"Assignment 1) NN matlab simulation for ‘Understanding how to solve pattern recognition problems using backpropagation’ You may find the following two exercises useful as an introduction to the neural network toolbox in matlab. Initial Exercises E1 E2 use matlab help documentation. E1) Read ‘Multilayer Shallow Neural Networks and Backpropagation Training’ for general background. E2) Perform exercise in 'Classify Patterns with a shallow Neural Network' - cancer\_dataset with trainscg and default parameters. Save the ‘advanced script’ at the end of the exercise and use this code as a basis for carrying out the assignment. In particular, you need classification error rather than cross-entropy for Exp 1). (Note that there is a panopto help lecture if needed, see ‘matlab help for using graphical interface to generate code for assignment’) If you are having difficulty with the initial exercises E1,E2 please contact me. For Exp 1) 2) 3) 4) I am happy to clarify what is required but cannot help you to carry out the implementation of assignment. ASSIGNMENT - EXPERIMENTS For help see eeem005\_NNassignment info.pptx Exp 1) Using cancer dataset (as in E2) and ‘trainscg’ or an optimiser of your choice, vary nodes and epochs (that is using early stopping for epochs) over suitable range, to find optimal choice in terms of classification test error rate of node/epochs for 50/50% random train/test split (no validation set). It is suggested that you initially try epochs = [ 1 2 4 8 16 32 64], nodes = [2 8 32], so there would be 21 node/epoch combinations. (Hint 1: from the ‘advanced script’ in E2, nodes and epochs can be changed), Repeat each of the 21 node/epoch combinations at least thirty times, with different 50/50 split and take average and report classification error rate and standard deviation(std). Graph classification train and test error rate and std as node-epoch changes, that is plot error rate vs epochs for different number of nodes. Report the optimal value for test error rate and associated node/epoch values. (Hint 2: as epochs increases you can expect the test error rate to reach a minimum and then start increasing, you may need to set the stopping performance criteria to zero to achieve the desired number of epochs – Hint 3: to find classification error rates for train and test set, you need to check the code from E2. Note: classification error rate = number of mis-classifications divided by total number of patterns, fraction or percentage e.g. predict the class labels from trained network and compute number of differences with target labels. Note that train and test patterns/targets may be obtained using tr.trainInd and tr.testInd (see documentation for train.m) and simulate with y = net(x) and convert to labels as shown in the advanced script from E2. As an alternative, it is also possible to obtain classification error rate from the confusion matrix). Exp 2) For cancer dataset, choose an appropriate value of node and epochs, based on Exp 1) and use ensemble of individual (base) classifiers with random starting weights and Majority Vote to see if performance improves - repeat the majority vote ensemble at least thirty times with different 50/50 split and average and graph (Each classifier in the ensemble sees the same training patterns). Repeat for a different odd number (prevents tied vote) of individual classifiers between 3 and 25, and comment on the result of individual classifier accuracy vs ensemble accuracy as number of base classifiers varies. Consider changing the number of nodes/epochs (both less complex and more complex) to see if you obtain better performance, and comment on the result with respect to why the optimal node/epoch combination may be different for an ensemble compared with the base classifier, as in Exp 1). ( Hint 4: to implement majority vote you need to determine the predicted class labels – probably easier to implement yourself rather than use the ensemble matlab functions. For the graph, use average value over thirty runs for y-axis. The x-axis should be Number of classifiers. Also for complexity, another graph with x-axis as Number of epochs) Exp 3) Repeat Exp 2) for cancer dataset with two different optimisers of your choice e.g. ‘trainlm’ and ‘trainrp’. Comment and discuss the result and decide which is more appropriate training algorithm for the problem. In your discussion, include in your description a detailed account of how the training algorithms (optimisations) work. Exp 4) Please note that you should only attempt this final section if you have time and after you have completed all other sections – only counts ten percent. The requirement is to distinguish between two equi-probable classes of ‘overlapping’ two-dimensional Gaussians’, where class 1 has mean 𝜇1=[0,0] variance 𝜎12=1 and class 2 has 𝜇2=[2,0] variance 𝜎22=4. It can be proved that the Bayes (optimum) boundary for this problem is a circle whose centre is located at [−23⁄,0] and radius 2.34. Find the optimal choice of node/epochs in terms of test error rate, using 300/3000 pattern train/test split, possibly using an ensemble, and plot the decision boundary on the same plot as the Bayes boundary, and discuss your result. (Hint 5:You may find the matlab function mvnrnd.m useful for generating random numbers) It is expected that the final report will be 9-12 pages, but can be longer. Put your matlab code in an appendix. Assignment 1) Marking Scheme matlab simulation Executive Summary [5] Introduction [5] Main body/Content Exp 1) [20] Exp 2) [20] Exp 3) [30] Exp 4) [10] Use of clear/concise English, Report Presentation/ layout [5] Conclusions [5]" - How to do this assignment

Hello! How can I assist you with this assignment on NN matlab simulation for understanding how to solve pattern recognition problems using backpropagation?

