Computer Science Courses for Undergraduate Programme of study with Computer Science discipline Elective

<div style="text-align:center; background-color:#4472C4; padding:10px; border-radius:20px;">

DISCIPLINE SPECIFIC ELECTIVE COURSE: Data Mining - I

</div>

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		
Data Mining - I	4	3	0	1	Pass in Class XII	DSC01 Programming using Python / GE1b Programming with Python

Learning Objectives

This course aims to introduce data mining techniques and their application on real-life datasets. The students will learn to pre-process the dataset and make it ready for application of data mining techniques. The course will focus on three main techniques of data mining i.e. Classification, Clustering and Association Rule Mining. Different algorithms for these techniques will be discussed along with appropriate evaluation metrics to judge the performance of the results delivered.

Learning outcomes

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

1. Pre-process the data for subsequent data mining tasks
2. Apply a suitable classification algorithm to train the classifier and evaluate its performance.
3. Apply appropriate clustering algorithm to cluster the data and evaluate clustering quality
4. Use association rule mining algorithms and generate frequent item-sets and association rules

SYLLABUS OF DSE

Unit 1 (5 hours)

Introduction to Data Mining: Motivation and challenges for data mining, Types of data mining tasks, Applications of data mining, Data measurements, Data quality, Supervised vs. unsupervised techniques

Unit 2 (10 hours)

Data Pre-processing: Data aggregation, sampling, dimensionality reduction, feature subset selection, feature creation, variable transformation.

Unit 3 (10 hours)

Cluster Analysis: Basic concepts of clustering, measure of similarity, types of clusters and clustering methods, K-means algorithm, measures for cluster validation, determine optimal number of clusters

Unit 4 (10 hours)

Association Rule mining: Transaction data-set, frequent itemset, support measure, rule generation, confidence of association rule, Apriori algorithm, Apriori principle

Unit 5 (10 hours)

Classification: Naive Bayes classifier, Nearest Neighbour classifier, decision tree, overfitting, confusion matrix, evaluation metrics and model evaluation.

Essential/recommended readings

1. Tan P.N., Steinbach M, Karpatne A. and Kumar V. *Introduction to Data Mining,*2nd edition, Pearson, 2021.
2. Han J., Kamber M. and Pei J. *Data Mining: Concepts and Techniques*, 3rd edition, 2011, Morgan Kaufmann Publishers.
3. Zaki M. J. and Meira J. Jr. *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*, 2nd edition, Cambridge University Press, 2020.

Additional References

1. Aggarwal C. C. *Data Mining: The Textbook*, Springer, 2015.
2. Dunham M. *Data Mining: Introductory and Advanced Topics,* 1st edition, Pearson Education India, 2006.

Suggested Practical List (If any): (30 Hours)

Practical exercises such as

All topics covered in theory will be implemented using Python. The operations may be performed on the datasets loaded through scikit, seaborn libraries or can be downloaded from from Open Data Portal (https:// data.gov.in/, UCI repository http://archive.ics.uci.edu/ml/).

Recommended Datasets for :

Classification: Abalone, Artificial Characters, Breast Cancer Wisconsin (Diagnostic)

Clustering: Grammatical Facial Expressions, HTRU2, Perfume data

Association Rule Mining: MovieLens, Titanics

Additional Suggested Practicals List

1. Apply data cleaning techniques on any dataset (e,g, wine dataset). Techniques may include handling missing values, outliers, inconsistent values. A set of validation rules can be prepared based on the dataset and validations can be performed.
2. Apply data pre-processing techniques such as standardization / normalization, transformation, aggregation, discretization/binarization, sampling etc. on any dataset
3. Run Apriori algorithm to find frequent itemsets and association rules on 2 real datasets and use appropriate evaluation measures to compute correctness of obtained patterns
   a) Use minimum support as 50% and minimum confidence as 75%
   b) Use minimum support as 60% and minimum confidence as 60 %

4. Use Naive bayes, K-nearest, and Decision tree classification algorithms and build classifiers on any two datasets. Divide the data set into training and test set. Compare the accuracy of the different classifiers under the following situations:
   I.   a) Training set = 75% Test set = 25% b) Training set = 66.6% (2/3rd of total), Test set = 33.3%
   II. Training set is chosen by i) hold out method ii) Random subsampling iii) Cross-Validation. Compare the accuracy of the classifiers obtained.
   Data is scaled to standard format.
5. Use Simple K-means algorithm for clustering on any dataset. Compare the performance of clusters by changing the parameters involved in the algorithm. Plot MSE computed after each iteration using a line plot for any set of parameters.

