AI Artificial Intelligence --HW4 Machine Learning

Stu: Song Jun Park Zewei Li

Part 1. Decision tree

The brief steps about how we think about the decision method on this problem:

1.This part we use decision tree to help use make classification on the dataset that we use. In the tree model, we know that leaves represent classes labels and branches represent the conjunctions of features that lead to those class labels. The rule of choosing attributes accords to that we choose the attribute that produces the “purest” nodes.

2.The information gain is used to decide which feature to split on at each step in building the tree. We will use the highest information gain on the first spilt and the process will continue all the children nodes are pure and the information gain equals to zero.

When we also use the information gain ratio which is a modification of the information gain to reduce its bias. It takes number and size of branches into account when choosing an attribute.

3.Pruning

In order to prevent to build the over-complex tree, we need to add pruning to stop growing the tree when there is no statistically significant association between that any attribute and the class at particular node.

In our code we use function to help us choose the best feature to make

classification to split nodes into subset.

Evaluation of decision tree method:

It is easier for us to understand the mechanism of decision tree. We think it is a better way to solve or perform on large dataset.

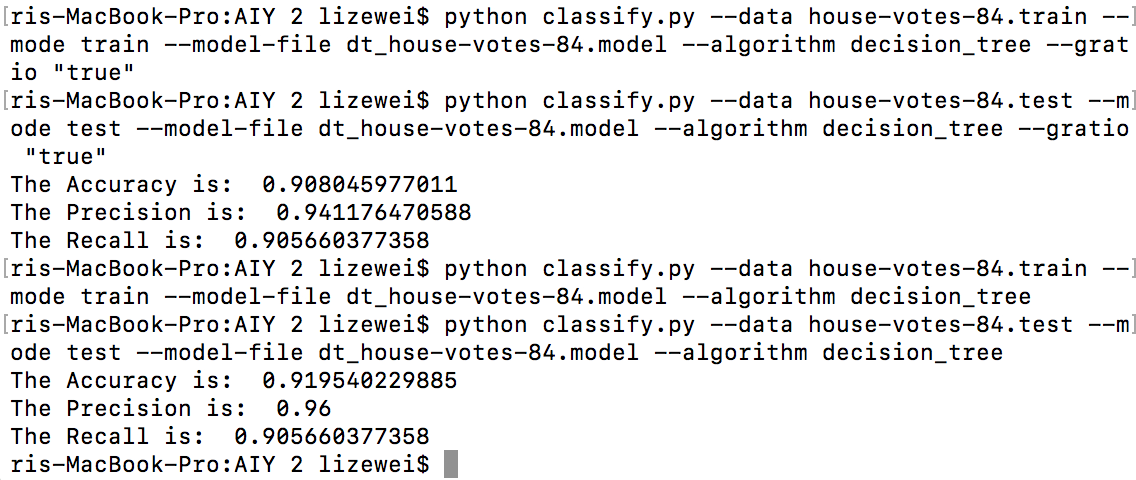
The decision tree is a good way to deal with missing data.

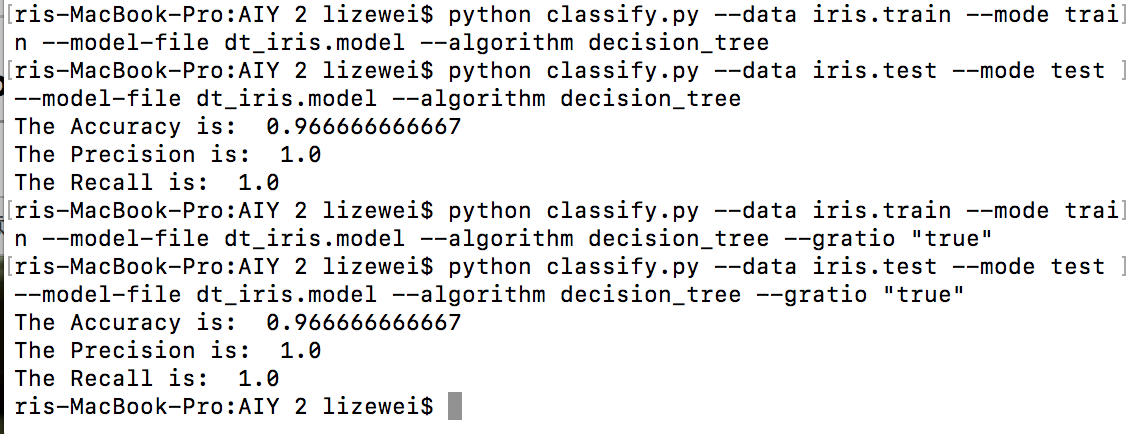
The limitation of decision tree method:

