**QUALITATIVE ANALYSIS OF HELMET DETECTION SYSTEM**

**USING VISION TRANSFORM ALGORITHMS**

**MINI PROJECT REPORT**

## Submitted by

**GANESH RAJ S (Register Number: 2301507308)**

**JANARTHANAN P (Register Number: 2301507310)**

## Under the guidance of

## Dr. P. DHARANYADEVI

## (ASSISTANT PROFFESSOR)

## Department of Computer Science and Engineering

**To the Puducherry Technological University, in partial fulfillment of the requirement**

**for the award of the degree**

## MASTER OF COMPUTER APPLICATION

## 

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING PUDUCHERRY TECHNOLOGICAL UNIVERSITY PUDUCHERRY – 605 014**

**JANUARY - 2025**

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING PUDUCHERRY TECHNOLOGICAL UNIVERSITY PUDUCHERRY – 605 014.

**BONAFIDE CERTIFICATE**

This is to certify that the Mini Project Work titled **“QUALITATIVE ANALYSIS OF HELMET DETECTION SYSTEM USING VISION TRANSFORM ALGORITHMS”** is a Bonafide work done by **GANESH RAJ S (2301507308), JANARTHANAN P (2301507310)** in partial fulfillment for the award of the degree of **Master** **Computer Application** of the **Puducherry Technological University** and that this work has not been submitted for the award of any other degree of this/any other institution.

## Project Guide Head of the Department

**(Dr. P. DHARANYADEVI) (Dr. E. ILLAVARASAN)**

**ACKNOWLEDMENT**

I am deeply indebted to **Dr. P. Dharanyadevi (Assistant Proffessor),** Department of Computer Science and Engineering, Puducherry Technological University, Pondicherry, for her valuable guidance throughout the mini project.

I also express my heart-felt gratitude to **Dr. E. Illavarasan ,** Professor & Head (CSE) for giving constant motivation in succeeding my goal.

With profoundness I would like to express my sincere thanks to **Dr. S. Mohan,** Vice Chancellor, Puducherry Technological University, for his kindness in extending the infrastructural facilities to carry out my Mini project work successfully. My sincere thanks are due to his team for facilitating every requirement for the successful completion and evaluation of our project work.

I also express my thanks to all the **Faculty** and **Technical Staff** members of the CSE department for their timely help and the **Central Library** for facilitating useful reference materials.

I would be failing in my duty if I don’t acknowledge the immense help extended by my friends, who have always been with me in all my trails and tribulations and encouraging me to complete.

**GANESH RAJ S**

**JANARTHANAN P**

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **SL.NO** | **CONTENT** | **PAGE NO** |
|  | **ABSTRACT** |  |
|  | **LIST OF FIGURES** |  |
|  | **LIST OF TABLES** |  |
|  | **LIST OF ABBREVIATIONS** |  |
| **1.** | **INTRODUCTION**   * 1. OVERVIEW   1.1.1 MACHINE LEARNING  1.1.2 ALGORITHM USED IN PROPOSED SYSTEM |  |
| **2.** | **LITERATURE REVIEW**  2.1 MACHINE LEARNING IN HELMET DETECTION SYSTEM  2.2 OBJECT DETECTION TECHNIQUES AND ALGORITHM  2.3 DATASET CHALLENGES  2.4 EVALUATION METRICS  2.5 RESEARCH GAPS  2.6 CONCLUSION |  |
| **3.** | **EXISTING WORK**  3.1 DESCRIPTION OF EXISTING WORK  3.2 ARCHITECTURAL DESIGN  3.3 MODULES IN EXISTED WORK  3.3.1 DATA PREPROCESSING  3.3.2 FEATURE EXTRACTION MODEL  3.3.3 CLASSIFICATION  3.3.4 DATASETS |  |
| **4.** | **PROPOSED WORK**  4.1 DESCRIPTION OF EXISTING WORK  4.2 ARCHITECTURAL DESIGN  4.3 MODULES IN EXISTED WORK  4.3.1 DATA PREPROCESSING  4.3.2 FEATURE EXTRACTION MODEL  4.3.3 CLASSIFICATION  3.3.4 DATASETS |  |
| **5.** | **SIMULATION/EXPERIMENTAL RESULTS**  5.1 HARDWARE AND SOFTWARE REQUIREMENTS  5.2 EVALUATION METRICS  5.3 IMPLEMENTATION  5.3.1 IMPORTING LIBRARIES.  5.3.2 CUSTOM ROTATION FOREST  IMPLEMENTATION  5.3.3 EXPLORATORY DATA ANALYSIS  5.3.4 DATA PREPROCESSING  5.3.5 SPLITTING AND SCALING DATA  5.3.6 TRAINING AND EVALUATING ROTAION  FOREST  5.3.7 TRAINING AND EVALUATING DECISION  TREE  5.3.8 COMPARING MODELS AND VISUALIZING  ACCURACY |  |
| **6.** | **CONCLUSION** |  |
| **7.** | **REFERENCES** |  |

