Machine Learning & NLP – Week 2 Assignments

(NOT A GROUP ASSIGNMENT)

Word Limit for each question: 300.

Goal:
This assignment will help you explore basic concepts in Machine Learning and Natural Language Processing, while learning to use Python libraries like **pandas**, **scikit-learn**, **nltk**, and **matplotlib**.

Section A – Theory Questions (Research + Write-Up)

Write your answers in your own words. Include short examples wherever possible.

1. ML Algorithms in the Wild

- Research: *Decision Trees*, *Linear Regression*, *K-Means*, *Neural Networks*.
- For each:
    - How it works (3–4 lines)
    - One real-world example
    - Name of the **scikit-learn** function/class used to implement it

Hint: Search “scikit-learn decision tree classifier” or “scikit-learn linear regression” in the documentation.

---

2. Model Evaluation Metrics

- Define: *Accuracy*, *Precision*, *Recall*, *F1-score*, *Confusion Matrix*.
- Write the **formula** for each (in simple math form).
- Explain **when each metric is most useful**.

Hint: Look up “precision vs recall tradeoff” in ML tutorials.

---

3. Data Preprocessing in Pandas

Explain how to:

- Load a CSV file into a DataFrame.
- Handle missing values.
- Rename columns.
- Filter rows based on a condition.
  Include the exact **pandas function names** in your answer.

---

4. NLP Basics in NLTK

Explain:

- What is tokenization?
- What is stopword removal?
- What is POS tagging?
  For each, mention the **NLTK function** that can perform it.

Hint: Search for “nltk.tokenize”, “nltk.corpus stopwords”, and “nltk.pos_tag”.

---

5. Matplotlib Visualization Types

List at least **5 different types of plots** in matplotlib.
For each:

- Name of the plot (e.g., scatter plot)
- Name of the matplotlib function (e.g., `plt.scatter`)
- One example use case

---

6. Scikit-learn & NLTK Library Exploration

Part A – Scikit-learn Models
List at least **5 different machine learning models** available in scikit-learn.
For each model:

- **Name of the model** (e.g., Decision Tree Classifier)
- **scikit-learn class/function name** (e.g., `DecisionTreeClassifier`)
- **Type of learning** (Supervised / Unsupervised)
- **One example use case** (e.g., Classifying flower species based on petal measurements)

Part B – NLTK Functions
List at least **5 different NLP tasks** you can perform with NLTK.
For each task:

- **Name of the task** (e.g., Tokenization)
- **NLTK function name** (e.g., `word_tokenize`)
- **One example use case** (e.g., Splitting a sentence into words before analysis)

Section B – Practical Questions (Code + Output)

Write Python scripts for each task. Submit both your **code** and **output screenshots**.

1. CSV Data Exploration & Visualization

Download any small CSV dataset (e.g., Titanic, Iris, Wine Quality). Using **pandas** and **matplotlib**:

1. **Load & Inspect**
    - Load CSV file into a DataFrame.
    - Show first 10 rows.
    - Display shape, column names, and data types.
2. **Summary Statistics**
    - `.describe()` for numeric columns.
    - Count missing values per column.
    - Fill missing numeric values with the mean.
3. **Filter & Sort**
    - Filter rows by a numeric condition (e.g., Age > 30).
    - Sort dataset by a column in descending order.
4. **Group & Aggregate**
    - Group by a categorical column, calculate mean of a numeric column.
5. **Visualize**
    - Create a histogram for a numeric column.
    - Create a bar chart of group averages.

Extra Challenge: Save the cleaned dataset as `processed_data.csv`.

---

2. Decision Tree Classifier

- Use the **Iris dataset** (`load_iris` from scikit-learn).

- Train a **DecisionTreeClassifier**.
- Print the accuracy score.
- Plot the tree using `plot_tree`.

---

3. Text Processing with NLTK

- Take a short paragraph from a news article.
- Tokenize it.
- Remove stopwords.
- POS tag the remaining words.
- Count how many nouns, verbs, and adjectives are in the text.

Hint: Explore `nltk.pos_tag` and `collections.Counter`.

---

4. K-Means Clustering with Visualization

- Generate a random 2D dataset using `make_blobs` (scikit-learn).
- Apply **KMeans** with 3 clusters.
- Plot results using matplotlib, with each cluster in a different color.

---

5. Confusion Matrix Plot

- Train any classifier (Decision Tree, Logistic Regression, etc.) on the Iris dataset.
- Predict on test set.
- Plot confusion matrix using matplotlib or `ConfusionMatrixDisplay`.

📌 **Submission Guidelines:**

- Write answers for **Section A** in a single `.pdf`. (should be handwritten)
- Submit **Section B** as `.py` files and screenshots of output. (attach it to your github repo)
- Mention your dataset source in each practical question. (V.V. Important)