Neural Net Report

Your Name: Aryan Bhatia

Your GT Username: 903754724

Question 5: Learning With Restarts

1. **testPenData**:
   - Max accuracy: 0.9053744997141223
   - Average accuracy: 0.898913664951400 7
   - Standard deviation: 0.0054858720714972 94
2. **testCarData**:
   - Max accuracy: 0.965
   - Average accuracy: 0.96
   - Standard deviation: 0.003162277660168382

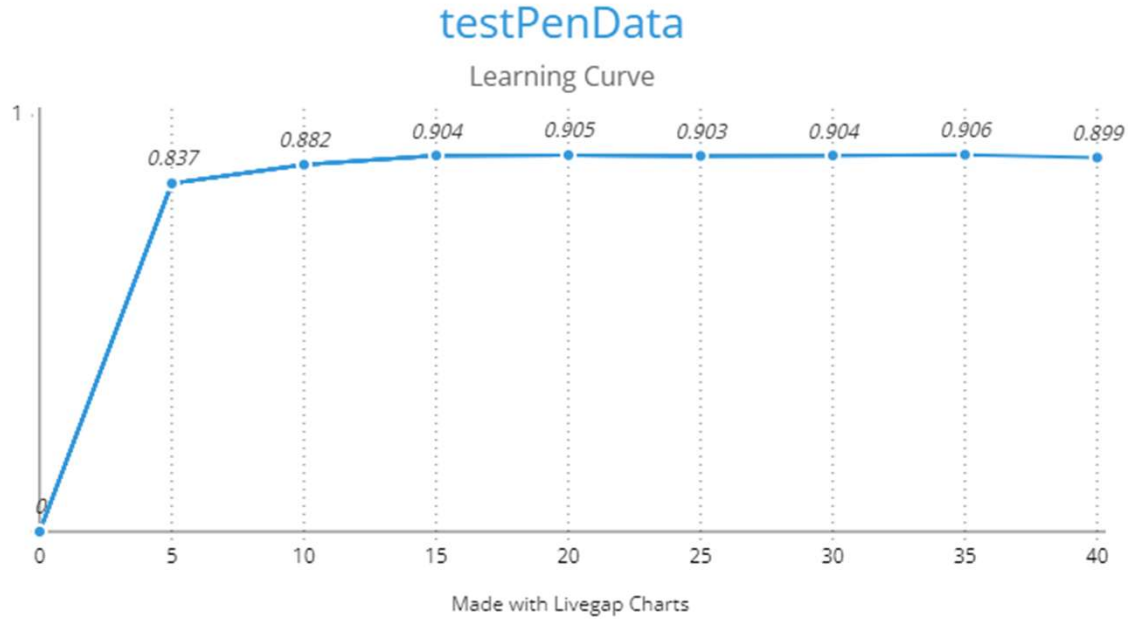
Question 6: Varying the Hidden Layers

Statistic table for **testPenData** – report the max, average, and standard deviation at various amount of perceptrons.

	Number of Perceptrons at the Hidden Layer								
	0	5	10	15	20	25	30	35	40
Max Accuracy	0.0	0.8505	0.8985	0.9057	0.9105	0.9068	0.9085	0.9117	0.9070
Avg Accuracy	0.0	0.8369	0.8819	0.9039	0.9051	0.9034	0.9039	0.9055	0.8989
Standard Deviation	0.0	0.0126	0.0101	0.0015	0.0029	0.0029	0.0028	0.0034	0.0074

Question 6: Varying the Hidden Layers

Create a learning curve for **testPenData** where the number of hidden layer perceptrons is the independent variable and the average accuracy is the dependent variable.



Question 6: Varying the Hidden Layers

For **testPenData**, discuss any notable trends you saw related to increasing the size of the hidden layers in your neural net.

Answer: For testPenData, the accuracy is lowest at the starting 0 neurons. As the neurons increase by 5, there is a rapid growth in the accuracy up to .88. Around the 15-20 neuron mark, the accuracy levels out at around a max of .905. From 15-40, the average accuracy is relatively level. However, as more neurons are introduced, it seems that the average fluctuates and drops a little, while the standard deviation increases.

