

Project Description

The goal of this project is to create a machine learning model that effectively predicts the final grades of students based on:

- **Attendance:** The percentage of classes attended.
- **Study Hours:** The amount of time spent studying per week.
- **Socioeconomic Factors:** Information that may include income level, parental education, and other relevant factors. We will utilize two machine learning algorithms: Linear Regression and Random Forest.
- **Requirements**
 - To run this project, you need to have Python 3.x installed along with the following libraries:
 - pandas
 - numpy
 - scikit-learn
 - matplotlib
 - seaborn
 - You can install these dependencies using pip:

Dataset

The dataset used in this project consists is **student performance**, which contains information about students' attendance, study hours, socioeconomic factors, and their corresponding grades. The dataset should include the following columns:

- **attendance:** Percentage of classes attended by the student.
- **study_hours:** Number of hours spent studying per week.
- **socioeconomic_status:** Categorical variable indicating the socioeconomic background (e.g., low, medium, high).
- **grade:** Final grade achieved by the student

Preprocessing

1. The preprocessing steps involved in the project include: **Load Data:** Use pandas to load the dataset.
2. **Handle Missing Values:** Check for any missing values and handle them appropriately (e.g., imputation or removal).

3. **Encoding Categorical Variables:** Convert categorical variables into numerical format using techniques like one-hot encoding.
4. **Feature Scaling:** Normalize or standardize features if necessary.

Here's a sample code snippet for preprocessing:

- .

Model Development

We experimented with several machine learning algorithms, including:

- Linear Regression
- **Linear Regression**
- Linear regression is a simple yet effective algorithm for predicting continuous outcomes.
- **Split Data:** Divide the dataset into training and testing sets.
- **Train Model:** Fit a linear regression model using the training data.
- **Predict:** Use the model to make predictions on the test set.
- Here's how you can implement it

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

- **Random Forest**
- Random Forest is an ensemble method that can improve prediction accuracy by combining multiple decision trees.
- **Train Model:** Fit a random forest model using the training data.
- **Predict:** Make predictions on the test set.
- Implementation example:

```
from sklearn.ensemble import RandomForestRegressor
```

Evaluation Metrics Results

The final model achieved an MAE score of 7.432 on the test. This suggests that ,on average,the model's predictions deviate form the actual student performance scores by approximately 7.43 points

Installation

To set up this project locally, follow these steps:

1. Clone the repository

```
bash

git clone [repository-url]
cd [repository-directory]
```

2. Install the required packages:

```
bash

pip install -r requirements.txt
```

Usage

To run the model and make predictions, use the following

```
bash

python predict_grades.py --input [input-file]
```

Replace [input-file] with the path to your input data.

Contributing

Contributions are welcome! Please feel free to submit a pull request or raise an issue for any suggestions or enhancements.