**SEMESTER III**

| Course Code | *PAD21301J* | Course Name | DEEP LEARNING FOR DATA SCIENCE | Course Category | C | Professional Core Course | L | T | P | C |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | 4 | 0 | 4 | 6 |

| Pre-requisite Courses | Nil | Co-requisite Courses | Nil | Progressive Courses | Nil |
|---|---|---|---|---|---|
| Course Offering Department | Computer Applications | | Data Book / Codes/Standards | Nil | |

| Course Learning Rationale (CLR): | The purpose of learning this course is to, |
|---|---|
| CLR-1 : | Develop knowledge of Neural Network |
| CLR-2 : | Understand and analyze deep learning |
| CLR-3 : | Perform Optimization Techniques |
| CLR-4 : | Implement Deep Learning models |
| CLR-5 : | Get Familiar with Keras library |
| CLR-6 : | Implement Deep Q-Learning |

| Course Learning Outcomes (CLO): | At the end of this course, learners will be able to: | Learning | | | Program Learning Outcomes (PLO) | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| | | Level of Thinking (Bloom) | Expected Proficiency (%) | Expected Attainment (%) | Fundamental Knowledge | Application of Concepts | Link with Related Disciplines | Procedural Knowledge | Skills in Specialization | Ability to Utilize Knowledge | Skills in Modeling | Analyze, Interpret Data | Investigative Skills | Problem Solving Skills | Communication Skills | Analytical Skills | ICT Skills | Professional Behavior | Life Long Learning |
| CLO-1 : | Build a Perceptron model | 2 | 85 | 80 | H | H | H | H | H | H | - | M | M | L | - | H | - | M | H |
| CLO-2 : | Build Neural Network model using BP algorithm | 3 | 85 | 80 | L | H | H | H | H | H | - | M | M | L | - | H | - | M | H |
| CLO-3 : | Using Pytorch to build a prediction model | 3 | 85 | 80 | L | H | H | H | H | H | - | M | M | L | - | H | - | M | H |
| CLO-4 : | Fine Tuning the Deep Learning models for performance optimization | 3 | 85 | 80 | L | H | H | H | H | H | - | M | M | L | - | H | - | M | H |
| CLO-5 : | Build, Compile, Test, and evaluate model in Keras | 3 | 85 | 80 | L | H | H | H | H | H | - | M | M | L | - | H | - | M | H |
| CLO-6 : | Work with Generative Adversarial Networks | 3 | 85 | 80 | L | H | H | H | H | H | - | M | M | L | - | H | - | M | H |

| Duration (hour) | | 24 | 24 | 24 | 24 | 24 |
|---|---|---|---|---|---|---|
| S-1 | SLO-1 | *Introduction to Neural Network* | Fine Tuning NN models<br>ANN Processing Components | Fine Tuning NN Models<br>What is Fine Tuning? | Keras and DL<br>Overview of TFMA | Interactive Applications of DL |
| | SLO-2 | *Fundamentals* | Learning and Training in ANN | Regularization | Practical Consideration for DL | Machine Vision – CNN |

| | | | | | | |
|---|---|---|---|---|---|---|
| S-2 | SLO-1 | *Biological NN Vs ANN* | Cluster analysis in ANN | What is Vector Quantization? | DL parameters | Pooling Layers |
| | SLO-2 | *ANN Architecture* | NN Building blocks | The Encoder-Decoder Model | Data Loading and Preprocessing | Lenet-5 in Keras |
| S-3 | SLO-1 | *Computational Models in NN* | Perceptron to Deep NN | The Generalized Lloyd Algorithm | Data Preprocessing with Keras | Alexnet and vggnet in keras |
| | SLO-2 | *Neurons Interconnection* | Model and Hyper parameters | Relation between SOM and noisy encoder-decoder | Keras Layers | Natural Language processing |
| S-4 | SLO-1 | *Threshold Functions* | Classification with NN | Voronoi Tesselation | Training Models with fit() | Creating word embeddings with word2vec |
| | SLO-2 | *Activation functions & ANN* | Deep Learning Frameworks | LVQ Introduction | Monitoring Performance Metrics | Natural Language Classification with familiar networks |
| S 5-8 | SLO-1 | Lab 1: Implement a Feed Forward Neural Network with Back propagation training algorithm for realizing XOR problem | Lab 4: Build a NN model using PyTorch | Lab 7: Implement LVQ Network for Pattern Classification | Lab 10: Build a model for Credit Card Fraudulence Detection | Lab 13: Build a CNN model for Image Classification |
| | SLO-2 | | | | | |
| S-9 | SLO-1 | *Implementing Neural Networks Building Neural Networks Models* | NN Categorization | The LVQ Algortihm | Checkpointing | Generative Adversarial Networks |
| | SLO-2 | *Use case of ANN* | NN Computational Model | The LVQ2 Algorithm | Debugging the model with eager execution | Essential GAN Theory |
| S-10 | SLO-1 | Perceptrons | NN Building Components | Hebbian Learning | Speed Up process with multiple GPUs | The Discriminator Network |
| | SLO-2 | Single Layer Perceptron Model | Evolutionary Algorithm & Gradient Descent | Hebbian Learning Rule | Multiple GPU and distributed trainings | The Generator Network |
| S-11 | SLO-1 | Least Mean Square Algorithm | Object Image Classification | Competitive Learning | Transfer Learning | The Adversarial Network |
| | SLO-2 | Learning Curves | Learning rates and Optimization | Optimizing NN | Image classification | GAN Training |
| S-12 | SLO-1 | Learning Rates | Optimizing Speed | Debugging NN | Keras Metrics | Reinforcement Learning |
| | SLO-2 | Perceptron | Dense Network Tuning using Hyper parameters | Learning rate optimization | Jupyter notebooks | Reinforcement Learning Process steps |
| S 13-16 | SLO-1 | Lab 2: Implement a Perceptron in Python | Lab 5: Implement ANN Training in Python for MNIST Digit Classification problem | Lab 8: Using Keras, perform rate adaption schedule. | Lab 11: Work on a text classification problem with Keras API | Lab 14: Design and build a Game environment |
| | SLO-2 | | | | | |
| S-17 | SLO-1 | Multilayer Perceptron | Linear Model with Estimators | Optimizing Networks | Dataset for NN | Deep Reinforcement Learning Applications |
| | SLO-2 | The XOR Problem | NN for Predictions | Rate adaption schedule | Exploring the Dataset | Deep RL Use cases |
| S-18 | SLO-1 | Back Propagation Algorithm | Optimization approaches for prediction | Scaling | Preparing the dataser | Deep-Q Learning Introduction |
| | SLO-2 | Heuristics for improving BP algorithm | NN algorithms | Scaling methods | Visualizing the dataset | The DQN Agent |