Project: Students should be promoted to take up one project on any UCI/kaggle/data.gov.in or a dataset verified by the teacher. Preprocessing steps and at least one data mining technique should be shown on the selected dataset. This will allow the students to have a practical knowledge of how to apply the various skills learnt in the subject for a single problem/project.

DISCIPLINE SPECIFIC ELECTIVE COURSE: Combinatorial Optimization

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		
Combinatorial Optimization	**4**	**3**	**1**	**0**	Pass in Class XII	NIL

Learning Objectives

This course is designed to introduce the fundamentals of combinatorial optimization to the students in terms of both theory and applications, so as to equip them to explore the more advanced areas of convex and non-convex optimizations.

Learning outcomes
On successful completion of the course, students will be able to:
- Model problems using linear and integer programs
- Apply polyhedral analysis to develop algorithms for optimization problems
- Use the concept of duality for design of algorithms

SYLLABUS OF DSE

Unit 1 (9 hours)

Introduction: Introduction to Combinatorial Optimization Problems, Linear and Integer Programs- LP Formulation, understanding integer programs, computational complexities of IP vs LP, using LP to find optimal or approximate integral solutions, concept of integrality gap.

Unit 2 (14 hours)

Theory of Linear Programming and Algorithmic Perspective to Simplex Method: standard vs. equational form, basic feasible solutions, convexity and convex polyhedra, correspondence between vertices and basic feasible solutions, geometry of Simplex algorithm, exception handling (unboundedness, degeneracy, infeasibility), Simplex algorithm, avoiding cycles.

Unit 3 (12 hours)

Primal-Dual Algorithms: interpretation of dual, optimality conditions for primal and dual, weak and strong duality, complementary slackness, primal-dual algorithm for the shortest path problem.

Unit 4 (10 hours)

Network Flows: linear programming formulations for network flows and bipartite matching, totally unimodular matrices.

Essential/recommended readings

2. Papadimitriou, C.H. & Steiglitz, K. *Combinatorial Optimization: Algorithms and complexity*, New edition, Dover Publications inc., 2000.

Additional References

(i) Bazaraa, M.S., Jarvis, J.J., & and Sherali, H.D. *Linear Programming and Network Flows*, 4th edition, Wiley, 2010.
(ii) Korte, B., & Vygen, J. *Combinatorial Optimization,* 6th edition, Springer, 2018.

Tutorials

Tutorials based on Theory

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		
Network Security		3	1	0	Pass in Class XII	DSC 04 Object Oriented Programming with C++/ GE 1a Programming using C++ / GE1b Programming with Python/ DSC 01 Programming using Python/ GE 3b: Java Programming

Learning Objectives

This course will provide students with an understanding of the fundamental concepts, principles, and techniques of network security. Students will learn how to assess, design, and implement secure networks using various tools and technologies.

Learning outcomes
On successful completion of the course, students will be able to:

- Describe the importance of network security and the principles of the CIA triad (confidentiality, integrity, and availability), types of security threats and attacks

- Describe the basics of cryptography, including symmetric and asymmetric encryption, hash functions, digital signatures, and public key infrastructure (PKI).

- Apply authentication and access control techniques, including password-based, token-based, and biometric authentication, as well as authorization models and single sign-on (SSO).

- Design and implement secure networks using network segmentation, security zones, and VPNs for remote access.

- Implement and manage firewalls, intrusion detection systems (IDS), and intrusion prevention systems (IPS) to protect network resources, secure wireless networks,
- Implement endpoint security and malware protection measures, including antivirus, patch management, and host-based firewalls.

