Programme Outcomes

PO 1 Domain Expertise:
- Acquire comprehensive knowledge and skills.
- Make use of the knowledge in an innovative manner.
- Effectively apply the knowledge and skills to address various issues.

.

PO 2 Modern equipment Usage
- Use ICT effectively.
- Access, retrieve and use authenticated information.
- Access, retrieve and use authenticated information. Have knowledge of software applications to analyze data.

PO 3 Computing Skills and Ethics
- Develop rationale and scientific thinking process.
- Use technology intelligently for communication, entertainment and for the benefit of mankind.
- Ensure ethical practices throughout ones endeavors for the wellbeing of human race.

PO 4 Complex problem Investigation & Solving
- Predict and analyze problems.
- Frame hypotheses.
- Investigate and interpret empirical data.
- Plan and execute action.

PO 5 Perform effectively as Individuals and in Teams
- Work efficiently as an individual
- Cooperate, coordinate and perform effectively in diverse teams/groups.
- Prioritize common interest to individual interest.

PO 6 Efficient Communication & Life Skills
- Express thoughts in an effective manner
- Listen, understand and project views in a convincing manner.
- Decide appropriate media to share information
- Develop skills to present significant information clearly and concisely to interested groups.

PO 7 Environmental Sustainability
- Understand the Environmental challenges.
- Think critically on environmental sustainability measures.
- Propagate and follow environment friendly practices.

PO 8 Societal contribution
- Render service for the general good of the society.
- Involve voluntarily in social development activities at Regional, National, global levels.
- Have pride in volunteering to address societal issues viz: calamities, disasters, poverty, epidemics.
- Be a patriotic citizen to uphold the values of the nation

PO 9 Effective Project Management
- Identify the goals, objectives and components of a project and decide the appropriate time of completion.
- Plan, organize and direct the endeavors of teams to achieve the set targets in time.
- Be competent in identifying opportunities and develop strategies for contingencies.

<div align="center">

Semester-III
Subject: Data Science
Paper–III: Data Engineering with Python
Course Code :DATA301

</div>

Course Outcome**:**
At the end ofthe course the student will be able to:

- Handle different typesof files and work with text data
- Use regular expressionoperations
- Use relational databases via SQL
- Usetabular numeric data
- Use the datastructures:data series and frames
- Use PyPlot for visualization

<div align="center">

SYLLABUS

</div>

Unit–I
Data Science: Data Analysis Sequence, Data Acquisition Pipeline, Report Structure
Files and Working with Text Data: Types of Files, Creating and Reading Text Data, FileMethods to Read and Write Data, Reading and Writing Binary Files, The Pickle Module,Reading and Writing CSV Files, Python os and os.pathModules. **Working With TextData:** JSONandXML in Python

Unit–II
Working with Text Data: Processing HTML Files, Processing Texts in Natural Languages
Regular Expression Operations: Using Special Characters, Regular Expression Methods,Named Groups in Python Regular Expressions, Regular Expression with *glob* Module

Unit–III
Working with Databases: Setting Up a MySQL Database, Using a MySQL Database:Command Line, Using a MySQL Database, Taming Document Stores: MongoDB
Working with Tabular Numeric Data(Numpy with Python): NumPy Arrays CreationUsing *array()* Function, Array Attributes, NumPy Arrays Creation with Initial PlaceholderContent, Integer Indexing, Array Indexing, Boolean ArrayIndexing, Slicing and Iterating inArrays, Basic Arithmetic Operations on NumPy Arrays, Mathematical Functions in NumPy,Changing the Shape of an Array, Stacking and Splitting of Arrays, Broadcasting in Arrays.

