Assignment No. 2

1.1 Title: Data Preprocessing on dataset for handling missing values.

1.2 Problem Definition: Data preprocessing is the essential phase to bring dataset into required standard format it includes finding the missing values and to impute those values through available features.

1.3 Prerequisite: Install Anaconda Python, Jupyter Notebook, Ubuntu 18.04.

1.4 Software Requirement: Jupyter Notebook, Spyder on Ubuntu.

1.5 Hardware Requirement: 2GB RAM, 500 GB HDD

1.6 Objectives: Understand the and implement data preprocessing.

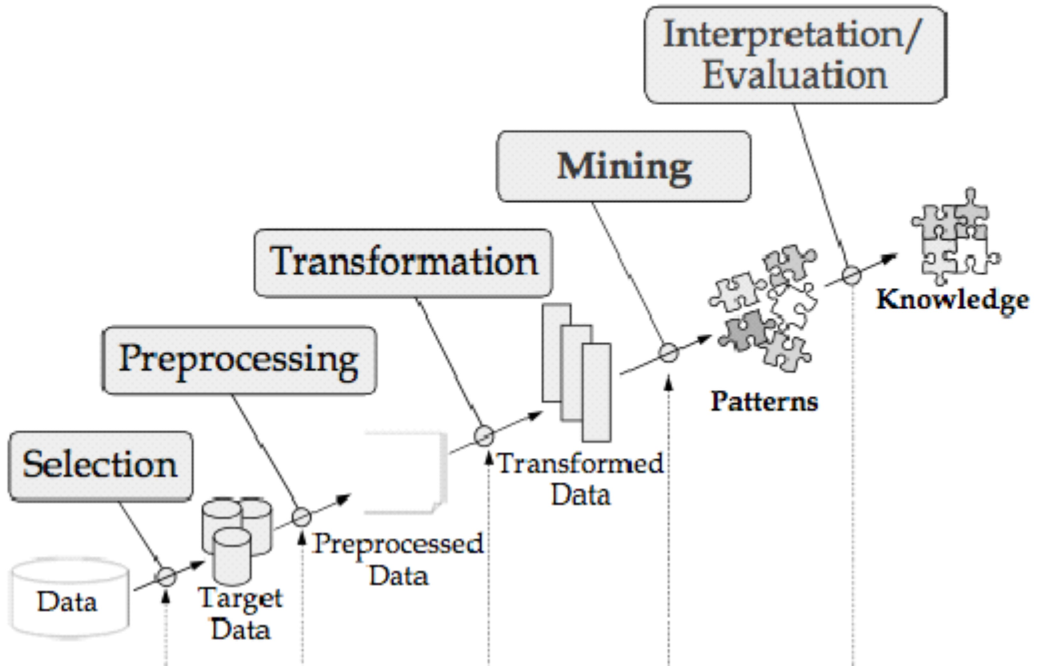
1.7 Outcomes: After completion of this assignment students can analyze and impute missing values in dataset also can handle categorial data.

1.8 Theory Concepts:

- Data Preprocessing

Data preprocessing is a data mining technique that involves transforming raw data into an understandable format. Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors. Data preprocessing is a proven method of resolving such issues.

In Real world data are generally incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data. Noisy: containing errors or outliers. Inconsistent: containing discrepancies in codes or names.



- Steps in Data Preprocessing

Step 1 : Import the libraries

Step 2 : Import the data-set

Step 3 : Check out the missing values

Step 4 : See the Categorical Values

Step 5 : Splitting the data-set into Training and Test Set

Step 6 : Feature Scaling

Step 1 : Import the Libraries

This is how we import libraries in Python using import keyword and this is the most popular libraries which any Data Scientist used.

- NumPy is the fundamental package for scientific computing with Python. It contains among other things: a powerful N-dimensional array object, Sophisticated (broadcasting) functions, tools for integrating C/C++ and FORTRAN code, useful linear algebra, fourier transform and random number capabilities

- Pandas is for data manipulation and analysis. Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. Pandas is a NumFOCUS sponsored project. This will help ensure the success of development of pandas as a world-class open-source project, and makes it possible to donate to the project.

- Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hard copy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers, and four graphical user interface toolkits.

- Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

Step 2 : Import the Data-set

By using Pandas we import our data-set and the file I used here is .csv file It’s not necessarily every-time you deal with CSV file, sometimes you deal with Html or Xlsx(Excel file). However, to access and to use fastly we use CSV files because of their light weights. After importing the dataset, you can see we use head function This function returns the first n rows for the object based on position. It is useful for quickly testing if your object has the right type of data in it. By default it returns 5 rows.

