Intrusion Detection System (IDS) - Machine Learning Models

# Project Overview

This repository contains three key machine learning models implemented for an Intrusion Detection System (IDS). The goal of the IDS is to detect malicious activities within a network using various classification techniques, including Logistic Regression, Naive Bayes, and Random Forest.

# Repository Structure

.  
├── logistic.py # Code for Logistic Regression model  
├── naive.py # Code for Naive Bayes model  
├── random\_forest.py # Code for running multiple models including Random Forest  
└── README.md # Project documentation

# Prerequisites

To run the project, ensure that you have the following dependencies installed:  
- Python 3.x  
- Required Libraries:  
 - numpy  
 - pandas  
 - scikit-learn  
 - matplotlib (for any visualizations)

You can install these using the following command:  
```bash  
pip install numpy pandas scikit-learn matplotlib  
```  
Or create a `requirements.txt` file with the necessary libraries and install them as shown:  
```bash  
pip install -r requirements.txt  
```

# Execution Guide

You can run each of the models separately or execute the overall pipeline using the summary script.

## 1. Logistic Regression Model

To run the Logistic Regression model, execute the following command:  
```bash  
python logistic.py  
```  
This script will:  
- Load the dataset.  
- Preprocess the data.  
- Train the Logistic Regression model.  
- Output the performance metrics (accuracy, precision, recall, F1-score).

## 2. Naive Bayes Model

To run the Naive Bayes model, use this command:  
```bash  
python naive.py  
```  
The script will follow a similar flow as the Logistic Regression model:  
- Load the dataset.  
- Preprocess the data.  
- Train the Naive Bayes model.  
- Output the performance metrics.

## 3. Random Forest Model

To run the Random Forest model (or execute multiple models), run the `random\_forest.py` script:  
```bash  
python random\_forest.py  
```  
This script:  
- Can train and evaluate the Random Forest model.  
- May also provide comparisons between Logistic Regression and Naive Bayes models.

# Dataset

Ensure the dataset used for the IDS is properly loaded within each script. Update the paths to the dataset if required inside each `.py` file. For example, update the `load\_data()` or `read\_csv()` function to point to your dataset location.

# Results

Each model will output key performance metrics to the console, including:  
- Accuracy  
- Precision  
- Recall  
- F1-Score  
  
The results will help in comparing the efficiency of different models for intrusion detection.

# Conclusion

This project demonstrates the use of three machine learning models for classifying network activity. The Random Forest model generally provides robust results due to its ensemble nature, but performance may vary depending on the dataset used.

# License

This project is licensed under the MIT License - see the LICENSE file for details.