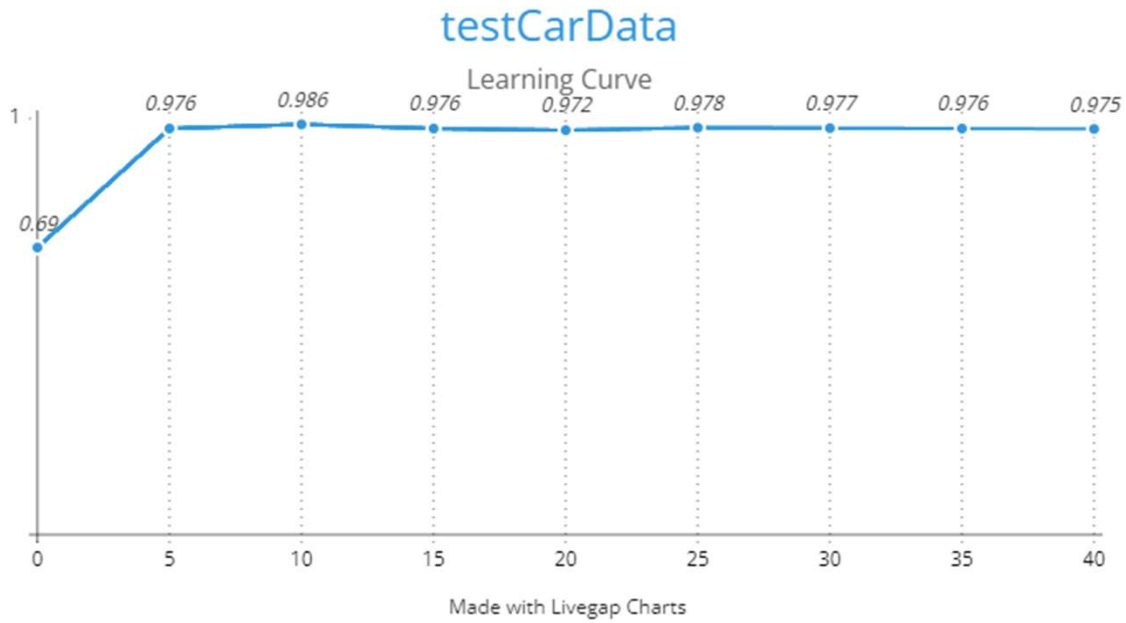
Question 6: Varying the Hidden Layers

Statistic table for **testCarData** – report the max, average, and standard deviation at various amount of perceptrons.

	Number of Perceptrons at the Hidden Layer								
	0	5	10	15	20	25	30	35	40
Max Accuracy	0.69	0.98	0.995	0.985	0.975	0.985	0.985	0.98	0.985
Avg Accuracy	0.69	0.976	0.986	0.976	0.9720	0.978	0.977	0.976	0.975
Standard Deviation	0.00	0.0037	0.0049	0.0058	0.0040	0.0051	0.0051	0.0037	0.0080

Question 6: Varying the Hidden Layers

Create a learning curve for **testCarData** where the number of hidden layer perceptrons is the independent variable and the average accuracy is the dependent variable.



Question 6: Varying the Hidden Layers

For **testCarData**, discuss any notable trends you saw related to increasing the size of the hidden layers in your neural net.

Answer: For testCarData, the starting accuracy is a lot higher with the average being around .7. As the number of neurons increases, so does the average accuracy, rising to a high of 0.986 at 10 neurons. However, after reaching that peak, the average accuracy seems to steadily decline as more neurons are added on. The max seems to fluctuate very little, settling at an average of 0.985, while the standard deviation seems to vary between trials.

Question 7 (extra credit): Learning XOR

Report the max accuracy, average accuracy, and standard deviation of the neural net that you have trained with 1) no hidden layer, and 2) a hidden layer with various amount of perceptrons (at least 3 different amounts)

	No Hidden Layer	Hidden Layer		
		5 perceptrons	25 perceptrons	35 perceptrons
Max Accuracy	0.5	0.75	1.0	1.0
Avg Accuracy	0.25	0.35	0.8	1.0
Standard Deviation	0.2236	0.2550	0.1870	0.0

Question 7 (extra credit): Learning XOR

Report the behavior of the trained neural net **without a hidden layer**.

Answer: **After running multiple trials with no hidden layer involved, it seems that the max accuracy that it reaches is 0.5, without fail. The average will fluctuate up to around 0.35, but does not go higher than that. The standard deviation also seems to be relatively stable.**

Question 7 (extra credit): Learning XOR

Report the behavior of the trained neural net **with a hidden layer**. Are the results what you expected? Explain your observation.

Answer: **After applying a hidden layer, the max quickly jumps from a low of 0.5 max accuracy at 0 neurons to 0.75 max accuracy at 5 neurons. From there, it is a relatively steady increase in the max and average accuracies up to 25 neurons. Here, the max caps at an accuracy of 1.0, while the average remains below it. Surprisingly, when then neuron count reached 35, both the average and max accuracy capped at 1.0, and remained that way regardless of how many more neurons were introduced.**

Question 8 (extra credit): Novel Dataset

List the name and the source of the dataset that you've chosen.

- Name: Poker Hand
- Source (e.g., URLs): https://archive.ics.uci.edu/dataset/158/poker+hand
- Briefly describe the dataset: The dataset consists of a hand of 5 playing cards drawn from a standard deck. The data itself consists of numbers that represent numerical values for the cards. Each number is paired with another that indicates the suit of the card. (Suits are numbered 1-4 and hands are numbered 1-13)

Question 8 (extra credit): Run Stats

- Max accuracy: 0.5125
- Average accuracy: 0.2345740000000002
- Standard deviation: 0.2274293803887264

Question 8 (extra credit): Novel Dataset

Describe how to run the code that you've set up to train the selected dataset.

Answer: I used the given code for getNNPenData and buildExamplesFromPenData to shape my code, with various alterations made to fit to my needs. Regardless, first I refactored the datasets from the library into the databases folder. After referencing those and making my buildExamplesFromExtraData with its get method, I went to Testing.py where I added lines:

- from NeuralNetUtil import buildExamplesFromExtraData

-extraData = buildExamplesFromExtraData()

def testExtraData(hiddenLayers = [24]):

return buildNeuralNet(extraData,alpha=10,maxItr=200,hiddenlayerList=hiddenlayers

After this, I implemented a separate .py file that would import Testing, and had it print the testing average, standard deviation, and max accuracy after 5 iterations.