Step 3 : Check out the Missing Values

The concept of missing values is important to understand in order to successfully manage data. If the missing values are not handled properly by the researcher, then he/she may end up drawing an inaccurate inference about the data. Due to improper handling, the result obtained by the researcher will differ from ones where the missing values are present.Two ways to handle Missing Values:

1. This method commonly used to handle the null values.

Here, we either delete a particular row if it has a null value for a particular feature and a particular column if it has more than 75% of missing values. This method is advised only when there are enough samples in the data set. One has to make sure that after we have deleted the data, there is no addition of bias. Removing the data will lead to loss of information which will not give the expected results while predicting the output.

2. Drop the Missing Values

This strategy can be applied on a feature which has numeric data like the year column or Home team goal column. We can calculate the mean, median or mode of the feature and replace it with the missing values. This is an approximation which can add variance to the data set. But the loss of the data can be negated by this method which yields better results compared to removal of rows and columns. Replacing with the above three approximations are a statistical approach of handling the missing values. This method is also called as leaking the data while training. Another way is to approximate it with the deviation of neighbouring values. This works better if the data is linear.

Step 4 : See the Categorical Values

Analyze the categorical values of column and assign the numeric values for each category which can be useful in furthur processing of data and mapping of data. Suppose we have gender as a column having values male and female then we can assign 1 for male and 0 female using LabelEncoder class.

Step 5 : Splitting the data-set into Training and Test Set

In any Machine Learning model is that we're going to split data-set into two separate sets

1. Training Set

2. Test Set

Using train_test_split class we can split the given dataset into train and test data. Train data is the input data for the processing and test data is the data used for comparison with train data hence machine can predict on basis of this data.

1.9 Code

```
#!/usr/bin/env python
# coding: utf-8
# In[16]:
#import required libraries.

import numpy as np
import matplotlib.pyplot as py
import pandas as pd


# In[17]:
#Read the dataset through pandas read_csv() function.

dataset = pd.read_csv(r"C:\\Users\\shivani\\Desktop
\\Mall Customers.csv")


# In[18]:
#Display dataset.

dataset


# In[19]:
#Display first five records of dataset

dataset.head()


# In[20]:
#Extracting array from given dataset that is entire rows and
entire columns except last column in x and all data of column 4
in y.

x = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 4].values


# In[21]:
#printing x values.

x


# In[22]:
#printing y values.

y


# In[25]:
#impute missing values using Imputer class and mean strategy.
```

```
from  sklearn.preprocessing import Imputer
imputer = Imputer(missing_values='NaN', strategy='mean',axis=0)
imputer=imputer.fit(x[:, 2:3])
x[:, 2:3]=imputer.transform(x[:, 2:3])


# In[26]:
#printing x values

x


# In[27]:
#Handling Categorical data (bivariant).Using LabelEncoder class
we can assign numeric value to categorial data.

from  sklearn.preprocessing import LabelEncoder
labelencoder_x=LabelEncoder()
x[:,1]=labelencoder_x.fit_transform(x[:,1])


# In[28]:
#Observe change in x, conclude

x


# In[30]:
#splitting dataset into training and testing dataset

from  sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=
0.2,random_state=0)


# In[31]:
#length of train data for x values

len(x_train)


# In[32]:
length of test data for x values

len(x_test)


# In[33]:
length of train data for y values

len(y_train)
```

```
# In[34]:
#length of test data for y values


len(y_test)


# In[35]:
#Feature Scaling: Standardization: Xstand = [X - MEAN(X)] /
[STANDARD DEVIATION(X) Normalization: Xnorm = [x - min(x)] /
[max(x) - min(x)]
#Feature Scaling: Change the value of salary (few rows to
thousands or lakhs)

from sklearn.preprocessing import StandardScaler
sc_x = StandardScaler()
x_train = sc_x.fit_transform(x_train)
x_test = sc_x.fit_transform(x_test)


# In[36]:
#To display test and train data for x

x_train
x_test

# In[37]:
#To display test and train data for y

y_train
y_test
```

1.10 Code with output:

1.11 Conclusion:

Thus, after successfully completing this assignment, we were able to understand & implement data preprocessing that is handling missing values and handling categorial data of dataset for furthur processing.