $\mathcal{N}$ and $\mathcal{N}'$ differ only by obvious equivalences.

The Proof

(HUMAN ELEMENT).

IDEA OF THE PROOF

ANALYTIC CONTINUATION !

Think about how the NEURAL NET responds to Complex inputs.

WLOG,

OUR NEURAL NETS

HAVE A SINGLE

INPUT & A SINGLE

OUTPUT.

So we deal with fns.

of ONE complex variable.

The function

$\sigma(t) = \tanh(t)$

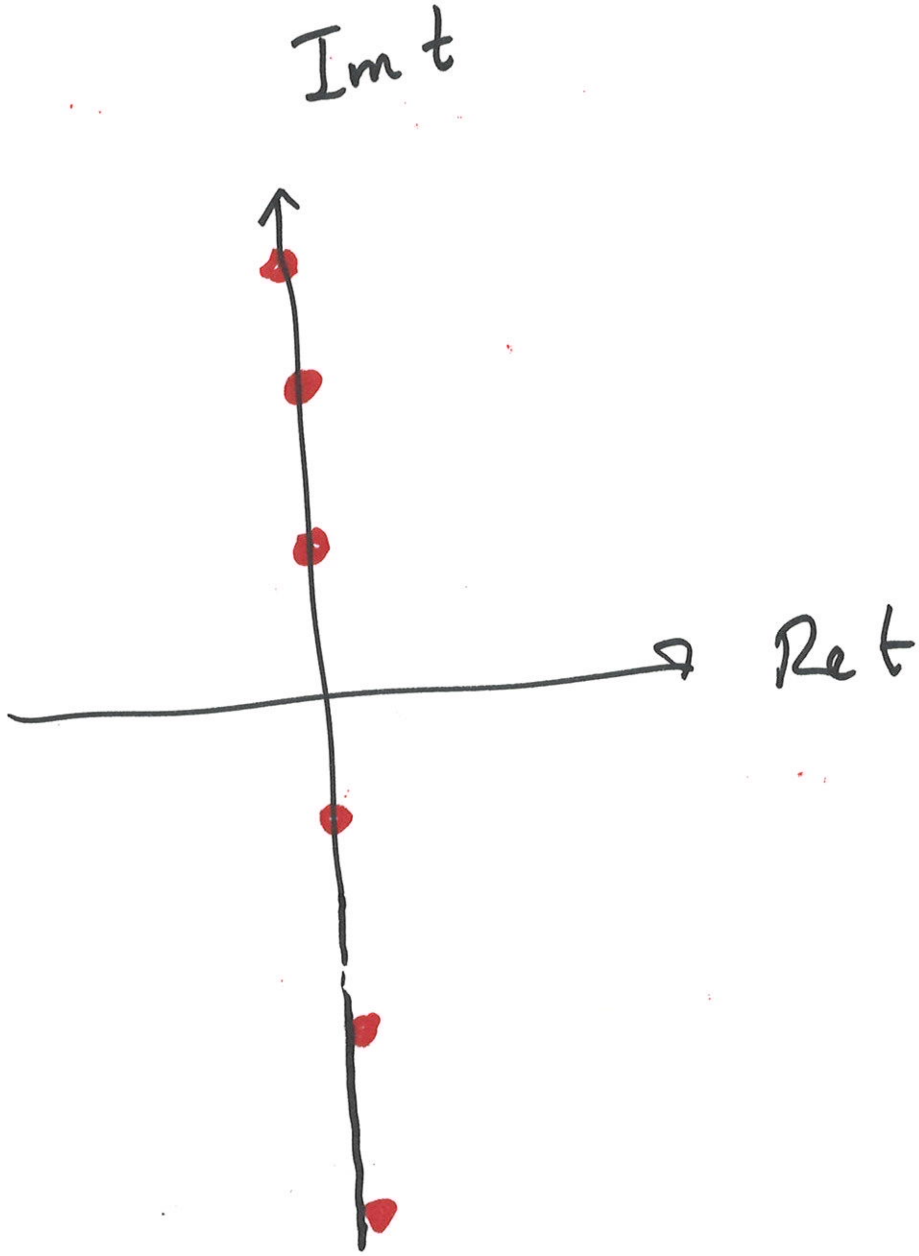
has poles at the

points of an

arithmetic progression

in the complex plane

$t = (2k+1)\cdot\pi i$


Im t
Re t

So, if the input is t,

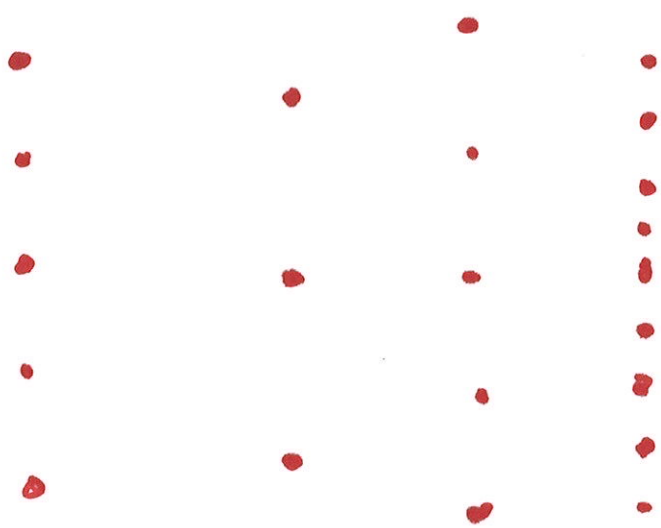
and a node in the first hidden layer has wt w & threshold θ,

then the output of that node has a pole at the points $t \in \mathbb{C}$ such that

$$wt + \theta = (2k+1) \cdot i\pi$$

INTEGER

So the set of poles of all the outputs from the nodes of the 1rst layer looks like this:



From that pattern, we can read off the 1rst layer.

Now let's look at a node in the SECOND LAYER.

It receives inputs $x_1, \ldots, x_K$ from the 1^{st} LAYER

and outputs

$$\tanh\left(w_1 x_1 + \cdots + w_K x_K + \theta \right),$$

where $x_1 = x_1(t)$, $x_2 = x_2(t)$, etc.

Let t_0 be a pole of $x_1(t)$.

With any luck, t_0 is **NOT** a pole of $x_2(t), \ldots, x_K(t)$.

