GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Master of Engineering

Level: PG

Branch: Information Technology

Course / Subject Code : ME01000701

Course / Subject Name : Artificial Neural Network

w. e. f. Academic Year:	A.Y. 2024-25
Semester:	1st Semester
Category of the Course:	PEC

Prerequisite:	Python
Rationale:	This course is offered for understanding artificial neural network and their implementations to efficiently solve complex computational problems and optimize software performance.

Course Outcome:

After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level
1	Understanding of basic artificial neural network models and their relation to artificial intelligence	U
2	Evaluate the practical considerations in applying ANNs to real classification and pattern recognition problems	A
3	Use the most common ANN architectures and their learning algorithms	A
4	Application of supervised and unsupervised learning, and generalization ability	E

**Revised Bloom's Taxonomy (RBT)*

Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
				Theory		Tutorial / Practical		
L	T	PR	C	ESE (E)	PA / CA (M)	PA/CA (I)	ESE (V)	
3	0	2	4	70	30	20	30	150

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	Introduction to artificial neural nets, motivation and applications Basic neuron model, Hebbian learning rule, neural network structures	3	10

2.	Single layer perceptron, nonlinear activation function, Delta rule The least mean square (LMS) algorithm, perceptron convergence theorem	5	10
3.	Supervised learning: multilayer networks. Backpropagation Learning General practices for network training and testing, applications of multilayer perceptron	6	15
4.	Recurrent Backprop0gation networks; Backpropagation through time learning algorithm, Radial basis function (RBF) networks and regularization theory	4	15
5.	Kernel methods, support vector machine and its variant	4	10
6.	Mixture of experts, the EM (Expectation Maximization) algorithm	4	10
7	Unsupervised learning, principal components analysis, competitive networks Hopfield networks and Boltzmann machines	4	5
8	Kohonen's self organizing feature maps: Algorithms and applications	4	5
9	Information theoric approach to model selection, independent components analysis	3	5
10	Neural networks for pattern recognition	4	5
11	Artificial Neural Systems: Overview of Actual Models	4	10
	Total	**45**	**100**

Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
10	**30**	**30**	**20**	**10**	**---**

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)

References/Suggested Learning Resources:

(a) Books:

1. James A. Freeman and David M. Skapura, "Neural Networks: Algorithms, Applications, and Programming Techniques", Addison Wesley Publishing Co., 1991, ISBN 0201513765
2. .S. Haykin, "Neural Networks: A Comprehensive Foundation", Prentice Hall, 2nd Ed., 1999, ISBN 8178083000.
3. C. M. Bishop, "Neural Networks for Pattern Recognition", Oxford University Press, 1996, ISBN13: 9780198538646.
4. Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville, Francis Bach

(b) Open source software and website:

1. https://nptel.ac.in/courses/117105084

2. https://courses.cs.washington.edu/courses/cse455/
3. https://nptel.ac.in/courses/108108148
4. https://archive.nptel.ac.in/courses/106/106/106106184

Suggested Course Practical List: (List can be change according to Latest Development)

1. Develop proficiency in writing efficient vectorized code with numpy.
2. Understand Neural Networks and how they are arranged in layered architectures.
3. Implement and apply a Two layer neural network classifier.
4. Understand and be able to implement (vectorized) backpropagation.
5. Implement and apply a k-Nearest Neighbor (kNN) classifier.
6. Implement and apply a Multiclass Support Vector Machine (SVM) classifier.
7. Implement and apply a Softmax classifier.
8. Understand the train/val/test splits and the use of validation data for hyperparameter tuning.
9. Understand and implement RNN.
10. Understand how to leverage self-supervised learning techniques to help with image classification tasks

List of Laboratory/Learning Resources Required: Programming Languages – Python / JAVA etc.

Suggested Project List:

- Understand the basic **Image Classification pipeline** and the data-driven approach (train/predict stages)
- Practice putting together a simple image classification pipeline based on the k-Nearest Neighbor or the SVM/Softmax classifier.

* * * * * * *