Iris Dataset Classification Using Machine Learning

---

1. Problem Overview and Motivation

The objective of this project is to automate plant species identification using the **Iris dataset.** By applying **supervised machine learning techniques**, flowers are classified into three species—**Iris Setosa, Iris Versicolor, and Iris Virginica**—based on morphological measurements.

This problem demonstrates **pattern recognition in biological data** and serves as a foundational classification task in machine learning. Due to its structured nature and well-separated classes, the Iris dataset is widely used to understand preprocessing, model training, and evaluation techniques.

---

2. Dataset Description and Preprocessing

The dataset is obtained from the **UCI Machine Learning Repository** and consists of **150 samples**, equally distributed among three species.

Each sample contains four numerical features:

- Sepal Length
- Sepal Width
- Petal Length
- Petal Width

To prevent **data leakage**, class labels were separated from the feature matrix before splitting the dataset. An **80/20 train-test split** was applied, resulting in 120 training samples and 30 testing samples. The dataset was verified to contain no missing values, and feature scaling was applied where required.

---

3. Mathematical Formulation

Logistic Regression (Linear Model)

Logistic Regression models the probability of class membership using the sigmoid function:

σ(z)=11+e−z\sigma(z) = \frac{1}{1 + e^{-z}}σ(z)=1+e−z1

where

z=wTx+bz = w^T x + bz=wTx+b

Here, xxx is the input feature vector, www represents the weights, and bbb is the bias.

Random Forest (Non-Linear Model)

Random Forest is an ensemble learning method that combines multiple decision trees. Each tree is trained on a random subset of the dataset using **bootstrap aggregation**, and only a subset of features is considered at each node. The final classification is determined using **majority voting**, enabling the model to handle non-linear decision boundaries.

---

4. Loss Function and Training Process

To optimize classification performance, **Cross-Entropy Loss** was minimized:

L=−∑[ylog(p)+(1−y)log(1−p)]L = - \sum \left[y \log(p) + (1 - y)\log(1 - p)\right]L=−∑[ylog(p)+(1−y)log(1−p)]

Training involved forward propagation, loss computation, and iterative parameter updates. The trained model was evaluated on unseen test data to assess generalization.

---

5. Model Architecture and Justification

Logistic Regression Architecture

- Single-layer linear model

- Weighted sum of input features
- Sigmoid activation function

Random Forest Architecture

- Ensemble of decision trees
- Bagging for variance reduction
- Random feature selection for decorrelation

Using both models enables comparison between **interpretable linear models** and **powerful non-linear models**, strengthening the validity of results.

---

6. Evaluation Methodology and Results

Model performance was evaluated using **accuracy and confusion matrix analysis**.

- **Accuracy achieved: 100%**
- Confusion matrix showed:
  - 10 Setosa samples correctly classified
  - 9 Versicolor samples correctly classified
  - 11 Virginica samples correctly classified
- No misclassifications were observed, indicating strong class separability.

---

7. Limitations and Future Improvements

Despite excellent performance, the dataset size is small and may not represent real-world variability. More complex datasets may introduce noise and overlapping class boundaries.

Future improvements include:

- Cross-validation for robust evaluation
- Hyperparameter tuning
- Feature importance analysis
- Testing on larger biological datasets

Conclusion

This project successfully demonstrates an end-to-end machine learning classification pipeline using both linear and non-linear models. The results confirm the effectiveness of supervised learning techniques and provide a strong foundation for more advanced machine learning applications.