:

KRISHNA UNIVERSITY

BOARD OF STUDIES (BoS) – DEPARTMENT OF COMPUTER SCIENCE

MINUTES OF THE BOARD OF STUDIES MEETING

Agenda:

Framing and Approval of Curriculum and Syllabus for I, II, III and IV Semesters of Diploma Programmes

- Diploma in Computer Applications
- Diploma in Artificial Intelligence and Robotics
- Diploma in Artificial Intelligence and Machine Learning
- Diploma in Cybercrime and Cyber Security
- Diploma in Data Analytics

Members Present:

S. No.	Name	Designation	Role
1	Prof. R. Kiran Kumar	Dept. of Computer Science, Krishna University	Chairman
2	Smt. P. Durga Bhavani	HOD, Dept. of Computer Science, Nalanda Degree College, Vijayawada	Member
3	Kondapalli Ramkumar	HOD, Dept. of Electronics and Internet of Things, Nalanda Degree College, Vijayawada	Member
4	Bommareddy Sudheer Reddy	Lecturer, Dept. of Computer Science, Nalanda Degree College & PG Centre, Vijayawada	Member
5	Prof. P. Suresh Varma	Dept. of CSE, Adikavi Nannaya University, Rajahmundry	Member
6	Dr. M.H.M. Krishna Prasad	Dept. of CSE, JNTUK, Kakinada	Member
7	Prof. D. Lalitha Bhaskari	Dept. of CS & Systems Engineering, Andhra University	Member
8	Assoc. Prof. M. Babu Reddy	HOD, Dept. of Computer Science, Krishna University	Member

Proceedings

The meeting commenced with the Chairman welcoming all the members. The proposal for introducing the above **Diploma programmes** was placed before the Board for detailed discussion.

The members discussed the proposed **curriculum, syllabus course structure, credits, teaching scheme, practical components, Course Outcomes, and assessment pattern** of the five Diploma programmes.

The Board members emphasized the importance of providing students with **practical, skill-oriented, and industry-relevant knowledge** in emerging areas of Computer Applications, Artificial Intelligence, Robotics, Machine Learning, Cyber Security, Cybercrime, and Data Analytics.

The members also suggested incorporating adequate **laboratory work, hands-on training, project-based learning, and application-oriented components** to enhance the employability and technical skills of students.

Resolution

After detailed discussion, the **Board of Studies approved/recommended the introduction of the following Diploma programmes**:

1. **Diploma in Computer Applications**
2. **Diploma in Artificial Intelligence and Robotics**
3. **Diploma in Artificial Intelligence and Machine Learning**
4. **Diploma in Cybercrime and Cyber Security**
5. **Diploma in Data Analytics**

The Board recommended that the proposed **curriculum and syllabus , course structure, credits, Course Outcomes, teaching scheme, practical components, and assessment pattern** be finalized by incorporating the suggestions given by the members.

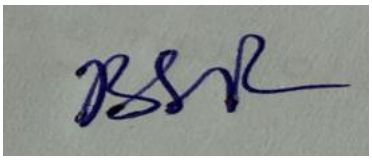
The Board further recommended that the finalized curriculum and syllabi be submitted to the **competent academic authority of Krishna University** for approval and implementation.

Conclusion

The Chairman thanked all the members for their valuable suggestions and active participation in the discussion. The meeting concluded with a vote of thanks.

Chairman, Board of Studies
Prof. R. Kiran Kumar
Signature: _____











Model Syllabus for Diploma in Artificial Intelligence and Machine Learning

COURSE STRUCTURE (for Semester I to IV)

Year	Semester	Course	Title of the Course	No. of Hrs /Week	No. of Credits
I	I	1	Mathematical Foundation for AI	4	4
	II	2	Python Programming and Data Structures	3	3
			Python Programming and Data Structures practical	2	1
II	III	3	Artificial & Computational Intelligence	3	1
			Artificial & Computational Intelligence Practical	2	1
	IV	4	Introduction Machine Learning	3	3
			Introduction Machine Learning Lab	2	1

SEMESTER-I

COURSE 1 : MATHEMATICAL FOUNDATION FOR AI

Theory **Credits: 4** **4 hrs/week**

Course Objectives

1. Develop a strong foundation in linear algebra, set theory, and functions essential for AI.
2. Understand and solve systems of linear equations using matrix methods.
3. Gain knowledge of eigenvalues, eigenvectors, and matrix diagonalization.
4. Learn fundamental concepts of probability and statistics for data analysis.
5. Explore functions and their applications relevant to AI problem-solving.

