DISCIPLINE SPECIFIC CORE COURSE – DSC-13: Machine Learning

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
Machine Learning	4	3	0	1	Class XII	DSC-01

Learning Objectives:

1. *To understand the basic theory underlying machine learning.*
2. *To be able to formulate machine learning problems corresponding to different applications.*
3. *To understand a range of machine learning algorithms along with their strengths and weaknesses.*
4. *To be able to apply machine learning algorithms to solve problems of moderate complexity.*
5. *To apply the algorithms to a real-world problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.*

Learning Outcomes:

1. *Differentiate between supervised and unsupervised learning tasks.*
2. *Appreciate the need of preprocessing, feature scaling and feature selection.*
3. *Understand the fundamentals of classification, regression and clustering*
4. *Implement various machine learning algorithms learnt in the course.*

Unit I **(9 Hours)**

Introduction: Basic definitions and concepts, key elements, supervised and unsupervised learning, introduction to reinforcement learning, applications of ML.

Unit II **(9 Hours)**

Preprocessing: Feature scaling, feature selection methods. Dimensionality reduction (Principal Component Analysis).

Unit III **(9 Hours)**

Regression: Linear regression with one variable, linear regression with multiple variables,gradient descent, over-fitting, regularization. Regression evaluation metrics.

Unit IV **(9 Hours)**

Classification: Decision trees, Naive Bayes classifier, logistic regression, k-nearest neighbor classifier, perceptron, multilayer perceptron, neural networks, back-propagation algorithm, Support Vector Machine (SVM). Classification evaluation metrics.

Unit V **(9 Hours)**

Clustering: Approaches for clustering, distance metrics, K-means clustering, hierarchical clustering.

References

1. Mitchell, T.M. Machine Learning, McGraw Hill Education, 2017.
2. James, G., Witten. D., Hastie. T., Tibshirani., R. An Introduction to Statistical Learning with Applications in R, Springer, 2014.
3. Alpaydin, E. Introduction to Machine Learning, MIT press, 2009.

Practical List: (30 Hours)

Use Python for practical labs for Machine Learning. Utilize publically available datasets from online repositories like https://data.gov.in/ and
https://archive.ics.uci.edu/ml/datasets.php
For evaluation of the regression/classification models, perform experiments as follows:

- Scale/Normalize the data
- Reduce dimension of the data with different feature selection techniques
- Split datasets into training and test sets and evaluate the decision models
- Perform k-cross-validation on datasets for evaluation

Report the efficacy of the machine learning models as follows:

- MSE and R2 score for regression models
- Accuracy, TP, TN, FP, TN, error, Recall, Specificity, F1-score, AUC for classification models.

DISCIPLINE SPECIFIC CORE COURSE – DSC 14: FULL STACK WEB DEVELOPMENT -2

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
FULL STACK WEB DEVELOPMENT -2	4	3	0	1	Class XII	DSC-11

***Learning objectives*:**

1. *Assimilate and master latest framework like frameworks like js, Node.js, and Mongo DB.*

***Learning outcomes*:**

1. *Able to use basic to advanced Node js.*
2. *Integrate Node js with mongo database*
3. *Install and use different tools like Github, Maven and Jenkins.*
4. *Develop a fully functioning website and deploy on a web server.*

UNIT I **(5 hours)**

Introduction to Node JS: What is Node.js, Why Node.js, Node in-built packages (buffer, fs, http, os, path, util, url), Node.js Modules, Import your own Package, Node Package Manager (NPM), Local and Global Packages, File System: Get Input from Users, Pass Multiple Arguments with Yargs, File System Module.