SYLLABUS OF DSE

Unit 1 (10 hours)

Introduction to Network Security and Network Fundamentals: Importance of network security, Confidentiality, integrity, and availability (CIA) triad, Types of security threats and attacks. OSI and TCP/IP models. IP addressing and subnetting. Networking devices (hubs, switches, routers, firewalls). Network protocols and services (HTTP, HTTPS, FTP, SSH, etc.).

Unit 2 (12 hours)

Cryptography Basics, Authentication and Access Control, Secure Network Design: Symmetric and asymmetric encryption, Hash functions and digital signatures, Public key infrastructure (PKI), Common cryptographic algorithms (AES, RSA, SHA, etc.). Authentication techniques (passwords, tokens, biometrics), Authorization and access control models (RBAC, ABAC, MAC, DAC), Single sign-on (SSO) and multi-factor authentication (MFA). Defense-in-depth strategy, Network segmentation and isolation, Security zones and DMZ, VPNs and secure remote access.

Unit 3 (12 hours)

Firewalls and Intrusion Detection/Prevention Systems, Wireless Network Security: Types of firewalls (packet filtering, stateful inspection, application layer), IDS and IPS concepts and deployment, Signature-based and anomaly-based detection, Honeypots and honeynets. Wireless standards and technologies (802.11, Bluetooth, RFID), Wireless security protocols (WEP, WPA, WPA2, WPA3), Rogue access points and wireless attacks, Securing wireless networks.

Unit 4 (8 hours)

Endpoint Security and Malware Protection, Security Monitoring and Incident Response: Antivirus and antimalware solutions, Patch management and software updates, Host-based firewalls and intrusion detection, Mobile device management (MDM). Security Information and Event Management (SIEM) systems, Log management and analysis, Incident response process and procedures, Forensic analysis and evidence handling.

Unit 5 (3 hours)

Network Security Best Practices and Compliance: Security policies and procedures, Risk assessment and management, Security awareness training, Regulatory compliance (HIPAA, GDPR, PCI-DSS, etc.).

Essential/recommended readings

1. Behrouz Forouzan, Cryptography and network security. 3rd edition (2015), McGraw Hill Education.

2. Stallings, W. (2021). Cryptography and Network Security: Principles and Practice (8th Edition). Pearson.
3. Harris, S. (2018). All-in-One CISSP Exam Guide (8th Edition). McGraw-Hill Education.
4. Atul Kahate, Cryptography and Network Security, McGraw-Hill; Fourth edition (8 May 2019); McGraw Hill Education (India).

Additional References

i. Conklin, W. A., White, G., Williams, D., Davis, R., & Cothren, C. (2021). Principles of Computer Security: CompTIA Security+ and Beyond (6th Edition). McGraw-Hill Education.
ii. Chapple, M., & Seidl, D. (2020). Network Security For Dummies. Wiley.
iii. Gibson, D. (2021). CompTIA Security+ Get Certified Get Ahead: SY0-601 Study Guide. YCDA Publishing.

Online Additional Reference Materials:

1. NIST Special Publications: https://csrc.nist.gov/publications/sp
   a. SP 800-53: Security and Privacy Controls for Federal Information Systems and Organizations
   b. SP 800-82: Guide to Industrial Control Systems (ICS) Security
   c. SP 800-115: Technical Guide to Information Security Testing and Assessment
2. ISO/IEC 27000 series: Information Security Management Systems (ISMS)
   a. ISO/IEC 27001: Information Security Management
   b. ISO/IEC 27002: Code of Practice for Information Security Controls
   c. ISO/IEC 27005: Information Security Risk Management
3. Center for Internet Security (CIS) Critical Security Controls: https://www.cisecurity.org/controls/
   a. A prioritized set of actions to improve network security.
4. OWASP Top Ten Project: https://owasp.org/www-project-top-ten/
   a. A list of the most critical web application security risks.
5. SANS Institute Reading Room: https://www.sans.org/reading-room/
   a. A collection of whitepapers and articles on various network security topics.
6. Vendor documentation and best practices guides (Cisco, Juniper, Palo Alto Networks, etc.)

