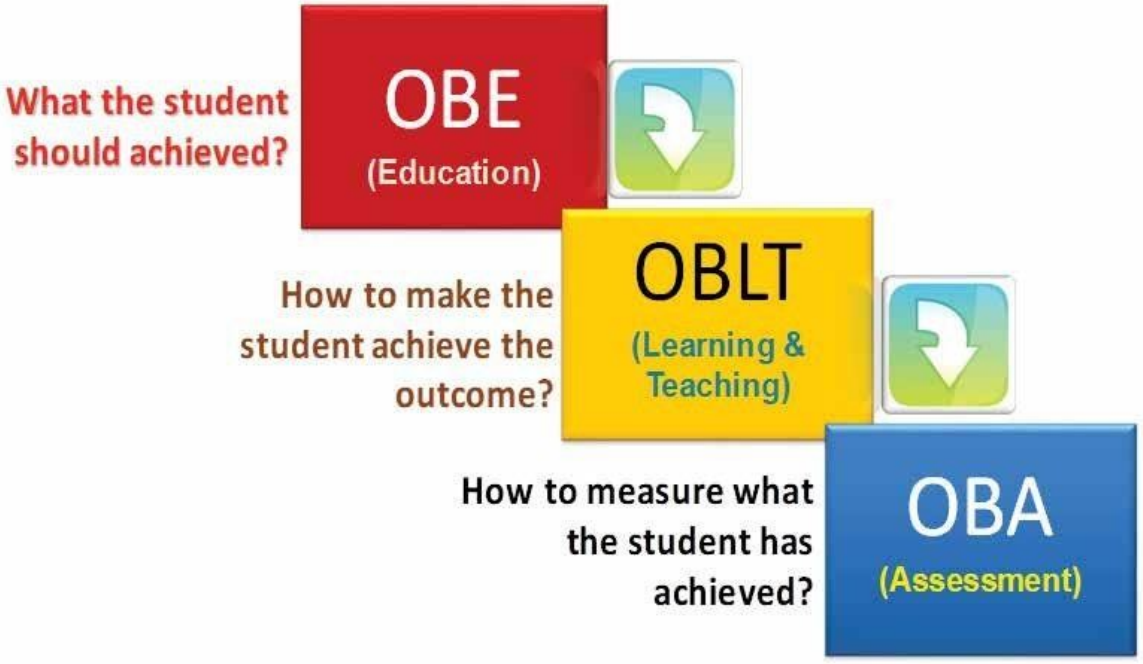
GALGOTIAS UNIVERSITY

GUSOP-01

Course-Pack Framework

A comprehensive instructional delivery document



Prepared by:

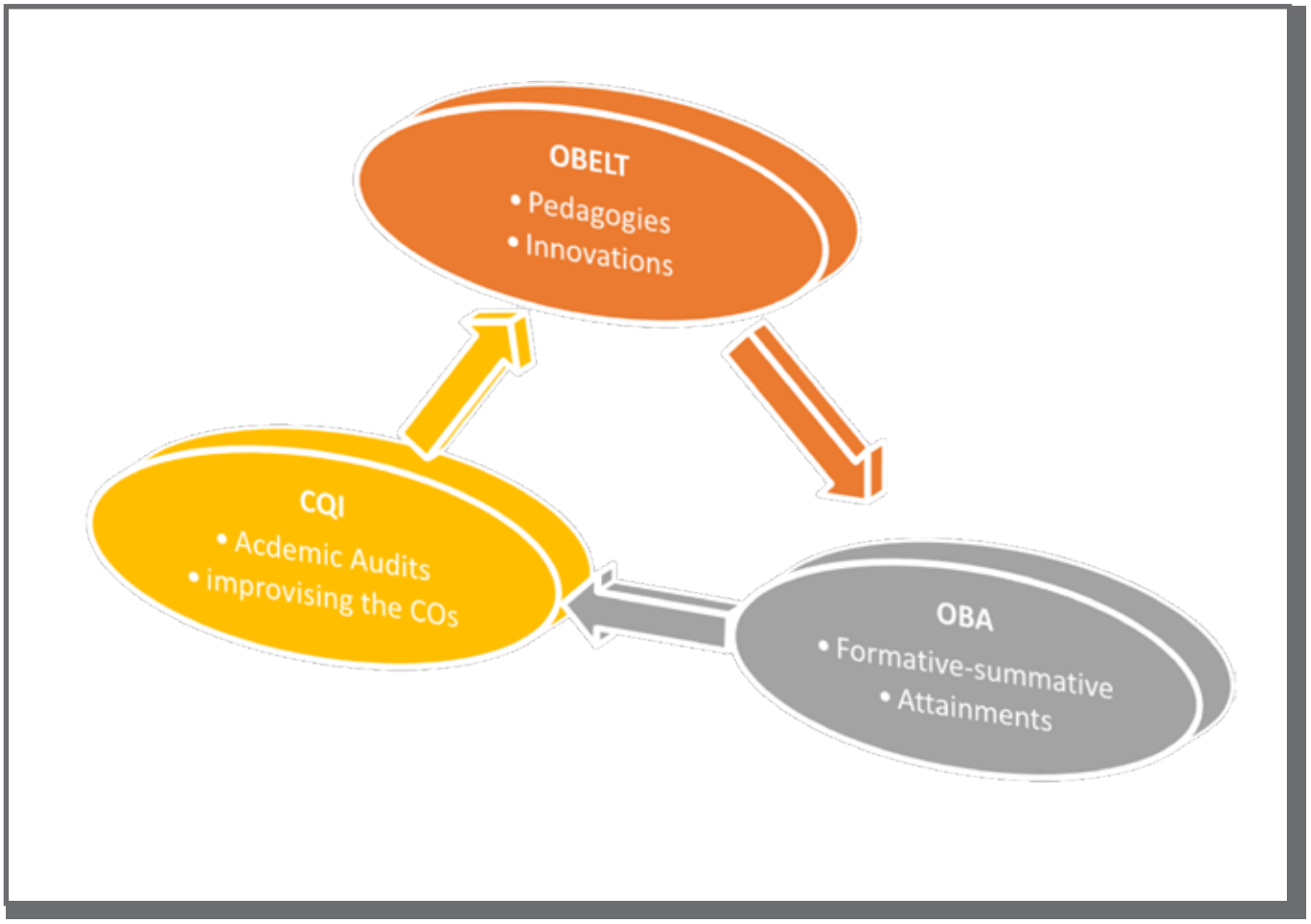
VCO

(Revised on Aug-2023)

The Course Pack is a comprehensive and complete pedagogical guideline document that describes the components of instruction delivery by a faculty member. It consists of the scheme of the course, Course Overview, Course Objectives, Prerequisite course, Program-specific Outcomes (PSOs), Course outcomes (COs), Bloom's taxonomy (Knowledge Levels), Types of Courses, Course articulation matrix, Course assessment patterns, Course content, Lesson Plan, Bibliography, Problem-based learning/case-studies/clinical, and Student-Centred learning (self-learning towards life-long-learning). It not only provides a uniform design of Course delivery across the University but also ensures freedom and flexibility to introduce innovations in learning and teaching and create vivid kinds of assessment tools (alternate assessment tools) by a faculty member.

The course pack is developed by the faculty member teaching a course. If more than one faculty teaches the same course, all the faculty members teaching the course shall be formed as a cluster, and a senior faculty member (Course-lead) lead the Course delivery design in a team effort. The Course Pack provides ample scope and opportunity to bring innovations in teaching pedagogies in a school/department.

Hence, the Course pack is a comprehensive learning-teaching strategy framework to be followed by all the faculty members in schools/departments in the university. It is not only a tool for measuring the learning of a class but also analyses the achievement levels (learning outcomes of the course) of all the students in a class in a continuous manner.



COURSEPACK (Even 2023-24)

SCHEME

The scheme is an overview of work-integrated learning opportunities and gets students out into the real world. This will give what a course entails.