Therefore,

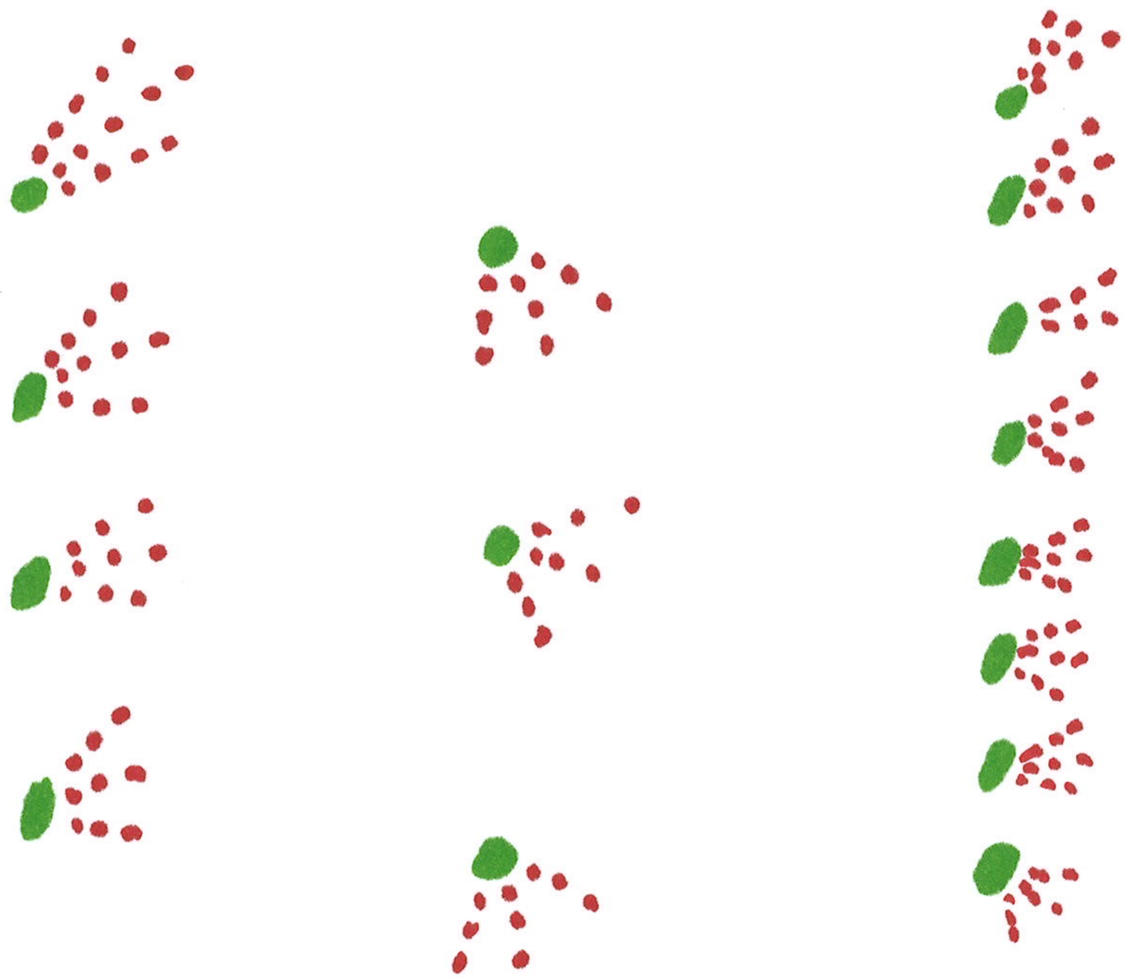
$$\boxed{\begin{aligned} &W_1 x_1(t) + \cdots + W_K x_K(t) + \theta \\ &= \frac{\lambda}{t - t_0} + \text{ANALYTIC}(t) \\ &\text{NEAR } t = t_0 \end{aligned}}$$

Therefore, the output of the node in the 2nd layer has a pole at values of t for which

