Summary

Clément Dell'Aiera, Clément Prévosteau , David Wahiche

This document aims at describing algorithms involving neural networks, and particularly new and cutting-edge methods from information geometry and deep learning.

1 Goal

Our aim within this article, is to understand new methods in the supervised learning field. For some years now, several teams, Geoffrey Hinton's in the lead, have been able to perform really low error rate on such problems. They applied what is now called *deep learing*, particularly on handwritten digits. More recently, Yann Ollivier has put online during 2013 a preprint in which he uses methods of information geometry on neural networks.

We had a threefold goal : to study and code neural networks, to understand and apply geometrical ideas to learning problems, and to be able to implement deep learning methods.

2 Work in a nutshell

2.1 Neural Network

Neural networks can be described as regression models. They are made of several layers stacked in pile, linked one another by synaptical connexions which control the response of the input via a weight and an activation function. These are the parameters of the neural network, as a statistical model. These are the value that the engineer want to tune in order to make the network learn, training it on a data set via a supervised algortihm.

Formally, a neural network is just an integer N, the total number of layers, and $N-1$ matrices which contains the weights, and as so many activation functions f_j. If the input is denoted by $x \in \mathcal{D}$, the networks acts by induction

$$\begin{cases} x_0 = x \\ x_{j+1} = f_j(W_j x_j) \end{cases}$$

In the formula above, the activation function is applied term by term to the vector $W_j x_j$. The activation function are to choose in a sample know from the

specialists : *tanh*, sigmoïd,... The weight matrices are the parameters of the model.

These networks are used in supervised learning, that is we give it an input x and the target value t. The network compare then the target value with the response it computes, and changes its weights accodingly to a algorithm, usually a gradient descent on a loss function.

2.2 Methods of information geometry

This is where information geometry comes into play. The main idea is to conceive the space of the model's parameters as a riemannian manifold, i.e. a topological space together with a chart system and a metric which enables us to compute length between variations of the paramters. (the intuitive picture of the tangent space is that of infinitesimal variations of parameters) We can then, following Ollivier [10], choose «invariant » or «intrisinc» metrics, in the sense that they depend only on what the network does, not on the choice of the parameters. E.g. Ollivier choose the metric cooresponding to the Fisher matrix : it is symetric positive and defines a metric. Ollivier prooves that it is intrinsic. We tested riemannian and euclidean gradient descents, and the reimannian ones show greater speed of convergence.

Information geometry uses more sophisticated tools. The book of Amari on this topic [12] present an introduction to affine connexions. However, even if we began to study it, we decided to limit ourselves to the riemannian gradient descent, for several reasons. For one thing, the technical level was high in geometry, more than what we could do. Then, it appears that these methods, promissing as they are, have not proved themselves source of a statistical breakthrough, for now.

2.3 Deep learning

Deep learning is, above the multilayer neural network technique, preparation of the weights of the networks. Using statistical models born in statistical physics, named Restricted Boltzmann Machine, we can train each layer before the gradient descent.

3 Results

We coded a Python class to handle multilayer neural networks. We could then compare these with other classical methods as Support Vector Machines,

for example.

We also coded information geometry techniques, and deep learning methods.

We encountered difficulties, msot arguably in the litterature. This field is actually brand new (some articles are from 2013), and it was difficult to find clear expositions.

We tested the algorithms on one of the benchmark in supervised learning : the MNIST database, available on Lecun's site, which contains 70000 pictures of handwritten digits, all 28×28 pixels. We proceeded by increasing order of complexity : at first, a simple logistic classifier on MNIST, then a two-layered network, the last layer being a logistic classifier, and finally we coupled weights preparation with RBM and the latter two-layered network. These enabled us to achieve an error rate on the MNIST test set of $7,489\%$, $1,65\%$ et $1,34\%$. The reader can compare our results with scores of research teams on the page `http://yann.lecun.com/exdb/mnist/`

We tested also differences between riemannian descent and euclidean descent on examples that we simulated. Mainly set non linearly separable (XOR in the article). We would have linked these two notions together (deep learning and neural networks) but we were unsuccessful.

Références

[1] G.Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4) :303–314, 1989.

[2] Yee-Whye Teh Geoffrey E.Hinton, Simon Osindero. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7) :1527–1554, 2006.

[3] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 2(4) :251–257, 2006.

[4] Shun ichi Amari. Information geometry of the em and em algorithm for neural networks. *Neural Networks*, 8(9) :1379–1408, 1994.

[5] Shun ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2) :251–276, 1998.

[6] Patrick Kenny. Notes on boltzmann machines. 2012.

[7] Jacques Lafontaine. *Introduction aux variétés différentielles*. EDP Sciences, 1996.

[8] Herbert Lee. *Bayesian Nonparametrics via Neural Networks*. Society for industrial and applied mathematics, 2004.

[9] Seymour Papert Marvin Minsky. *Perceptrons*. MIT Press, 1969.

[10] Yann Ollivier. Riemanian metrics for neural networks. *Preprint*, 2013.

[11] Frank Rosenblatt. *Principles of neurodynamics : perceptrons and the theory of brain mechanisms.* Washington, Spartan Books, 1962.

[12] Hiroshi Nagaoka Shun-ichi Amari. *Methods of information geometry.* Oxford University Press, 1993.