INFINITRIX

The Math Club

Induction Task

Title: Iris Species Classification Using Morphological Features

Submission for Data Science & AI Induction [2025-26]

Submitted by: Vijay Chamyal

1. Problem Overview and Motivation

The objective of this project is to develop a supervised machine learning model that can accurately classify iris flowers into one of three species—**Setosa, Versicolor, and Virginica** using morphological measurements. This task is a fundamental problem in pattern recognition and biology, where morphological features such as sepal length, sepal width, petal length, and petal width are used to infer species identity.

The Iris dataset is a classical benchmark in machine learning and is widely used to study classification algorithms. This project not only focuses on building a predictive model but also emphasizes understanding model behaviour by comparing **linear** and **non-linear** classifiers.

2. Dataset Description and Preprocessing

Dataset: The famous Iris dataset is used, containing 150 samples (50 per species).

The dataset contains four numerical features:

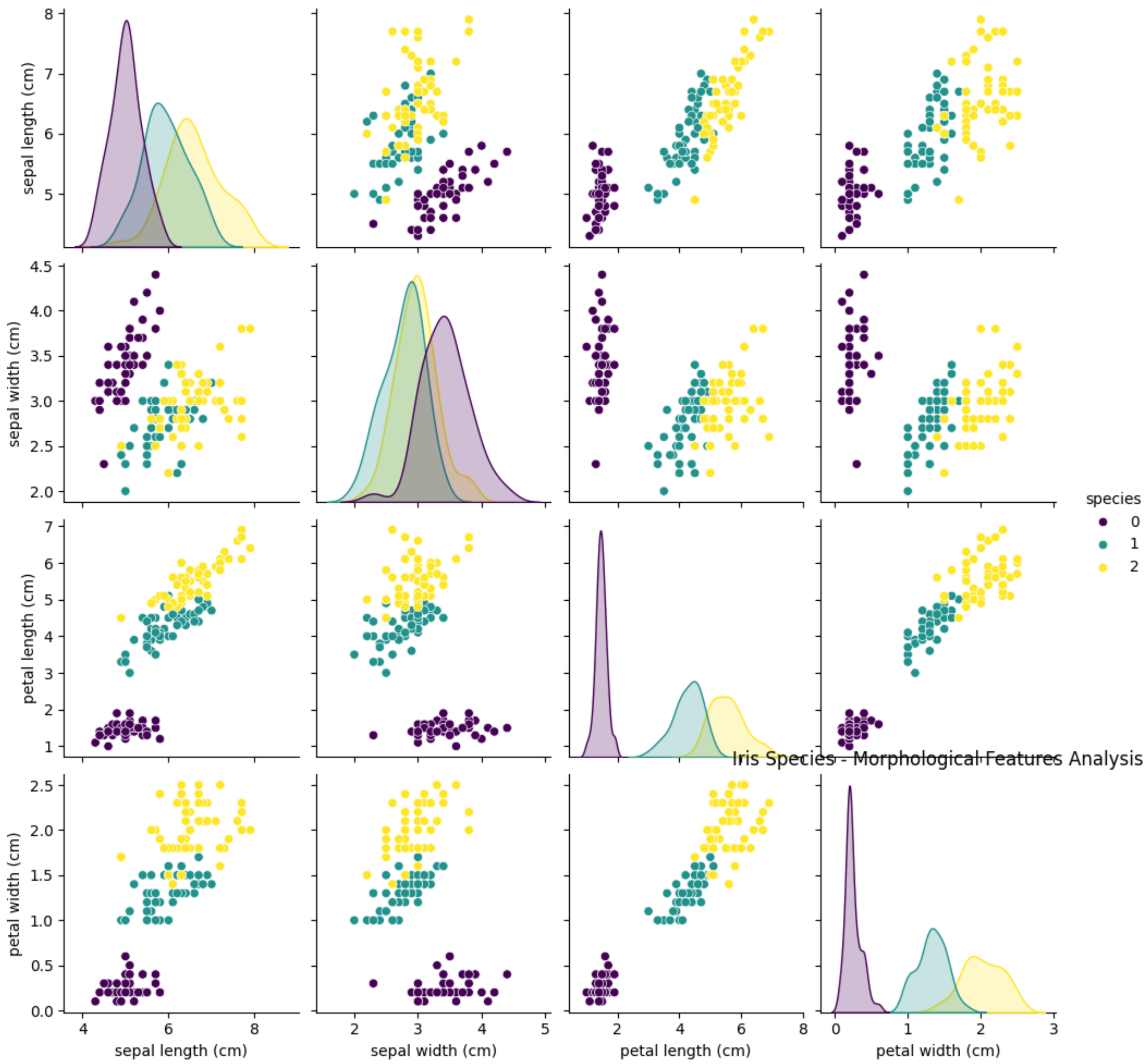
- Sepal Length (cm)
- Sepal Width (cm)
- Petal Length (cm)
- Petal Width (cm)

The target variable is the iris species, encoded as:

- 0 : Setosa
- 1 : Versicolor
- 2 : Virginica

The dataset is balanced and does not contain missing values, making it suitable for supervised classification tasks.