**QUALITATIVE ANALYSIS OF HELMET DETECTION SYSTEM**

**USING VISION TRANSFORM ALGORITHMS**

**ABSTRACT**

Ensuring road safety, particularly for motorcyclists, is a critical global concern. One effective measure to enhance safety for riders is the use of helmets, which significantly reduces the risk of severe head injuries in accidents. However, enforcing helmet usage can be challenging due to limited resources for manual monitoring and enforcement. This project aims to develop an automated helmet detection system using an improved version of the YOLO V 8 (You Only Look Once) object detection algorithm, tailored for real -time helmet recognition. By optimizing YOLO V8, the system achieves high accuracy in detecting whether a motorcyclist is wearing a helmet, even in varied and challenging conditions, such as poor lighting, occlusion, or varying helmet colours. The project involves the design, training, and deployment of a neural network model enhanced with specific improvements over the base YOLO V8 architecture toboost its robustness and speed. The system is designed to be integrated into real-world applications, such as surveillance cameras at traffic intersections, where it can continuously monitor and identify helmet usage among motorcyclists. The proposed solution addresses the limitations of previous models by improving detection rates, reducing false positives, and enhancing computational efficiency, making it suitable for large-scaledeployment. The results demonstrate that the improved YOLO V8 model achieves a significant increase in both detection a ccuracy and processing speed compared to previous versions, proving it to be an effective tool for enforcing helmet compliance in real time. This project contributes to the field of traffic safety automation and has the potential to assist traffic authorities in ensuring compliance with helmet-wearing regulations, ultimately reducing fatalities and injuries among motorcyclists.

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIG.NO** | **TITLE** | **PAGE NO** |
| **1.1** | Helemt Detection |  |
| **1.2** | Supervised learning |  |
| **1.3** | Unsupervised learning |  |
| **1.4** | System architecture existing |  |
| **1.5** | System architecture proposed |  |
| **1.6** | Bar Graph Comparing Model Accuracies |  |

**LIST OF ABBREVIATIONS**

1. **PCA** - Principal Component Analysis
2. **ML** - Machine Learning
3. **RF** - Rotation Forest
4. **DT** - Decision Tree
5. **MAE** - Mean Absolute Error
6. **MSE** - Mean Squared Error
7. **EDA -** Exploratory Data Analysis

