Scikit-Learn Syllabus

A Machine Learning Library for Training Models

→ Maths Required (Before Phase 0)

Goal: Understand the foundational maths behind ML algorithms

Topics Covered:

- Basic Statistics (mean, median, mode, variance)
- Probability Concepts
- Linear Algebra (vectors, matrices, dot product)
- Derivatives and Gradient
- Distance Metrics (Euclidean, Manhattan)
- Concept of Error and Cost Functions

→ Phase 0: Foundations of Machine Learning

Goal: Understand what Machine Learning is and where it is used

Topics Covered:

- What is Machine Learning?
- Difference: AI vs ML vs Deep Learning
- Types of ML: Supervised, Unsupervised, Reinforcement
- Core ML Concepts:

  → Features (X) and Target (y)

  → Model, training, testing, prediction
- Where Scikit-learn fits into the ML workflow
- Why learn Scikit-Learn for real-world projects

Assignment/Project:

List 3 real-life ML applications and break each into X and y

→ Phase 1: Python, NumPy & Pandas for ML

Goal: Gain basic programming and data handling skills

Topics Covered:

- Python Refresher:

  → Variables, loops, conditions, functions, lists, dictionaries

- NumPy:

  → Creating arrays, indexing, slicing, reshaping
  → Array-level operations

- Pandas:

  → Series vs DataFrame
  → Loading datasets (CSV)
  → Filtering, slicing, subsetting
  → Descriptive statistics: `.head()`, `.info()`, `.describe()`

Assignment/Project:

Load a dataset and explore its structure: shape, head, summary, and apply filters

---

→ Phase 2: Data Preprocessing

Goal: Clean, transform, and prepare data for training

Topics Covered:

- Handling Missing Values:

  → `.dropna()`, `.fillna()`, checking with `.isnull()`

- Encoding Categorical Data:

  → Label Encoding
  → One-Hot Encoding

- Feature Scaling:

  → StandardScaler
  → MinMaxScaler

- Splitting Data:

  → `train_test_split(X, y)`
  → Understanding test_size, random_state, shuffle

Assignment/Project:

Take a CSV file with missing and categorical data → clean, encode, scale, and split

---

→ Phase 3: Supervised Machine Learning

Goal: Train predictive models for classification and regression tasks

Topics Covered:

- Regression:
  → Linear Regression using `LinearRegression()`

- Classification:

  → Logistic Regression
  → K-Nearest Neighbors (KNN)
  → Decision Tree Classifier

- Model Training & Prediction:

  → `.fit(X_train, y_train)`
  → `.predict(X_test)`

- Underfitting vs Overfitting
  → Concepts and visual understanding

Assignment/Project:

1. Predict house prices using size (regression)
2. Predict student pass/fail using study hours (classification)

→ Phase 4: Model Evaluation & Metrics

Goal: Evaluate model performance using appropriate metrics

Topics Covered:

For Classification:

- Accuracy
- Precision
- Recall
- F1 Score
- Confusion Matrix
- `classification_report` and `ConfusionMatrixDisplay`

For Regression:

- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- R² Score

Scikit-learn Modules:

- `sklearn.metrics`

Assignment/Project:

Use classification and regression models and evaluate them using 3+ metrics each

---

→ Phase 5: Unsupervised Learning (Clustering)

Goal: Group unlabeled data using similarity-based learning

Topics Covered:

- K-Means Clustering:
  → `.fit()`, `.predict()`, `.inertia_`
  → Elbow Method to determine `k`

- Principal Component Analysis (PCA):
  → Reducing dimensions
  → `explained_variance_ratio_`, visualizing clusters

- Visualizing clusters using scatter plots

- Cluster Labeling and Interpretability

Assignment/Project:

Cluster customers by Age and Spending Score using Mall Customer dataset

Visualize clusters and apply PCA

→ Phase 6: Model Tuning & Deployment

Goal: Improve model accuracy and save models for reuse or deployment

Topics Covered:

Hyperparameter Tuning:

- `GridSearchCV`
- `RandomizedSearchCV`
- Cross-validation

Model Saving & Loading:

- Using `joblib`
- Using `pickle`

Pipelines:

- Automate preprocessing + modeling
- `Pipeline()`, `ColumnTransformer()`

Assignment/Project:

Tune a KNN or Decision Tree model using GridSearch → save it → reload it → use for prediction on unseen data

---

→ Final Project (Capstone)

Goal: Apply all concepts to a single end-to-end ML problem

- Load raw data
- Clean and preprocess it
- Train multiple models
- Tune the best model
- Evaluate using metrics
- Save and reload final model
- Optional: create a web interface using Streamlit

1. → Data]
2. Suggest a real dataset where supervised learning can be applied.
3. From an online shopping site, list 3 features (X) and 1 output (y) to predict delivery time.
4. Convert a real-world problem into X and y: Predict exam results from hours studied.
5. hich of these tools help in model training? [Scikit-learn, Pandas, Excel, SQL]
6. Which part of ML process does `.predict()` belong to?
7. From the diagram of a model lifecycle (not shown), identify where Scikit-Learn is used.
8. State whether these are input, output, or model: `data`, `model.predict()`, `target`