Explain in detail each of the steps required to finish this assignment

This assignment involves using the neural network toolbox in MATLAB to carry out experiments on a cancer dataset and a two-dimensional Gaussian dataset. The following are the steps required to finish this assignment:

1. Initial Exercises E1 E2: The first step is to complete the initial exercises E1 and E2. E1 involves reading ‘Multilayer Shallow Neural Networks and Backpropagation Training’ for general background. E2 involves performing an exercise in 'Classify Patterns with a shallow Neural Network' using the cancer dataset with trainscg and default parameters. The advanced script at the end of the exercise should be saved and used as a basis for carrying out the assignment.
2. Exp 1) Varying Nodes and Epochs: For this experiment, the nodes and epochs are varied over a suitable range to find the optimal choice in terms of classification test error rate of node/epochs for a 50/50% random train/test split (no validation set). Initially, epochs = [1 2 4 8 16 32 64] and nodes = [2 8 32], so there would be 21 node/epoch combinations. Each of the 21 node/epoch combinations should be repeated at least thirty times with different 50/50 split and the average and standard deviation of the classification error rate should be reported. The classification train and test error rate and std as node-epoch changes should be graphed. The optimal value for the test error rate and associated node/epoch values should be reported.
3. Exp 2) Ensemble of Individual Classifiers: For this experiment, an appropriate value of node and epochs is chosen based on Exp 1), and an ensemble of individual (base) classifiers with random starting weights and Majority Vote is used to see if performance improves. The majority vote ensemble should be repeated at least thirty times with different 50/50 split and averaged and graphed. Each classifier in the ensemble should see the same training patterns. The experiment should be repeated for a different odd number (prevents tied vote) of individual classifiers between 3 and 25, and the result of individual classifier accuracy vs ensemble accuracy as the number of base classifiers varies should be commented on. Changing the number of nodes/epochs (both less complex and more complex) to see if better performance is obtained should be considered, and the result should be commented on with respect to why the optimal node/epoch combination may be different for an ensemble compared with the base classifier, as in Exp 1).
4. Exp 3) Two Different Optimisers: This experiment involves repeating Exp 2) for the cancer dataset with two different optimisers of your choice, e.g. ‘trainlm’ and ‘trainrp’. The result should be commented on and discussed, and the more appropriate training algorithm for the problem should be decided. The discussion should include a detailed account of how the training algorithms (optimisations) work.
5. Exp 4) Distinguishing Between Two Equi-Probable Classes of Overlapping Two-Dimensional Gaussians: This final experiment should only be attempted if there is time and after all other sections have been completed. The requirement is to distinguish between two equi-probable classes of overlapping two-dimensional Gaussians. The optimal choice of node/epochs in terms of test error rate should be found, using a 300/3000 pattern train/test split, possibly using an ensemble. The decision boundary should be plotted on the same plot as the Bayes boundary, and the result should be discussed.
6. Report: The final step is to write a report on the experiments carried out. The report should be 9-12 pages long and can be longer. The MATLAB code used in the experiments should be included in the report. The report should also include the results

How to do E2 which involves performing an exercise in 'Classify Patterns with a shallow Neural Network' using the cancer dataset with trainscg and default parameters. The advanced script at the end of the exercise should be saved and used as a basis for carrying out the assignment.

To perform exercise E2, which involves using a shallow neural network to classify patterns in the cancer dataset using the trainscg algorithm with default parameters, follow these steps:

1. Open MATLAB and go to the Neural Network Toolbox by typing "nntool" in the Command Window and pressing enter. This will open the Neural Network Fitting app.
2. In the Neural Network Fitting app, select "File" -> "Open" -> "Examples" -> "Neural Network Toolbox" -> "nndemos".
3. In the "nndemos" window, select "Classify Patterns with a Shallow Neural Network" and click on "Open".
4. In the "Classify Patterns with a Shallow Neural Network" window, select "Load Data" and choose the "cancer\_dataset.mat" file. This file contains the cancer dataset that we will use for this exercise.
5. Set the "Network Training Settings" as follows:

* Algorithm: trainscg
* Validation checks: 6
* Maximum epochs: 1000
* Minimum gradient: 1e-6
* Maximum validation failures: 6

1. Set the "Network Architecture" as follows:

* Number of hidden layers: 1
* Number of neurons in the hidden layer: 10
* Activation function: tansig

1. Click on "Train" to start the training process. This will train the neural network using the cancer dataset and the specified settings.
2. Once the training is complete, click on "Test Network" to test the neural network using the same dataset.
3. The "Test Results" window will show the performance of the neural network on the training, validation, and test sets.
4. At the end of the exercise, copy the advanced script that is displayed in the "Advanced" tab and save it as a MATLAB script file. This script can be used as a basis for carrying out the assignment.

To use this script for the assignment, you will need to modify it to meet the specific requirements of the assignment. You may need to change the algorithm, adjust the network architecture, or tune the training parameters to achieve better performance on the given dataset. You should also test the neural network on a separate validation dataset to ensure that it is not overfitting to the training data.