<table>
<tr><td>Course Title</td><td colspan="3">Deep Learning</td><td colspan="2">Course Type</td><td colspan="4">Comprehensive</td></tr>
<tr><td>Course Code</td><td colspan="3">R1UC604C</td><td colspan="2">Class</td><td colspan="4">B. Tech.</td></tr>
<tr><td rowspan="6">Instruction delivery</td><td>Activity</td><td>Credits</td><td>Credit Hours</td><td colspan="4" rowspan="2">Total Number of Classes per Semester</td><td colspan="2" rowspan="2">Assessment in Weightage</td></tr>
<tr><td>Lecture</td><td>3</td><td>3</td></tr>
<tr><td>Tutorial</td><td>0</td><td>0</td><td rowspan="3">Theory</td><td rowspan="3">Tutorial</td><td rowspan="3">Practical</td><td rowspan="3">Self-study</td><td rowspan="3">CIE</td><td rowspan="3">SEE</td></tr>
<tr><td>Practical</td><td>1</td><td>2</td></tr>
<tr><td>Self-study</td><td>1</td><td>4</td></tr>
<tr><td>Total</td><td>5</td><td>9</td><td>45</td><td>0</td><td>30</td><td>60</td><td>50%</td><td>50%</td></tr>
<tr><td>Course Lead</td><td colspan="2">Dr. Saurabh Singh</td><td>Course Coordinator</td><td colspan="6">Dr. Aditya Kishor Saxena</td></tr>
<tr><td rowspan="2">Names Course Instructors</td><td colspan="3">Theory</td><td colspan="6">Practical</td></tr>
<tr><td colspan="3">1. Dr. Saurabh Singh
2. Dr. Aditya K Saxena
3. Dr. Abdul Aleem
4. Dr. Deepa Joshi
5. Dr. Garima Pandey
6. Dr. Siksha Singh
7. Dr. Vimal Singh
8. Dr. Himanshu Sharma
9. Dr. Suman Mann
10.Dr. Sandeep Kumar</td><td colspan="6">1. Dr. Saurabh Singh
2. Dr. Aditya K Saxena
3. Dr. Abdul Aleem
4. Dr. Deepa Joshi
5. Dr. Garima Pandey
6. Mr. Siksha Singh
7. Dr. Vimal Singh
8. Dr. Himanshu Sharma
9. Dr. Suman Mann
10.Dr. Sandeep Kumar</td></tr>
</table>

COURSE OVERVIEW

Deep learning is a sub-field of machine learning that focuses on learning complex, hierarchical feature representations from raw data. The dominant method for achieving this, artificial neural networks, has revolutionized the processing of data (e.g. images, videos, text, and audio). In this course, students will learn the fundamental principles, underlying mathematics, and implementation details of deep learning. This includes the concepts and methods used to optimize these highly parameterized models, the modules that make them up (linear, convolution, and pooling layers, activation functions, etc.), and common neural network architectures (convolutional neural networks, recurrent neural networks, etc.). Through in- depth programming assignments, students will learn how to implement these fundamental building blocks as well as how to put them together using Python.

PREREQUISITE COURSE

PREREQUISITE COURSE REQUIRED	YES	
	Prerequisite course code	**Prerequisite course name**
		Machine Learning

COURSE OBJECTIVE

- To understand the fundamentals of neural networks.
- To apply the concept of neural network in real-time use of deep learning.
- To demonstrate real-time development of applications of deep learning.
- To apply the basic concept of deep learning in the implementation of convolutional neural network over various datasets.
- To implement various automatic model of Deep Learning neural networks and evaluate their performance.

COURSE OUTCOMES(COs)

After the completion of the course, the student will be able to:

CO No.	Course Outcomes
R1UC604C.1	Understand the fundamentals of Neural Networks.
R1UC604C.2	Apply the concepts of Neural Networks in the development of deep learning models.
R1UC604C.3	Develop and Analyze the deep learning models like RNN, CNN and LSTM.
R1UC604C.4	To be able to evaluate and optimize deep learning models

BLOOM'S LEVEL OF THE COURSE OUTCOMES

Bloom's taxonomy is a set of hierarchical models used for the classification of educational learning objectives into levels of complexity and specificity. The learning domains are cognitive, affective, and psychomotor.

