ELU $\gg$ ReLU

Aidan Rocke

May 30, 2017

Abstract

While the choice of activation function in the hidden layers of a feedforward neural network is essential for controlling the rate and stability of learning, there appears to be widespread confusion in the deep learning community as to whether ELU networks are superior to ReLU networks. In the following analysis, this issue is clarified by using a theoretically-motivated experimental analysis instead of using fashionable benchmarks as is the present tradition in the deep learning community.

1 Introduction

1.1 Why are activation functions important?

The choice of activation function in the hidden layers is important as this has a profound impact on every aspect of neural network training. In particular, the properties of an activation affect the following:

1. Rate of training:

   Ideally, learning would be fast and computationally efficient.

2. Stability:

   We want all components to learn at similar rates and we want the derivative of the activation function to be stable to input perturbations.

3. Internal covariate shift:

   Small changes to the inputs of a neural networks hidden layers get amplified as we go deeper into the network. This creates an internal data set shift within the neural network, and impedes the process of learning the true joint distribution.

Today there's a large choice of activation functions for the hidden layers of a feedforward neural network ranging from the sigmoid, hyperbolic tangent to the ReLU. However, most researchers justify the use of a particular activation function(ex. ReLU) with performance on popular benchmarks rather than principled experimental analysis. Given that in most instances prior knowledge of the joint distribution of the training data is unknown, the present tradition is both scientifically unsound and unacceptable.

2 Background on training feedforward neural networks:

2.1 feedforward neural networks:

In general, a feedforward neural network f is a non-linear function consisting of affine transformations(W_i) interleaved with differentiable activation functions(σ_i):

$$\begin{aligned} f &: \widetilde{X} \to Y \\ f(x) &= \sigma_n W_n ... \sigma_2 W_2 \sigma_1 W_1 x \end{aligned} \tag{1}$$

It's useful to note that f represents the composition of differentiable functions so it's also fully differentiable. This leads me to my next point.

2.2 function approximation:

The task of training a feedforward neural network is essentially to obtain successively better approximations f_i to a desired but unknown mapping g:

3 References

One of the nice things about using LaTeX is that it makes internal references easy. For example, if I want to remind you where I discussed math mode, I can mention that it was in Section **??**. If you're looking at the pdf file, you see the correct reference, but in the TeX file I typed a label that I had attached to that section. (You may need to typeset your document more than once to make the references show up correctly.) Labels work for definitions, theorems, questions, sections, diagrams, and equations, among others.