import pandas as pd

from sklearn.model\_selection import train\_test\_split

# Load dataset

data = pd.read\_csv('/road\_accident - Copy.csv')

# Display first few rows

print(data.head())

# Checking for missing values

print(data.isnull().sum())

# Dropping rows with missing target values (severity)

data = data.dropna(subset=['Accident\_Severity'])

# Filling or dropping missing values for features (independent variables)

data.fillna(0, inplace=True)

# Convert categorical variables (e.g., road type, weather) to numeric using one-hot encoding

data = pd.get\_dummies(data, columns=['Weather\_Condition', 'Road\_Condition'], drop\_first=True)

# Display cleaned data

print(data.head())

import pandas as pd

import numpy as np

from sklearn.model\_selection import train\_test\_split

# Dependent variable

Y = data['Accident\_Severity']

X = data.drop(columns=['Accident\_Severity'])

# X = data[['Speed\_Limit', 'Num\_Lanes', 'Driver\_Alcohol\_Level', 'Weather\_Condition\_Sunny', 'Road\_Condition\_Wet']]

X\_array = X.to\_numpy()

Y\_array = Y.to\_numpy()

# Splitting the dataset into training and testing sets

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error, r2\_score

# Initialize and train the model

model = LinearRegression()

model.fit(X\_train, Y\_train)

# Make predictions

Y\_pred = model.predict(X\_test)

# Evaluate the model

mse = mean\_squared\_error(Y\_test, Y\_pred)

r2 = r2\_score(Y\_test, Y\_pred)

print(f"Mean Squared Error: {mse}")

print(f"R-squared: {r2}")