M. Sc. (Artificial Intelligence and Machine Learning)

Syllabus

Semester-I

Course Name: Mathematical Foundation

Course Code: MSCAI 111

Objectives:

To review and strengthen important Mathematical concepts required for AI & ML which the student has already been exposed to in previous programs.

Prerequisites:

1. Undergraduate level course in Linear Algebra
2. Undergraduate level course in Probability and Statistics

Contents:

1. Linear Equations [20%]
   Gaussian Elimination and Matrices, Gauss-Jordon Method, Ill-Conditioned Systems, Row-echelon forms, Homogeneous Systems, Linearity, Matrix Multiplication, Matrix Inversion.

2. Vector Spaces [10%]
   Spaces and Subspaces, Null Space, Rank, Basis and Dimension, Classical Least Square, Linear Transformations, Change of Basis, Invariant Subspaces.

3. Norms and Inner Product Spaces [20%]
   Vector Norms, Matrix Norms, Inner Product Spaces, Orthogonal Vectors, Gram-Schmidt Procedure, Orthogonal Decomposition.

4. Eigenvalues and Eigenvectors [10%]
   Properties of Eigenvalues and Eigenvectors, Positive definite Matrices, Singular Value Decomposition, Principal Component Analysis.

5. Probability Theory [20%]
   Concept of Measure, Sigma algebra, Sample Spaces, Joint distribution, Conditional probability, Marginal Distribution, Independence, Conditional Independence, Continuous Probability Distribution, Gaussian Distribution

6. Probabilistic Graphical Models [20%]
   Bayesian Networks, Markov Models, Independencies, MAP Inference, Sampling

Methods, MCMC (Markov Chain Monte Carlo) Methods, Metropolis-Hastings Algorithm

Reference Books:

a) Linear Algebra and Its Application, 3rd Edition, David C. Lay
b) Linear Algebra, Kenneth Hoffman
c) Linear Algebra and Its Applications, Gilbert Strang
d) A First Course in Probability, 9th Edition, Sheldon Ross
e) An Introduction to Probability and Statistics, 2nd Edition, Rohatgi & Saleh
f) Probabilistic Graphical Models, Daphne Koller & Nir Friedman

Course Name: Introduction to Programming with Python

Course Code: MSCAI 112

Objectives:

Introduce students to Python, the modern language useful for writing compact codes specifically focussing on Data Analysis and Scientific Computing. Equal weightage has been given to both theory and practical.

Prerequisites:

No Prerequisites

Contents:

1. Introduction [5%]
   Basic elements of python, branching process, Control Structures, String and Inputs, iterations

2. Functions and Abstractions [5%]
   Functions and scoping, recursions, global variables, modules, files, system functions and parameters

3. Structured Types, Mutability, Higher order functions [10%]
   Strings, tuples, lists, dictionaries, mutability, Functions as Objects

4. Testing, Debugging, Exceptions and Assertions [10%]
   Types of testing, Debugging, Handling exceptions, Assertions

5. Classes and Objects [10 %]
   Abstract Data Types, Inheritance, Encapsulation

6. Simple Algorithms [10%]
   Search and Sorting Algorithms, Hash Tables

7. Practical Lab [50%]
   Practical implementation of relevant algorithms, e.g. Building a Neural Network from scratch, Implementation of K- Nearest Neighbours Algorithms, etc.

Reference Books:

a) Introduction to Computer Programming using Python, John V Guttag
b) Core Python Programming, R. Nageswara Rao
c) Data Structures and Algorithmic Thinking with Python, N. Karumanchi

Course Name: Introduction to Artificial Intelligence

Course Code: MSCAI 113

Objectives:

Introduce and define the meaning of Intelligence and explore various paradigms for knowledge encoding in computer systems. Introduce subfields of AI such as NLP, Game Playing, Bayesian Models, etc.