Course Outcomes

1. Solve complex linear algebra problems including matrix properties.
2. Apply set theory rigorously and compute eigenvalues/eigenvectors for intermediate examples.
3. Understand differentiation rules and solve constrained optimization problems.
4. Calculate probabilities in varied scenarios and explore discrete random variables.
5. Analyze data using comprehensive statistical measures and interpret visualizations.

Unit 1: Basic Linear Algebra and Systems of Linear Equations

Vectors and matrices: basics and operations (addition, multiplication, transpose, inverse)
Elementary row operations: row swapping, scalar multiplication, row addition
Row Echelon Form (REF), Reduced Row Echelon Form (RREF), Rank of matrix
System of linear equations: coefficient and augmented matrix representation
Types of solutions: unique, infinite, no solution
Gaussian elimination method using REF and back substitution

Unit 2: Set Theory and Eigen Concepts

Sets, subsets, set operations (union, intersection, difference, complement), Venn diagrams, Cartesian products
Relations and functions: definitions and properties, linear transformations, Eigenvalues, eigenvectors, characteristic polynomial, Diagonalization of matrices and symmetric matrices

Unit 3: Functions and their Properties

Definition, types of functions (polynomial, rational, exponential, logarithmic), Domain, range, and inverses of functions, Composition of functions, Continuity and basic limits, Graphical representation of functions, Maxima & Minima of functions.

Unit 4: Vector Differentiation

Vector differentiation –ordinary – derivatives of vectors – Differentiability – Gradient – Divergence - Curl operators - Directional derivatives of functions

Unit 5: Fundamentals of Probability & Basic Statistics

Probability: Concept of Uncertainty, Axioms and rules of probability, Conditional probability and independence, Law of total probability and Bayes' theorem

Measures of central tendency: Mean, Median, Mode

Measures of dispersion: range, interquartile range, variance, standard deviation

Introduction to correlation and covariance

Data representation: histograms, bar charts, scatter plots

Textbooks and References

1. Mathematics for Machine Learning, M. P. Deisenroth, A. A. Faisal, C. S. Ong, Cambridge University Press, 2020.
2. Introductory Linear Algebra, Howard Anton, Wiley.
3. Probability and Statistics for Engineers and Scientists, Ronald E. Walpole, Wiley.
4. Discrete Mathematics and its Applications, Kenneth H. Rosen, McGraw Hill.
5. Online Resources: Khan Academy, MIT OpenCourseWare (Linear Algebra, Probability, Statistics, Functions).

Activities:

Unit 1

Activity: Solve advanced linear equation systems using elementary row operations; explore matrix rank with concrete examples; interpret solutions graphically.

Evaluation Method: Assess problem-solving accuracy, clarity of solution process, and ability to classify solution types.

Unit 2

Activity: Practice set theory problems including Venn diagrams, unions, intersections; compute eigenvalues/eigenvectors for 3x3 matrices; perform matrix diagonalization exercises.

Evaluation Method: Evaluate completeness of set operations, correctness of eigen computations, and accuracy in diagonalization.

Unit 3

Activity: Plot and analyze various types of functions (polynomial, exponential, logarithmic); solve problems on function composition and inverses; perform simple graphical interpretations.

Evaluation Method: Assess quality of function plots, conceptual clarity of compositions and inverses, and accuracy of graphical analyses.

Unit 4

Activity: Calculate conditional probabilities; simulate discrete probability distributions; apply Bayes' theorem in practical scenarios (e.g., medical testing, reliability analysis).

Evaluation Method: Evaluate correct application of probability laws and rules, and logical use of Bayesian inference.

Unit 5

Activity: Analyze sample datasets to calculate central tendency measures and dispersion; compute correlation coefficients; create histograms and scatter plots; interpret data insights.

Evaluation Method: Assess accuracy of statistical calculations, clarity and correctness of visual data representation, and quality of interpretation.

SEMESTER-II

COURSE 2 : PYTHON PROGRAMMING AND DATA STRUCTURES

Theory **Credits: 3** **3 hrs/week**

Course Objectives

1. To introduce the fundamentals of Python programming, including environment setup, syntax, and core concepts.
2. To develop problem-solving skills using control flow, functions, and modules.
3. To provide knowledge of Python data structures, file handling, and exception handling for effective programming.
4. To impart object-oriented programming concepts and GUI development skills for building applications.

