# Create a linear regression model for analyzing road accident severity using the relevant dataset related to the scenario. Please specify the dependent variable (the variable you want to predict) and the independent variables (the factors that influence the accident severity). After creating the model, save it for future use. Then, provide an example of using the model to predict accident severity for a hypothetical set of independent variables, and explain how such a model could be beneficial for traffic accident analysis and prevention in underdeveloped countries. Add all relevant screen shots as well from your program. Also share the URL of your GITHUB (Where you have uploaded your work) so that I can simulate the same

## Variables

- Dependent Variable (Target): Accident_severity

## This is what we want to predict: how severe an accident might be based on other conditions.

- Independent Variables (Features):
- These are factors that likely influence accident severity. From the dataset, relevant variables could include:
- Age_band_of_driver: Age range of the driver, like '18-30' or '31-50'.
- Sex_of_driver: Gender of the driver.
- Educational_level: Education level of the driver.
- Driving_experience: Years or level of driving experience.
- Lanes_or_Medians: Type or number of lanes.

- Road_surface_type: Type of road surface (e.g., asphalt).
- Light_conditions: Lighting conditions at the time of the accident.
- Weather_conditions: Weather at the time of the accident

I am importing the necessary libraries: Pandas for data handling, sklearn for model creation and evaluation and for the joblib to save the trained model. Also i am loading the data

In [2]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import pickle
data = pd.read_csv("RTA.csv")
```

In [3]:
```
data.head()
```

Out[3]:

| | Age_band_of_driver | Sex_of_driver | Educational_level | Vehicle_driver_relation | Driving_exp |
|---|---|---|---|---|---|
| **0** | 18-30 | Male | Above high school | Employee | |
| **1** | 31-50 | Male | Junior high school | Employee | Abo |
| **2** | 18-30 | Male | Junior high school | Employee | |
| **3** | 18-30 | Male | Junior high school | Employee | |
| **4** | 18-30 | Male | Junior high school | Employee | |

In [5]:
```
data.describe(include="all")
```

Out[5]:

| | Age_band_of_driver | Sex_of_driver | Educational_level | Vehicle_driver_relation | Drivin |
|---|---|---|---|---|---|
| count | 12316 | 12316 | 12316 | 12316 | |
| unique | 5 | 3 | 7 | 4 | |
| top | 18-30 | Male | Junior high school | Employee | |
| freq | 4271 | 11437 | 7619 | 9627 | |
| mean | NaN | NaN | NaN | NaN | |
| std | NaN | NaN | NaN | NaN | |
| min | NaN | NaN | NaN | NaN | |
| 25% | NaN | NaN | NaN | NaN | |
| 50% | NaN | NaN | NaN | NaN | |
| 75% | NaN | NaN | NaN | NaN | |
| max | NaN | NaN | NaN | NaN | |

In [10]:
```
X = data[['Age_band_of_driver', 'Sex_of_driver', 'Educational_level',
          'Driving_experience', 'Lanes_or_Medians', 'Road_surface_type',
          'Light_conditions', 'Weather_conditions']]
y = data['Accident_severity']
```

In [12]:
```
X = pd.get_dummies(X, drop_first=True)
```

In [14]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_sta
```

In [16]:
```
from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(X_train, y_train)
```

Out[16]:

LinearRegression

LinearRegression()

In [17]:
```
from sklearn.metrics import mean_squared_error
y_pred = model.predict(X_test)
error = mean_squared_error(y_test, y_pred)
print("Mean Squared Error:", error)
```

```
Mean Squared Error: 0.16991887234245445
```

In [19]:
```
import pickle
with open("accident_severity_model.pkl", "wb") as file:
    pickle.dump(model, file)
```

- I am importing the necessary libraries: Pandas for data handling, sklearn for model creation and evaluation and for the joblib to save the trained model. Also i am loading the data.
- Assigning the X and y variables y to the target variable and indepent variables and converting them into the numeric form
- Diving, training and testing and training them with linearregression and checking the performance using the MSE and R squared values(for accuracy)
- Finally I am saving the so as it can be used again

In [31]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import joblib


data = pd.read_csv("RTA.csv")

X = data[['Age_band_of_driver', 'Sex_of_driver', 'Educational_level',
          'Vehicle_driver_relation', 'Driving_experience', 'Lanes_or_Medians',
          'Types_of_Junction', 'Road_surface_type', 'Light_conditions',
          'Weather_conditions', 'Type_of_collision', 'Vehicle_movement',
          'Pedestrian_movement', 'Cause_of_accident']]
y = data['Accident_severity']


X = pd.get_dummies(X, drop_first=True)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta


model = LinearRegression()
model.fit(X_train, y_train)


mse = mean_squared_error(y_test, model.predict(X_test))
r2 = r2_score(y_test, model.predict(X_test))
print('Mean Squared Error:', mse)
print('R-squared:', r2)


joblib.dump(model, 'accident_severity_model.pkl')
```

```
Mean Squared Error: 0.17460386613293943
R-squared: 0.007548901555093135
```

Out[31]:
```
['accident_severity_model.pkl']
```

In [25]:
```
new_data = pd.DataFrame(columns=X_train.columns)


new_data.loc[0] = [1 if col == 'Age_band_of_driver_18-30' else
```

```
                    0 if col == 'Sex_of_driver_Male' else
                    0 for col in X_train.columns]
```

we are loading the data and use to predict the severity

In [26]:
```
loaded_model = joblib.load('accident_severity_model.pkl')

predicted_severity = loaded_model.predict(new_data)
print('Predicted Severity:', predicted_severity)
```

```
Predicted Severity: [1.89958898]
```