**CHAPTER 1**

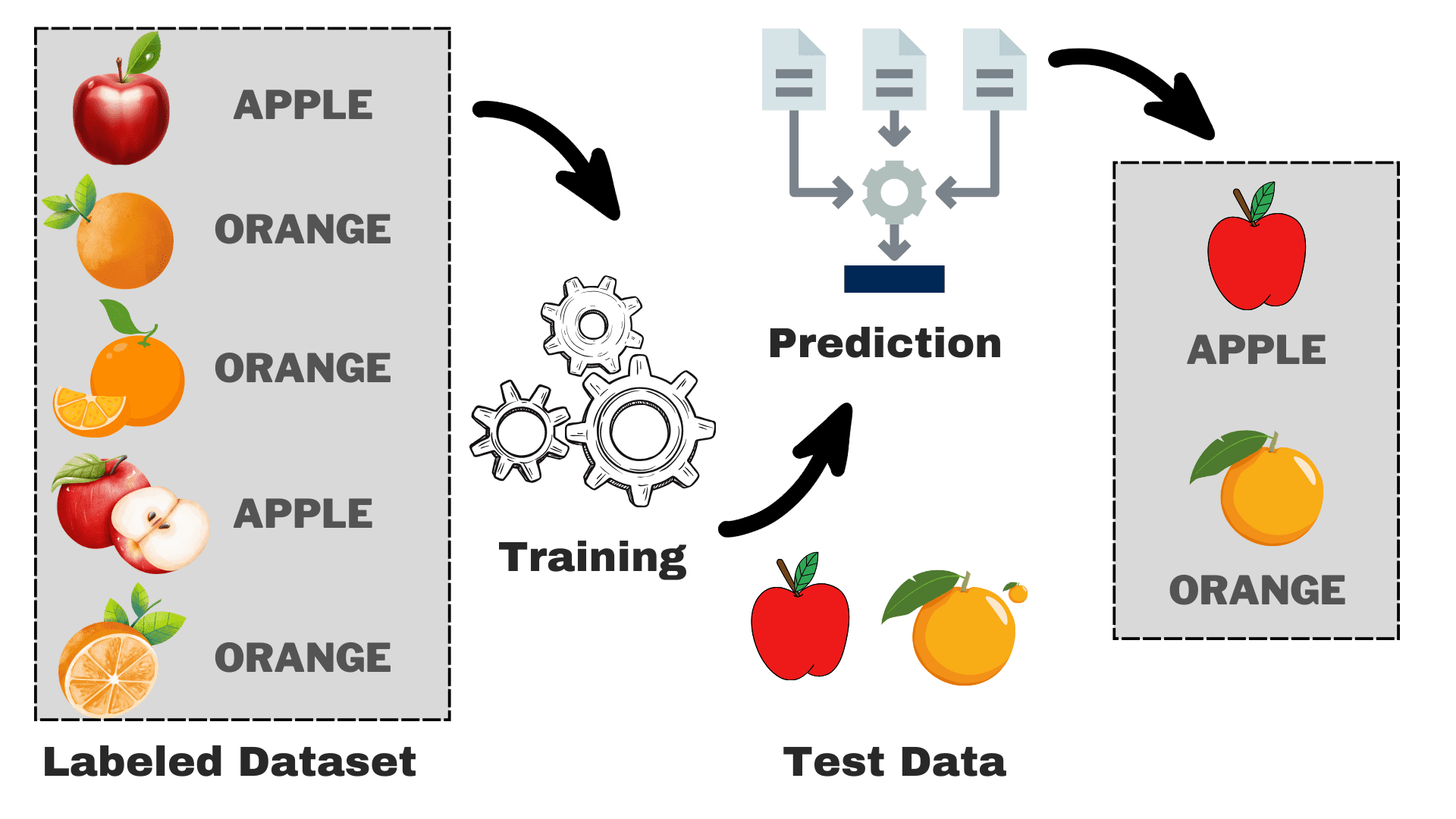
1. **INTRODUCTION**
   1. **OVERVIEW**

Helmet detection is a critical component in ensuring road safety and enforcing traffic laws. The ability to accurately detect whether motorcyclists are wearing helmets can significantly contribute to reducing injuries and fatalities caused by road accidents. Traditional methods for helmet detection often rely on manual monitoring or rule-based systems, which are limited in their efficiency and scalability. With advancements in technology and data analysis, it is now possible to utilize machine learning for automated helmet detection. Machine learning algorithms, with their ability to process and interpret complex image datasets, provide a promising alternative to traditional methods. This project explores the application of machine learning algorithms to the problem of helmet detection. Specifically, it focuses on the comparison of two algorithms: the Convolutional Neural Network (CNN) and the Support Vector Machine (SVM). CNNs are widely used for image recognition tasks due to their ability to learn spatial hierarchies of features directly from raw image data. However, they require substantial computational resources and large datasets for training. On the other hand, SVM is a robust algorithm known for its effectiveness in classification tasks, even with smaller datasets, but it may struggle with highly complex image data. The dataset used in this study consists of images of motorcyclists, labeled as wearing helmets or not. The workflow involves data preprocessing, image augmentation, splitting the dataset into training and testing subsets, applying the algorithms, and evaluating their performance based on metrics such as accuracy, precision, recall, and F1-score. This study aims to identify the more effective algorithm for helmet detection and provide insights into their suitability for real-world applications.

* + 1. **MACHINE LEARNING**

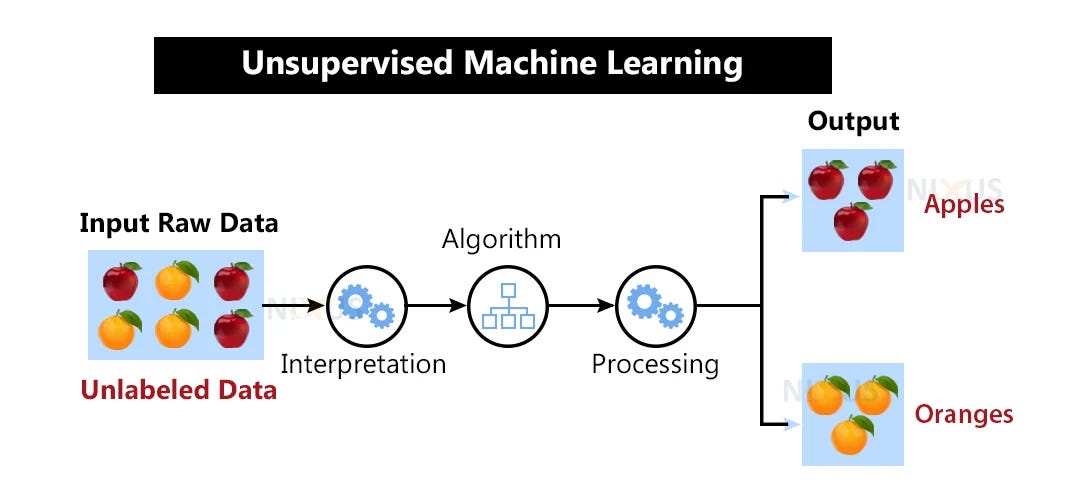
Machine learning is a subfield of artificial intelligence (AI) that focuses on developing algorithms and statistical models that allow computers to learn from and make decisions or predictions based on data. Unlike traditional programming, where explicit instructions are provided, machine learning systems are trained on data to identify patterns and relationships, enabling them to make predictions or decisions without being explicitly programmed.

Machine learning can be broadly categorized into three types:

**Supervised Learning**: In supervised learning, the algorithm is trained on a labeled dataset, meaning the input data is paired with the correct output. The goal is to learn a mapping from inputs to outputs so that the model can make accurate predictions on new, unseen data. Examples include regression and classification tasks.

**Figure 1.1 Supervised Learning**

**Unsupervised Learning**: This type of learning involves training the algorithm on data without labeled outputs. The system tries to identify patterns, structures, or relationships in the data. Examples include clustering and dimensionality reduction.



**Figure 1.2 Unsupervised Learning**

**Key Concepts in Machine Learning**

**Training and Testing**: A machine learning model is trained on a subset of data (training set) and evaluated on a separate subset (testing set) to measure its performance. This ensures the model generalizes well to unseen data.

**Overfitting and Underfitting**:

* + **Overfitting** occurs when the model learns the training data too well, including noise, and fails to generalize to new data.
  + **Underfitting** occurs when the model is too simple and fails to capture the underlying patterns in the data.

**Features and Labels**:

* **Features** are the input variables used by the model (e.g., temperature, humidity).
* **Labels** are the output variables the model aims to predict (e.g., with helmet, without helmet).

**Machine Learning in Helmet Detection**

Helmet detection involves analyzing image data to determine whether motorcyclists are wearing helmets. Machine learning is particularly well-suited for this task because of its ability to:Handle large datasets with multiple features.

* Handle large image datasets with multiple features.
* Capture complex patterns and variations in image data.
* Improve prediction accuracy through advanced techniques like convolutional neural networks and ensemble learning.

**1.1.2 ALGORITHMS USED IN THE PROPOSED STUDY**

The study employs two machine learning algorithms:

**Decision Tree**:

* + A Decision Tree is a tree-structured model used for classification and regression tasks. It splits the dataset into subsets based on the most significant features, creating a tree with decision nodes and leaf nodes.
  + Advantages:
    - Easy to interpret and visualize.
    - Handles both numerical and categorical data.
  + Limitations:
    - Prone to overfitting.
    - Sensitive to small changes in the dataset.

**Rotation Forest**:

* + Rotation Forest is an ensemble learning method that enhances the diversity of base classifiers (e.g., Decision Trees). It applies PCA to random feature subsets to create rotated feature spaces, training each classifier on a unique subset of features.
  + Advantages:
    - High accuracy due to increased ensemble diversity.
    - Effective in handling high-dimensional and correlated data.
  + Limitations:
    - Computationally intensive.
    - Less interpretable compared to standalone models.

