Dataset Write-Up

For this project I chose to modify a dataset from the UCI Machine Learning Repository called the QSAR biodegradation data set. This dataset takes 41 inputs to find out if a chemical is readily biodegradable. In the way that I have modified the dataset, however, there are only 3 inputs (reasons and process will be discussed shortly). The 3 inputs,in order, are the number of atoms of type ssssC, frequency of C - N at topological distance 2, and leading eigenvalue from Burden matrix weighted by mass respectively. The output is the last column, which determines if the chemical is readily biodegradable by either a 0 or 1.

The way that I have modified this dataset is as follows:I found that many of the values between different columns varied largely so I applied sklearn's StandardScaler to scale all values in all columns of the inputs by $z = (x-u)/s$ where u is the mean of the training samples in that column, s is the standard deviation, and x is the training sample's value. Afterwards, I found that there was a significant difference between amounts of 0s and 1s in the output column so I applied an imbalancing technique known as SMOTE that oversamples datasets on binary columns using the K Nearest Neighbors (KNN) approach so that the amount of 0s and 1s are equal to each other (e.g. generating samples using KNN approach until the 0s and 1s equal each other). The drawback, however, is that new samples are added to the dataset that may not be entirely accurate. Due to this, I applied a Gradient Boosted Trees classifier and, from this method, only used the columns with an importance greater than 0.1, which ended up being the three inputs that I had previously talked about.

As for the initial weights of the neural network, I have generated the weights using numpy's np.random.uniform function that returns a random number between 0 and 1 from a uniform distribution.

Lastly, upon much experimentation, the optimal parameters using my dataset is 20 hidden nodes in the hidden layer, 200 epochs, and a 0.1 learning rate. As for the specific files pertaining to my dataset, descriptions are listed below:

biodeg.results.txt - results file
biodeg.train.txt - train file
biodeg.test.txt - test file
biodeg.weights.txt - initial weights
Biodeg.trained.weights.txt - weights after training w/ optimal params