Prerequisites:

1. Basic knowledge of Mathematical Logic

Contents:

1. AI Fundamentals [5%]
   Defining Artificial Intelligence, Defining AI techniques,

2. State Space Search and Heuristic Search Techniques [10%]
   Defining problems as State Space search, Production systems and characteristics, Hill Climbing, Breadth first and depth first search, Best first search

3. Knowledge Representation Issues [10%]
   Representations and Mappings, Approaches to knowledge representation

4. Using Predicate Logic and Representing Knowledge as Rules [20%]
   Representing simple facts in logic, Computable functions and predicates, Procedural vs Declarative knowledge, Logic Programming, Forward vs backward reasoning

5. Symbolic Logic under Uncertainty [15%]
   Non-monotonic Reasoning, Logics for non-monotonic reasoning

6. Statistical Reasoning [25%]
   Probability and Bayes Theorem, Certainty factors, Probabilistic Graphical Models,

Bayesian Networks, Markov Networks, Fuzzy Logic

7. Important Applications [15%]
   Introduction to Natural Language Processing, Hopfield Networks, Neural Networks, Recurrent Networks, Symbolic AI

Reference Books:

1. Artificial Intelligence: A Modern Approach, Stuart Russel, Peter Norvig
2. Artificial Intelligence, 2nd Edition, Rich and Knight

Course Name: Object Oriented Concepts and Programming using C++

Course Code: MSCAI 114

Objectives:

The programming for small devices like mobile phones, networking devices like routers, coding for graphics and multimedia, requires efficient coding as well as object oriented programming. The C++ language fits perfectly as a tool for this type of work. How this important language is to be mastered and how to use this knowledge in building efficient and flexible code is one of the prime requirements today. The course presented here is targeting to enable the student to master such skills. Aim of the course is to enable students to

1. Differentiate between procedural and object oriented programming.
2. Learn C++ as a language and various features of it.
3. Learn Object Oriented principles and their application using C++.

Prerequisites:

1. Knowledge of C language
2. Programming concepts including algorithm building and logical problem solving

Contents:

1. Introduction to C++, Overview of Core C++ Language, Classes and Objects [10%]

Identifiers and constants (Literals), Keywords, Data Types, The Operators, New Casting Operators, typeid and throw, The Conditional structures and Looping Constructs, , The Difference between struct and class in C++,The difference between Union and Class, Static Data members of a class, Pointer to objects and pointer to members of class, The local classes, Assigning Objects

2. Functions [10%]

Introduction, The inline function, Default Arguments to the function,Functions with object as parameters,Call by reference and return by reference, Prototyping and Overloading, Friend functions, Const and Volatile functions, Static functions, Private and Public

functions, Function Pointers, Adding C functions to the C++ program

3. Constructors and Destructors [10%]

Introduction to constructors, The explicit constructors, Parameterized constructors, Having multiple constructors, Constructors with default arguments, Dynamic Initialization, Constructor with dynamic allocation, copy constructors, The member initialization list, destructors

4. Operator Overloading and User Defined Conversions [10%]

Introduction, Unary Operators, Binary Operators, Using Friends as operator functions, Overloading other Operators, The need for user defined conversion, Four different cases where user defined conversions are needed, Comparison of both the methods of conversion

5. Templates [5%]

Function Templates, Non Generic (Non Type) Parameters in Template functions, Template function and specialization, Overloading a template function, Using Default Arguments, Class Templates, Classes with multiple generic data types, Static data members, Primary and Partial Specialization, The Export Keyword, The other use of typename

6. Inheritance [10%]

The need, Defining derived class using single base class, Derivation using public, private and protected access modifiers, The implementation of inheritance in the C++ object model, The Access Control, The Access Declaration, The multiple-inheritance, Abstract classes, Composite objects (container objects)

7. Runtime polymorphism by virtual functions [10%]

Compile Time and Runtime Polymorphism, Pointers to Objects, This pointer, Compatibility of Derived and base class pointers, The subobject concept, Virtual functions, Static invocation of virtual function, Default arguments to virtual functions, Virtual destructors, Pure virtual functions

8. IO Streams [5%]

Need for streams, Advantages of using C++ I/O over C IO, The C++ Predefined streams, Formatting IO, Formatting using ios members, Manipulators, Creating our own manipulator