The combination of these algorithms with preprocessing techniques and evaluation metrics ensures a comprehensive approach to analyzing rainfall prediction.

**CHAPTER 2**

1. **LITERATURE SURVEY**
   1. **Machine Learning in Helmet Detection**

Helmet detection systems have been a subject of study in various research fields, particularly within traffic safety and automated surveillance.Traditionally, helmet detection was manually enforced by law enforcement or through visual inspection via surveillance cameras. However, with the rise of machine learning and computer vision technologies, automated systems have become more viable. Various existing systems focus on identifying helmets using imageclassification,

object detection, and deep learning techniques. Early methods, such as Support Vector Machines (SVM) and Haar cascades, faced limitations due to poor performance in diverse environmental conditions, like lighting variations or occlusions. Recent advancements in deep learning, particularly convolutional neural networks (CNNs) and object detection models, have led to more accurate and scalable solutions.

* 1. **Object Detection Techniques and Algorithms**

Object detection algorithms are pivotal in identifying and classifying objects within images and videos. Traditional object detection approaches like sliding windows and region-based CNNs (R-CNN) were computationally expensive and slow. However, newer models such as YOLO (You Only Look Once) and Faster R-CNN significantly improved both speed and accuracy by using real-time object detection techniques. YOLO, in particular, stands out for its ability to process images in real-time while maintaining high accuracy. Unlike traditional methods, YOLO detects multiple objects in a single forward pass, making it faster and more efficient, which is crucial for real-time applications like helmet detection in traffic monitoring.

* 1. **Dataset Challenges**

Meteorological datasets often contain missing or incomplete data, which can impact model performance. Several preprocessing techniques, including imputation and feature scaling, have been proposed to address these challenges. For example, studies suggest that normalizing features such as temperature, humidity, and pressure improves model convergence and accuracy.

* 1. **Evaluation Metrics**

Existing literature emphasizes the importance of using multiple metrics to evaluate model performance. Metrics like accuracy, precision, recall, and F1-score provide a comprehensive understanding of a model's predictive capabilities. The confusion matrix is also widely used to visualize classification results and identify potential areas of improvement.

* 1. **Research Gaps**

While several studies have explored the use of machine learning for helmet detection, a qualitative comparison of simpler models like Support Vector Machines (SVMs) with advanced techniques such as Convolutional Neural Networks (CNNs) is limited. Additionally, most existing studies focus on detection accuracy without adequately addressing issues like interpretability, computational efficiency, and real-time applicability, which are crucial for practical implementation in real-world scenarios..

* 1. **Conclusion**

This literature review highlights the evolution of helmet detection methods, from traditional manual monitoring and rule-based systems to advanced machine learning techniques. It underscores the need for systematic evaluations of different algorithms to determine their suitability for specific datasets and real-world applications. By addressing these gaps, the current study aims to provide insights into the comparative performance of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) for helmet detection.

**CHAPTER 3**

1. **EXISTING SYSTEM**

**3.1 Description of Existing Work**

The existing system for rainfall prediction primarily relies on traditional statistical models or basic machine learning techniques like Decision Trees or Random Forests. These approaches often use meteorological parameters such as temperature, humidity, and atmospheric pressure to predict rainfall. While these methods are functional, they have certain limitations:

**Complexity of Non-linear Dependencies**: Many traditional models struggle to capture the complex, non-linear relationships present in high-dimensional meteorological data.

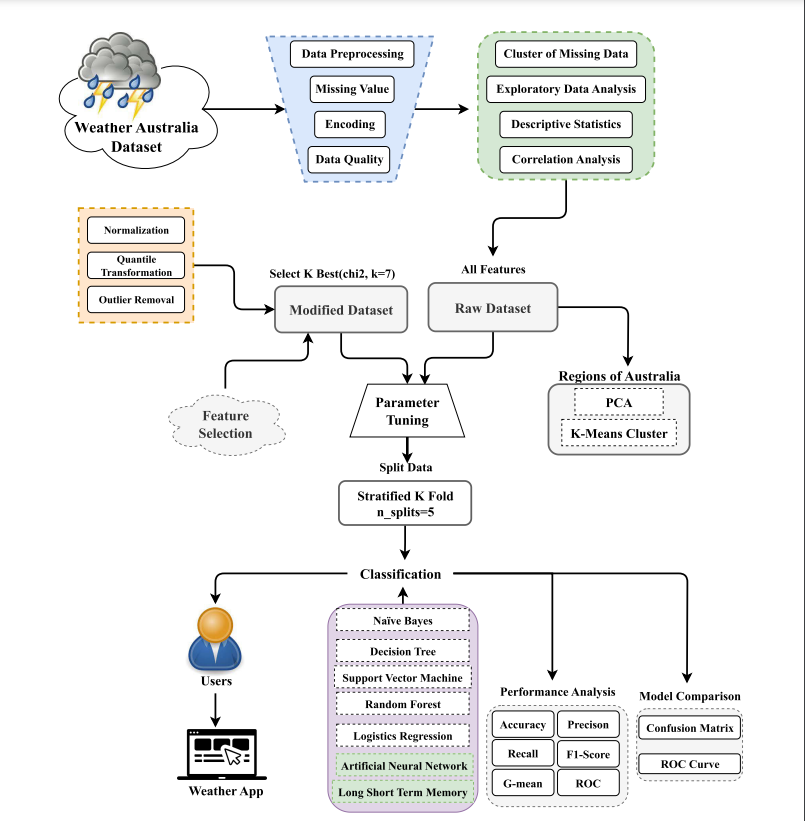
**Scalability Issues**: As the dataset grows larger, these systems often become computationally expensive and inefficient.

