



Machine Learning Using Python

Duration: 1 Month



Introduction to Machine Learning

- ✓ Introduction to Machine Learning and its types (supervised, unsupervised, reinforcement learning)
- ✓ Setting up the development environment (Python, Jupyter Notebook, libraries: NumPy, Pandas, Scikit-learn)
- ✓ Overview of the Machine Learning workflow and common data preprocessing techniques





Introduction to data science and its applications

- ✓ Definition of data science and its role in various industries.
- ✓ Explanation of the data science lifecycle and its key stages.
- ✓ Overview of the different types of data: structured, unstructured, and semi-structured.
- ✓ Discussion of the importance of data collection, data quality, and data preprocessing.



Exploratory data analysis (EDA) using Pandas and NumPy

- ✓ Introduction to Pandas, a Python library for data manipulation and analysis.
- ✓ Overview of NumPy, a fundamental package for scientific computing with Python.
- ✓ Explanation of key data structures in Pandas: Series and DataFrame.
- ✓ Hands-on exploration of data using Pandas to summarize, filter, and transform data.
- ✓ Data cleaning techniques, handling missing values, and dealing with outliers.
- ✓ Statistical analysis of data using NumPy functions.



Data visualization using Matplotlib, Seaborn, and Plotly

- ✓ Introduction to data visualization and its importance in data analysis.
- ✓ Overview of Matplotlib, a popular plotting library in Python.
- ✓ Exploring different types of plots: line plots, scatter plots, bar plots, histogram, etc.
- ✓ Customizing plots with labels, titles, colors, and styles.
- ✓ Introduction to Seaborn, a Python data visualization library based on Matplotlib.
- ✓ Advanced plotting techniques with Seaborn: heatmaps, pair plots, and categorical plots.
- ✓ Introduction to Plotly, an interactive plotting library for creating web-based visualizations.
- ✓ Creating interactive and dynamic visualizations with Plotly

Hands-on: Instagram Reach Analysis



Data Engineering and Preprocessing

- ✓ Introduction to Data Engineering: Data cleaning, transformation, and integration
- ✓ Data cleaning and Handling missing values: Imputation, deletion, and outlier treatment
- ✓ Feature Engineering techniques: Creating new features, handling date and time variables, and encoding categorical variables
- ✓ Data Scaling and Normalization: Standardization, min-max scaling, etc.
- ✓ Dealing with categorical variables: One-hot encoding, label encoding, etc.



Web Scraping

- ✓ Introduction to web scraping: Tools, libraries, and ethical considerations
- ✓ Scraping data from websites using libraries like BeautifulSoup and requests: HTML parsing, locating elements, and extracting data
- ✓ Handling different types of data on websites: Tables, forms, etc.
- ✓ Storing scraped data in appropriate formats: CSV, JSON, or databases

Hands-on: Working on Scraping Data from Static / Dynamic Website



Supervised Learning - Regression

- ✓ Introduction to Regression: Definition, types, and use cases
- ✓ Linear Regression: Theory, cost function, gradient descent, and assumptions
- ✓ Polynomial Regression: Adding polynomial terms, degree selection, and overfitting
- ✓ Lasso and Ridge Regression: Regularization techniques for controlling model complexity
- ✓ Evaluation metrics for regression models: Mean Squared Error (MSE), R-squared, and Mean Absolute Error (MAE)

Hands-On - Video Game Sales Prediction



Supervised Learning - Classification

- ✓ Introduction to Classification: Definition, types, and use cases
- ✓ Logistic Regression: Theory, logistic function, binary and multiclass classification
- ✓ Decision Trees: Construction, splitting criteria, pruning, and visualization
- ✓ Random Forests: Ensemble learning, bagging, and feature importance
- ✓ Evaluation metrics for classification models: Accuracy, Precision, Recall, F1-score, and ROC curves
- ✓ Implementation of classification models using scikit-learn library

Hands-On - Heart Disease Detection & Food Order Prediction



SVM, KNN & Naive Bayes

- ✓ Support Vector Machines (SVM): Study SVM theory, different kernel functions (linear, polynomial, radial basis function), and the margin concept. Implement SVM classification and regression, and evaluate the models.
- ✓ K-Nearest Neighbors (KNN): Understand the KNN algorithm, distance metrics, and the concept of K in KNN. Implement KNN classification and regression, and evaluate the models.
- ✓ Naive Bayes: Learn about the Naive Bayes algorithm, conditional probability, and Bayes' theorem. Implement Naive Bayes classification, and evaluate the model's performance.

Hands-On - Contact Tracing & Sarcasm Detection



Ensemble Methods and Boosting

- ✓ AdaBoost: Boosting technique, weak learners, and iterative weight adjustment
- ✓ Gradient Boosting (XGBoost): Gradient boosting algorithm, Regularization, and hyperparameter tuning
- ✓ Evaluation and fine-tuning of ensemble models: Cross-validation, grid search, and model selection
- ✓ Handling imbalanced datasets: Techniques for dealing with class imbalance, such as oversampling and undersampling

Hands-On - Medical Insurance Price Prediction



Unsupervised Learning - Clustering

- ✓ Introduction to Clustering: Definition, types, and use cases
- ✓ K-means Clustering: Algorithm steps, initialization methods, and elbow method for determining the number of clusters
- ✓ DBSCAN (Density-Based Spatial Clustering of Applications with Noise): Core points, density reachability, and epsilon-neighborhoods
- ✓ Evaluation of clustering algorithms: Silhouette score, cohesion, and separation metrics

