

Napa (Tan) Vananupong  
GTID 903235545  
Nvananupong3  
CS3600 Project 4

### Question 5.

Pen Dataset:  
Max: 0.9080177545802215  
Average: 0.9053028  
Standard Deviation: 0.0017925030543907

Car Dataset:  
Max: 0.897560208425  
Average: 0.880582150831  
Standard Deviation: 0.0203446725069

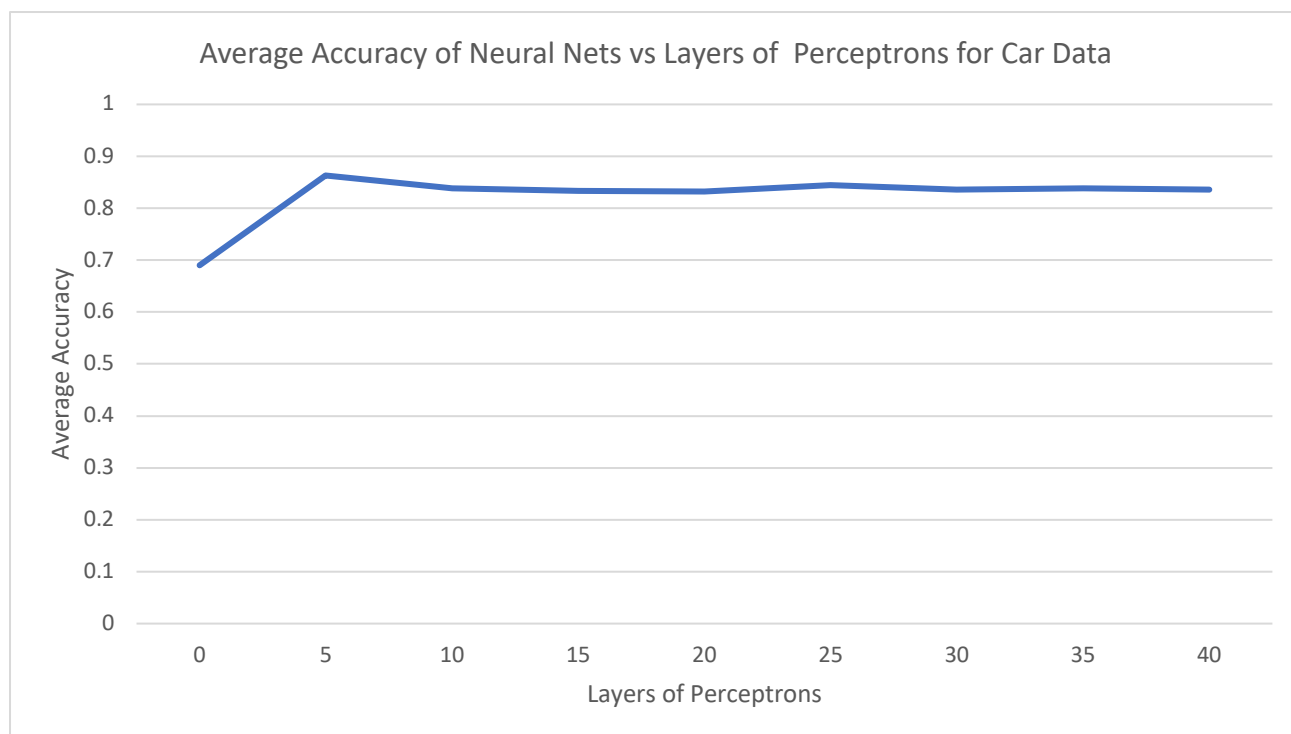
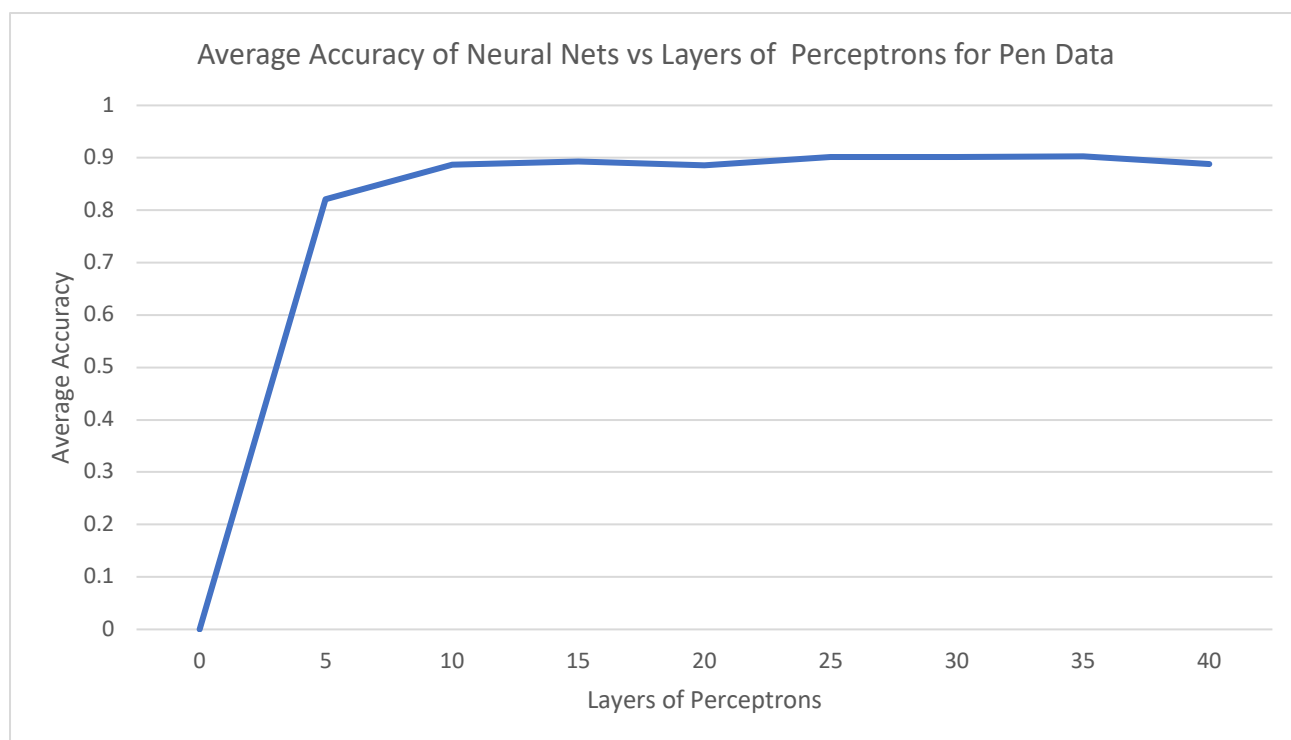
### Question 6.