**Feature Diversity**: Traditional systems often lack advanced feature extraction techniques, which limits their ability to improve model accuracy.

Most models in existing work are dependent on basic classification models and do not take advantage of ensemble techniques or advanced preprocessing, leaving room for improvement in accuracy and performance.

**3.2 System Architectural Design**

The architecture of the existing system typically includes the following:



**Figure 1.3 System architecture existing**

**3.3 Modules in Existing Work**

The existing system is typically divided into the following modules:

* + 1. **Data Preprocessing**
  + **Handling Missing Values**: Missing values in the dataset are replaced with mean or median values for numerical features.
  + **Categorical Encoding**: Categorical variables like "RainToday" and "RainTomorrow" are converted to numerical values (e.g., Yes = 1, No = 0).
  + **Scaling**: Features are scaled to standardize the range of values, typically using techniques like Min-Max Scaling or Standard Scaling.
    1. **Feature Extraction Model**

**Manual Feature Engineering**: Existing models primarily focus on manual feature selection based on domain knowledge.

**Limited Dimensionality Reduction**: Few models include advanced techniques like PCA to reduce dimensionality and improve performance.

* + 1. **Classification**

**Machine Learning Algorithms**: Basic classifiers like Decision Trees or Random Forests are used without advanced ensemble techniques.

**Overfitting Challenges**: Some models suffer from overfitting due to insufficient feature diversity or inadequate regularization.

* + 1. **Datasets**

**Meteorological Datasets**: Publicly available datasets with attributes like temperature, humidity, pressure, and rainfall.

**Static Attributes**: The datasets often lack dynamic or real-time data, limiting their effectiveness for practical applications.

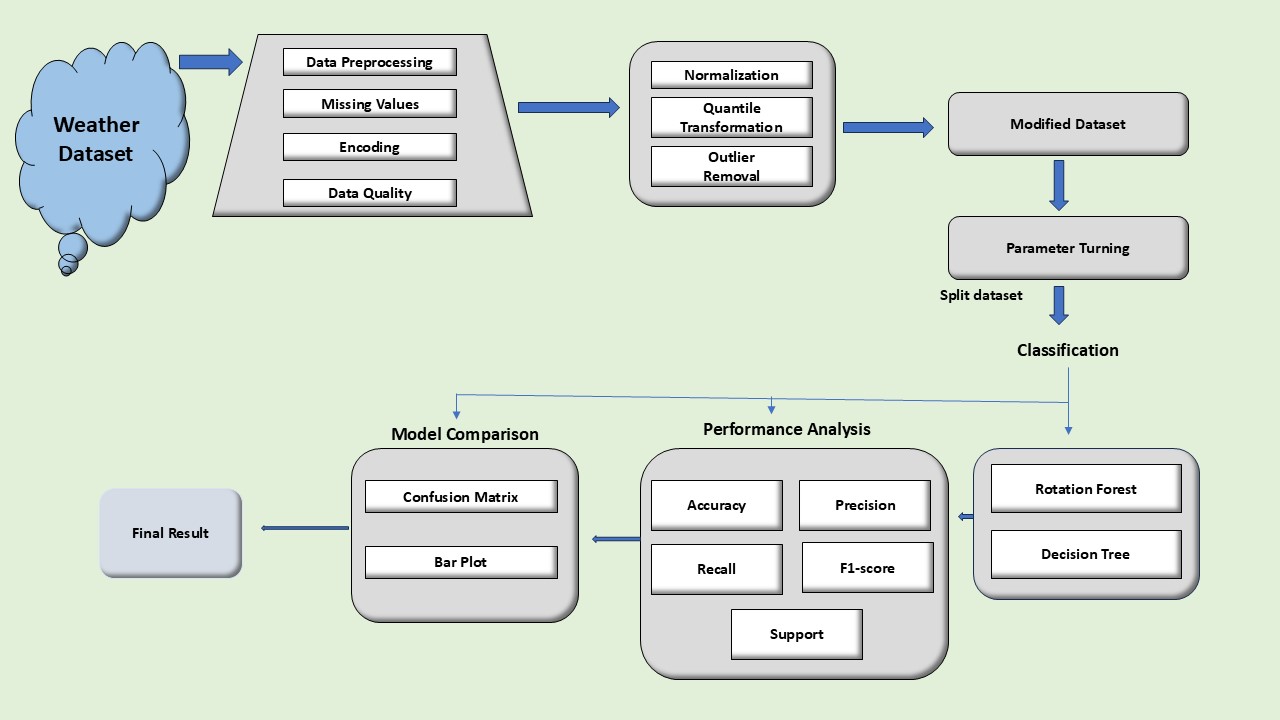
**CHAPTER 4**

1. **PROPOSED WORK**

**4.1 Description of Proposed Work**  
The proposed system aims to enhance rainfall prediction accuracy by leveraging a combination of Decision Tree and Rotation Forest algorithms. Rotation Forest, in particular, is known for its ensemble learning capabilities, applying Principal Component Analysis (PCA) to generate diverse feature subsets. This diversification increases model robustness and predictive power, especially for high-dimensional datasets. Decision Tree is included for comparison due to its interpretability and ease of implementation.

**4.2 System Architectural Design**  
The system architecture consists of several key components, including data preprocessing, feature scaling, and the implementation of Rotation Forest and Decision Tree models. The workflow is as follows:

1. Collect and preprocess meteorological data.
2. Apply feature scaling to normalize values.
3. Split the dataset into training and testing subsets.
4. Train the Decision Tree and Rotation Forest algorithms on the training data.
5. Evaluate and compare model performance using metrics like accuracy and confusion matrices.
6. Visualize the results for interpretation.