Course Outcomes (COs)

After successful completion of the course, students will be able to:

1. Explain the basic features, syntax, data types, and operators of Python programming.
2. Apply control flow constructs, functions, and modules to develop structured Python programs.
3. Demonstrate the use of sequences, sets, and dictionaries for effective data handling and manipulation.
4. Implement file handling techniques and apply exception handling mechanisms for robust applications.
5. Develop object-oriented and GUI-based applications using Python.

Unit 1: Basics of Python Programming:

Introduction to Python, Features of Python, Programming Modes - Interactive Mode & Script Mode, Identifiers, Naming Conventions, Keywords (Reserved Words), Built-in Data Types, Literals - Integer, Float, Complex, Boolean, String, Variables, Operators, Expressions, Assignment Statements, Input/Output Statements, Python Syntax (Lines, Comments, Indentation),

Operators & Operands, Classification of Operators - Arithmetic Operators, Relational Operators, Logical Operators, Bitwise Operators, Assignment, Augmented Assignment, Identity Operators, Expressions & Precedence Rules

Unit 2: Control Flow, Functions & Modules:

Control Flow - if Statement, if-else, if-elif-else. Iterative Statements – while, for, Nested Loops, Loop Control Statements – break, continue, pass; else with loops

Need for Functions, Defining & Invoking User-defined Functions, Return Statement, Function Input/Output Cases, Scope of Variables - Local, Global, Nested Functions, Function Arguments - Required, Positional, Default, Variable-length, main() Function, Documentation Strings, Recursive Functions, Anonymous Functions (Lambda), Library Functions

Modules - Import, from..import, Creating & Using Modules, Namespaces

Unit 3: Core Data Types and Python Collections

Strings: Representation, Indexing, Slicing, Immutability, Operators, Methods, Formatting

Lists: Creation, Indexing, Slicing, Mutability, Common Methods, List Comprehension

Tuples: Immutability, Operations, Tuple Assignment

Sets and Frozensets: Methods, Mathematical Operations, Comprehension

Dictionaries: Key-Value Structure, Methods, Traversal, Nested Dictionaries

Unit 4: File Handling, Exception Management & Object-Oriented Programming

File Handling: Types, Opening, Reading, Writing, Closing, CSV Files, OS/Pathlib

Error Types, Exception Handling: try-except, raise, User-defined Exceptions, Assertions

OOP Concepts: Classes, Objects, Attributes, Methods, Constructor and Destructors

Encapsulation: Private and Public Members

Inheritance: Single, Multilevel, Multiple, Method Overriding

Unit 5: Abstract Data Structures and GUI Programming

Abstract Data Structures (ADTs): Concepts and Importance

Linked Lists: Definition, Types- Singly, Doubly, Circular; Node Structure, Insertion, Deletion, Traversal (Single Linked list implementation only)

Stacks: LIFO Principle, Implementation using List, Applications

Queues: FIFO Principle, Implementation using List, Priority Queues

GUI Programming with Tkinter: Widgets (Label, Button, Entry, Menu, Listbox, Canvas etc.), Event Handling, Building Simple GUI Apps

Textbooks:

1. Python Programming-An Object Oriented approach, Anita Goel, Universities Press
2. Python Programming using Problem Solving Approach Reema Thareja Oxford University Press 2020
3. Exploring Python, Budd T A, McGraw-Hill Education, 1st Edition, 2011.

Reference Book:

1. Python: The Complete Reference, Martin C. Brown, Mc Graw-Hill, 2018
2. Fundamentals of Python, Kenneth A. Lambert. (2019), First Programs,2nd Edition, CENGAGE Publication.

Activities:

Outcome: Explain the basic features, syntax, data types, and operators of Python programming.

Activity: Conduct a "Python Basics Lab" where students write small programs to demonstrate literals, variables, data types, and operators (e.g., swapping numbers, simple calculator).

Evaluation Method:

- Lab performance checklist (execution of 3 mini tasks)
- Short quiz with multiple-choice and fill-in-the-blanks on syntax, data types, and operators

Outcome: Apply control flow constructs, functions, and modules to develop structured Python programs.

Activity: Group activity - "Python Problem Solving Challenge": Students solve real-life problems (e.g., finding prime numbers, grade calculator, menu-driven calculator) using control structures, functions, and importing standard modules.