UNIT II **(10 hours)**

Advanced Node JS : Express Framework, Run a Web Server using Express Framework, Routes, Deploy application using PM2 and Nginx, Asynchronous Programming- Call Stack, Callbacks, Callback Queue and Event Loop , Callback Abstraction , Callback Chaining

UNIT III **(10 hours)**

Integration of Node.js with Mongo DB: Inserting Documents, Querying, Updating and Deleting Documents, Connect Mongo DB and Node.js Application, REST API

UNIT IV **(10 hours)**

Overview of Git, Jenkins and Maven: Git- Understand the differences between Git, Github and Gitlab, Install and configure Git for use, Use Git to manage files using CLI commands, Create, Clone and manage repositories.

Jenkins- Jenkins and its architecture, Jenkins tools management, user management in Jenkins

Maven - Maven project structure, maven plugins, Project object model (POM), maven build lifecycle, adding external dependencies to maven pom.xml, maven build and test project

UNIT V **(10 hours)**

Introduction to Docker: Comparing VM and Docker, Docker- an Architectural overview, The Docker Hub A brief Introduction, Preparing docker - machine- Installation and configuration, Start containerizing, Play with docker images, Customizing container on your own, Running Container with Docker - commands, Port forwarding with docker container.

References:

1. Node.js, Mongo DB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications (Developer's Library) - by Brad Dayley , Addison-Wesley; 2nd edition

2. DevOps. Building CI/CD Pipelines with Jenkins, Docker Container, AWS ECS, JDK 11, Git and Maven by John Edward Cooper Berg , Kindle Edition

List of Practicals **(30 hours)**

A web development project implementing technologies such as Node JS, Mongo DB, Angular JS, JQuery, JavaScript, Git, Jenkins and Maven.

DISCIPLINE SPECIFIC CORE COURSE – DSC 15: Minor Project-1

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
Minor Project-1	4	0	0	4	Class XII	NIL

Learning Objectives:

The students will undergo one semester of project work based on the concepts studied in a subject of their choice. The objective is to train the students for the industry by exposing them to prototype development of real life software.

Learning Outcomes:

On successful completion of this course, a student will be able to:

1. *Develop a project plan based on informal description of the project.*
2. *Implement the project as a team.*
3. *Write a report on the project work carried out by the team and defend the work done by the team collectively.*
4. *Present the work done by the team to the evaluation committee.*

Each student shall carry out a minor project in the fifth semester. The students will work on any project based on the concepts studied in core/elective/ skill based elective courses. Specifically, the project could be a research study, or a software development project.

In case the student is opting for research project, students are required to select a relevant topic, carryout a detailed literature review followed by a critical analysis or implementation. The conclusions drawn from the analysis/ implementation must also be brought out in the form of a research paper.

PROJECT GROUP ORGANIZATION/PLAN

- Students will initially prepare a synopsis (500 words) and submit it to their respective department/supervisor. Only after obtaining the approval of supervisor the student can initiate the Project work.
- For a given project, the group size could be a maximum of four (04) students.
- Each group will be assigned a teacher as a supervisor who will be responsible for their lab classes.
- A maximum of four (04) projects would be assigned to one teacher.

PROJECT EVALUATION

The project will be evaluated as follows:

(a) Mid-semester evaluation 25% weightage

(b) End-semester evaluation

(i) External Examination 50% weightage

Thesis/Project report - 25% of total marks.

Software Coding

i) Documentation - 10% of total marks.

ii) Software - 15% of total marks.

(ii) Viva-voce **25% weightage**

- Practical/discussion sessions based on the area of the project. Work carried out in each lab session will be assessed out of five marks (zero for being absent). Finally, the marks obtained will be scaled out of a maximum marks of mid-semester evaluation (i.e. 25% of total marks).
- The end-**semester evaluation marks** to be awarded jointly by the examiner and supervisor / mentor.
- The **Mid-semester evaluation** to be awarded by the supervisor/mentor. Work carried out in each lab session will be assessed.
- The students will submit both the soft copy and the hard copy of the report.
- The reports may be retained by the examiners.

PROJECT REPORT

Two copies of the Project Report certified by the supervisor shall be submitted to the Department. The format of report can be downloaded from the website/guide/ coordinator.