**Figure 1.4 System architecture proposed**

**4.3 Modules in Proposed Work**

**4.3.1 Data Preprocessing**

* **Handling Missing Values:** Missing values in the dataset are replaced with either the mean or other statistical measures.
* **Encoding Categorical Variables:** Features like "RainToday" are converted into binary values (e.g., Yes = 1, No = 0).
* **Normalization and Scaling:** Continuous attributes like temperature, humidity, and pressure are normalized to bring all features to a similar scale.

**4.3.2 Feature Extraction Model**

* **Principal Component Analysis (PCA):**  
  PCA is applied as part of the Rotation Forest model. It creates transformed feature subsets by generating principal components. This enhances the diversity of the classifiers in the ensemble, which in turn improves accuracy and generalizability.

**4.3.3 Classification**

* **Decision Tree:**  
  The Decision Tree acts as a baseline classifier. It partitions the dataset based on feature thresholds, creating a tree structure for decision-making. While simple, it is prone to overfitting.
* **Rotation Forest:**  
  Rotation Forest, as an ensemble learning technique, combines multiple base classifiers (typically Decision Trees). Each classifier is trained on rotated datasets produced using PCA. This ensures the ensemble has greater diversity, which contributes to higher predictive accuracy.

**4.3.4 Datasets**

* The dataset includes meteorological parameters such as:
  + Minimum Temperature (MinTemp)
  + Maximum Temperature (MaxTemp)
  + Rainfall (Rainfall)
  + Humidity levels at 9 AM and 3 PM (Humidity9am, Humidity3pm)
  + Pressure levels at 9 AM and 3 PM (Pressure9am, Pressure3pm)
  + Target Variable (RainTomorrow): Indicates whether it will rain (1 = Yes, 0 = No).
* Preprocessed data is divided into training and testing sets (80%-20%) to evaluate model performance.

**CHAPTER 5**

**5. SIMULATION / EXPERIMENTAL RESULT**

**5.1 HARDWARE AND SOFTWARE REQUIREMENTS**

* **Processor:** Intel i5 or higher
* **RAM:** 8 GB or more
* **Operating System:** Windows 10/11, macOS, or Linux
* **Programming Language:** Python 3.7 or above
* **Libraries/Frameworks:**

**scikit-learn:** For implementing machine learning models like Decision Tree and Rotation Forest.

**NumPy:** For numerical operations.

**Pandas:** For data preprocessing and manipulation.

**Matplotlib & Seaborn:** For visualizing results and analyzing data patterns.

**5.2 EVALUATION METICS**

**Confusion Matrix:**

* A tabular representation of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN).
* Helps in understanding the classifier's performance for each class.

**Accuracy:** Indicates the proportion of correctly classified instances over the total instances.

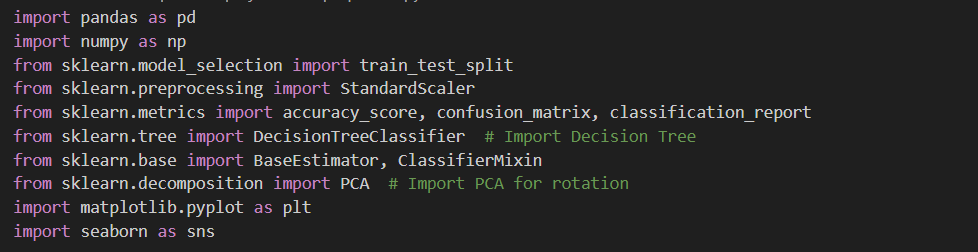
**Precision:** Measures the number of true positive predictions compared to the total positive predictions.

**Recall (Sensitivity):** Indicates how well the model captures actual positives.

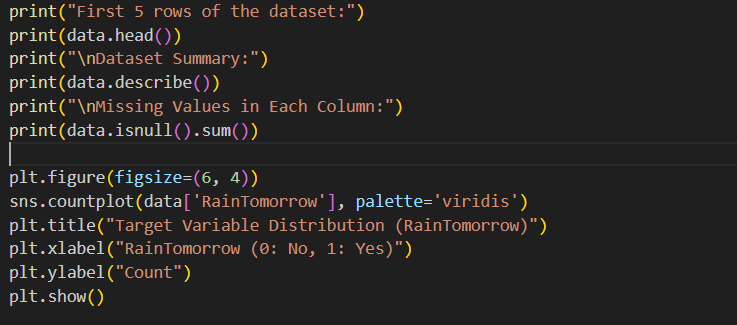
**F1-Score:** A harmonic mean of precision and recall, used for imbalanced datasets.