Evaluation Method:

- Code submission with proper use of functions/modules (20%)
- Viva-voce to explain logic and flow of control (40%)
- Unit test with scenario-based programming questions (40%)

Outcome: Demonstrate the use of sequences, sets, and dictionaries for effective data handling and manipulation.

Activity: Hands-on mini project – "Student Data Manager": Students create a program using lists, tuples, sets, and dictionaries to store and manipulate student records (e.g., marks, courses, hobbies).

Evaluation Method:

- Practical demo of program with at least 5 data operations (add, search, delete, update, traverse)
- Evaluation rubric for correctness, efficiency, and use of appropriate data structure

Outcome: Implement file handling techniques and apply exception handling mechanisms for robust applications.

Activity: Individual assignment - "File-Based Address Book": Students create a program to store, update, and retrieve data from files, with exception handling for invalid inputs or missing files.

Evaluation Method:

- Assessment of program correctness (file read/write, append, delete, exception handling)
- Short quiz with error-tracing and debugging questions (given code with errors, students identify and correct)

Outcome: Develop object-oriented and GUI-based applications using Python.

Activity: Mini Project – "Student Information System with GUI": Students design a simple Tkinter-based application with classes/objects for handling student data, including basic GUI widgets (Entry, Button, Listbox).

Evaluation Method:

- Project demo and presentation (50%)
- Rubric-based evaluation for OOP concepts (classes, inheritance, encapsulation) and GUI design (widgets, event handling) (30%)
- Peer review/feedback on usability (20%)

SEMESTER-II

COURSE 2: PYTHON PROGRAMMING AND DATA STRUCTURES

Practical **Credits: 1** **2 hrs/week**

1. Basic Python Programs:
    a. Write a program to display basic details (name, roll number, department) using print() and demonstrate different literal types (int, float, string, boolean, complex).
    b. Write a program to perform arithmetic, relational, logical, bitwise, and assignment operations on given inputs.
2. Control Flow Practice
    a. Write a program to find the largest of three numbers using if-elif-else.
    b. Write a program to check whether a number is prime or not using loops.
    c. Write a program to illustrate the use of loop control statements (break, continue, pass).
3. Functions and Recursion
    a. Write a program to define a function to calculate factorial of a number (using recursion).
    b. Write a program to demonstrate different types of function arguments (default, positional, keyword, variable-length).
4. Write a program to illustrate string slicing, concatenation, repetition, and built-in methods.
5. Write a program to create a list of numbers, perform insertion, deletion, searching, sorting, and list comprehension.
6. Write a program to demonstrate tuple packing, unpacking, and immutability.
7. Write a program to implement set operations (union, intersection, difference, subset, superset).
8. Write a program to create a dictionary of student roll numbers and marks, and perform add, update, delete, and traversal operations.
9. Write a program to read and display count of vowels, consonants, digits, and spaces of a text file.
10. Write a program to copy the contents of one file into another file.
11. Write a program to read and process student marks from a CSV file (calculate average, highest, lowest).

12. Write a program to demonstrate exception handling using try-except-finally.
13. Write a program to create a class Student with attributes and methods to display details.
14. Write a program to demonstrate single and multilevel inheritance.
15. Implement stack (LIFO) and queue (FIFO) using lists.
16. Implement singly linked lists: node creation, insertion, deletion, traversal.
17. Write a Tkinter program with Label, Entry, and Button widgets to take user input and display it.
18. Write a Tkinter program to create a simple calculator application.

SEMESTER-III

COURSE 3: ARTIFICIAL & COMPUTATIONAL INTELLIGENCE

Theory **Credits: 3** **3 hrs/week**

Course Objectives

- Introduce foundational concepts and history of Artificial Intelligence (AI).
- Teach the PEAS framework and agent-based problem solving.
- Develop understanding of uninformed and informed search methods.
- Provide basic knowledge of machine learning concepts and computational intelligence.
- Instill awareness of ethical considerations in AI.
- Introduce logic programming fundamentals using Prolog as a practical tool.

Course Outcomes

Students completing this course will be able to:

- Explain basic AI terminology, agent models, and the PEAS framework.
- Formulate AI problems using search strategies and implement basic algorithms conceptually.
- Understand core machine learning categories and their applications.
- Describe computational intelligence approaches and their role in AI.
- Analyze ethical issues related to AI technologies.
- Write and execute simple Prolog programs demonstrating logic programming fundamentals.

