

Practical Machine Learning Using Python

Course 3585 – 40 Hours

Overview

Machine learning is the science of getting computers to act without being explicitly programmed. Autonomic cars, face recognition, effective web search, and analyzing human genome are all examples of Machine Learning Solutions.

In this course, you will learn about the most effective machine learning techniques, and gain practice implementing them and getting them to work for yourself.

Python is one of the most popular programming languages for data analysis and Machine Learning. Using Python, you'll learn about not only the theoretical underpinnings of learning, but also gain the practical know-how needed to quickly and powerfully apply these techniques to new problems.

Who Should Attend

This course is mainly intended for Data Analysts, Developers, Business Intelligence professionals, Data Engineers, and other roles responsible for analyzing the organization's data.

Prerequisites

- Basic understanding of Programming concepts

Course Contents

• Python Refresher/Crash Course

- Introduction
- IDEs and Tools for data analysis
- Data types and strings
- Control structures
- Functions and functional programming
- Collections
- Object Oriented
- Modules and Packages

• Introduction to Data Analysis

- Data Science
- Machine Learning Overview
- Data Analysis and ML Packages overview
- Demo – Complete Machine Learning task

- **Machine Learning Basic Concepts**

- Overview
- Why Learn
- Applications
- Machine Learning Process
- Learning Types:
 - Supervised Learning
 - Unsupervised Learning
 - Semi-Supervised Learning
 - Active Learning
 - Reinforcement learning
- Batch vs Online Learning
- Instance-based vs model-based learning
- CRISP – DM Methodology

- **Business and Data Understanding**

- ML tasks
- Algorithms
- Variables and Features
- Training, Validating and Testing Data
- Exploratory Analysis
- Types of data/features
- Pandas package (python)
 - Series and Dataframes
 - Selecting data
 - Grouping and aggregation
- Data visualization
- Matplotlib and Seaborn packages (python)
 - Line and bar graphs
 - Scatter graphs
 - Histograms and distributions
 - Bar plots and Box-plot
 - Heat map and correlation
 - Other useful graphs for statistics
 - Animations and events

- **Data Preparation**

- Data preparation and cleaning
- Dealing with Missing values
- Central tendency
- Mean/median/mode
- Bias
- Variance and standard deviation
- Standard scores
- Feature scaling – standardization and normalization
- Numpy package (python)

- Vectors and matrixes
- Multi-dimensional arrays
- Functions
- Slicing and fancy indexing
- Linear algebra
- Feature Engineering
- Feature selections
- Dummy variables
- Converting continuous variable to discrete

• Supervised Learning

- Overview
- Regressions
- Classification
- Linear Regression
 - Least Squares using numpy
 - Using scikit-learn package
 - Gradient descent using TensorFlow package
- Non-Linear Regression
- SciPy package (python)
 - Interpolation
 - Curve fit
 - Optimization
 - Statistics
 - Image processing
 - Integration and more
- Logistic Regression
- Model evaluation using metrics
- SVM
- Tuning hyper parameters
- Cross validation and grid search
- Decision trees and random forests
- Naïve Bayes
- KNN
- Classification
 - Multi class
 - Multi label
 - One vs All
 - All vs All
 - Error correcting codes

• Unsupervised Learning

- Overview
- Clustering
- K-Means
- Recommender systems

- **Deep Learning and Neural Networks**

- Deep Learning
- Neural networks overview
- The perceptron
- Network structure and hidden layers
- Activation functions
- Training the network
- Optimization
- Forward and back propagation
- Gradient decent
- Convergence
- Learning rate
- Overfitting and underfitting
- Adding bias
- Boltzmann Machines
- Convolutional Neural Networks
- Recurrent Neural Networks
- SOM
- Auto Encoders

- **Advanced topics**

- NLP
- Anomaly detection
- Handling imbalanced data
- Ensemble methods