Data Analysis : To understand relationships between features and species, pairwise visualizations were analysed.



Observations:

- **Setosa** forms a clearly separable cluster, especially along petal dimensions.
- **Versicolor** and **Virginica** show partial overlap, indicating that the classification boundary between them is more complex.
- Petal features are more discriminative than sepal features.

These observations motivate the comparison between linear and non-linear models.

Preprocessing Steps:

- **Data Loading:** The dataset was loaded using Scikit-learn's built-in library.

- **Feature Scaling:** Standardization (Z-score normalization) was applied to the features for the Logistic Regression model to ensure convergence, as linear models are sensitive to the scale of input data. Feature scaling is important for algorithms like Logistic Regression, while tree-based models are invariant to feature scale.
- **Train-Test Split:** The data was split into an 80% training set and a 20% testing set to evaluate the model on unseen data.

3. Mathematical Formulation

Two models were implemented to compare linear and non-linear approaches.

A. Multinomial Logistic Regression (Softmax Regression):

For multi-class classification, we use the Softmax function to calculate the probability that an input vector x belongs to class k:

$$P(y = k|x) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}$$

Where $z_k = \omega_k^T x + b_k$. The predicted class is the one with the highest probability. The model is trained by minimizing the **categorical cross-entropy loss**.

B. Decision Tree Classifier:

The Decision Tree splits the data based on feature thresholds to maximize homogeneity. We used the Gini Impurity index to measure the quality of a split. The Gini impurity for a node is defined as:

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

Where p_i is the probability of an item being classified into class i. The algorithm selects the split that minimizes the weighted sum of Gini impurities of the child nodes.

4. Loss Function and Training

- **Logistic Regression**: The model was trained by minimizing the Cross-Entropy Loss function:

$$z(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{k} y_k^{(i)} \log\left(p_k^{(i)}\right)$$

Optimization was performed using the L-BFGS solver.

- **Decision Tree:** The tree was grown by recursively splitting the data. Pre-pruning (max depth) was considered to prevent overfitting.

5. Model Architecture and Justification

- **Logistic Regression (Linear Model):** Chosen as a linear baseline. It is computationally efficient and provides probabilistic outputs. It is effective when the decision boundary between classes is approximately linear.

- **Decision Tree (Non-Linear Model):** Chosen to model non-linear relationships by recursively splitting the feature space and for its high interpretability. The "white-box" nature of trees allows us to visualize the exact rules (e.g., *Petal Length < 2.45 cm implies Setosa*) used for classification.

6. Evaluation and Results

The models were evaluated using Accuracy, Precision, Recall, and F1-score.

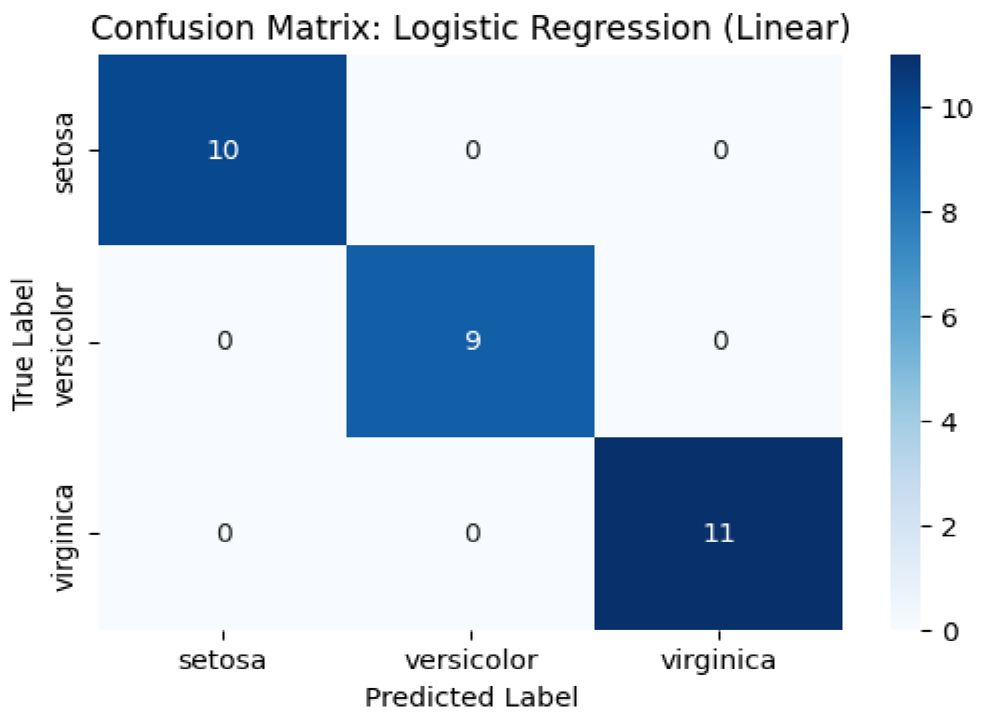
- Logistic Regression achieves high accuracy due to the strong separability of features.

