James Hurt

Project 1

Your report should contain a sections on

1. The technical description of all techniques utilized,

2. The design of the algorithms (pseudo-code, flowcharts, or some

other structured descriptive means),

3. The results of the algorithms,

4. An analysis of the results, i.e., did you obtain what you expected?

Were there any surprises? What conclusions can you draw from the

experiments? etc.

5. Well documented, structured, modular program listings.

## Technical Description

I chose to implement my project in Python3 due to its large number of libraries and the relaxed syntax of the language. The libraries used:

* NumPy is a library specializing in complicated mathematical structures. NumPy was the vector and ndarray handler, which is mathematically equivalent to a matric. I also used it for finding v = wTx as well as storing the weights and biases throughout the program
* Pandas is a library that has a simple and elegant way to parse CSV files into a structure called a Dataframe which resembles an SQL table. I used this library to bring in initial values for w and b as well as the training data with labels from CSV files. Pandas also has a function that converts a dataframe to a NumPy ndarray, making it a very easy library to implement
* PPrint is a Python library that allows for stylized printing such as how many items per line. I used this library to make the results of the network training (weights and bias) more legible.
* ArgParse has functions for passing arguments to the python file via command line. I wanted the user to be able to pass in the dimensions of the network and then input filenames for a general-purpose program. Then I went ahead and built in -a and -b parameters to allow for the script to automatically load the correct files for parts A and B of the assignment.
* MatPlotLib. This library is used for building graphs and plotting points and curves. I used this library to build the graphs asked for in part A, such as error per epoch and the data itself.

## Algorithm Design