Hands-On - Credit Card Clustering



Unsupervised Learning - Dimensionality Reduction

- ✓ Introduction to Dimensionality Reduction: Curse of dimensionality, feature extraction, and feature selection
- ✓ Principal Component Analysis (PCA): Eigenvectors, eigenvalues, variance explained, and dimensionality reduction
- ✓ Implementation of PCA using scikit-learn library

Hands-On - MNIST Data



Model Evaluation and Hyperparameter Tuning

- ✓ Cross-validation and model evaluation techniques
- ✓ Hyperparameter tuning using GridSearchCV and RandomizedSearchCV
- ✓ Model selection and comparison



Natural Language Processing (NLP)

- ✓ Introduction to NLP: Understand the basics of NLP, its applications, and challenges.
- ✓ Text Preprocessing: Learn about tokenization, stemming, lemmatization, stop word removal, and other techniques for text preprocessing.
- ✓ Text Representation: Explore techniques such as Bag-of-Words (BoW), TF-IDF, and word embeddings (e.g., Word2Vec, GloVe) for representing text data.
- ✓ Sentiment Analysis: Study sentiment analysis techniques, build a sentiment analysis model using supervised learning, and evaluate its performance.

Hands-On - Real Time Sentiment Analysis



Recommendation Systems

- ✓ Introduction to Recommendation Systems: Understand the concept of recommendation systems, different types (collaborative filtering, content-based, hybrid), and evaluation metrics.
- ✓ Collaborative Filtering: Explore collaborative filtering techniques, including user-based and item-based approaches, and implement a collaborative filtering model.
- ✓ Content-Based Filtering: Study content-based filtering methods, such as TF-IDF and cosine similarity, and build a content-based recommendation system.
- ✓ Deployment and Future Directions: Discuss the deployment of recommendation systems and explore advanced topics in NLP and recommendation systems.

Hands-On - News Recommendation System

PAGE NO : 05



Reinforcement Learning

- ✓ Introduction to Reinforcement Learning: Agent, environment, state, action, and reward
- ✓ Markov Decision Processes (MDP): Markov property, transition probabilities, and value functions
- ✓ Q-Learning algorithm: Exploration vs. exploitation, Q-table, and learning rate
- ✓ Hands-on reinforcement learning projects and exercises

Hands-On - Working with OpenAI Gym



Developing API using Flask / Webapp with Streamlit

- ✓ Introduction to Flask / Streamlit web framework
- ✓ Creating a Flask / Streamlit application for ML model deployment
- ✓ Integrating data preprocessing and ML model
- ✓ Designing a user-friendly web interface



Deployment and Web Development

- ✓ Building a web application for Machine Learning models: Creating forms, handling user input, and displaying results
- ✓ Deployment using AWS (Amazon Web Services): Setting up an AWS instance, configuring security groups, and deploying the application
- ✓ Deployment using PythonAnywhere: Uploading Flask application files, configuring WSGI, and launching the application



Project Work and Consolidation

- ✓ Work on a real-world Machine Learning project: Identify a problem, gather data, and define project scope
- ✓ Apply the learned concepts and algorithms: Data collection, preprocessing, model building, and evaluation
- ✓ Deployment of the project on AWS or PythonAnywhere: Showcase the developed application and share the project with others
- ✓ Presentation and discussion of the project: Demonstrate the project, explain design decisions, and receive feedback



Projects & Case Study



Real-Time Rain Prediction

- Install necessary libraries
- Obtain an API key
- Fetch live weather data
- Preprocess the data
- Train a machine learning model
- Evaluate the model
- Integrate the model with Flask
- Display the results
- Test and debug
- Deploy the application
- Continuously update the weather data



Netflix Recommendation System

- Gather the dataset
- Explore and preprocess the dataset
- Define the problem
- Feature engineering
- Build the recommendation model
- Train the model
- Evaluate the model
- Generate recommendations
- Deploy the recommendation system
- Continuously update the dataset and retrain the model



House Price Prediction

- ✓ Identify a reliable source for house price data
- ✓ Understand the website structure
- ✓ Perform web scraping
- ✓ Preprocess the scraped data
- ✓ Explore and preprocess additional data sources (if applicable)
- ✓ Define the problem
- ✓ Split the data
- ✓ Train the model
- ✓ Evaluate the model
- ✓ Fine-tune the model (optional)
- ✓ Deploy the model
- ✓ Continuously update the dataset and retrain the model



Stock Price Prediction

- ✓ Import necessary libraries
- ✓ Collect stock price data
- ✓ Preprocess the data
- ✓ Explore and visualise the data
- ✓ Feature engineering
- ✓ Split the data
- ✓ Choose an ML algorithm
- ✓ Train the model
- ✓ Evaluate the model
- ✓ Predict stock prices
- ✓ Analyze the results
- ✓ Iterate and improve



GDP Prediction

- ✓ Import necessary libraries
- ✓ Collect GDP data
- ✓ Preprocess the data
- ✓ Explore and visualize the data
- ✓ Feature engineering
- ✓ Split the data
- ✓ Choose an ML algorithm
- ✓ Train the model
- ✓ Evaluate the model
- ✓ Predict GDP
- ✓ Analyze the results
- ✓ Iterate and improve



Course Outcome:

- ✓ By the end of this course, you will have a strong understanding of machine learning concepts, algorithms, and techniques. You will be able to build, evaluate, and deploy machine learning models for various tasks. Additionally, you will have the skills to perform data engineering, data preprocessing, and web scraping. You will also be proficient in deploying your ML models using AWS and PythonAnywhere, and building user-friendly web interfaces with Flask.