 Unit–IV
WorkingwithDataSeriesandFrames:
PandasDataStructures,ReshapingData,HandlingMissing Data, Combining Data, Ordering and Describing Data, Transforming Data, TamingPandasFileI/O
Plotting: Basic Plotting with PyPlot, Getting to Know Other Plot Types, MasteringEmbellishments,PlottingwithPandas

Practical   **2 Hours/Week**   **1 credit**

DataEngineeringwithPython(Lab)

Libraries
In this course students are expected to extract, transform and load input data that can be textfiles, CSV files, XML files, JSON, HTML files, SQL databases, NoSQL databases etc.,. Fordoingthis, they should learnthe following Pythonlibraries/modules:
pandas,numpy, BeautifulSoup, pymysql, pymongo, nltk, matplotlib

Datasets
For this laboratory, appropriate publicly available datasets, can be studied and used.Example:
MNIST(http://yann.lecun.com/exdb/mnist/),
UCI Machine Learning
Repository(https://archive.ics.uci.edu/ml/datasets.html),Kaggle(https://www.ka ggle.com/datasets)TwitterData

Exercises
1. Write programs to parse text files, CSV, HTML, XML and JSON documents and extractrelevantdata.After retrievingdatacheck anyanomaliesin thedata,missing valuesetc.
2. Writeprogramsforreading andwritingbinaryfiles
3. Write programs for searching, splitting, and replacing strings based on pattern matchingusingregularexpressions
4. Design a relational database for a small application and populate the database. Using SQLdothe CRUD(create, read, update and delete)operations.
5. Create aPython MongoDB client using the Python module pymongo. Using a collectionobject practice functions for inserting, searching, removing, updating, replacing, andaggregatingdocuments, as well as forcreating indexes
6. Write programs to create numpy arrays of different shapes and from different sources,reshape and slice arrays, add array indexes, and apply arithmetic, logic, and aggregationfunctionsto someor all array elements
7. Write programs to use the pandas datastructures: Frames and series as storage containersandforavariety of data-wrangling operations, such as:
   - Single-levelandhierarchicalindexing
   - Handlingmissingdata
   - ArithmeticandBoolean operationsonentirecolumnsandtables
   - Database-typeoperations(suchasmerging and aggregation)

   - Plottingindividualcolumnsandwholetables
   - Readingdatafromfilesandwritingdatato files

Semester – III
Subject: Data Science
Paper- Operating Systems – 1 (SEC-2)
Syllabus
COURSE CODE:300/SEC/E

Course Outcome:
- Understands the use of different process scheduling algorithm and synchronization techniques to avoid deadlock.
- They will learn different memory management techniques like paging, segmentation and demand paging etc.

Unit – I

Introduction: Computer-System Architecture, Computing Environments.
Operating-System Structures: Operating-System Services, User Interface for Operating-System, System Calls, Types of System Calls, Operating System Structure.
Process Management: Process Concept, Process Scheduling, Operations on Processes, Inter process Communication, Examples–Producer-Consumer Problem.
Process Synchronization: Critical-Section Problem, Peterson's Solution, Synchronization, Semaphores, Monitors.

Unit – II

CPU Scheduling: Concepts, Scheduling Criteria, Scheduling Algorithms.
Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

B.Sc. (Data Science) II Year
Semester-IV
Subject: Computer Science
Paper–IV: MachineLearning
Course Code :DATA401

Theory	4 Hours/Week	4 credits
Practical	3 Hours/Week	1 credit

CourseOutcome:

At The end of the course the student will be able to understand

- BasicsofMachine Learning and its limitations
- MachineLearningAlgorithms:supervised,unsupervised,bio-inspired
- ProbabilisticModelingandAssociationRuleMining

SYLLABUS

Unit-I

Introduction: What does it mean to learn, Some canonical Learning Problems, The DecisionTree Model of Learning, Formalizing the Learning Problem, ID3 Algorithm

Limits of Learning: Data Generating Distributions, Inductive Bias, Not Everything islearnable, Underfitting and Overfitting, Separation of training and test Data, Models,parameters and Hyperparameters, Real World Applications of Machine Learning

Geometry and Nearest Neighbors: From Data to Feature Vectors, k-Nearest Neighbors,DecisionBoundaries,k-means Clustering,High Dimensions

