INFORMATION BROCHURE

OF

Internship on

Data Science, Machine Learning and AI using Python

National Institute of Electronics & Information Technology
A Deemed to be University
under
Ministry of Communications & Information Technology, Govt. of India
Centre of Excellence in Chip Design, Noida
PS-1D, Behind Brahampurtra Shopping Complex Sector 29, Arun Vihar, Sector 37, Noida, Uttar Pradesh 201303
Phone: 7706009303
E-Mail: pankaj.shukla@nielit.gov.in, suklap@gmail.com

NATIONAL INSTITUTE OF ELECTRONICS AND INFORMATION TECHNOLOGY

Course Name: Internship on Data Science, Machine Learning and AI using Python

(Course Start Date: 5 November 2024 at 3:00 PM)
(Timing: 3 PM to 5 PM Daily)

Course Overview:

This course aims to equip students with the foundational skills needed to analyze and model data using Python. It introduces learners to essential techniques and tools in Data Science and Machine Learning (ML), emphasizing hands-on learning through projects and case studies. By the end of the course, students will be able to build and evaluate machine learning models, understand data analysis, and apply these techniques to real-world problems.

Objective:

- Understand the Data Science workflow and how to structure data analysis projects.
- Explore Python libraries such as NumPy, Pandas, Matplotlib, Scikit-learn, and TensorFlow for data manipulation, visualization, and machine learning.
- Gain proficiency in data pre-processing, feature engineering, and exploratory data analysis.
- Implement various machine learning algorithms and models in Python.
- Learn how to evaluate model performance and optimize machine learning workflows.
- Apply ML techniques to solve real-world problems through project-based learning.

Projects to be covered in the course:

1. **Regression: Admission Prediction Problem**
2. **Classification: Loan Prediction Problem**
3. **Image Classification Problem**
4. **Natural Language Processing: Spam Detection Problem**

5. **Clustering: Customer Segmentation**
6. **Neural Network/Deep Learning: Image Classification Problem**
7. **Deployment of Machine Learning Model using Flask or any other web framework**
8. **Final Capstone Project:** Choose a real-world problem (e.g., predicting house prices, or sentiment analysis). Apply the entire data science pipeline: data cleaning, EDA, feature engineering, model building, evaluation, and deployment. Present your solution in a final presentation and submit a report.

Duration:

90 Hours to be covered in 9 weeks (10 Hours per week)

Fees:

Course Fees: Rs. 2700/- (Online)

Course Fees: Rs. 4500/- (Offline)

Note: Recorded Lecture up to 7 days will be provided.

Registration: Interested candidates can get registration done on following web site:

https://regn.nielitvte.edu.in/

Certification: At the end of the course, online assessment will be done. After assessment, Certificate will be provided to successful candidates.

Eligibility: Any one with interest in Data Science, Machine Learning and AI is eligible for this course

Faculty for the course:

Shri Pankaj Shukla
M. Tech. (CSE)- IIT Delhi
Scientist 'E', NIELIT
Mobile/WhatsApp: 7706009303
E-Mail: pankaj.shukla@nielit.gov.in, suklap@gmail.com

Detailed Syllabus of the Course

Module 1: Introduction to Data Science and Python for Data Analysis

Lecture Material:

- Introduction to Data Science concepts
- Overview of Python libraries:
    - Pandas, NumPy, Matplotlib, and Seaborn

Code Exercises:

1. **Basic Python Practice**:
    - Python syntax, loops, conditionals
    - Data structures: lists, dictionaries, sets
    - Example: Write a Python script that reads in a CSV file and prints its contents.
2. **Pandas Basics**:
    - Loading data from CSV
    - Manipulating DataFrames: adding/removing columns, filtering data
    - Exercise: Import a dataset, such as the Titanic dataset from Kaggle, and perform basic manipulations (sorting, filtering, and summarizing).
3. **Numpy Basics**:
    - Working with arrays, broadcasting, matrix operations
    - Exercise: Implement basic matrix operations like dot product and matrix addition.