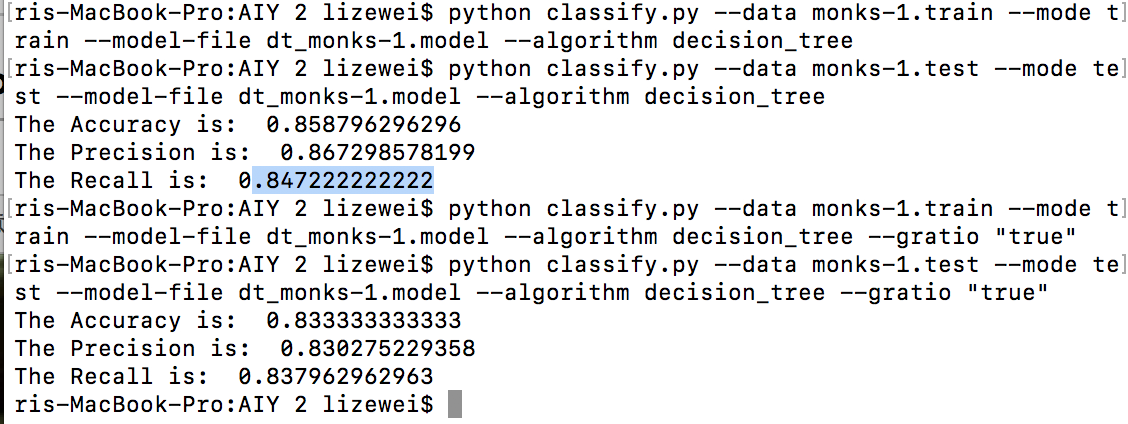
Because it is may be too much mechanical, the programmer maybe build the decision tree over-complex however it will not generate better results. Hence, sometimes, we need to use pruning to avoid this problem.

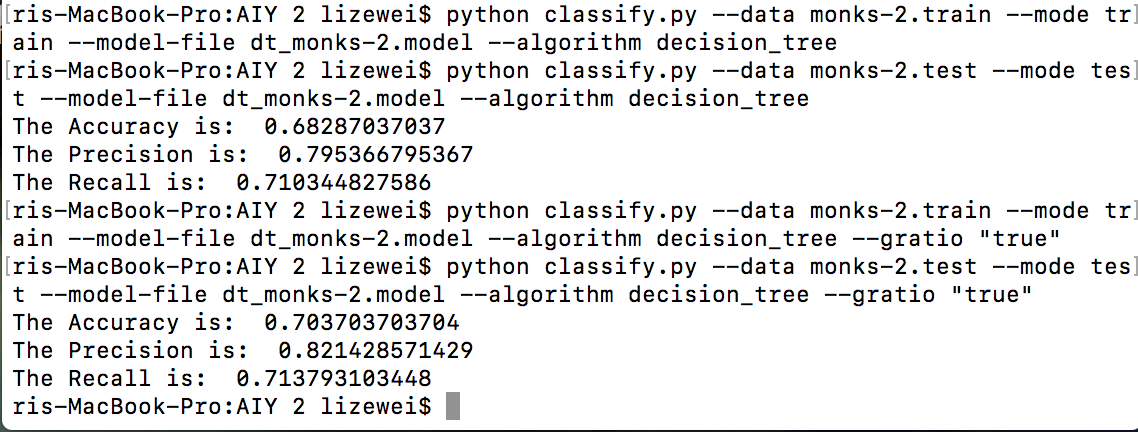
The screenshot of the result:

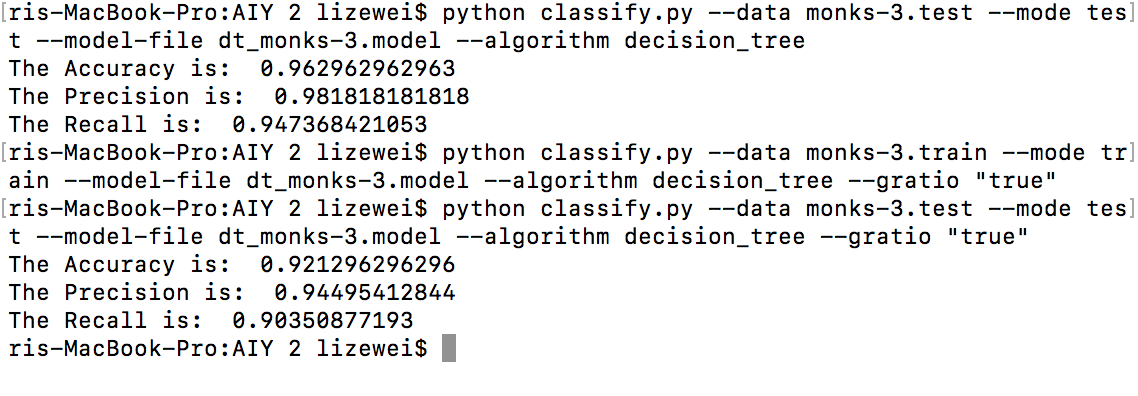
1.



2.

3.

4.

5.

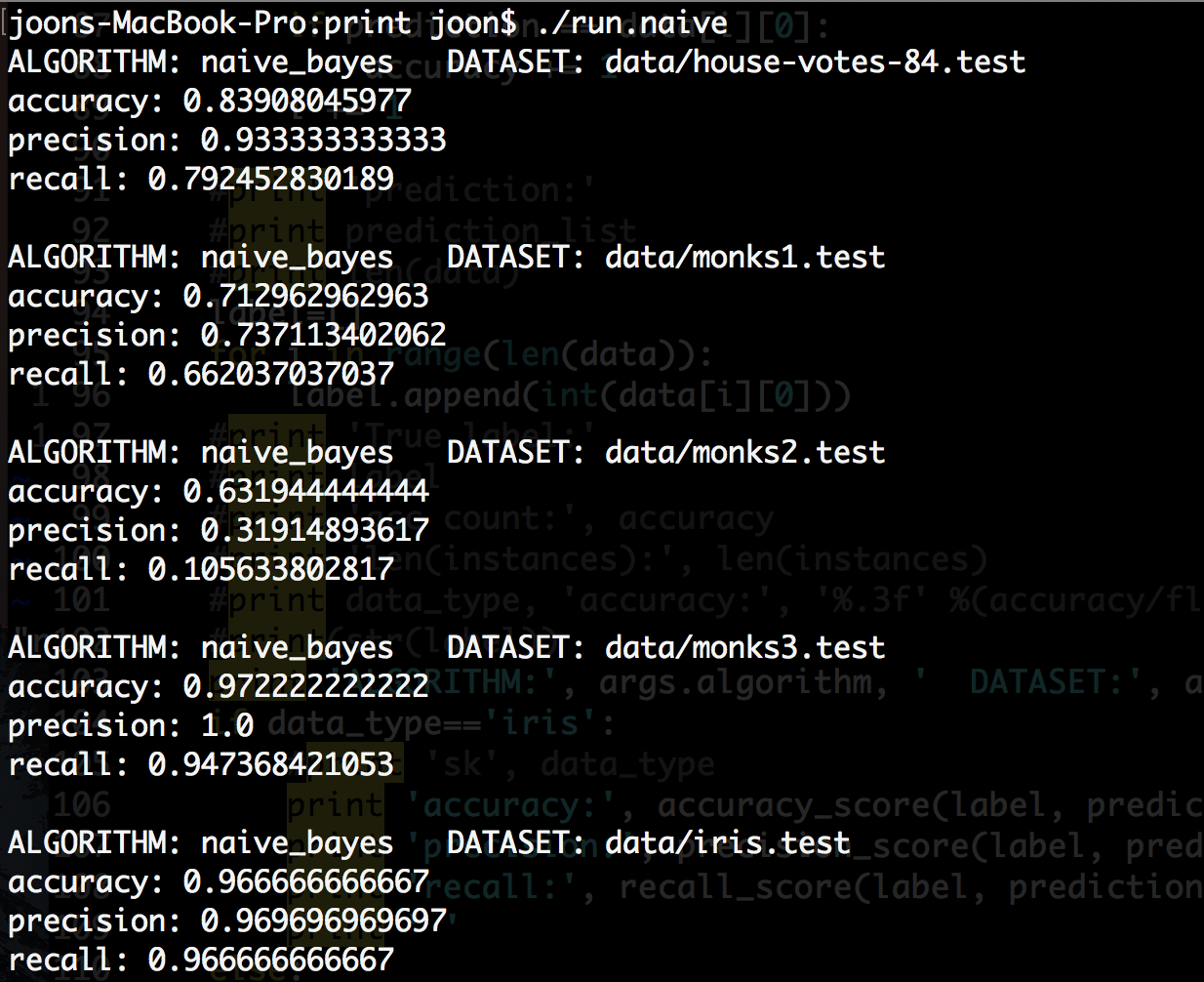
Part2 Naive Bayes:

The Naive Bayes method is to use transfer the dataset into a frequency table and we can use Bayes' theorem and formula to calculate the posterior probability.

Pros: From the quantity of coding, it is easier and faster compared to other methods.

Cons: It may cause “zero frequency” when the some variables which are in the test data but not in the train dataset. So it will make no prediction .

The screen shots of result:



Part3 Neural Network:

In this program, we just use one hidden layer to achieve the neural networks. I and my partner try to use different activation function to test our dataset in order to get better results. We also try to change the number of nodes which are at middle hidden layer to see the difference of the results that we get using the same dataset. We mostly use sigmoid function to compute our hidden layer.

The dataset travels from input layer, then to the hidden layer and finally gets to output layer.

We’ve decided to use 10 hidden layer nodes and by trial and error, we observed that the iris dataset worked most optimally with 0.5 learning rate and 1.0 for house-votes and monks. As for the number of iterations, the epoch value was set to 500 for the accuracy results.

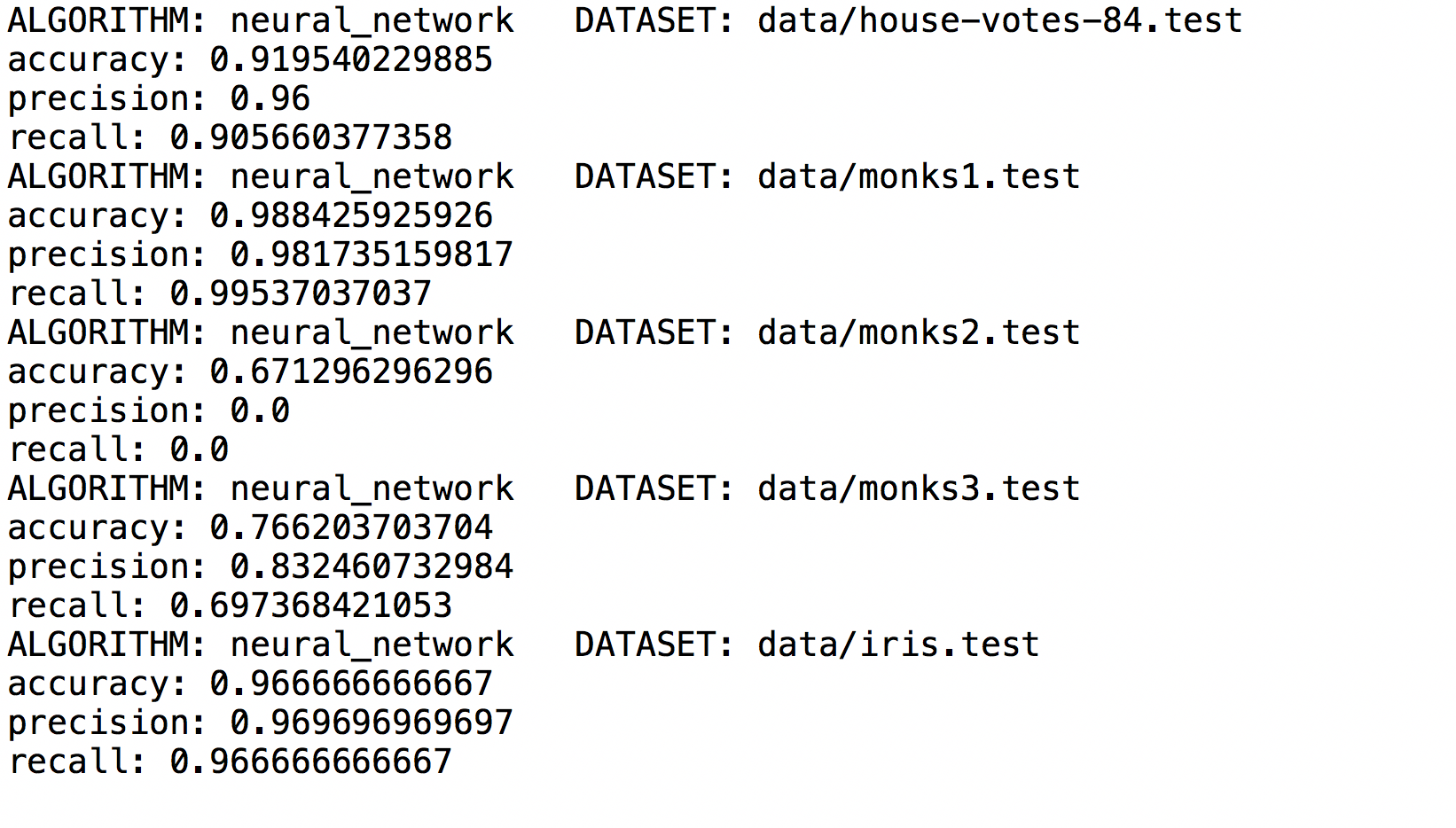
In my own opinion, I really like this method to make classification on dataset because it is really likes the process that how a real person is thinking problem. It lets computer to simulate the action of biological neuron working.