CO No.	Remember **KL1**	Understand **KL 2**	Apply **KL 3**	Analyse **KL 4**	Evaluate **KL 5**	Create **KL 6**
R1UC604C.1	√	√				
R1UC604C.2			√			
R1UC604C.3				√		√
R1UC604C.4					√	

PROGRAM OUTCOMES (POs):

PO1	The program enhances analytical, managerial, communication and computing Application skills besides inculcating the virtues of self-study
PO2	Analyze and synthesis computing systems through quantitative and qualitative techniques.
PO3	Design and develop computer programs in the areas related to algorithm, web design, networking and AI
PO4	Inculcating essential skills as demanded by Indian and Global Software industries through latest tools of learning. This also includes team-building skills, audio- visual presentations and personality development programs.
PO5	To develop inter-twining competence in the field of Digital Marketing and Commerce, Computing Skill and Computational tools.
PO6	To Develop practical skills to provide solutions to industry, society and business.
PO7	Understand environmental issues and lead a life with concerns for environment.
PO8	Apply ethical principles and commit to professional ethics and responsibilities and norms of the computing science practice.
PO9	To make graduates understand cross cultural, societal, professional, legal and ethical issues prevailing in industry.
PO10	Communicate effectively in both verbal and written form
PO11	To apply standard software engineering practices and strategies in software project development using open-source programming environment to deliver a quality of product for business success
PO12	Recognize the need for lifelong learning for continuous enhancement and up gradation of technological changes in society.

Program Specific Outcomes (PSOs):

PSO1	Have the ability to work with emerging technologies in computing requisite to Industry 4.0.
PSO2	Demonstrate Engineering Practice learned through industry internship and research project to solve live problems in various domains.

COURSE ARTICULATION MATRIX

The Course articulation matrix indicates the correlation between Course Outcomes and Program Outcomesand their expected strength of mapping in three levels(low,medium,and high).

COs#/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
R1UC604C.1	1													
R1UC604C.2	3					1							1	
R1UC604C.3		3		3										2
R1UC604C.4			3		2								3	

Note: 1-Low, 2-Medium, 3-High

COURSE ASSESSMENT

The course assessment patterns are the assessment tools used both in formative and summative examinations.

Type of Course (C)	CIE			Total Marks		Final Marks CIE*0.5+SEE*0.5
	LAB@ (Work+ Record)	MTE	Course-based Project^	CIE	SEE	
COMPREHENSIVE	25	50	25	100	100	100

COURSE CONTENT

THEORY+ PRACTICAL

THEORY:

Machine Learning Brief: Introduction, Types of Machine Learning, Supervised Learning, Unsupervised Learning, Reinforcement Learning. Applications of Machine Learning, Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thresholding logic, Linear Perceptron, Perceptron Learning Algorithm, Linear separability, Multilayer Perceptron.

ANN: Architecture of ANN, Types of Artificial Neural Network, Feed forward Neural Network, Back propagation Algorithm, Gradient Descent, Activation Functions, Types of Activation Functions, Deep Neural Network, Difficulty of training deep neural networks, Greedy layer wise training, Regularization.

DL Models: Recurrent Neural Network (RNN), Long Short-Term Memory, Gated Recurrent Units, Bidirectional LSTMs, Convolutional Neural Network, Introduction to images, Layers in CNN, Convolution Layer, Min-Max Pooling Layer, Types of Pooling, Flattening Layer, Padding Layer, Fully Connected Layer, Transfer Learning, Using pre-trained learning models-YOLO.

Advance Techniques: Advanced optimization methods for neural networks (Adagrad, rmsprop, adam), Saddle point problem in neural networks, Regularization methods (dropout, drop connect, batch normalization). Generative Adversarial Networks, Multi-task Deep Learning, Multi-view Deep Learning.

LESSON PLAN

FOR THEORY 15 weeks * 3 Hours = 45 Classes) (1credit = 1Lecture Hour)
FOR PRACTICAL 15 weeks * 2Hours = 30 Hours lab sessions (1 credit = 2 lab hours)

Lec. No	Topic	Theory/Practical	Skills	Competency (CO)
1	Machine Learning Brief: Applications, Types of Machine Learning, Supervised Learning	Theory	Understanding Basics of Machine Learning	CO1
2	Unsupervised Learning	Theory		
3	Reinforcement Learning	Theory		
4	Python Programming Fundamental Revision.	Practical		
5	Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thresholding logic	Theory		
6	Linear Perceptron, Perceptron Learning Algorithm, Linear separability.	Theory		
7	Multilayer Perceptron	Theory		
8	Python Programming Libraries Revision.	Practical		
9	Tutorial/Revision	Theory		
10	Tutorial/Revision	Theory		
11	Architecture of ANN, Types of Artificial Neural Network, Feed forward Neural Network	Theory	Applying Machine Learning for Problem Solving	CO1, CO2
12	Python program to implement k-Nearest Neighbor algorithm to classify the iris dataset. Print both correct and wrong predictions.	Practical		
13	Activation Functions, Types of Activation Functions	Theory		
14	Back propogation Neural Network	Theory		
15	Gradient Descent	Theory		
16	Python program to implement the non-parametric locally weighted regression algorithm in order to fit the data point. Select the appropriate data set for your experiment and draw graphs.	Practical		
17	Deep Neural Network, Difficulty of training deep neural networks	Theory		
18	Greedy layer wise training	Theory		
19	Regularization methods (dropout, drop connect, batch normalization)	Theory		
20	Python program to build a machine learning model which will predict whether or not it will rain tomorrow by studying past data.	Practical		
21	Tutorial/Revision	Theory		

