



## AI/ML /Data Science (40 Hours)

### AI ML Concepts:

What is machine learning?

- 1) Traditional Programming
- 2) Machine Learning

Applications of Machine Learning

- 3) Web search
- 4) Computational
- 5) Finance
- 6) E-commerce
- 7) Space exploration
- 8) Robotics
- 9) Information extraction
- 10) Social networks
- 11) Debugging

Key Elements of Machine Learning

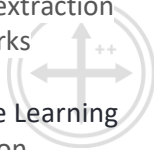
- 12) Representation
- 13) Evaluation
- 14) Optimization

Types of Learning

- 15) Supervised learning
- 16) Unsupervised learning
- 17) Semi-supervised learning
- 18) Reinforcement learning

Machine Learning in Practice

- 19) Understand the domain, prior knowledge and goals
- 20) Data integration, selection, cleaning and pre-processing.
- 21) Learning models
- 22) Interpreting results.
- 23) Consolidating and deploying discovered knowledge





### **AI ML Basics:**

- 1) Myth busting Data Science
- 2) Myth busting ML and AI - GCP for ML applications and Alexa cloud architecture
- 3) Applications in ML vs DL - When not to use ML/DL
- 4) Analytics - Types/ Roles/ Applications - Financial Use cases.
- 5) Machine Learning Pipeline
- 6) Machine Learning tools eco-system in the cloud and lab
- 7) Problem Framing - Understanding predictability and traceability
- 8) EDA - Various Use cases.
- 9) Linear Regression
- 10) Logistic Regression
- 11) Loss and Producing ML pipeline.
- 12) Linear Regression and Training and Loss
- 13) Reducing Loss
- 14) 15) Generalization Techniques
- 16) Validation Set
- 17) Representation
- 18) Feature Crossing and Optimisation
- 19) Regularisation - L2 and Lambda
- 20) K-Means
- 21) Gaussian Naive Bayes
- 22) SVM
- 23) Classification - Precision, Accuracy, Recall, ROC and AUC, Prediction Bias study
- 24) Neural networks
- 25) Activation Functions and Propagation cycles
- 26) Training neural nets
- 27) Embeddings
- 28) ML engineering fundamentals - production level deployment and stack.

### **Python Basic:**

An understanding of how to use the Python standard library to write programs, access various tools, and document and automate analytical processes.

- Ø Types (strings, lists, dictionaries, and more)
- Ø Control Flow (if-then statements, looping)
- Ø Organizing code (functions, modules, packages)
- Ø Reading and writing files
- Ø Overview of Object-Oriented Programming (OOP)



### **NumPy & 2D Plotting Library:**

Introduction to NumPy and 2D plotting. The NumPy package is presented as a tool for rapidly manipulating and processing large data sets. 2D plotting is introduced with matplotlib.

- Ø Understanding the N-dimensional data structure
- Ø Creating arrays
- Ø Indexing arrays by slicing or more generally with indices or masks
- Ø Basic operations and manipulations on N-dimensional arrays
- Ø Plotting with matplotlib

### **Python Pandas & Data Analysis:**

the Python Data Analysis Library (Pandas) is a powerful and convenient package

- Ø Tabular Datasets
- Ø Data Aggregation & Data Exploration
- Ø Labelling data for each dimensionBasic operations and manipulations on N-dimensional arrays
- Ø Dealing with missing values, and time series manipulations.

### **Accessing Data from & multiple sources:**

- Ø Reading and writing data from local files (.txt,.csv,.xls, json, etc.)
- Ø Reading data from remote files
- Ø Scraping tables from web pages (.html)
- Ø Making the most of the powerful read table method

### **Data Preparation & Cleaning:**

- Ø Working with Pandas data structures: Series and Data Frames.
- Ø Accessing your data: indexing, slicing, fancy indexing, Boolean indexing.
- Ø Data wrangling, including dealing with dates and times and missing data.
- Ø Adding, dropping, selecting, creating, and combining rows and columns.



### **Data Access & Databases:**

- Ø Database access with DB-API2 and SQL Alchemy.
- Ø Executing SQL commands from Pandas
- Ø Loading database data into a Data Frame.
- Ø Combining and manipulating Data Frames: merge, join, concatenate

### **Data Visualization:**

- Ø Understanding the structure of a Figure
- Ø Data visualization: scatter plots, line plots, box plots, bar charts, and histograms with matplotlib
- Ø Customizing plots: important attributes and arguments

### **Data Analysis:**

- Ø Split-apply-combine with Data Frames
- Ø Data summarization and aggregation methods
- Ø Pandas powerful group by method
- Ø Reshaping, pivoting, and transforming your data
- Ø Simple and rolling statistics

### **Real World Modelling & Problem Solving:**

- Ø Deep learning of the data analysis tools through lectures, Q&A, and hands-on exercises
- Ø Develop transferable skills through application to authentic data sets
- Ø Predict the future with time series analysis
- Ø And more!

### **Python Data Science:**

- Ø Linear Regression
- Ø SVM (Support Vector Machine)
- Ø KNN (K-Nearest Neighbors)
- Ø Logistic Regression



- Ø Decision Tree
- Ø K-Means
- Ø Random Forest
- Ø Naive Bayes
  
- Ø Dimensional Reduction Algorithms
- Ø Gradient Boosting Algorithms

#### **Python Forecasting Modelling in Data Science:**

- Ø Autoregression (AR)
- Ø Moving Average (MA)
- Ø Autoregressive Moving Average (ARMA)
- Ø Autoregressive Integrated Moving Average (ARIMA)
- Ø Seasonal Autoregressive Integrated Moving-Average (SARIMA)
- Ø Seasonal Autoregressive Integrated Moving-Average with Exogenous Regressors (SARIMAX)
- Ø Vector Autoregression (VAR)
- Ø Vector Autoregression Moving-Average (VARMA)
- Ø Vector Autoregression Moving-Average with Exogenous Regressors (VARMAX)
- Ø Simple Exponential Smoothing (SES)
- Ø Holt Winter's Exponential Smoothing (HWES)