9. Using Files for IO [10%]

Why IO is special, Text and binary streams, Opening and closing files, Dealing with text files Dealing with binary files, Providing Random Access using seek, IO Modes, Handling Errors

10. Namespaces [5%]

Introduction and need, Use the using syntax, Defining namespaces, Extending the namespace, Unnamed namespaces, Nested Namespaces, Namespace aliases, The std namespace, The Koenig lookup, Overhead with namespaces

11. The Standard Template Library [5%]

The STL (Standard Template Library) Introduction, Generic Programming , Generic Software Components, Generic Algorithms, Iterators, Containers, Algorithms.

12. GNU Debugger [5%]

Tracing and altering the execution of programs using gdb.

Reference Books:

1. Programming with ANSI C++ by Bhushan Trivedi, Oxford University Press
2. C++ Primer by Stanley Lippmann Pearson Education
3. The C++ Programming Language by Bjarne Stroustrup, Pearson Education
4. Effective C++ by Scott Mayer Addison Wesley
5. Complete Reference C++ by Herbert Schildt McGraw Hill Publications
6. C++ FAQs by Pearson Education

Accomplishments of the student after completing the course:

1. He/She should be able to understand and appreciate the Object Oriented approach of programming
2. He/She should be aware of the working and architectural model of C++.

He/She should be able to solve problems given to him/her using C++ keeping balance between efficiency and flexibility.

Semester-II

Course Name: Numerical Optimization

Course Code: MSCAI 121

Objectives:

To teach the student fundamental concepts of optimization both from the point of view of theory as well as practical implementation of algorithms relevant to Machine Learning applications.

Prerequisites:

3. Undergraduate level course in Linear Algebra
4. Undergraduate level course in Multivariable Calculus

Contents:

1. Review of Multivariable Calculus [10%]
   Multivariable functions, Partial Derivatives, Total Derivative, Vector Functions, Gradient, Physical interpretation of Gradient, Existence of Minimum and a Maximum, Continuity of Functions, Taylor's Theorem, Convex Functions

2. Optimization Problem Formulation [10%]
   Statement of an Optimization problem, Historical development, Classification of Optimization problems and techniques, Single variable optimization problem, Iterative algorithmic approach

3. One Dimensional Unconstrained Optimization [10%]
   Unimodality and bracketing, Fibonacci Method, Golden Section Method, Line search

4. Unconstrained Optimization [30%]
   Necessary and Sufficient conditions for optimality, Convexity, Steepest Descent Method, Hessian Matrix, Conjugate Gradient Method, Newton's Method, Quasi-Newton Method, Approximate Line Search

5. Constrained Optimization [20%]
   Necessary conditions for optimality, sufficient conditions for optimality, sensitivity of solution, Sequential Quadratic Programming, Duality, Exterior penalty functions, interior penalty functions

6. Direct Search methods [20%]
   Hooke-Jeeves Pattern Search, Powell's Methods of Conjugate directions, Nelder-Mead's Simplex methods, Simulated Annealing, Genetic Algorithms

Reference Books:

a) Optimization Concepts and Applications in Engineering, Belegundu
b) Engineering Optimization, 2nd Edition, Ravindran & Reklaits
c) Practical Methods of Optimization, R. Fletcher

Course Name: Advanced Python

Course Code: MSCAI 122

Objectives:

This course will introduce students to advanced python implementations and the latest Machine Learning and Deep learning libraries, Scikit-Learn and TensorFlow. The course will be hands on with major focus on practical implementation of concepts. The course will have two mini projects to help the students master the concepts.