22	Tutorial/Revision	Theory		
23	Tutorial/Revision	Theory		
24	Python program to build a machine learning model which will predict whether or not it will rain tomorrow by studying past data.	Practical	Basics of Multi layer Perceptron Learning	CO2, CO3
25	Recurrent Neural Network (RNN)	Theory		
26	Recurrent Neural Network (RNN)	Theory		
27	Long Short-Term Memory, Gated Recurrent Units	Theory		
28	Python program to implement AND, OR gates using perceptron.	Practical		
29	Bidirectional LSTMs	Theory		
30	Image processing basics, Types of Digital Images	Theory		
31	Convolutional Neural Network	Theory		
32	Python program for classification of an XOR problem with multi-layer perceptron.	Practical		
33	Layers in CNN	Theory	Solving Real- world problems using Perceptron Models	CO3
34	Pooling Layer, Types of Pooling	Theory		
35	Flattening Layer, Padding Layer, Fully Connected Layer,	Theory		
36	Python program to implement classification linearly separable data with perceptron.	Practical		
37	Transfer Learning	Theory		
38	Using pre-trained learning models-YOLO	Theory		
39	Tutorial/Revision	Theory		
40	Python program to recognize handwritten digits using neural network.	Practical	Understandi g Neural Networks Optimizatio Methods	CO3
41	Tutorial/Revision	Theory		
42	Tutorial/Revision	Theory		
43	Advanced optimization methods for neural networks (Adagrad, rmsprop, adam)	Theory		
44	Python program to study a bank credit data set and determine whether a transaction is fraudulent or not based on past data.	Practical		
45	Saddle point problem in neural networks,	Theory		
46	Saddle point problem in neural networks,	Theory		

47	Python program for implementation of CNN.	Practical		
48	Regularization methods (dropout, drop connect, batch normalization)	Theory	Understanding and Analyzing Variations of ANN for different tasks	CO3, CO4
49	Regularization methods (dropout, drop connect, batch normalization)	Theory		
50	Generative Adversarial Networks	Theory		
51	Python program for implementation of RNN.	Practical		
52	Generative Adversarial Networks	Theory		
53	Multi-task Deep Learning	Theory		
54	Multi-view Deep Learning.	Theory		
55	Python program for implementation of LSTM.	Practical		
56	Tutorial/Revision	Theory		
57	Tutorial/Revision	Theory		
58	Tutorial/Revision	Theory		
59	Python program for implementation of Bidirectional LSTM.	Practical		CO4
60	Tutorial/Revision	Theory		

	List of course based projects	
1	AI-based Game Playing Agent: Create an AI agent capable of playing and winning games like chess, Go, or Atari games using reinforcement learning.	KL6
2	Handwritten Digit Recognition: Implement a neural network to classify handwritten digits from the MNIST dataset.	KL6
3	Sentiment Analysis: Build a model to analyze sentiment in text data, such as movie reviews or tweets.	KL6
4	Image Classification: Develop a convolutional neural network (CNN) to classify images from datasets like CIFAR-10 or ImageNet.	KL6
5	Predictive Maintenance: Create a model to predict equipment failure or maintenance needs based on sensor data.	KL6
6	Healthcare Diagnosis: Use machine learning techniques to assist in medical diagnosis based on patient data or medical images.	KL6
7	Stock Price Prediction: Build a model to predict stock prices using historical data and technical indicators.	KL6
8	Spam Email Detection: Develop a classifier to identify spam emails from a given dataset.	KL6
9	Recommendation Systems: Create a recommendation system for movies, products, or music based on user preferences.	KL6
10	Anomaly Detection: Build a model to detect anomalies in data, such as fraudulent transactions or network intrusions.	KL6