$$\frac{\lambda}{t - t_0} + \text{ANALYTIC}(t) = (2k+1) \cdot \pi i$$

INTEGER

HERE'S A PICTURE OF THE POLES OF THE OUTPUTS OF THE NODES IN THE SECOND LAYER



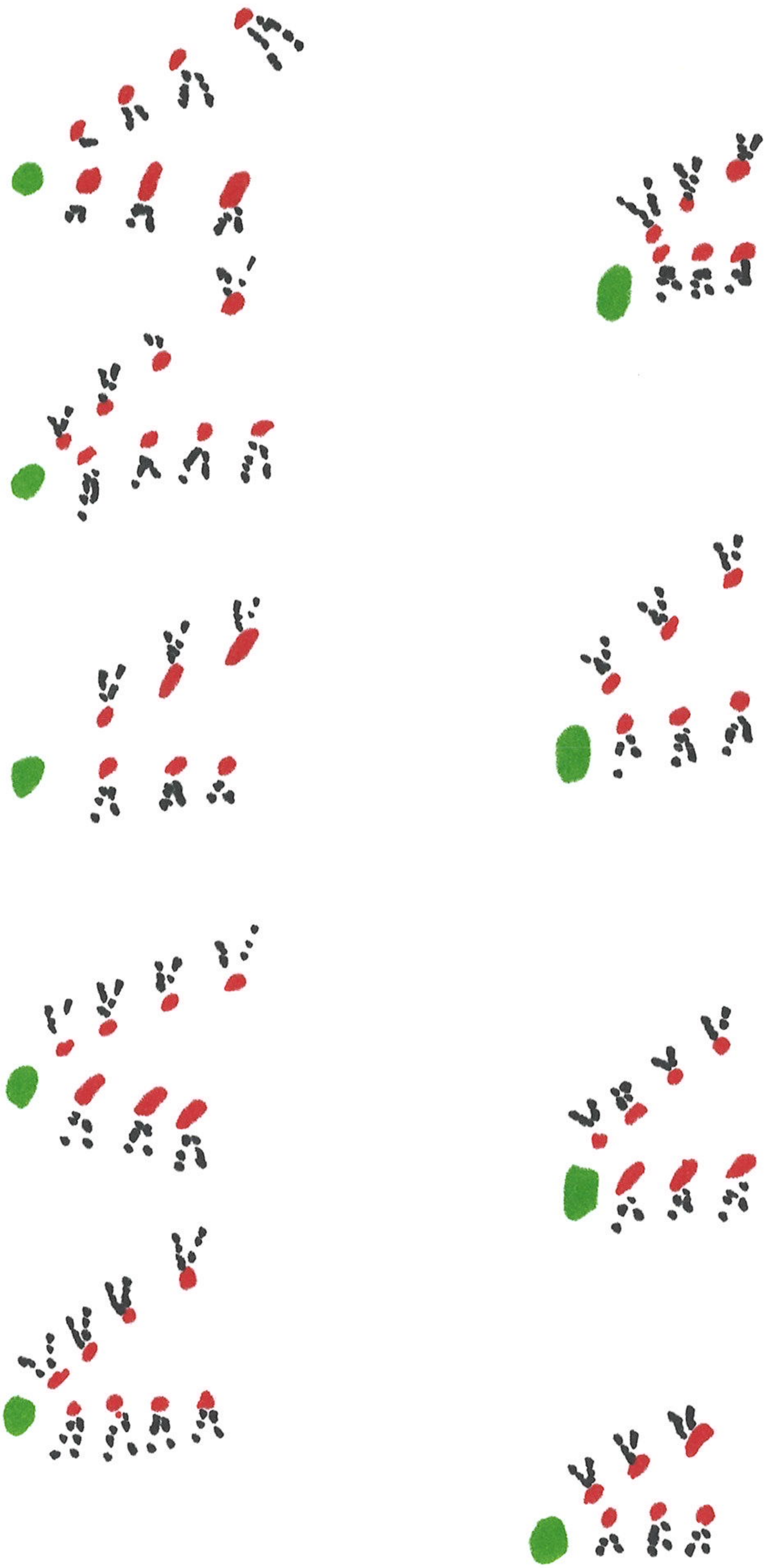
• POLES

• ESSENTIAL SINGULARITIES

From the picture, we obtain COMPLETE INFORMATION about the 1rst & 2nd Layers.

Continue in this way.

The function computed by the neural net has singularities in the complex plane that look like this:



So the singularities of the output map let us read off complete information about our neural net.

This is only an intuitive discussion. It included the phrase "WITH ANY LUCK".

But with some effort, one can make it into a rigorous proof.

FITTING SMOOTH FUNCTIONS TO DATA

THE PROBLEM

$\mathbb{X} = \left[\begin{array}{l}\text{Our favorite Banach space} \\ \text{of fns on } \mathbb{R}^n\end{array}\right]$

Maybe

$\mathbb{X} = C^m(\mathbb{R}^n)$,

$\mathbb{X} = W^{m,p}(\mathbb{R}^n)$,

$\mathbb{X} = C^{m,\alpha}(\mathbb{R}^n)$

Given:

$E \subset \mathbb{R}^n$ finite

$f : E \to \mathbb{R}$.

Want to find $F \in \mathbb{X}$ s.t.