Project 1:

- **Title**: Exploring a Dataset (Titanic or Iris dataset)
- **Description**: Perform basic data loading, cleaning, and exploration using Pandas. Summarize statistics of the dataset and visualize distributions using Matplotlib.

Module 2: Data Exploration and Visualization

Lecture Material:

- Exploratory Data Analysis (EDA) process:
    - Descriptive statistics (mean, median, variance)
    - Visualization techniques (histograms, scatterplots, boxplots)
    - Reading: Exploratory Data Analysis

Code Exercises:

1. **Data Exploration**:
    - Use Pandas to explore dataset statistics
    - Exercise: Generate summary statistics and check for missing values in a dataset

2. **Data Visualization with Matplotlib and Seaborn**:
    - Generate histograms, box plots, scatter plots
    - Exercise: Plot various relationships in a dataset using Seaborn's `pairplot()` function.

Project 2:

- **Title**: Exploratory Data Analysis of a Real-World Dataset (e.g. admission prediction data or sales data)
- **Description**: Perform a full exploratory analysis, summarize important features, and visualize key relationships using Matplotlib and Seaborn.

Module 3: Data Cleaning and Preprocessing

Lecture Material:

- Handling missing values, outliers, and data transformations
    - Reading: Data Cleaning with Python

Code Exercises:

1. **Handling Missing Data**:
    - Use Pandas to fill in or drop missing values
    - Exercise: Clean the Titanic dataset by filling in missing values for `Age`, `Embarked`, and other columns.
2. **Scaling and Encoding**:
    - Scaling data (MinMaxScaler, StandardScaler), one-hot encoding categorical variables
    - Exercise: Apply scaling and one-hot encoding to a dataset (e.g., Kaggle's House Prices dataset).

Project 3:

- **Title**: Data Preprocessing Pipeline
- **Description**: Build a preprocessing pipeline that includes missing value handling, outlier detection, and data scaling on a dataset such as house prices.

Module 4: Introduction to Machine Learning

Lecture Material:

- Introduction to Supervised Learning and the ML workflow

Code Exercises:

1. **Simple Linear Regression**:

    - Implement linear regression using Scikit-learn.
    - Exercise: Build a linear regression model to predict housing prices.
2. **Cross-Validation and Model Splitting**:
    - Splitting data into training and test sets, using `train_test_split()`
    - Exercise: Split the dataset into training and testing sets, and evaluate performance using Mean Squared Error (MSE).

Project 4:

- **Title**: Simple Linear Regression for Housing Prices
- **Description**: Build and evaluate a linear regression model to predict house prices using the Boston Housing dataset.

Module 5: Regression Models

Lecture Material:

- Linear vs. Polynomial regression, Ridge, Lasso
    - Reading: Regularization Techniques

Code Exercises:

1. **Polynomial Regression**:
    - Implement polynomial regression with Scikit-learn.
    - Exercise: Build a polynomial regression model and compare its performance to linear regression on a dataset like the Boston Housing dataset.
2. **Regularization**:
    - Implement Ridge and Lasso regularization.
    - Exercise: Apply Ridge and Lasso to regularize a regression model.

Project 5:

- **Title**: Predicting House Prices with Regularization
- **Description**: Build a house price prediction model using Ridge and Lasso regularization. Compare performance and analyze the effects of regularization parameters.

Module 6: Classification Models

Lecture Material:

- Logistic Regression, Decision Trees, and Random Forests

Code Exercises:

1. **Logistic Regression**:
    - Implement binary classification using logistic regression.
    - Exercise: Classify passengers as survivors or not on the Titanic dataset.
2. **Decision Trees**:
    - Build decision trees using Scikit-learn.
    - Exercise: Apply decision trees to classify wine quality from the Wine dataset.
3. **Random Forests**:
    - Implement Random Forests in Scikit-learn.
    - Exercise: Use Random Forests to predict a classification problem and evaluate performance with accuracy and confusion matrix.

Project 6:

- **Title**: Predicting Loan prediction with Logistic Regression and Decision Trees
- **Description**: Build classification models (logistic regression, decision trees, and random forests) to predict Titanic survivors. Compare model performance and visualize the decision tree.