Unit 1: Introduction to Artificial Intelligence and PEAS Framework

Introduction to AI: Definition, history, applications, and scope

The PEAS framework: Performance Measure, Environment, Actuators, Sensors, Examples of PEAS in real-world AI systems

Intelligent agents: Intelligent agents and their environments, **Types of intelligent agents:** Simple reflex, model-based, goal-based, utility-based, Agent architectures and rationality

Unit 2: Expert Systems

Definition and components of Expert Systems (Knowledge Base, Inference Engine, User Interface), Rule-based systems and knowledge representation, Examples of expert systems: medical diagnosis, decision support, Limitations and comparison with AI agents, Role and significance of expert systems in AI evolution

Unit 3: Search Strategies in AI

Problem-solving as search: problem formulation, states, actions, goal test, Traveller's problem

Uninformed (Blind) Search: Breadth-first search, Depth-first search, Uniform-cost search

Informed (Heuristic) Search: Greedy best-first search, A* algorithm

Applications of search in AI problems

Unit 4: Introduction to Machine Learning

What is machine learning, definitions, Types of learning: Supervised, Unsupervised, Reinforcement learning (basic ideas), classification, Regression , clustering and Association, Basic learning algorithms overview and applications

Unit 5: Computational Intelligence and Ethics in AI

Overview of computational intelligence (Basics of fuzzy logic, neural networks), Role of computational intelligence in AI, Ethics and societal challenges in AI, Responsible AI, fairness, transparency, and safety concerns

Unit 1: Introduction to AI, PEAS Framework & Intelligent Agents

Activities:

Interactive lecture with real-world AI examples

Group discussion on PEAS framework for different environments

Case study analysis of intelligent agents in everyday AI applications

Quiz on AI fundamentals and agent types

Outcomes:

Students will explain AI basics and describe the PEAS components.

Students will identify types of intelligent agents and their roles.

Evaluation Method:

Quiz and participation: 10%

Assignment on PEAS and agent modeling: 10%

Unit 2: Expert Systems

Activities:

Lecture with multimedia explaining components of expert systems

Hands-on group activity designing rule-based systems for simple decision problems

Case study review of medical diagnosis expert systems

Class debate on limitations and advantages of expert systems

Outcomes:

Understand expert system architecture and knowledge representation.

Develop simple rule-based systems for decision making.

Analyze real-world expert system examples critically.

Evaluation Method:

Group assignment designing rule base: 15%

Written test on expert system concepts: 10%

Unit 3: Search Strategies in AI

Activities:

Demonstration of uninformed search algorithms

Interactive exercises formulating search problems

Simulation of heuristic search and A* algorithm

Problem-solving sessions applying search to puzzles

Outcomes:

Formulate search problems and apply uninformed and heuristic algorithms conceptually.

Explain the working and use cases of different search strategies.

Evaluation Method:

Problem formulation assignment: 10%

Class test on search algorithms: 10%

Unit 4: Basics of Machine Learning

Activities:

Video lectures on different machine learning paradigms

Simple data classification exercises/classification demos

Group presentations on various ML types and applications

Quiz on ML types and terminology

Outcomes:

Explain supervised, unsupervised, and reinforcement learning basics.

Identify sample use cases for machine learning.

Evaluation Method:

Quiz and presentation: 10%

Written assignment on ML overview: 5%

Unit 5: Computational Intelligence & Ethics

Activities:

Lecture introducing fuzzy logic, neural networks overview

Ethical dilemma discussions and case studies in AI technology

Role play on AI responsibility and fairness

Group project: Develop guidelines for responsible AI use

Outcomes:

Understand computational intelligence basics and ethical AI challenges.

Propose responsible AI practices for diverse scenarios**.**

Evaluation Method:

Participation in ethics discussions: 5%

Group project report and presentation: 15%

Recommended Textbooks and References

- Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition
- Elaine Rich, Kevin Knight, Artificial Intelligence, 3rd Edition
- Michael Negnevitsky, Artificial Intelligence: A Guide to Intelligent Systems
- Ivan Bratko, Prolog Programming for Artificial Intelligence, 4th Edition
- Online resources: AI course materials from Coursera, NPTEL, GeeksforGeeks AI tutorials.

