A white robot hand with red and blue stripes

Description automatically generated

**Lab Activity 5.1**



**LAB MANUAL**

**Disclaimer: The content is curated from online/offline resources and used for educational purpose only**

**Lab Activity 5.1: Preprocess the Indian Road Accident Dataset and Split into Training & Test Sets**

**Objective:** The objective of this lab is to preprocess the Indian Road Accident dataset and split it into training (80%) and test (20%) sets while ensuring a balanced class distribution. Suitable techniques will be applied to verify the balance.

**Step 1: Load the Dataset**

1. Import the necessary libraries:

import pandas as pd

import numpy as np

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

from sklearn.utils import resample

from collections import Counter

1. Load the dataset:

df = pd.read\_csv("indian\_road\_accident\_dataset.csv")

1. Display the first few rows:

print(df.head())

**Step 2: Data Preprocessing**

1. Check for missing values:

print(df.isnull().sum())

Handle missing values (e.g., fill missing numerical values with mean/median and categorical values with mode).

df.fillna(df.median(), inplace=True)

df.fillna(df.mode().iloc[0], inplace=True)

1. Convert categorical variables to numerical format if necessary:

df = pd.get\_dummies(df, drop\_first=True)

1. Identify the target variable:

target\_column = 'Severity' # Change based on dataset

X = df.drop(columns=[target\_column])

y = df[target\_column]

**Step 3: Split the Data While Maintaining Class Balance**

1. Check class distribution:

print(Counter(y))

1. Perform a stratified split to maintain class balance:

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, stratify=y, random\_state=42)

1. Verify the class distribution in the split data:

print("Training set class distribution:", Counter(y\_train))

print("Test set class distribution:", Counter(y\_test))

**Step 4: Verify Data Splitting & Balancing**

1. Check shape of the datasets:

print("Training Set Shape:", X\_train.shape, y\_train.shape)

print("Test Set Shape:", X\_test.shape, y\_test.shape)

1. Ensure target class distribution is approximately similar before and after splitting:

train\_class\_dist = y\_train.value\_counts(normalize=True)

test\_class\_dist = y\_test.value\_counts(normalize=True)

print("Training Set Distribution:\n", train\_class\_dist)

print("Test Set Distribution:\n", test\_class\_dist)

If the distributions are significantly different, resampling techniques (oversampling/undersampling) can be applied.

**Step 5: Conclusion**

By completing this lab, you have successfully:

* Preprocessed the Indian Road Accident dataset.
* Handled missing values and categorical variables.
* Split the dataset into training (80%) and test (20%) sets.
* Ensured a balanced class distribution using stratified sampling.
* Verified the class balance in both training and test sets.

This dataset is now ready for model training and evaluation.