Can we analytically solve neural networks?

no.

I don't care tho

Thomas Narramore

Department of Mathematics and Computer Science
Western Colorado University

Spring 2024

Consider a shallow neural network with input $\vec{x}$, output $\vec{y}$, and hidden layer $\vec{h}$. Using activation function a.

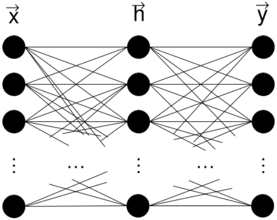

Figure: first set of weights is W_h, second set is W_y with bias $\vec{b_h}$, $\vec{b_y}$

In matrix notation:

$$f(\vec{x}) = W_y \left(a(W_h\vec{x} + \vec{b_n}) + \vec{b_y} \right)$$

Using

$$f(\vec{x}) = W_y \left(a(W_h \vec{x} + \vec{b_n}) + \vec{b_y} \right)$$

given a set functions inputs

$$data = \{(\vec{x_1}, \vec{y_1}), \ldots (\vec{x_n}, \vec{y_n})\}$$

we want $W_h,\ W_y,\ \vec{b_h},\ \vec{b_y}$ such that

$$f(\vec{x}) = \vec{y}$$

For all $(\vec{x},\ \vec{y}) \in data$

One approach: minimize loss, $L = (\vec{y} - f(\vec{x}))$ Using calculus:

1. Take the partial derivatives $\frac{\partial L}{\partial w_{l,i,j}}$ and $\frac{\partial L}{\partial b_{l,i}}$ for every weight and bias
2. Set them to 0.
3. Solve nonlinear system of $O\left(|\vec{h}| \times (|\vec{x}| + |\vec{y}|) \times |data|\right)$ equations.

Our matrix equation for the whole network is

$$\vec{x} = W_y \left(a(W_h \vec{x} + \vec{b_n}) + \vec{b_y} \right)$$

Writing out the equation for a single element of $\vec{y}$:

$$\vec{y_k} = a \left(\sum_{j=1}^{|\vec{h}|} w_{jk} \times a \left(\sum_{i=1}^{|\vec{x}|} (w_{ij} \times x_i) + b_j \right) + b_k \right)$$

So the loss L of each input/output equation y_k looks like:

$$L = (f(x) - y_k)^2$$

$$= \left(a \left(\sum_{j=1}^{|\vec{h}|} w_{jk} \times a \left(\sum_{i=1}^{|\vec{2}|} (w_{ij} \times x_i) + b_j \right) + b_k \right) - y_k \right)^2$$

Taking the derivative with respect to a weight in the output layer:

$$\frac{\partial L}{\partial w_{j,k}} = 2a' \left(f(\vec{x}) \right) a \left(\sum_{i=1}^{|\vec{x}|} (w_{ij} x_i) + b_j \right) - 2y_k$$

Loss:

$$L = (f(x) - y_k)^2$$

$$= \left(a \left(\sum_{j=1}^{|\vec{h}|} w_{jk} \times a \left(\sum_{i=1}^{||\vec{x}||} (w_{ij} \times x_i) + b_j \right) + b_k \right) - y_k \right)^2$$

Taking the derivative with respect to a weight in the hidden layer:

$$\frac{\partial L}{\partial w_{i,j}} =$$

$$2a' \left(f \left(\sum_{j=1}^{|\vec{h}|} w_{jk} \times a \left(\sum_{i=1}^{||\vec{x}||} (w_{ij} x_i) + b_j \right) + b_k \right) \right) a' \left(\sum_{i=1}^{||\vec{x}||} (x_i) \right) - 2y_k$$

To solve, set derivative to zero.
We have $O\left(|data| \times |\vec{h}| \times |\vec{y}|\right)$ equations of the form

$$2a'\left(f(\vec{x})\right) a\left(\sum_{i=1}^{|\vec{x}|}(w_{ij}x_i) + b_j\right) - 2y_k = 0$$

and $O\left(|data| \times |\vec{h}| \times |\vec{x}|\right)$ of the form

$$2a'\left(f\left(\sum_{j=1}^{|\vec{h}|} w_{jk} \times a\left(\sum_{i=1}^{||\vec{x}||}(w_{ij}x_i) + b_j\right) + b_k\right)\right) a'\left(\sum_{i=1}^{||\vec{x}||}(x_i)\right) - 2y_k = 0$$

- The difficulty of this depends on the activation function.
- Generally solving a non-linear systems is intractable (NP-Hard? Undecidable?)

- We know we need non-linearity to approximate non-linear functions.
- What if we use a quadratic activation function?
- Get a linear derivative and simplify the chain rule expressions.

$$y = \left(\theta_0 + \theta_1(\theta_{10} + \theta_{11}x)^2 + \theta_2(\theta_{20} + \theta_{21}x)^2\right)^2$$

$$=$$

$$\left((\theta_0 + \theta_1\theta_{10}^2 + \theta_2\theta_2^2) + (\theta_{10}\theta_{11}\theta_1 + \theta_{20}\theta_{21}\theta_2)x + (\theta_1\theta_{11}^2 + \theta_2\theta_{21})x^2\right)^2$$

You just get 4th degree polynomial output.