Experiment #		Student ID	
Date		Student Name	

Experiment # 8: Implement various Data preprocessing techniques on a given data set

Aim/Objective:

This experiment aims to implement data pre-processing techniques to clean, transform, and prepare raw data for further analysis or machine learning tasks

Description:

In this experiment, students will learn the importance of data pre-processing in the data science workflow. They will understand the various steps involved in cleaning and transforming raw data to make it suitable for analysis or model building. Students will implement a data pre-processing pipeline using Python and relevant libraries, gaining hands-on experience in handling missing values, outliers, categorical variables, feature scaling, and more.

Pre-Requisites:

Basic understanding of data types, including numerical and categorical variables.

Familiarity with Python programming and data manipulation libraries such as pandas

Pre-Lab:

1. Why data are dirty?
2. What is data preprocessing? Why is it important in machine learning?
3. What are some common problems that occur during data processing? How can they be fixed?
4. How do you handle the missing data?
5. What is the difference between missing value treatment and outliers treatment?

1. Data is considered "dirty" when it contain errors, inconsistencies, (or) is incomplete leading to ~~our~~ inaccurate (or) unrealible result in analysis:-

reasons data become dirty:- i) Missing values
ii) Duplicates
iii) Incorrect data

(2) Data Preprocessing is a Process of cleaning & transforming raw data before feeding it into a machine learning model. It involves handling missing values, normalizing data, encoding categorical variables.

(3) common problems:- i) Missing Data
ii) outliers
iii) Inconsisent Data
iv) Noise
v) Duplicate data.

(4) Handling missing data can be done in several ways:- i) Removal
ii) Imputation
iii) Indicator for missing values.

Experiment #		Student ID	
Date		Student Name	

(5.)

Missing outlier treatment:- It focuses on filling in or dealing with missing data points.

Outlier treatment:- Deals with extreme or unusual data points that may skew the model results.

Experiment #		Student ID	
Date		Student Name	

In-Lab:

Implement a Python program to find and impute the missing data in the following dataset.

Dataset Link:

https://www.kaggle.com/datasets/bharatnatrayn/movies-dataset-for-feature-extracion-prediction/data?select=movies.csv

Procedure/Program:

```
import pandas as pd

from sklearn.impute import SimpleImputer

df = pd.read_csv('movies.csv')
print("missing values before imputation:"):
print(df.isnull().sum())

numerical_colowns=df.select_dtypes(include=float64, int64]
imputer_num=SimpleImputer(strategy='mean')

df[numerical_columns] = imputer_num.fit_transform

imputer_cat = SimpleImputer(strategy=most_frequent).
```

```
Print("missing values after imputation:"):
 Print(df.isnull().sum())
df.to_csv('movies_imputed.csv', index=false)
```

- **Data and Results:**

the dataset is loaded from csv files using Pandas. The Program Points the number of missing values for each column.

- **Analysis and Inferences:**

For numerical values, missing values are imputed using mean.

The imputed dataset is saved as movies_imputed.csv

Experiment #		Student ID	
Date		Student Name	

VIVA-VOCE Questions (In-Lab):

1. What is the difference between normalization and standardization?
2. What are the different encoding techniques for categorical data?
3. What are some common techniques for data reduction?
4. How do you preprocess time-series data?
5. What is data integration and what challenges are associated with it?

(1.) Normalization:- Rescales data to a range of [0, 1], useful when data needs to be on same scale for comparison.

standardization:- Rescal data to have a mean of 0 and a standard deviation of 1.

(2.) encoding techniques:- i) label encoding
ii) one-Hot Encoding
iii) ordinal Encoding
iv) target Encoding

③ Techniques:- i) Principle component Analysis
ii) feautre selection
iii) Sampling
iv) Aggregation

④ Preprocessing time-series data:-
i) Handling Missing data
ii) Smoothing
iii) Resampling
iv) Differencing
v) Normallization

⑤ Data integration:- combining data from different sources into a unfied dataset
challenges:- i) schema Mismatch
ii) Data Quality
iii) Duplicate data
iv) Scaling

Experiment #		Student ID	
Date		Student Name	

Post-Lab:

Implement a Python program to apply various data preprocessing techniques on the following dataset.

Dataset Link:

https://catalog.data.gov/dataset/electric-vehicle-population-data/resource/fa51be35-691f-45d2-9f3e-535877965e69

Procedure/Program:

import pandas as pid.

from Sklearn.impute import simpleImputer

from sklearn.preprocessing impo l labelEncoder

from Sklearn.model-selection import train-test-split.

url='https://data.wa.gov/api/views/frwt-q2d2/
rows.csv!acesstype=Download

df=df.read-csv(url)

Print(Intial dataset information:")

Print (df.info())

numerical-column=df.select-dtypes(include=int4)

inputer-num=simpleImputer(strategy=mean)

categorical-colums=df.selecttypes(include=['object'])

```
imputer-cat = simple Imputer(strategy=most-frequent)
Print("\n missing value after imputation:")
Print(df. isnull().sum())
label-encodel = Label Encoder()
for column in categorical-columns:
df[column]=label-encoder. fit-transform(df[column])
Print("\n first 5 rows after encoding:")
Print(df.head()).
```

Experiment #		Student ID	
Date		Student Name	

Data and Results:

Imputed missing values using mean for numerical data and the most frequent value for categorical data.

Analysis and Inferences:

the preprocessing steps ensured the dataset was free of missing values and duplicates.

Evaluator Remark (if Any):	Marks Secured:______out of 50
	... of the Evaluator with Date