#### Pen Dataset

Perceptrons	0	5	10	15	20	25	30	35	40
Maximum Accuracy	0.0000	0.8335	0.8971	0.9058	0.9012	0.9056	0.9063	0.9081	0.9053
Average Accuracy	0.0000	0.8213	0.8870	0.8931	0.8859	0.9013	0.9015	0.9028	0.8877
Standard Deviation	0.0000	0.0017	0.0073	0.0073	0.0037	0.0027	0.0053	0.0038	0.0058

#### Car Dataset

Perceptrons	0	5	10	15	20	25	30	35	40
Maximum Accuracy	0.6901	0.8813	0.8715	0.8602	0.8502	0.8584	0.8471	0.8455	0.8439
Average Accuracy	0.6901	0.8631	0.8388	0.8335	0.8327	0.8450	0.8357	0.8378	0.8356
Standard Deviation	0.0000	0.0112	0.0143	0.0128	0.0135	0.0067	0.0072	0.0068	0.0078



### **Noticeable Trends:**

For both the pen and car dataset, the accuracy was increasing steadily as the number of layers of perceptron increased until it hit 5 layers. After it hit 5 layers, growing the number of layers did not affect the accuracy of the neural net much as you can see from the table and charts, the line flattened by a lot, whereas from 0 to 5 the curve was very steep. After 5 layers the average accuracy of the pen dataset never really strayed from 0.9 – they all averaged around 0.9. After 5 layers the average accuracy of the car dataset stabilized around 0.85. I think this makes sense because the neural net was not able to function with just 0 layers, thus the 0.000 accuracy for 0 layers. Then the rapid increase in accuracy from 0 – 5 is most likely due to the fact increasing the layers up to that point allowed my neural net to model non-linearity. However, as we increased the hidden layers, the gradients get smaller and smaller and since the step in weight of the backpropagation algorithm is proportional to the magnitude of the gradient, this means that as the gradients get smaller, so does the weight of the backpropagation algorithm and eventually the gradient will be so small that the weight will be so small as well (i.e very close to 0) which means that the neural net will not be able to train anymore as we saw with the 0 layers situation.