

# CS 51. Project Proposal.

Angela Fan, Andre Nguyen, Vincent Nguyen, George Zeng.

Due April 10th, 2015.

## 1 Basics

We plan to work on a machine learning project with the ultimate goal of implementing a neural network by hand.

Our project members are:

- Angela Fan <huihuifan@college.harvard.edu>
- Andre Nguyen <andrenguyen@college.harvard.edu>
- Vincent Nguyen <vnguyen01@college.harvard.edu>
- George Zeng <gzeng@college.harvard.edu>

## 2 Brief Overview

As a group, we are very interested in learning more about machine learning methods. For this project, we would like to explore neural networks, as they have seen increasing use, particularly in deep learning.

We will implement a basic version of a Restricted Boltzmann Machine (RBM), a type of stochastic neural network that has been used in a variety of applications. Specifically, we are fascinated by the potential to apply neural networks to everyday pattern recognition in image recognition and classification. However, we will treat the goal of image recognition as an extension to our core project, a little something extra should we have time to explore it.

A restricted Boltzmann machine is a simplified version of a Boltzmann machine in which nodes at hidden layers are restricted to non-cyclical graphs. The RBM will tune the parameters of an energy-based probabilistic function (linear in its parameters) so that desirable states have lower energy configurations. The energy function will be defined as

$$E(v, h) = -b'v - c'h - h'Wv$$

where  $W$  represents the weights that connect the hidden and visible nodes, and  $b$  and  $c$  are offsets of the visible and hidden node layers.

We will then sample from the probability distribution generated by the energy function using a Gibbs Sampler. In order to speed up the sampling, we will use the Contrastive

Divergence algorithm with  $k = 1$ . The CD-k algorithm speeds up the sampling because it does not wait for the markov chain to converge, and allows us to initialize the markov chain with a known training example, avoiding the potential burn-in time.

For our project we would like to

### **3 Feature List**

We believe that implementing an RBM will be a decent amount of work. We will implement the RBM as well as a few functions to help assess its performance. We will train and test our RBM on a toy dataset of 10 users and the movies they like.

We will explore the following extensions, time permitting:

1. Persistent CD algorithm to approximate sampling from the energy function probability distribution, as an alternative to CD-k
2. Train and test our RBM on more complicated datasets, such as MNIST handwriting image recognition and Faces in the Wild, a set of labeled images of faces

### **4 Technical Specification**

We will implement this project in an entirely object oriented fashion.

### **5 Next Steps**