**GT**

**Programme :**

Novembre - Janvier : Bibliographie, livre de Amari, Sepulchre. Premiers codes.

Janvier – Mars : Implémentation de la méthode d’Ollivier.

Mars- : Implémentation distribuée.

**But :**

Appliquer les méthodes de géométrie de l’information aux réseaux de neurones « profonds » pour expliquer pourquoi ça marche. (cf travail de Yann Ollivier)

Implémenter en optimisation distribué : utilisation de MapReduce.

**1ère étape :** implémenter une descente de gradient sur la famille exponentielle.

Essayer d’implémenter un perceptron multicouche pour une reconnaissance de chiffres. ( codés en ondelettes)

**06/02/2014**

**Cours sur Coursera :** <https://www.coursera.org/>

Cours de Hinton sur le machine Learning, aborde le **dropout** technique pour accélérer la vitesse d’apprentissage (le propout évite l’over fît ?)

<https://www.coursera.org/course/neuralnets>

Cours de Andrew Ng sur le machine learning avec des tutos

<https://www.coursera.org/course/ml>

Hinton a gagné une compétition (kaggle competition) avec deus techniques : remplacer la fonction logistique par une fonction softmax, et le propout. Ou dropout…

<http://blog.kaggle.com/2012/11/01/deep-learning-how-i-did-it-merck-1st-place-interview/>

packages python à installer : theano <http://deeplearning.net/software/theano/> et pylearn qui fonctionne avec theano : <http://deeplearning.net/software/pylearn/> et permet de gérer les calculs via la carte graphique (rapide)

**Idée pour le rapport :**

Tracer de vraies ellipses bien aplaties et montrer qu’alors la méthode du gradient ne converge pas bien. Tracer le gradient Riemannien.

Ecrire un truc sur les machines de Boltzmann.