

A Few Tricks for Improving BP(back propagation) Network Performance

Author: Xiang

Date: 2017.1.16

In order to show this document correctly, please install the package(markdown-preview-enhanced) in Atom first.

The following story bases on a paper titled "Understanding the difficulty of training deep feedforward neural networks", written by Xavier Glorot and Yoshua Bengio in 2010.

The paper said there are several factors would benefit ANN's performance:

- **factor 1:** initialization of weights W_{ij}

$$W_{ij} \sim U \left[-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}} \right],$$

where $U[-a, a]$ is the uniform distribution in the interval $(-a, a)$ and n is the size of the previous layer(**the number of columns of W**).

- **normalized initialization**

$$W_{ij} \sim U \left[-\frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}}, \frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}} \right],$$

where n_i is the size of layer i .

- **factor 2:** selection of activation function
(Bergstra et al., 2009 Quadratic polynomials learn better image features (Technical Report 1337)) proposed a activation function called the **softsign**,

$$f(x) = \frac{x}{1 + |x|}.$$

The softsign is similar to the hyperbolic tangent (its range is -1 to 1) but its tails are quadratic polynomials rather than exponentials, i.e., it approaches its asymptotes much slower.

That's all so far for me.