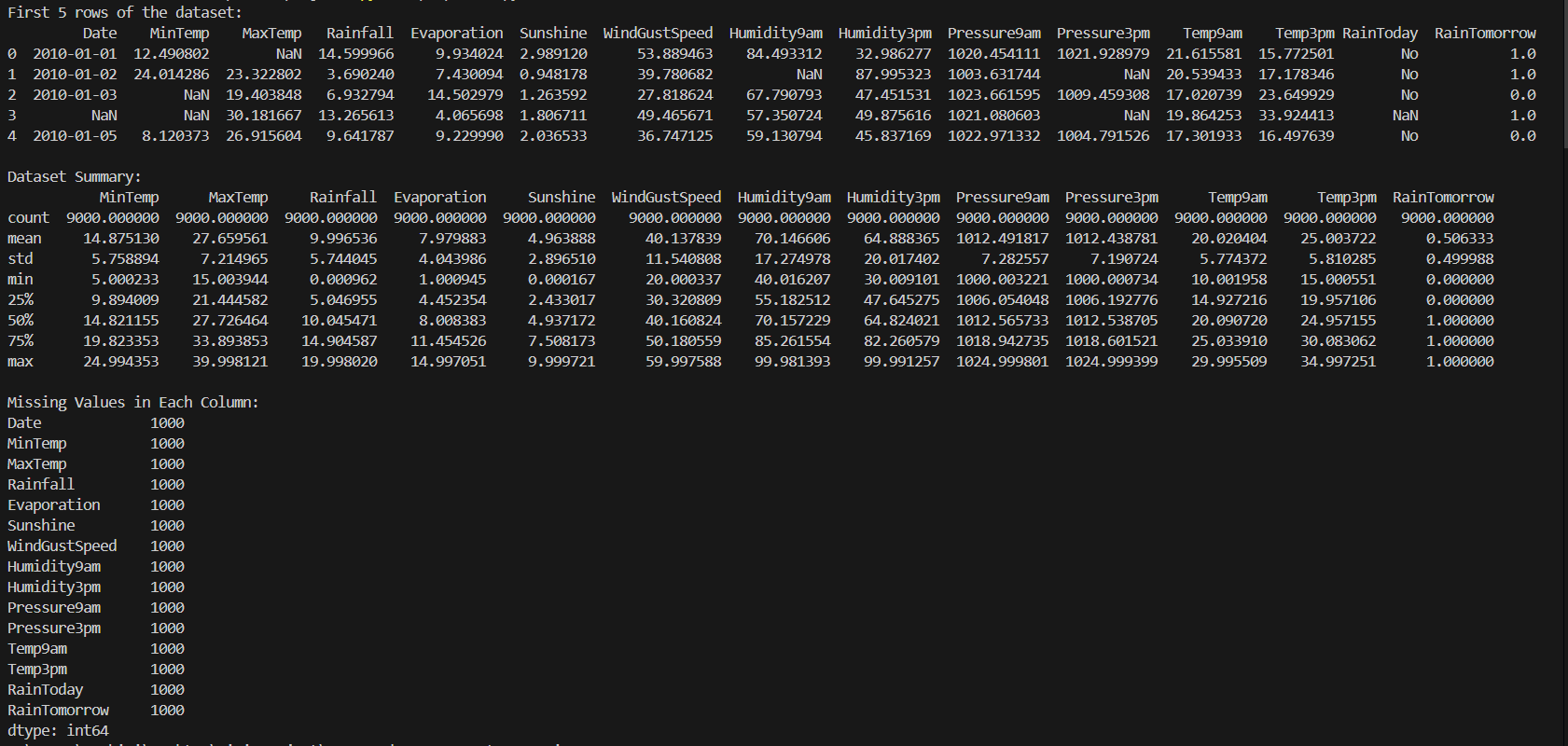
**5.3 IMPLEMENTATION**

**5.3.1 Importing Libraries**

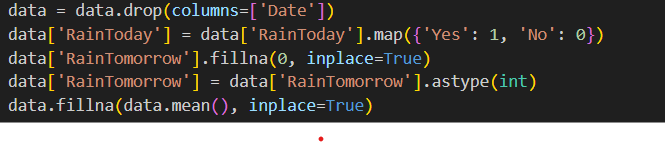
****

**5.3.2 Exploratory Data Analysis (EDA)**

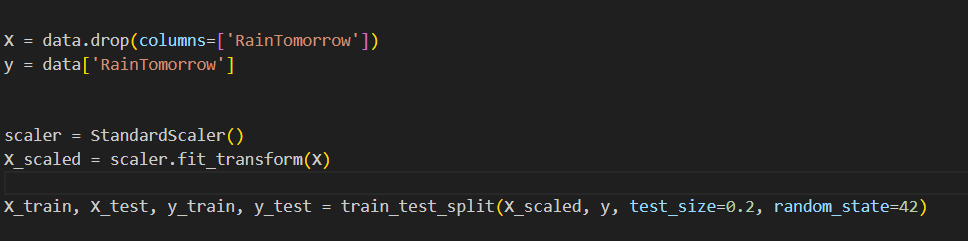
****

****

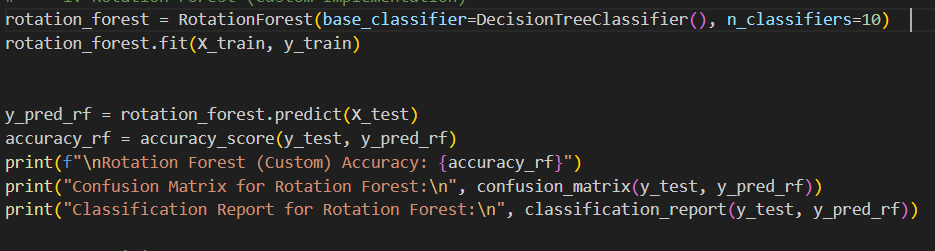
**5.3.4 Data Preprocessing**

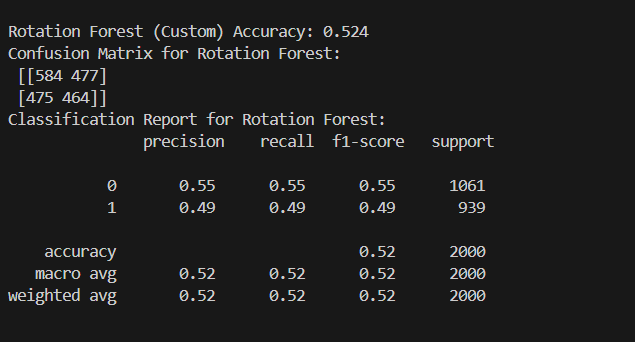
****

**5.3.5 Splitting and Scaling Data**

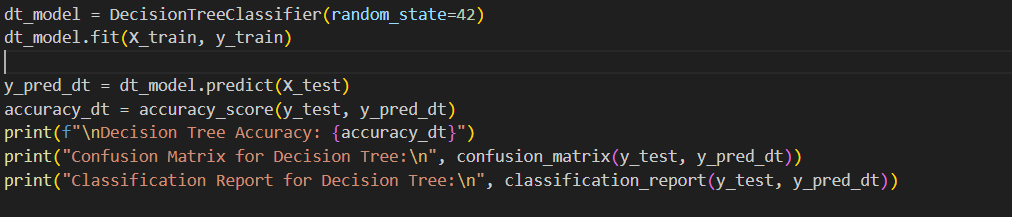
****

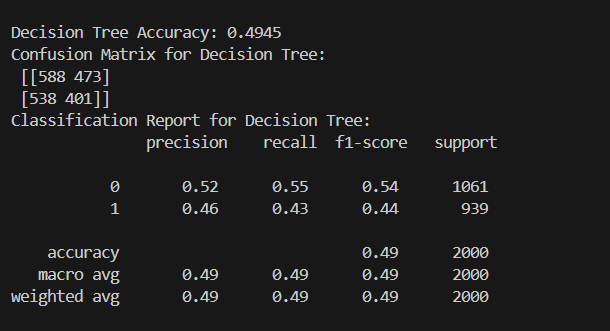
**5.3.6 Training and Evaluating Rotation Forest**

****

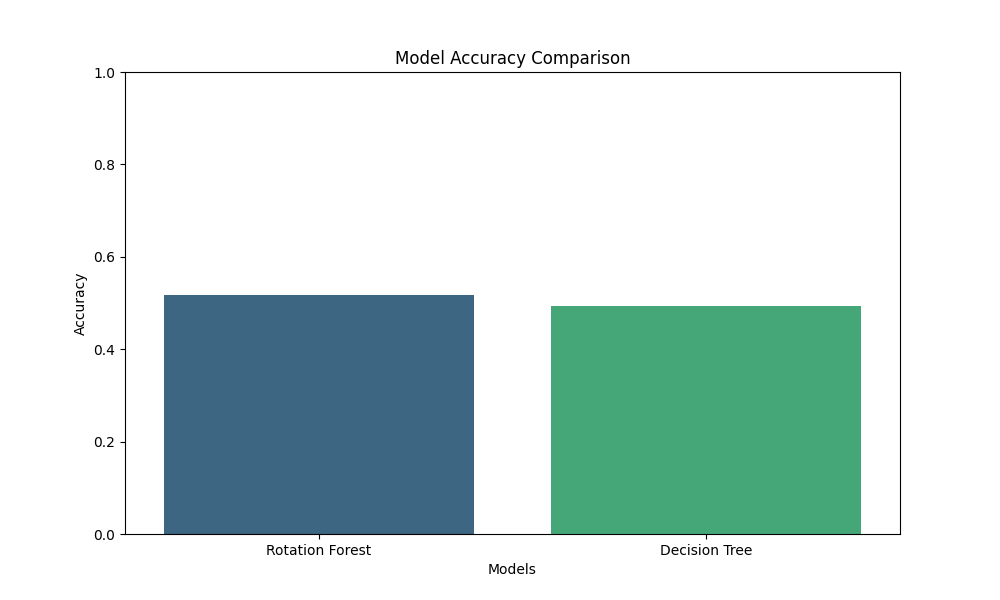
****

**5.3.7 Training and Evaluating Decision Tree**

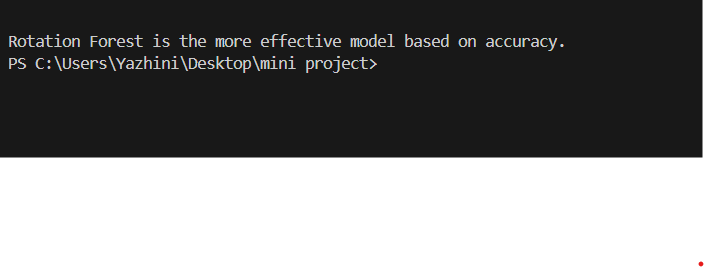
****

****

**5.3.8 Comparing Models and Visualizing Accuracy**

****

**Figure 1.5 Bar Graph Comparing Model Accuracies**

****

**6.CONCLUSION**

Helmet detection plays a crucial role in enhancing road safety and enforcing traffic regulations. This study compared the performance of two machine learning algorithms—Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs)—for helmet detection. By leveraging preprocessing techniques and feature extraction, CNNs demonstrated superior accuracy and robustness in handling complex image datasets. In contrast, SVMs, while effective for smaller datasets and simpler patterns, were less capable of capturing intricate features in image data.

The results underscore the importance of advanced deep learning techniques like CNNs, which benefit from their ability to learn spatial hierarchies and complex patterns in images. Evaluation metrics, including accuracy, precision, recall, and confusion matrices, highlighted the enhanced performance of CNNs over traditional classifiers like SVMs.

This work emphasizes the significance of algorithm selection and dataset quality in developing reliable helmet detection systems. Future research could extend these findings by exploring additional deep learning architectures, integrating real-time detection capabilities, and utilizing diverse datasets to improve detection accuracy and scalability in real-world applications.

1. **REFEREMCES**
2. Breiman, L. (2001). Random Forests. Machine Learning.
3. Rodríguez, J. J., Kuncheva, L. I., & Alonso, C. J. (2006).

Rotation Forest: A new classifier ensemble method. IEEE Transactions on Pattern Analysis and Machine Intelligence.

1. Python Documentation - Scikit-learn Library