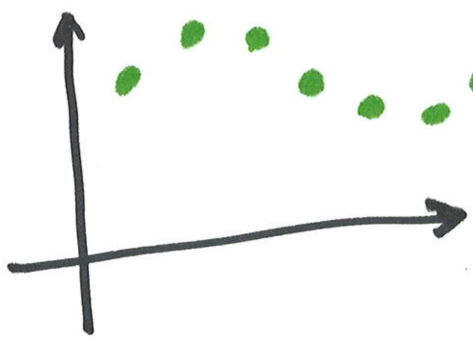
$F = f$ on E,

with

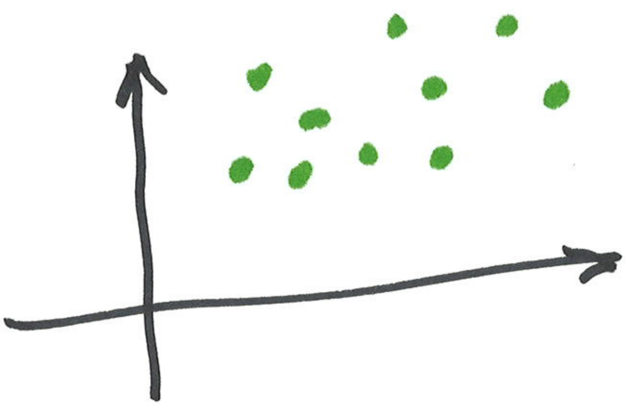
$\|F\|_{\mathbb{X}}$ as small as possible,

(or nearly so).

We'd also like to find the least possible $\|F\|_{\mathbb{X}}$.



← GOOD



← BAD

Oops!

Simple examples show that

$$\inf\{\|F\|_{\mathbb{X}} : F \in \mathbb{X},\ F = f \text{ on } E\}$$

may be an inf, NOT a minimum.

For $f: E \to \mathbb{R}$, define

$$\boxed{\|f\|_{\mathbb{X}(E)} = \inf\{\|F\|_{\mathbb{X}} : F \in \mathbb{X}, F = f \text{ on } E\}}$$

An <u>A-optimal interpolant</u>

for $f : E \to \mathbb{R}$

is a function $F \in \mathbb{X}$ s.t.

$F = f$ on E

and

$$\|F\|_{\mathbb{X}} \leq A \cdot \|f\|_{\mathbb{X}(E)} .$$

PROBLEM:

Given $f : E \to \mathbb{R}$

$(E \subset \mathbb{R}^n, \quad \#E = N < \infty)$,

Compute an A-optimal interpolant F,

with A depending only on the space $\mathbb{X}$.

COMPUTING A FUNCTION F

Say $\mathbb{X} = C^m(\mathbb{R}^n)$.

- WE ENTER THE DATA E, f
- The computer performs **ONE-TIME WORK** after which it can
- **ANSWER OUR QUERIES**

A QUERY CONSISTS of a point $\hat{x} \in \mathbb{R}^n$.

The RESPONSE to a query $\hat{x}$ consists of the values of F & its derivatives up to order m at $\hat{x}$.

We want **EFFICIENT** algorithms, making minimal use of computer resources.

The relevant computer resources are

- The number of operations needed to perform the one-time work
- The number of operations needed to answer a query
- The size of the memory.

WE WILL OPTIMIZE
ALL RESOURCES
SIMULTANEOUSLY—

NO NEED for
TRADEOFFS

INTERPOLATION ALGORITHM

FOR

$C^m(\mathbb{R}^n)$

{ JOINT WORK WITH

BO'AZ KLARTAG }

—

SET $\mathbb{X} = C^m(\mathbb{R}^n)$.

—

Given

$$f : E \to \mathbb{R}$$

$$(E \subset \mathbb{R}^n, \ \#E = N)$$

WE Compute an

A-optimal interpolant F

using

- $O(N \log N)$ ONE-TIME WORK
- $O(\log N)$ QUERY WORK
- $O(N)$ MEMORY.