Unit-II

The Perceptron: Bio-inspired Learning, The Perceptron Algorithm, GeometricInterpretation, Interpreting Perceptron Weights, Perceptron Convergence and LinearSeparability,Improved Generalization,Limitations of the Perceptron

Practical Issues: Importance of Good Features, Irrelevant and Redundant Features, FeaturePruning and Normalization, Combinatorial Feature Explosion, Evaluating ModelPerformance,CrossValidation,HypothesisTestingandStatisticalSignificance,Debuggin gLearningAlgorithms, Bias Variancetradeoff

Linear Models: The Optimization Framework for Linear Models,Convex Surrogate LossFunctions, Weight Regularization, Optimization and Gradient Descent,Support VectorMachines

Unit-III

Probabilistic Modeling: Classification by Density Estimation, Statistical Estimation,NaïveBayesModels, Prediction

Neural Networks: Bio-inspired Multi-Layer Networks, The Back-propagation Algorithm,Initialization and Convergence of Neural Networks, Beyondtwo layers,Breadth vs Depth,BasisFunctions

UnitIV

Unsupervised Learning: Clustering Introduction, Similarity and Distance Measures,AgglomerativeAlgorithms,DivisiveClustering,MinmumSpanningTree

AssociationRules: Introduction,large Itemsets,AprioriAlgorithm

Practical	**2 Hours/Week**	**1 credit**

MachineLearning(Lab)
<mark>MLToolkits</mark>

Students Are Expected To Learn
1. Scikit-learn(https://scikit-learn.org/) an open source machine learning Python library that supports supervised and unsupervised learning. It also provides various tools for model fitting,data preprocessing, model selection and evaluation, and many other utilities.
2. Weka(http://www.cs.waikato.ac.nz/ml/weka/)isanotherwidely usedMLtoolkit.

<mark>Datasets</mark>

1. The sklearn.datasets package embeds small toy datasets. It includes utilities to load these datasets. It also includes methods to load and fetch popular reference datasets and features some artificial data generators. Students are expected to study and makeuseof these datasets
2. Weka also provides various datasets.

Exercises

8. Write a Python program using Scikit-learn to split the iris dataset into 70% train data and30% test data. Out of a total 150 records, the training set will contain 120 records and thetestset contains 30 of those records. Print both datasets
9. Write a Python program to use sklearn'sDecisionTreeClassifier to build a decision tree for the sklearn's datasets. Implement functions to find the importance of a split (entropy,information gain, gini measure)
10. Write a Python program to implement your own version of the K-means algorithm. Then Apply It to different datasets and evaluate the performance.
11. Design a perceptron classifier to classify handwritten numerical digits (0-9). Implement Using Scikit or Weka.
12. Write a Python program to classify text as spam or not spam using the Naïve BayesClassifier
13. Use WEKA and experiment with the following classifiers: Association Rule Mining(Apriori),Agglomerative And Divisive Clustering

B.Sc II (Data Science)Year Examination
Semester – III
Subject: Data Science
Paper- Operating Systems – 2 (SEC-4)
Syllabus
Course Code :DATA/400/SEC/E

Theory   **2 Hours/Week**   **2 credits**

Course Outcome:
- Understands the different services provided by the Operating System at different levels.
- They learn real life applications of Operating Systems in every field.

Unit – I

Main Memory: Introduction, Swapping, Contiguous Memory Allocation, Segmentation, Paging.
Virtual Memory: Introduction, Demand Paging, Page Replacement, Allocation of Frames, Thrashing.
Mass-Storage Structure: Overview, Disk Scheduling, RAID Structure.
File Systems: File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, Protection

Unit – II

File System Implementation, Directory Implementation, Allocation Methods, Free-Space Management. Recovery, Network File System.
Protection and Security: Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Access Control, Revocation of Access Rights, The Security Problem, Program Threats, System and Network Threats, Cryptography as a Security Tool, User Authentication, Implementing Security Defenses, Firewalling to Protect Systems and Networks, Computer-Security Classifications.