11	Natural Language Processing (NLP): Implement NLP techniques for tasks like text summarization, named entity recognition, or language translation.	KL6
12	Facial Recognition: Develop a system to recognize faces in images or videos using deep learning techniques.	KL6
13	Autonomous Vehicles Simulation: Create a simulated environment where autonomous vehicles navigate using reinforcement learning algorithms.	KL6
14	Gesture Recognition: Build a model to recognize hand gestures from images or videos, which can be used for human-computer interaction.	KL6
15	Customer Churn Prediction: Develop a model to predict customer churn for businesses based on historical customer data.	KL6
16	Fake News Detection: Implement a model to classify news articles as real or fake based on their content.	KL6
17	Predictive Analytics for Energy Consumption: Use machine learning to predict energy consumption patterns and optimize energy usage.	KL6
18	Credit Risk Assessment: Build a model to assess credit risk for loan applicants using historical financial data.	KL6
19	Object Detection: Develop a model to detect and localize objects within images or videos.	KL6
20	Human Activity Recognition: Create a system to recognize different human activities, such as walking, running, or sitting, from sensor data.	KL6
21	Brain-Computer Interface: Explore EEG data and build a model to interpret brain signals for controlling devices or applications.	KL6
22	Customer Segmentation: Use clustering algorithms to segment customers based on their behavior or demographics.	KL6
23	Language Translation: Implement a neural machine translation system to translate text between different languages.	KL6
24	Document Classification: Build a model to classify documents into predefined categories based on their content.	KL6
25	Image Style Transfer: Develop a model to transfer the style of one image onto another, creating artistic effects.	KL6
26	Time Series Forecasting: Use recurrent neural networks (RNNs) or long short-term memory networks (LSTMs) to forecast time series data, such as stock prices or weather patterns.	KL6
27	Emotion Recognition: Build a model to recognize emotions from facial expressions in images or videos.	KL6
28	Chatbot Development: Create a conversational AI chatbot that can answer user queries or assist in tasks.	KL6
29	Semantic Segmentation: Develop a model to segment images at the pixel level, identifying different objects or regions within the image.	KL6
30	Speech Recognition: Build a system to transcribe spoken language into text using deep learning techniques.	KL6
31	Customer Lifetime Value Prediction: Develop a model to predict the lifetime value of customers for a business based on their past behavior.	KL6

32	Video Activity Recognition: Extend human activity recognition to videos, identifying activities performed over a sequence of frames.	KL6
33	Automated Essay Scoring: Build a model to automatically score essays based on their content, coherence, and grammar.	KL6
34	Fraud Detection: Develop a model to detect fraudulent transactions in banking or e-commerce systems.	KL6
35	Music Genre Classification: Create a model to classify music into different genres based on audio features.	KL6
36	Topic Modeling: Use techniques like Latent Dirichlet Allocation (LDA) to identify topics within a collection of documents.	KL6
37	Object Tracking: Implement algorithms to track objects in videos across frames, useful for surveillance or self-driving cars.	KL6
38	Satellite Image Analysis: Analyze satellite imagery for tasks like land cover classification, urban development monitoring, or disaster response.	KL6
39	Hand Gesture Recognition: Extend gesture recognition to recognize hand gestures in real-time for controlling devices or applications.	KL6
40	E-commerce Recommendation: Build a personalized recommendation system for an e-commerce platform based on user browsing and purchase history.	KL6
41	Drug Discovery: Use machine learning to predict the biological activity of molecules and assist in drug discovery processes.	KL6
42	Smart Home Automation: Develop a system to automate tasks in a smart home environment based on sensor data and user preferences.	KL6
43	Facial Attribute Detection: Build a model to detect facial attributes such as age, gender, or facial hair from images.	KL6
44	Speech Emotion Recognition: Develop a model to recognize emotions from speech signals, useful for applications like customer service or mental health monitoring.	KL6
45	Urban Mobility Prediction: Predict traffic flow or transportation demand in urban areas using machine learning techniques.	KL6

BIBLIOGRAPHY

- **Text Book**:
  1. Deep learning (Hard Cover)- Ian Goodfellow, Yoshua Bengio, Aaron
- **Reference Books:**
  1. Dive into Deep Learning - Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola, Open community
  2. Pattern Classification- Richard O. Duda, Peter E. Hart, David G. Stork, John Wiley & Sons Inc.
  3. Deep Learning: A practical Approach, R. Chopra.

