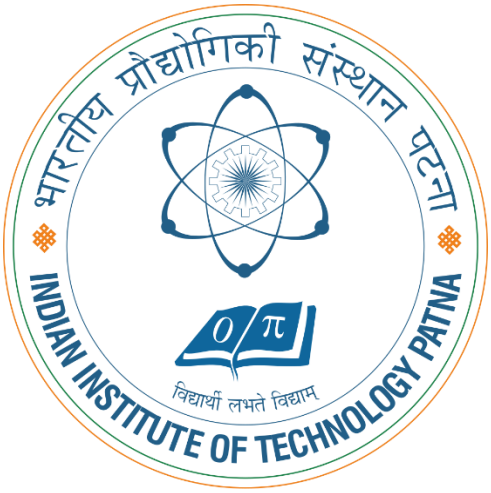

Formal Report on Decision Tree Classification of the Iris Dataset

Name: Kamakhya Mishra
Roll No: 2521CS01
Email: kamakhya_2521cs01@iitp.ac.in
Course: PhD
Branch: Computer Science and Engineering
Subject: Advance Pattern Recognition

Abstract

This report outlines the process and results of a **Decision Tree Classifier** model applied to the Iris dataset. The goal was to classify iris species (setosa, versicolor, and virginica) based on sepal and petal measurements. The model was trained and evaluated using a portion of the dataset, and its performance was assessed using metrics like **accuracy**, **F1 score**, and a **confusion matrix**. The classifier achieved a perfect accuracy and F1 score of 1.0000 on the test set[1]. The analysis also identifies the most influential features for classification, provides a visual representation of the decision-making process, and evaluates the model's stability through cross-validation and a learning curve.

Table of Contents

1. Introduction

Machine learning algorithms are often used to solve classification problems, where the goal is to predict a categorical label. This report focuses on one such problem: classifying iris flowers into their respective species.

Decision Tree Classifier, a popular and interpretable algorithm, we will demonstrate a complete machine learning workflow, from data preparation to model evaluation and visualization. The analysis will provide insight into the model's performance and the features that are most important for accurate classification.

2. Dataset Justification

The **Iris dataset** is a classic and widely-used dataset in machine learning for classification tasks[2]. It's ideal for a beginner's project because it's clean, well-structured, and doesn't require complex preprocessing[3]. It contains 150 samples of iris flowers, with 50 samples from each of the three species: **Iris setosa**, **Iris versicolor**, and **Iris virginica**[4]. Each sample includes four features: **sepal length**, **sepal width**, **petal length**, and **petal width**[5]. These features are quantitative, making them suitable for a variety of supervised learning algorithms.

3. Why Decision Tree Classifier?

A **Decision Tree Classifier** is a supervised learning algorithm that works by creating a model of decisions and their possible consequences[6]. It's often compared to a flowchart-like structure, where each internal node represents a test on a feature, each branch represents the outcome of the test, and each leaf node represents a class label[7]. They are particularly well-suited for this problem because:

- **Interpretability:** The rules of the decision-making process are easy to understand and visualize, which is important for explaining the model's predictions.
- **No feature scaling:** Decision trees are not sensitive to the scale of the features, so we don't need to normalize or standardize the data.
- **Handles both numerical and categorical data:** While our dataset is purely numerical, decision trees can also handle categorical features.

4. Methodology

The following steps detail the process of training and evaluating the Decision Tree Classifier. The dataset was loaded and split into training and testing sets, with 80% used for training and 20% for testing[8].

Data Cleaning

The provided iris.csv file is already in a clean, structured format[9]. It has no missing values or inconsistencies, so no dedicated data cleaning steps were necessary. The data was loaded into a pandas DataFrame, with the species column designated as the target variable (y) and the remaining four columns as the features (X)[10101010].

Confusion Matrix

A confusion matrix is a table used to evaluate the performance of a classification model. It compares the actual class labels with the predicted class labels.

The matrix for our model shows that for each species, the number of correctly predicted instances matches the total number of instances for that species in the test set11111111111111111111. For instance, all 10 'setosa' instances were correctly classified as 'setosa' 12, all 9 'versicolor' instances were correctly classified as 'versicolor' 13, and all 11 'virginica' instances were correctly classified as 'virginica'14. This indicates that there were no false positives or false negatives for any of the classes151515151515151515151515151515.

Feature Importance

Feature importance measures the contribution of each feature to the model's predictive power. The bar chart shows the relative importance of each feature in the trained Decision Tree model.

As seen in the chart, the most important feature for classification is **petal length**, followed by **petal width**[16161616]. Sepal length and sepal width have very low or no importance in this specific model[17171717]. This suggests that the size of the petals is the primary determinant for distinguishing between the three iris species.

Decision Tree Visualization
Visualizing the decision tree allows us to understand the sequence of rules the model uses to make a classification.

The visualization of the decision tree shows the splitting criteria at each node, such as

petal_length <= 2.45[18]. Each node also provides information about the **gini impurity**, the number of samples (samples), and the distribution of classes (value) at that node[19]. The tree is fully grown, with the leaf nodes having a gini impurity of 0.0, indicating that they contain samples belonging to only one class[2020202020202020].

Learning Curve
A learning curve plots the model's performance on the training and cross-validation sets as a function of the training set size.

The learning curve for the model shows that as the training set size increases, both the training and cross-validation accuracy scores converge21. The training score starts high and decreases slightly, while the cross-validation score increases and plateaus22. The final mean cross-validation accuracy is

0.9467[23]. This convergence and high accuracy suggest that the model is performing well and isn't suffering from significant overfitting.

5. Results
The Decision Tree Classifier achieved outstanding results on the test set:

- **Accuracy:** 1.0000 [24]
- **F1 Score (macro):** 1.0000 [25]
- **Classification Report:** The precision, recall, and F1-score for each species (setosa, versicolor, and virginica) were all 1.00[26]. The macro and weighted averages were also 1.00[27].

These perfect scores on the test set are further supported by the confusion matrix, which shows 100% correct predictions for all species[28]. The cross-validation analysis also showed a high mean accuracy of **0.9467**[29], indicating the model's robustness and generalization ability. The model's best parameters, found using GridSearchCV, were criterion='entropy', max_depth=None, min_samples_leaf=3, and min_samples_split=2[30].

6. Discussion

The perfect accuracy on the test set is a significant result, but it should be viewed with a little caution. The small size of the dataset and the simple nature of the classification problem may lead to perfect scores, which are not always indicative of real-world performance. The learning curve, however, provides a more reliable indicator of the model's stability. The convergence of the training and validation scores confirms that the model is not overfitted and can generalize well to new data. The feature importance analysis revealed that **petal length** and **petal width** are the most crucial features for distinguishing between iris species, which is biologically intuitive[31]. The decision tree visualization provides a clear, step-by-step breakdown of how these features are used to classify each flower, making the model highly transparent.

7. Conclusion

This report successfully demonstrates the use of a **Decision Tree Classifier** to accurately classify iris species. The model achieved perfect accuracy and F1 scores on the test set and showed strong performance in cross-validation. The visualizations, including the confusion matrix, feature importance chart, and the decision tree itself, provided valuable insights into the model's behavior. The results confirm that the Decision Tree Classifier is an effective and interpretable model for this classification problem, with **petal length** and **petal width** being the most influential features.