SEMESTER-III

COURSE 3: ARTIFICIAL & COMPUTATIONAL INTELLIGENCE

Practical **Credits: 1** **2 hrs/week**

***Note**: Experiments have to be conducted using Prolog*

1. Introduction to Prolog Environment and Syntax
   - Setting up Prolog, understanding facts, rules, and queries.
2. Defining Simple Facts and Queries
   - Write and test simple facts like family relationships, likes/dislikes.
3. Creating Rules in Prolog
   - Define logical rules with conditions and test queries.
4. List Handling in Prolog
   - Write programs to manipulate lists (head, tail, concatenation).
5. Recursion in Prolog
   - Implement recursive relations such as factorial and Fibonacci.
6. Search and Backtracking
   - Demonstrate Prolog's backtracking with sample queries and control cuts.
7. Family Relationship Programs
   - Model family trees and query relationships like siblings, ancestors.
8. Solve the Eight Queens Problem
   - Classic AI problem solved using backtracking.
9. Implement Simple Arithmetic Operations
   - Addition, subtraction, multiplication using Prolog predicates.
10. Monkey and Banana Problem
    - Logic problem modeling and solution.
11. Basic Expert System Prototype
    - Write rules for a simple medical diagnosis or recommendation system.
12. Implement Search Algorithms (Conceptual)
    - Demonstrate basic search algorithms like best-first search using Prolog rules.

SEMESTER-IV

COURSE 4 : INTRODUCTION TO MACHINE LEARNING

Theory **Credits: 3** **3 hrs/week**

Unit 1. Introduction to Machine Learning:

Introduction to Machine Learning: Types of human learning, What is machine learning?, Types of machine learning: supervised, unsupervised, semi-supervised and reinforcement learning, machine learning activities, applications of machine learning. Types of data in machine learning, Structure of data

Unit 2. Model Preparation, Evaluation and feature engineering:

Data pre-processing, Model selection and training (for supervised learning), Model representation and interpretability, Evaluating machine learning algorithms and performance enhancement of models. What is feature engineering?, Feature transformation, Feature subset selection. Principal component analysis.

Unit 3. Supervised Learning:

Classification: Classification model and learning steps, Classification algorithms: Naïve Bayes classifier, k-Nearest Neighbour (kNN)

Regression: Introduction of regression, Regression algorithms: Simple linear regression, Multiple linear regression, Polynomial regression model, Logistic regression

.**Unit 4. Unsupervised Learning:**

Introduction of unsupervised learning, Unsupervised vs supervised learning, Application of unsupervised learning, Clustering and its types, Partitioning method: k-Means and KMedoids, Hierarchical clustering

Unit 5.Neural Networks:

Introduction to neural networks, Structure of a neural network - Input layer, hidden layer, and output layer, types of neural networks, Weights and bias, Activation functions, Step function, Sigmoid function, ReLU, Applications of neural networks.

Text Books:

- Introduction to Machine Learning with Python, Andreas C. Müller and Sarah Guido
- Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurélien Géron
- Machine Learning, Tom M. Mitchell
- Pattern Recognition and Machine Learning, Christopher M. Bishop
- Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville

COURSE 4 : INTRODUCTION TO MACHINE LEARNING

Practical **Credits: 1** **2 hrs/week**

1. Write a python program to import and export data using Pandas library functions.
2. Demonstrate various data pre-processing techniques for a given dataset
3. Implement Dimensionality reduction using the Principal Component Analysis (PCA)
4. method.
5. Write a Python program to demonstrate various Data Visualization Techniques.
6. 5. Implement MLE on a Dataset
7. Implement Simple and Multiple Linear Regression Models.
8. Develop Logistic Regression Model for a given dataset.
9. Develop Decision Tree Classification model for a given dataset and use it to classify a
10. new sample.
11. Implement Naïve Bayes Classification in Python.
12. Develop K-Means for a Given Dataset
13. Build KNN Classification model for a given dataset.

KRISHNA UNIVERSITY

Diploma Model Paper for Core Subjects

Semester: I

Paper Title: Mathematical Foundation for AI

Time: 3hrs **Max.Marks:70**

Section–A(20Marks)

Answer any *Five* questions. All questions carry equal marks. 5X4=20M

1. Explain matrix multiplication with a suitable example.
2. Define REF.
3. Define a function.
4. Explain Venn diagrams and their applications.
5. Explain the graphical representation of functions.
6. Explain the relationship between directional derivative and gradient.
7. Explain the basic rules of probability.
8. What is a scatter plot?

Section–B(50Marks)

Answer ALL Questions. All questions carry equal marks. 5X10 =50M

9. a) Explain vectors and their basic operations with suitable examples.