Prerequisites:

1. Introductory course in Python

Contents:

1. Review of Important Python Concepts [5%]
   Classes, String, Tuples, Lists, Dictionaries, sorting, handling exceptions, using iPython

2. Machine Learning Algorithms with Scikit-learn [10%]
   Pandas Library, Using Scikit-Learn for Logistic Regression, Support Vector Machines, Building Neural Networks,

3. Introduction to Tensor Flow [30%]
   Concept of Computational Graph and Nodes, Virtual Environment and Anaconda, Installing TensorFlow with GPU support on a Linux System, TF Datatypes, Placeholders, TF Variables, TF Session, Softmax, One Hot Encoding, Dropout, building hidden layers, Batching, Stochastic Gradient Descent, Building an Optimizer, Training and displaying results

4. Building a Neural Network with Tensor Flow [5%]
   Using inbuilt TensorFlow functionality to build a Neural Network and train on MNIST Dataset for classification

5. Practical Implementation [50%]
   Mini Project on Machine Learning Application using Scikit-Learn, Mini Project on TensorFlow implementation for a classification problem

Reference Books:

1. Python Machine Learning, Sebastian Raschka
2. Getting Started with TensorFlow, Giancarlo Zaccone

Course Name: Introduction to Machine Learning

Course Code: MSCAI 123

Objectives:

Introduce the concept of learning patterns from data and develop a strong theoretical foundation for understanding state of the art Machine Learning algorithms. This course is broad in scope and gives the student a holistic understanding of the subject.

Prerequisites:

1 Undergraduate level course in Linear Algebra

2 Undergraduate level course in Calculus

Contents:

7. Introduction [5%]
   Idea of Machines learning from data, Classification of problem – Regression and Classification, Supervised and Unsupervised learning

8. Linear Regression [15%]
   Model representation for single variable, Single variable Cost Function, Gradient Decent for Linear Regression, Multivariable model representation, Multivariable cost function, Gradient Decent in practice, Normal Equation and non-invertibility

9. Logistic Regression [15%]
   Classification, Hypothesis Representation, Decision Boundary, Cost function, Advanced Optimization, Multi-classification (One vs All), Problem of Overfitting, Regularization

10. Neural Networks [20%]
    Non-linear Hypothesis, Biological Neurons, Model representation, Intuition for Neural Networks, Multiclass classification, Cost Function, Back Propagation Algorithm, Back Propagation Intuition, Weights initialization, Neural Network Training

11. Support Vector Machines [15%]
    Optimization Objective, Large Margin Classifiers, Kernels, SVM practical considerations

12. Unsupervised learning [20%]
    Unsupervised learning introduction, k-Means Algorithm, Optimization objective, Random Initialization, Choosing number of clusters

13. Recommender Systems [10%]
    Problem Formulation, Content based recommendations, Collaborative Filtering, Vectorization, Implementation details.

Reference Books:

1. Machine Learning, Tom M. Mitchell
2. Building Machine Learning Systems with Python, Richert & Coelho

Course Name: Introduction to Computer Vision

Course Code: MSCAI 124

Objectives:

A.I. has major applications in Computer Vision, especially in object detection, recognition & classification. This course introduces the student to the fundamentals of Image processing and Computer Vision to provide sufficient background so that the student can then tackle the hard problems in later courses.

Prerequisites:

1. Introductory course in Linear Algebra
2. Introductory course in Calculus
3. Introductory course in Probability

Contents:

1. Introduction [20%]
   Overview, Smoothing, Image Morphology, Flood Fill, Resize, Image Pyramids, Thresholding operation

2. Image Transforms [30%]
   Convolution, Gradients and Sobel Derivatives, Laplace, Canny & Hough Transforms, Remap, Stretch, Shrink, Warp, and Rotate, Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Integral Images, Distance Transform, Histogram Equalization

3. Image Parts and Segmentation [20%]
   Parts and Segments, Background Subtraction, Watershed Algorithm, Image Repair by Inpainting, Mean-Shift Segmentation, Delaunay Triangulation, Voronoi Tessellation.

4. Tracking and Motion [20%]
   The Basics of Tracking, Corner Finding, Subpixel Corners, Invariant Features, Optical Flow, Mean-Shift & Camshift Tracking, Motion Templates, Estimators

5. Camera Models and Calibration [10%]
   Developing Camera Model, Calibration, Un-distortion, Putting Calibration Together, Rodrigues Transform

Reference Books:

1. Image Processing: Analysis and Machine Vision, Sonka & Hlavac
2. Digital Image Processing, Gonzalez