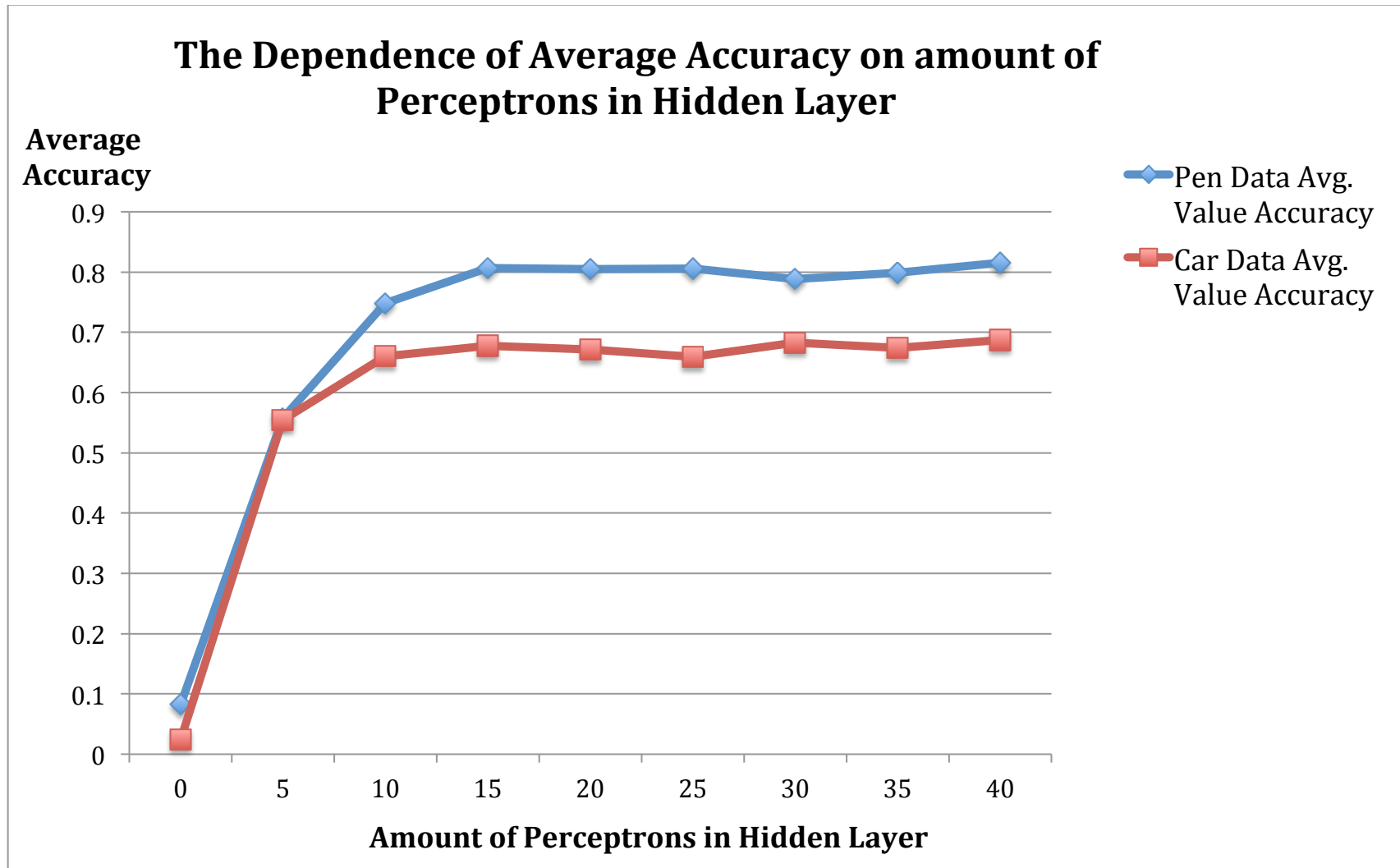


## Question 6: Varying the Hidden Layer

Here is a table of my results of running testPenData and testCarData with 5 iterations of incrementing the number of perceptrons in each layer by 5 from 0 to 40 inclusive. I recorded the Max Value, Standard Deviation, and Average Value of the accuracy in each iteration.

	Pen Data Max Value	Pen Data Avg. Value	Pen Data Std. Deviation	Car Data Max Value	Car Data Avg. Value	Car Data Std. Deviation
0 Perceptrons	0.10405946	0.08324757	0.041623785	0.041230366	0.02473822	0.020198672
5 Perceptrons	0.77472841	0.556889651	0.130114847	0.69895288	0.554973822	0.171635742
10 Perceptrons	0.80617495	0.747913093	0.060901306	0.731020942	0.660340314	0.038993375
15 Perceptrons	0.83133219	0.807089766	0.01539753	0.714005236	0.677225131	0.033224939
20 Perceptrons	0.855917667	0.804688393	0.035740148	0.747382199	0.671727749	0.044909977
25 Perceptrons	0.837621498	0.805717553	0.020612234	0.667539267	0.658900524	0.006377449
30 Perceptrons	0.822184105	0.788736421	0.030642563	0.69895288	0.682722513	0.014117935
35 Perceptrons	0.818181818	0.799142367	0.013497058	0.702225131	0.67421466	0.016397342
40 Perceptrons	0.84419668	0.815494568	0.018481586	0.704842932	0.687041885	0.015046

Here is a graph showing the dependence on the average value of the test accuracy on the number of perceptrons used in the hidden layer. It is applied to the Car and Pen Data.



When running the functions `testPenData` and `testCarData` using the default parameters, my neural network produced that the average value of the accuracy for the Pen Data was around 80% and for the Car Data around 70%. However, after running the test where the number of the perceptrons is varied and starts at 0, it seems that the neural network produces varied results. After analyzing the graph, there seems to be a direct correlation between the amount of perceptrons used in the hidden layer with getting the most accurate results. For instance using 0 and 5 perceptrons, the accuracy for both tests was much less than using 15 perceptrons, where 0 perceptrons obviously yielded the most inaccurate results. However, there is a drop-off in the correlation between increasing perceptrons and accuracy when you reach around 15-25 perceptrons, as once this level was reached, the test's accuracy did not decrease much at all, in fact they both hovered right around 80 and 70 percent. So there is an ideal number of hidden layer perceptrons one wants to use to get the most accurate results of their data, but also to produce those results in an efficient amount of time. In the case of the Car Data, it seems that this ideal number of perceptrons is around 15, where you are still getting very accurate results, and it is much more efficient than using 40 perceptrons. For the Pen Data the ideal number of perceptrons is around 20, as this is very close to the accuracy used for 40 perceptrons, and is much quicker to calculate the result