

## **Milestone 0: Charter**

Kyra Patton (kp483), Agrippa Kellum (ask252)

Luca Leaser (ll698), Stephanie Sinwell (sas575)

### **Meeting time**

We plan to meet as a group twice a week - once on Thursday evening and once on Saturday.

### **Proposal**

We will create a neural networks library for OCaml. Users will be able to generate classifiers by inputting labeled datasets and a loss function.

Key features:

1. Implementation of both fully connected neural networks
2. Interfaces with C, allowing for fast matrix operations
3. Simple interfaces and clear documentation to support client usage
4. Potentially implement CUDA library to allow people to run their networks on a GPU

We will need to find or make an implementation of matrices. Then, we can implement tensors as functions which input and output matrices. A neural net is simply a set of nested matrix operations (tensors) with a nonlinear function applied to their output. We will allow users to specify the nesting using a directed graph structure with tensors as nodes.

In order to make the neural net trainable, nodes in the network can have parameters. We will implement an algorithm for mutating these parameters based on the gradient of the loss function across the parameter space given a learning speed. Iterative mutation will produce parameters that minimize the neural network's loss for the given inputs.

We will interface with C++ and OpenBLAS to allow for fast matrix computations, and potentially use the CUDA library in C++ to allow for training on a GPU.

There already exists a library that implements ANNs in OCaml, called Owl. The documentation is not clear about how full the implementation is, but we do not intend to look at this package while developing our project.