OR

b) Explain the different types of solutions of a system of linear equations with suitable examples.

10. a) Explain sets and set operations with suitable examples and Venn diagrams.

OR

b) Explain eigenvalues and eigenvectors and describe the procedure for finding them.

11. a) Explain the different types of functions with suitable Example

OR

b) Describe about Maxima & Minima of functions.

12. a) Explain differentiability of vector-valued functions in detail.

OR

c) Discuss about Curl operators

13. a) State and explain the Law of Total Probability and solve a suitable problem.

OR

d) Explain Bayes' Theorem with a suitable numerical example.

KRISHNAUNIVERSITY

Diploma (AI&ML) Examinations Model Paper for Core Subjects

Semester: II

Paper Title: PYTHON PROGRAMMING AND DATA STRUCTURES

Time: 3hrs **Max.Marks:70**

Section–A(20Marks)

Answer any *Five* questions. All questions carry equal marks. **5X4=20M**

1. Built-in Data Types in Python.
2. Explain about Input/Output Statements in Python?
3. Describe Recursive Functions in Python.
4. What are Local, Global Variables?
5. Explain about Dictionaries?
6. Explain about Encapsulation.
7. Define Abstract Data Structures (ADTs)?
8. What is LIFO data structure.

Section–B(50Marks)

Answer ALL Questions. All questions carry equal marks. **5X10 =50M**

9. a) Explain about the Features of Python.

OR

b). Discuss about Python Operators.

10. a) Explain about the Control Statements in Python.

OR

b). Explain about Python Modules.

11. a) Briefly explain about Strings in Python.

OR

b) Write about Lists in Python.

12. a) Explain about Exception Handling.

OR

b) Briefly explain about OOP Concepts.

13. a) Briefly explain about Linked Lists.

OR

b) Explain about Event Handling in GUI Programming.

KRISHNAUNIVERSITY

Diploma (AI&ML) Examinations Model Paper for Core Subjects

Semester: III

Paper Title: ARTIFICIAL & COMPUTATIONAL INTELLIGENCE

Time: 3hrs **Max.Marks:70**

Section–A(20Marks)

Answer any *Five* questions. All questions carry equal marks. **5X4=20M**

1. Applications of AI.
2. What are Intelligence Agents?
3. What are the Examples of Expert Systems?
4. What are Rule-based systems?
5. Explain about Traveler's problem?
6. Explain about Regression.
7. Define Responsible AI.
8. Role of computational intelligence in AI.

Section–B(50Marks)

Answer ALL Questions. All questions carry equal marks. **5X10 =50M**

9. a) Explain about the PEAS framework.

OR

b) Discuss about Types of intelligent agents.

10. a) What is Expert System? Explain the components of Expert Systems

OR

b) Explain about Role and significance of expert systems in AI evolution.

11. a) Briefly explain about Uninformed (Blind) Search methods.

OR

b) Write about Informed (Heuristic) Search.

12. a) What is machine learning? Explain about the types of Machine Learning.

OR

b) What are the Basic Learning Algorithms in ML?

13. a) Overview of computational intelligence.

OR

b) What are the Ethics and societal challenges in AI?

KRISHNAUNIVERSITY

Diploma (AI&ML) Examinations Model Paper for Core Subjects

Semester: IV

Paper Title: INTRODUCTION TO MACHINE LEARNING

Time: 3hrs **Max.Marks:70**

Section–A(20Marks)

Answer any *Five* questions. All questions carry equal marks. **5X4=20M**

1. Applications of machine learning.
2. What is machine learning?
3. Describe Data pre-processing.
4. What is Classification model?
5. Explain about Logistic regression?
6. Explain about Hierarchical clustering.
7. Define ReLU?
8. Applications of neural networks.

Section–B(50Marks)

Answer ALL Questions. All questions carry equal marks. **5X10 =50M**

9. a) Types of machine learning.

OR

b) Discuss about Types of data in machine learning.

10. a) Explain about the data pre-processing.

OR

b) Explain about Feature Engineering.

11. a) Briefly explain about k-Nearest Neighbor (kNN) Algorithm.

OR

b) Write about Regression. Types of Regression Algorithms

12. a) Define unsupervised learning? unsupervised vs. supervised learning.

OR

b) What is Clustering? Explain K-Means Algorithm.

13. a) Briefly explain about Structure of a neural network.

OR

b) Explain about Activation functions in Neural Networks.