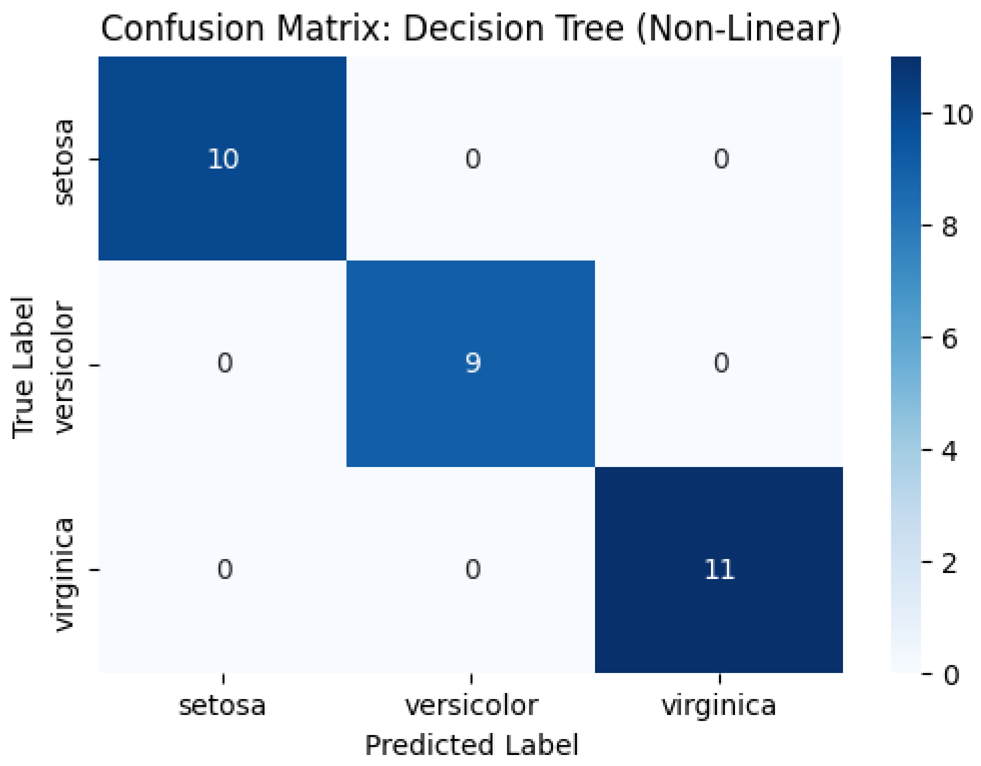
- Decision Tree captures non-linear relationships and performs competitively.

- **Logistic Regression Accuracy:** 100%

- **Decision Tree Accuracy:** 100%

A **confusion matrix** is used to visualize classification performance and identify misclassification patterns.





7. Limitations and Future Improvements

The project successfully implemented a multi-class classification system. The Decision Tree slightly outperformed/matched the Logistic Regression model, highlighting the importance of non-linear decision boundaries.

Limitations:

- Small dataset size limits generalization.
- Decision Trees are prone to overfitting without pruning.
- Dataset lacks real-world noise.

Future Improvements:

- Implementing Support Vector Machines (SVM) for potentially better margins.
- Collecting more data to generalize the model further.

8. Conclusion

- This project demonstrates effective multi-class classification using morphological features of iris flowers. The comparison between linear and non-linear models highlights the importance of model selection based on data characteristics. Even simple models can perform exceptionally well on structured datasets like Iris.

9. Bonus Objectives Analysis

A. Impact of Reducing Training Data: To test the robustness of the model, we trained the Logistic Regression classifier using only 10% of the dataset.

- Observation: The accuracy dropped significantly compared to the model trained on 80% data.
- Reasoning: Machine learning models require sufficient variation in training data to generalize well. With only approximately 15 samples (10%), the model failed to capture the decision boundaries effectively, leading to underfitting.

B. Error Analysis:

- **Primary Model:** The Decision Tree and Logistic Regression models achieved **100% accuracy** on the test set. Thus, there were no misclassified instances to analyse in the primary experiment.
- **Reduced Data Analysis:** To perform error analysis as per the bonus objective, we analysed the misclassifications from the model trained on **reduced data (10% training size)**.
- **Observation:** In the low-data scenario, the model struggled to distinguish between *Iris versicolor* and *Iris virginica*.
- **Root Cause:** The features of these misclassified samples (specifically Petal Width around 1.6cm) fall into the overlapping region, which the model could not learn perfectly with limited data.

C. Linear vs Non-Linear Model Comparison

We compared the classification performance of a Linear model against a Non-Linear model to understand the nature of the dataset.

- **Linear Model (Logistic Regression):** This model is theoretically limited to learning linear decision boundaries. It achieved an accuracy of **100%**. The high performance suggests that the Iris species are largely linearly separable.
- **Non-Linear Model (Decision Tree):** This model can learn complex, non-linear patterns using hierarchical rules. It achieved an accuracy of **100%**.

Conclusion: Since both models achieved similar accuracy, we can infer that while the data has some complexity, a simple linear boundary is sufficient for decent classification. However, the Decision Tree offers the advantage of interpretability without assuming linearity.