- **Webliography**:
  1. https://course.fast.ai/
  2. https://www.youtube.com/watch?v=7PiK4wtfvbA&list=PLBAGcD3siRDguyYYzhVwZ3tLvOyyG5k6K

- **SWAYAM/NPTEL/MOOCs Certification:**
  1. Course Name: Deep Learning, By Prof. Prabir Kumar Biswas, IIT Kharagpur. https://nptel.ac.in/courses/

PROBLEM-BASED LEARNING

Exercises in Problem-based Learning (Assignments) (Min 45 Problems*)

SNo	Problem	KL
1	Download the fashion-MNIST dataset and plot 1 sample image for each class. Use "from keras.datasets import fashion_mnist" for getting the fashion mnist dataset.	K3
2	Implement a feedforward neural network which takes images from the fashion-mnist data as input and outputs a probability distribution over the 10 classes.	K6
3	Implement the backpropagation algorithm with support for the following optimisation functions: • sgd • momentum based gradient descent • nesterov accelerated gradient descent • rmsprop • adam	K6
4	Use the train_test_split to divide the dataset and perform the classification.	K3
5	Evaluate the performance of the model using Precision, Recall and F1-score measures.	K5
6	Draw the confusion matrix.	K4
7	In the model above, we would have used cross-entropy loss. Now compare the cross-entropy loss with the squared error loss.	K4
8	Display the comparison Charts for MSE and Cross-Entropy loss.	K3
9	What is Deep Learning? How is it different than Machine Learning?	K2
10	What is Neural Network? Describe the Multi-Layer Perceptron.	K3
.11	What Is Data Normalization, and Why Do We Need It?	K2
12	What is the Boltzmann Machine?	K3
13	What Is the Role of Activation Functions in a Neural Network?	K2
14	What Is the Cost Function?	K2
15	Discuss the Gradient Descent.	K2
16	Discuss and demonstrate the process of Backpropagation?	K2
17	What Is the Difference Between a Feedforward Neural Network and Recurrent Neural Network?	K3
18	What Are the Applications of a Recurrent Neural Network (RNN)?	K3
19	Describe and compare the Softmax and ReLU Functions?	K3
20	What Are Hyperparameters tuned in the development of a deep neural network?	K3
21	What Will Happen If the Learning Rate Is Set Too Low or Too High?	K3
22	Explain the Dropout and Batch Normalization with example?	K3
23	What Is the Difference Between Batch Gradient Descent and Stochastic Gradient Descent?	K3

24	Discuss Overfitting and Underfitting, and How to Combat Them?	K3
25	Demonstrate how Are Weights Initialized in a Network?	K4
26	Explain the Different Layers on CNN?	K3
27	Describe is Pooling on CNN, and How Does It Work?	K4
28	How Does an LSTM Network Work?	K4
29	What are Vanishing and Exploding Gradients? Explain with example.	K3
30	What is the Difference Between Epoch, Batch, and Iteration in Deep Learning?	K2
31	What do you mean by a tensor? What Are the Programming Elements in Tensorflow?	K3
32	What Is an Auto-encoder?	K3
33	Compare and contrast Bagging and Boosting with example?	K4
34	What do you understand by transfer learning? Name a few commonly used transfer learning models.	K4
35	Discuss the difference between SAME and VALID padding in Tensorflow?	K3
36	What are some of the uses of Autoencoders in Deep Learning?	K3
37	What are the reasons for mini-batch gradient being so useful?	K3
38	What do you understand by Leaky ReLU activation function?	K3
39	What is Data Augmentation in Deep Learning?	K3
40	Explain the Adam optimization algorithm.	K3
41	Why is a convolutional neural network preferred over a dense neural network for an image classification task?	K4
42	Which of the following strategy does not prevent a model from over-fitting to the training data? 1. Dropout 2. Pooling 3. Data augmentation 4. Early stopping	K5
43	Explain two ways to deal with the vanishing gradient problem in a deep neural network.	K4
44	Why is a deep neural network better than a shallow neural network?	K5
45	What is the need to add randomness in the weight initialization process?	K4
46	How can you train hyperparameters in a neural network?	K4