A DEPENDS ONLY ON m & n.

The algorithm is

THEORETICALLY OPTIMAL

but

OF NO PRACTICAL USE

(because A is too big).

QUESTION: Can the proof or the algorithm be tweaked to yield something practical?

No time to explain the algorithm, but

It's based on the proof of an interesting, simple conjecture with an interesting history.

Will discuss the conjecture.

The history of the CONJECTURE starts in 1936 with WHITNEY'S EXTENSION PROBLEM.

WHITNEY'S PROBLEM:

Fix $m, n \geq 1$.

Given $f : E \to \mathbb{R}$

with $E \subset \mathbb{R}^n$ COMPACT,

how can we tell whether

f extends to a C^m function

F on $\mathbb{R}^n$?

That problem wasn't solved

until ~ 2003,

but already in 1936,

Whitney solved an

easier variant.

The Variant : Fix $m, n \geq 1$.

Let $E \subset \mathbb{R}^n$ be compact.

Suppose that for each $x \in E$ we are given P^x, an m^{th} degree polynomial on $\mathbb{R}^n$.

$\boxed{\text{How can we tell whether there exists } F \in C^m(\mathbb{R}^n) \text{ s.t. } \forall x \in E,\ P^x \text{ is the } m^{\text{th}} \text{ degree Taylor poly. of } F\text{?}}$

The Whitney Extension Thm:

The obvious necessary conditions

{ simple consequences of Taylor's thm. }

are SUFFICIENT.

The proof of the

Whitney extension thm

had far-reaching

consequences in analysis.

But for our story we

focus on two Soviet

mathematicians in the

1970's & 1980's.

THE HUMAN ELEMENT

(YURI BRUDNYI
&
PAVEL SHVARTSMAN)

SIMPLEST VERSION OF "FINITENESS PRINCIPLE"

Let $f: E \to \mathbb{R}$,

with $E \subset \mathbb{R}^2$ FINITE.

Suppose that for every 6-point subset $S \subset E$ there exists $F^S \in C^2(\mathbb{R}^2)$ s.t.

$F^S = f$ on S, and

$\|F^S\|_{C^2(\mathbb{R}^2)} \leq 1.$

Then there exists $F \in C^2(\mathbb{R}^2)$

s.t.

$F = f$ on E

and

$\|F\|_{C^2(\mathbb{R}^2)} \leq C$

UNIVERSAL CONSTANT.

The same holds for $C^2(\mathbb{R}^n)$,

with 6 REPLACED by

$3 \cdot 2^{n-1}$,

& the result IS FALSE

if $3 \cdot 2^{n-1}$ is replaced by

$(3 \cdot 2^{n-1}) - 1$.

The proof by

Brudnyi & Shvartsman

doesn't extend beyond C^2,

but Br & Shv made the

natural conjecture:

Given $m, n \geq 1$, $\exists k^{\#} = k^{\#}(m,n)$,

with the following property:

Let $f: E \to \mathbb{R}$

($E \subset \mathbb{R}^n$ finite).

Suppose that for each $S \subset E$

with ~~###~~ $\# S = k^{\#}$, there

exists $F^S \in C^m(\mathbb{R}^n)$ s.t.

$$F^S = f \text{ on } S$$

and

$$\|F^S\|_{C^m(\mathbb{R}^n)} \leq 1$$

Then there exists $F \in C^m(\mathbb{R}^n)$

s.t.

$F = f$ on S

and

$\|F\|_{C^m(\mathbb{R}^n)} \leq A$

DEPENDS ONLY ON m, n.

I worked on that conjecture,

& proved it ~2002.

The ideas in that proof underlie the interpolation algorithm mentioned before.

THE MORAL OF THE STORY

INACCURATE INFORMATION
CAN
PROMOTE SCIENTIFIC
DISCOVERY.

We are constantly surrounded by inaccurate information.

May discovery thrive!

THANKS

FOR

LISTENING!