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		
	4	3	0	1	Pass in Class XII	NIL

Learning Objectives

The course aims at introducing the basic concepts and techniques of client side web programming. The student shall be able to develop simple websites using HTML, CSS and Javascript.

Learning outcomes

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

- Build websites using the elements of HTML.
- Build dynamic websites using the client side programming techniques with CSS, Javascript and jQuery.
- Learn to validate client-side data.

SYLLABUS OF DSE

Unit 1 (5 hours)
Introduction: Introduction to Internet and web design. Basic concepts of web architecture.

Unit 2 (12 hours)
HTML: Introduction to hypertext mark-up language (html), creating web pages, lists, hyperlinks, tables, web forms, inserting images, frames.

Unit 3 (8 hours)
Cascading style sheet (CSS): Concept of CSS, creating style sheet, Importing style sheets, CSS properties, CSS styling (background, text format, controlling fonts), CSS rules, Style Types, CSS Selectors, CSS cascade, working with block elements and objects, working with lists and tables, CSS id and class, box model (introduction, border properties, padding

properties, margin properties).

Unit 4 (10 hours)
Javascript: Document object model, data types and variables, functions, methods and events, controlling program flow, JavaScript object model, built-in objects and operators, validations.

Unit 5 (10 hours)
jQuery and JSON: Introduction to jQuery, syntax, selectors, events. JSON file format for storing and transporting data.

Essential/recommended readings

1. Nixon, R. *Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5*, O'Rielly*, 2018.
2. Powell, T.A. *HTML & CSS: The Complete Reference, 5th edition*, Tata McGrawHill, 2010.
3. Duckett, J. *JavaScript and JQuery: Interactive Front-End Web Development*, Wiley, 2014.

Additional References

1. Minnick, J. *Web Design with HTML5 and CSS3*, 8th edition, Cengage Learning, 2015.
2. Boehm, A., & Ruvalcaba, Z. *Munarch's HTML5 and CCS,* 4th edition, Mike Murach & Associates, 2018.
3. J. A. Ramalho *Learn Advanced HTML 4.0 with DHTML*, BPB Publications, 2007.
4. Ivan Bayross *Web Enabled Commercial Application Development Using Html, Dhtml, Javascript, Perl CGI*, BPB Publications, 2009.

Suggested Practical List (If any): (30 Hours)

Practical exercises such as

HTML

1. Create an HTML document with following formatting – Bold, Italics, Underline, Colors, Headings, Title, Font and Font Width, Background, Paragraph, Line Brakes, Horizontal Line, Blinking text as well as marquee text.
2. Create an HTML document with Ordered and Unordered lists, Inserting Images, Internal and External linking
3. Create an HTML displaying this semester's time table.
4. Create a website with horizontal and vertical frames. Top horizontal frame showing your college's name and logo. Bottom horizontal frame split into two vertical frames. The left frame with hyperlinks to pages related to faculty, courses, student activities, etc. The right frame showing corresponding pages based on the link clicked on the left frame.
5. Create a student registration form using HTML which has the following controls:

I.   Text Box
II.   Dropdown box
III.   Option/radio buttons
IV.   Check boxes
V.   Reset and Submit button

CSS

Create a webpage for your department with drop down navigation menu for faculty, courses, activities, etc.. Implement the webpage using styles, rules, selectors, ID, class.

Javascript

1. Create event driven programs for the following:
   a. Enter a number and on click of a button print its multiplication table.
   b. Print the largest of three numbers entered by the user.
   c. Find the factorial of a number entered by the user.
   d. Enter a list of positive numbers using the prompt terminated by a zero. Find the sum and average of these numbers.
2. Create a student registration form using text, radio button, check box, drop down box, text field and all other required HTML elements. Customise the CSS and javascript to input and validate all data. Create functions to perform validation of each element, example:
   a. Roll number is a 7-digit numeric value
   b. Name should be an alphabetical value(String)
   c. Non-empty and valid fields like DOB

jQuery and JSON

1. Change text color and contents using button click events using jQuery
2. Select elements using ID, class, elements name, attribute name
3. Run code on click events in jQuery
4. Handle HTML form, store the data in JSON object, pass them to another page and display it there using jQuery/Javascript