Module 7: Unsupervised Learning and Clustering

Lecture Material:

- K-means clustering, hierarchical clustering, and PCA

Code Exercises:

1. **K-Means Clustering**:
    - Implement K-means clustering in Scikit-learn.
    - Exercise: Cluster customers based on purchasing behavior (e.g., from a Mall Customers dataset).
2. **PCA for Dimensionality Reduction**:
    - Implement PCA to reduce the dimensionality of a dataset.
    - Exercise: Apply PCA to visualize clusters of the Iris dataset.

Project 7:

- **Title**: Customer Segmentation using K-Means and PCA
- **Description**: Perform customer segmentation using K-means clustering and apply PCA for dimensionality reduction on a customer dataset.

Module 8: Advanced Machine Learning Techniques

Lecture Material:

- Support Vector Machines (SVM), Gradient Boosting, and Ensemble methods

Code Exercises:

1. **Support Vector Machines**:
    - Implement SVM for classification.
    - Exercise: Build an SVM model to classify images in the MNIST dataset.
2. **Gradient Boosting (XGBoost)**:
    - Implement XGBoost for classification tasks.
    - Exercise: Use XGBoost to solve a Kaggle competition dataset.

Project 8:

- **Title**: Implementing Ensemble Models (Random Forests and Gradient Boosting)
- **Description**: Use ensemble learning methods to build and optimize a classification model on a real-world dataset. Compare random forests, XGBoost, and other boosting techniques.

Module 9: Introduction to Deep Learning

Lecture Material:

- Neural Networks, Backpropagation, and TensorFlow/Keras

Code Exercises:

1. **Building a Neural Network**:
    - Create a basic neural network using TensorFlow/Keras.
    - Exercise: Implement a neural network to classify the MNIST handwritten digits dataset.
2. **Backpropagation and Optimization**:
    - Visualize backpropagation and weight updates.
    - Exercise: Tune hyperparameters and optimize the neural network model for better performance.

Project 9:

- **Title**: Image Classification with Neural Networks
- **Description**: Build a neural network for image classification using TensorFlow/Keras. Train it on the MNIST dataset and experiment with different architectures and optimization techniques.

Module 10: Model Evaluation and Tuning

Lecture Material:

- Cross-validation, Grid Search, and Hyperparameter Tuning

Code Exercises:

1. **Cross-Validation Techniques**:
    - Implement k-fold cross-validation using Scikit-learn.
    - Exercise: Evaluate model performance using k-fold cross-validation on a classification problem.
2. **Hyperparameter Tuning**:
    - Implement Grid Search for hyperparameter tuning.
    - Exercise: Use GridSearchCV to tune a decision tree model on a classification dataset.

Project 10:

- **Title**: Hyperparameter Tuning for Classification Models
- **Description**: Tune the hyperparameters of a random forest model using GridSearchCV to improve performance on a classification dataset (e.g., Titanic or MNIST dataset).

Module 11: Model Deployment and Making an AI Product

Lecture Material:

- Introduction to Model Deployment using Flask/Streamlit
    - Reading: Deploying ML Models with Flask

Code Exercises:

1. **Building a Flask Web App**:
    - Implement a simple Flask app to deploy a machine learning model.
    - Exercise: Deploy a classification model (e.g., a logistic regression model) as a web service using Flask.
2. **Model Monitoring and Updates**:
    - Implement basic monitoring for model performance.
    - Exercise: Build an app using Streamlit to visualize model predictions.

Project 11:

- **Title**: Deploying a Machine Learning Model
- **Description**: Deploy a machine learning model (e.g., house price prediction or image classification) using Flask or Streamlit and create a web app for users to interact with the model.

Module 12: Capstone Project

Capstone Project:

- **Title**: End-to-End Machine Learning Project
- **Description**: Choose a real-world problem (e.g., predicting customer churn, house prices, or sentiment analysis). Apply the entire data science pipeline: data cleaning, EDA, feature engineering, model building, evaluation, and deployment. Present your solution in a final presentation and submit a report.