The limitation of using neural network:

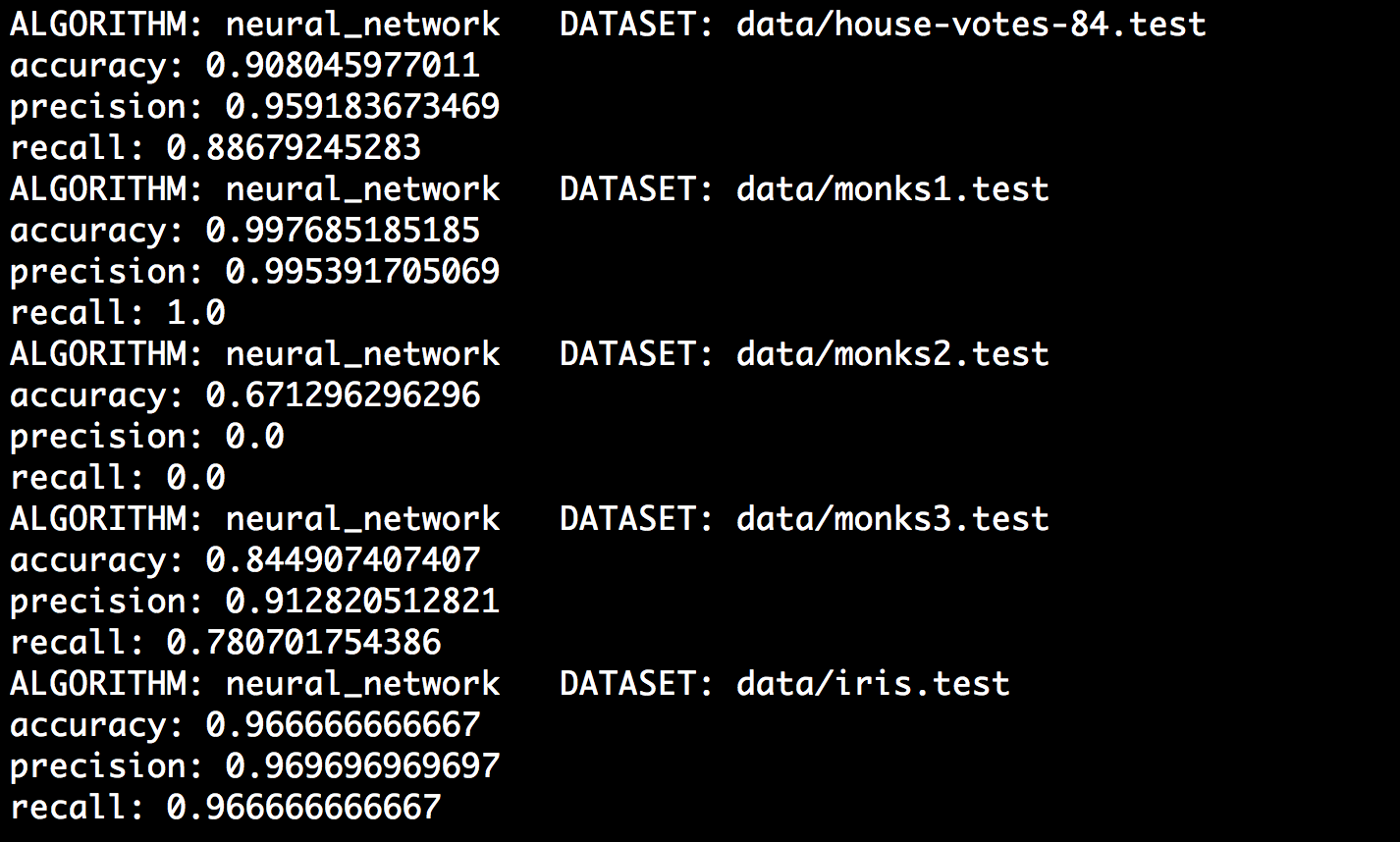
It is unpredictable because after training, some systems are good at solving problems while others are not.