| | | | | | | |
|---|---|---|---|---|---|---|
| S-19 | SLO-1 | Radial Basis Function Networks | Data preparation for NN | Batch Normalization | Compiling the model | Q-Learning |
| | SLO-2 | Interpolation | ANN Training in python | Mini Batch Normalization | Training the NN | Deep Q Learning |
| S-20 | SLO-1 | Regularization | Training Samples | Internal Covariate Shift | Testing the NN | Steps in Deep Q Learning |
| | SLO-2 | Learning Strategies | Overfitting and Underfitting | Implement Gradient learning | Evaluate the model | Experience Replay |
| S 21-24 | SLO-1<br>SLO-2 | Lab 3: Implement a Feed Forward Neural Network with Back propagation training algorithm for realizing Straight line e.g. y = 2x + 3 | Lab 6: Perform Hyper parameter tuning in the ANN model implemented in Lab 5 | Lab 9: Implement Batch Normalization and gauge its performance | Lab 12: Build a DL model for diabetes classification problem | Lab 15:Build and Train the Deep Q Neural Network |

| | | |
|---|---|---|
| Learning Resources | **1. Deep Learning with Python,By Francois Chollet, December 2017**<br>**2. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence,By Jon Krohn, Grant Beyleveld and Aglaé Bassens, September 2019**<br>**3. Hugo Larochelle's Video Lectures on Deep Learning** | **4. Introduction to Deep Learning by Sandro Skansi, Springer, 2018**<br>**5. Deep Learning withTensorFlow 2 and Keras - Second Edition,By Antonio Gulli, Amita Kapoor and Sujit Pal, December 2019** |

**Learning Assessment**

| | Bloom's Level of Thinking | Continuous Learning Assessment (50% weightage) | | | | | | | | Final Examination (50% weightage) | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | CLA – 1 (10%) | | CLA – 2 (10%) | | CLA – 3 (20%) | | CLA – 4 (10%)# | | | |
| | | Theory | Practice | Theory | Practice | Theory | Practice | Theory | Practice | Theory | Practice |
| Level 1 | Remember | 20% | 20% | 15% | 15% | 15% | 15% | 15% | 15% | 15% | 15% |
| | Understand | | | | | | | | | | |
| Level 2 | Apply | 20% | 20% | 20% | 20% | 20% | 20% | 20% | 20% | 20% | 20% |
| | Analyze | | | | | | | | | | |
| Level 3 | Evaluate | 10% | 10% | 15% | 15% | 15% | 15% | 15% | 15% | 15% | 15% |
| | Create | | | | | | | | | | |
| | Total | 100 % | | 100 % | | 100 % | | 100 % | | 100 % | |

# CLA – 4 can be from any combination of these: Assignments, Seminars, Tech Talks, Mini-Projects, Case-Studies, Self-Study, MOOCs, Certifications, Conf. Paper etc.,

**Course Designers**

| Experts from Industry | Experts from Higher Technical Institutions | Internal Experts |
|---|---|---|
| Mr.G.Muruganandam, Group Project Manager, HCL Technologies, Chennai | Dr.Muthu, Professor, Loyola College, Chennai | Mrs. M.Ramla, Assistant Professor,SRM IST |
| Mr.M. Hemachandar, Tech Lead, Wipro Limited, Chennai | Dr.Vincent, Associate Professor, VIT | |