CSE 474
Pattern Recognition Sessional

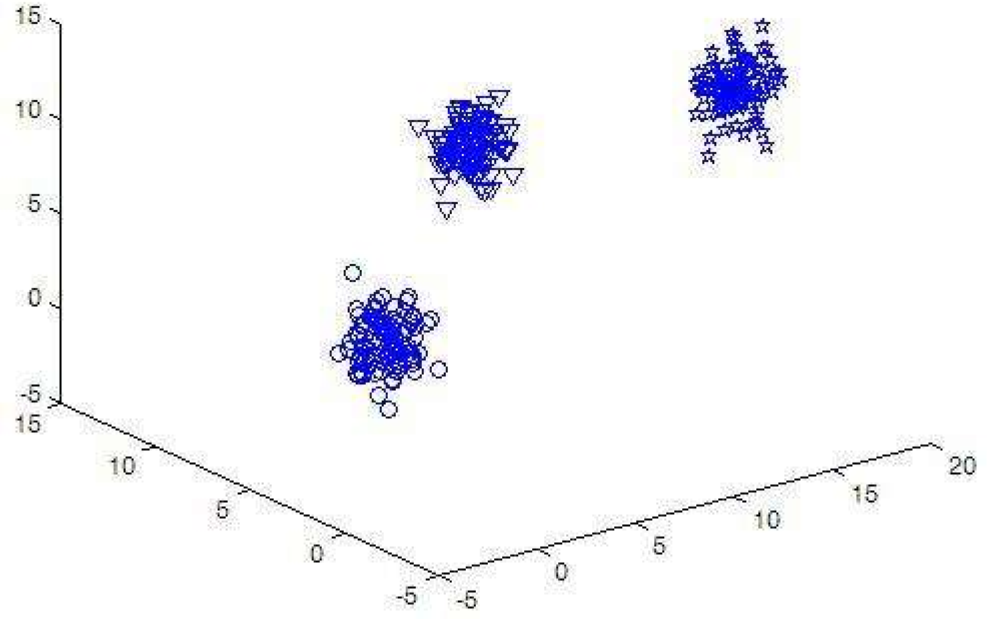
Lab# 3: Implementation of a Neural Network Classifier

Lab Objective

- Implement a multiclass classifier using Neural Network
  - Number classes: variable
  - Feature dimension: variable
  - Network structure: arbitrary

Training Data

- Assume the following training set
  - – Multiple classes
  - – Multiple features



What to do

- Assume the following training set
  - All numerical data
  - Features are real numbers
  - Classes are integers

F1	F2	F3	Class
9.4512	7.3199	6.4664	1
10.7276	9.6067	5.9398	2
10.1960	9.3145	8.3873	1
15.7777	1.5879	11.4440	3
15.8685	2.7902	11.2532	3
14.9448	0.7798	12.7481	2

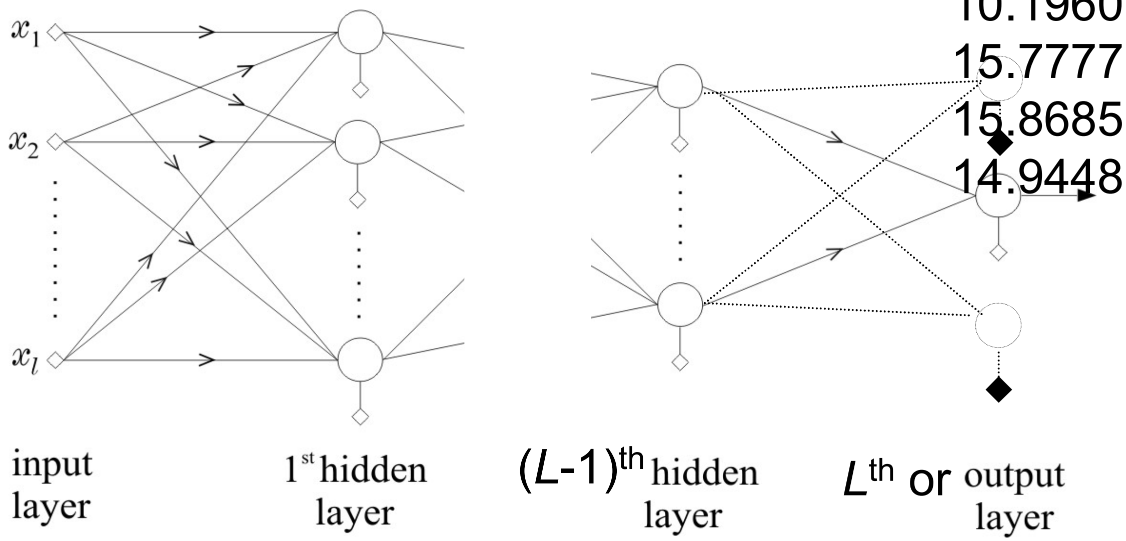
Training Set

What to do

- Use the training set to learn a neural network of *arbitrary structure* using *backpropagation* algorithm