The screen shots of the result:

1.Uniform Random Initialization:



2. Alternate Weight Initialization----Glorot and Bengio



Note:

We have talked about the details and comparsion of three methods separately in this pdf file. So you can search respective information from the whole file from page one to the end.

Answering question:

For decision tree:

I think using pruning on decision tree is better to not using the pruning the decision tree. The pruning will let classification to be more simple than original problem and try to avoid making problem over-complex.

After finishing the all tree methods as a team, the neural network looks promising as it is similar to the process of how a real human thinking or doing classification in our mind. As for the somewhat contrived dataset of monks, the decision tree gets the best results. The Naive Bayes performed the worst in all of these three method. We believe the size of the training data was not sufficient for Naïve Bayes classifier, which due to its simple implementation scales well. We analyze that because we just used one level of hidden layer, it limits the true capabilities of a neural network. Furthermore, it is evident from modern literature that deep neural networks outperform many machine learning models.

As the training dataset was extremely small, the models will be prone to overfitting. More training data is will prevent our methods from poor generalization to future data prediction. High accuracies might be an indicator that the models are overfitting to the small dataset.

Brief Explanation of classes:

1.DecisionTree

I build the class named "DecisionTreeBuild" to build up the method of decision tree, which is imported by classify.py.

In this class, I also define some function to build this class: first i split dataset, then build up or create the tree by function "createMyTree" and finally, we do classfication by function named "Makeclassify".

2.Naive Bayes

The class NaiveBayes implements the Gaussian probability and frequency based probability functions to calculate the conditional input probabilities assuming that the inputs are independent. 'calculateClassGaussianProb' is used for the iris dataset and the frequencies of feature values are counted for the house-votes and monks dataset.

3.Neural Network

The architecture of neural network is created by 'createNetwork' where size of the input layer, hidden layer, and output layer is specified. For forward pass, weighted sum of inputs and a sigmoid activation function was implemented. The backpropagation and updating weights are calculated after a forward pass according to a learning rate and epoch, which determines the number of iterations. For the house-votes and the monks dataset, one output node was defined whereas the iris classification used 3 output nodes.

The breakdown of partners:

1. Song Jun Park----Finish the part 2 of Naive Bayes

2. Zewei Li----Finish the part 1 of Decision Tree

3. Both of us work together to finish the part 3 of Neural Network

4. We work together to write the PDF file and ReadMe file.