Analysis for Simple Neural Nets Used to Test my Computational Methods

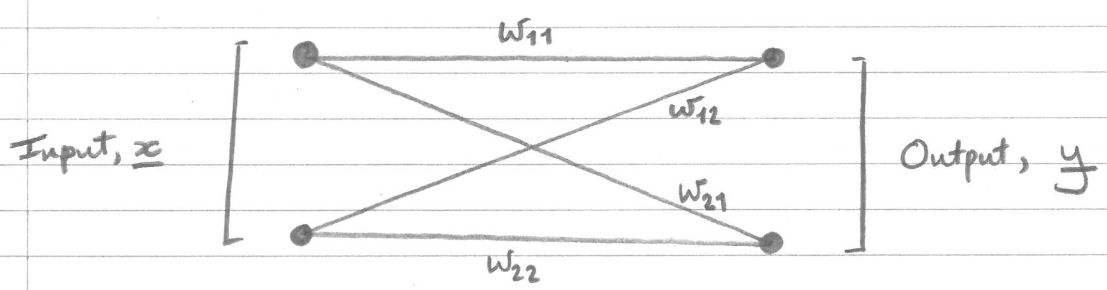
This document provides detail on the methods used to validate the accuracy of the computational tools developed to calculate the Hessian of a neural network.

It focuses on the comparison of the results obtained with the analytical expressions derived with "pen and paper".

Note:

Whilst these networks are very simple, the algebra for first- and second-order derivatives is somewhat tedious. The analytical values were instead obtained using the tools available on the website, desmos. This was found to be far quicker and more reliable.

Network 1



This is a simple neural net with no hidden layers. A tanh activation function acts on the final layer.

The network can be written as:

$$f: \mathbb{R}^2 \longrightarrow \mathbb{R}^2 \quad , \quad f(\underline{x}) = \tanh(\underline{\underline{w}}\,\underline{x} + \underline{b}) = \underline{y}$$

where $\underline{x} \in \mathbb{R}^2$, $\underline{y} \in \mathbb{R}^2$, $\underline{\underline{w}} \in \mathbb{R}^{2\times 2}$, $\underline{b} \in \mathbb{R}^2$.

PTO →

The loss is defined as:

$$L = \|\underline{y} - \underline{t}\|^2$$

where $\underline{t} \in \mathbb{R}^2$ is some "target" vector, i.e. a ground-truth.

This can be written in index notation:

$$L = (y_1 - t_1)^2 + (y_2 - t_2)^2$$

$$= \left[\tanh(w_{11}x_1 + w_{12}x_2 + b_1) - t_1\right]^2$$

$$+ \left[\tanh(w_{21}x_1 + w_{22}x_2 + b_2) - t_2\right]^2$$

Evaluating some of the first-order gradients:

$$\frac{\partial L}{\partial b_k} = 2\,\text{sech}^2(w_{kj}x_j + b_k)\left[\tanh(w_{kj}x_j + b_k) - t_k\right]$$

$$\frac{\partial L}{\partial w_{k1}} = 2x_1\left[\tanh(w_{k1}x_1 + b_k) - t_k\right]\text{sech}^2(w_{k1}x_1 + b_k)$$

(using the Einstein summation convention).

The following shows that the 2^{nd}-order gradients are somewhat more complex:

$$\frac{\partial^2 L}{\partial b_k \partial b_k} = 2\,\text{sech}^2(w_{kj}x_j + b_k)\left[2\tanh(w_{kj}x_j + b_k)(t_k - \tanh(w_{kj}x_j + b_k)) + \text{sech}^2(w_{kj}x_j + b_k)\right]$$

Fortunately, these 2^{nd}-order derivatives (as well as other information, such as the network output, loss and first-order derivatives) can all be computed easily using the tools available at desmos (as mentioned above).

These analytical values were verified as being equal to those predicted computationally.

Network 2

This network has the same structure as network 1, except it uses a sigmoid activation function.

The analytical and computational results were found to be the same.

Network 3

Works in exactly the same way as network 2, except with a simple modulus loss:

$$L = \| y - \underline{t} \|$$

Again, the analytical and computational results were found to be the same.