F1	F2	F3	Class
9.4512	7.3199	6.4664	1
10.7276	9.6067	5.9398	2
10.1960	9.3145	8.3873	1
15.7777	1.5879	11.4440	3
15.8685	2.7902	11.2532	3
14.9448	0.7798	12.7481	2

Training Set

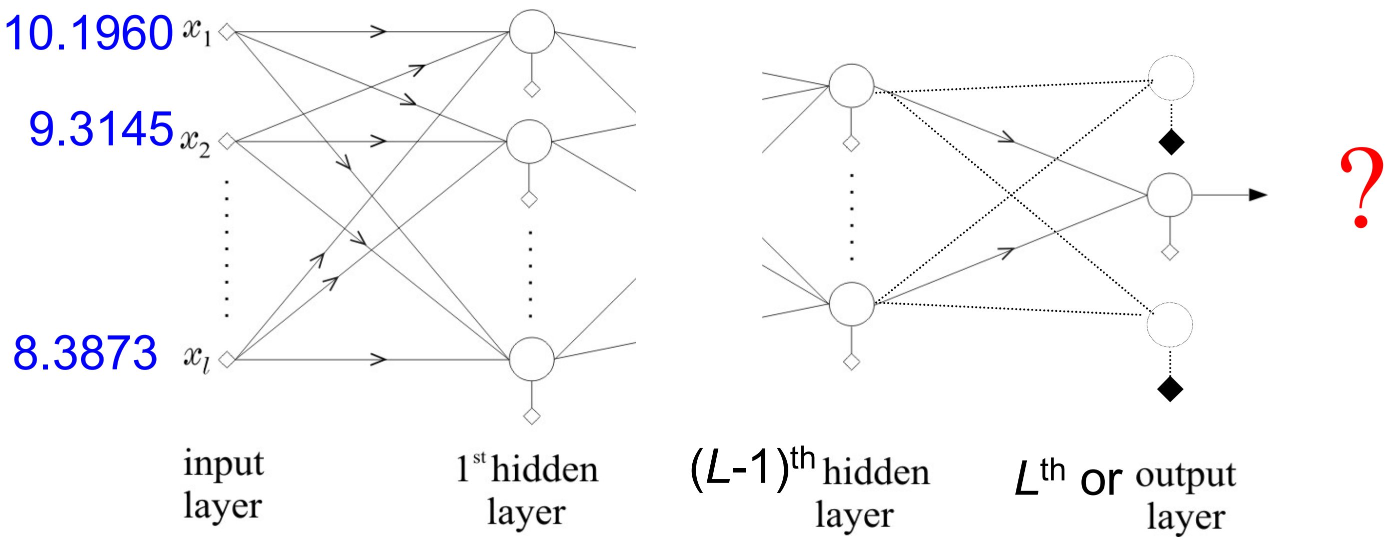


What to do

- Given an unknown sample,

  $[x_1, x_2, x_3]$ = [10.1960 9.3145 8.3873]

Predict its class!



Training and Testing Files

- Each file contains multiple lines
- Each line describes a sample
  - Except the last one, all are real valued features
  - Last number is the class of the sample in integer

- Analyze the training file to know the feature dimension and total number of classes
- You can assume necessary hyper parameters

Output Submission to Moodle (1)

- Change network structure, e.g., no. of layers and nodes in layers.
- For each network
  - Learn *different network parameters (e.g., weights*, etc) from the *supplied training file* using *backpropagation* algorithm
  - Store the network structure and learned parameters in a file.
  - Use the corresponding testing file to identify all misclassified samples and report as follows

no. of layers	no. of nodes/layer	accuracy

Output Submission to Moodle (2)

- Write a separate s/w module to use a learned network
- For each stored file
  - Load the network structure and learned parameters in memory.
  - Use the corresponding testing file to identify all misclassified samples and report as follows

no. of layers	no. of nodes/layer	accuracy

  - Compare this result with that found in the previous slide
- Compile all the reports in a separate word file
- Make a single zip file containing all source codes and the word file and submit at moodle.

Output during Evaluation

- The instructor may ask you to change network structure and to run the experiment using new training/testing files
  - Learn the network using the new training file
  - Use the corresponding testing file to identify all misclassified samples and report as follows

sample no.	feature values	actual class	predicted class

  - % of accuracy

Other information

- Your program must be able to handle variable no. of features, classes, layers and nodes per layer. Hard coded assumption will NOT be accepted.
- Submission deadline is Tuesday 11/01/2022 at 11:55 pm
- Sample training and testing files will be available in the moodle
- Follow the algorithms and notations of your text book (e.g., *Pattern Recognition* by S. Theodoridis)
- You can use your own data to judge your code
- Different files will be used during evaluation
- You can use feature normalization as necessary.