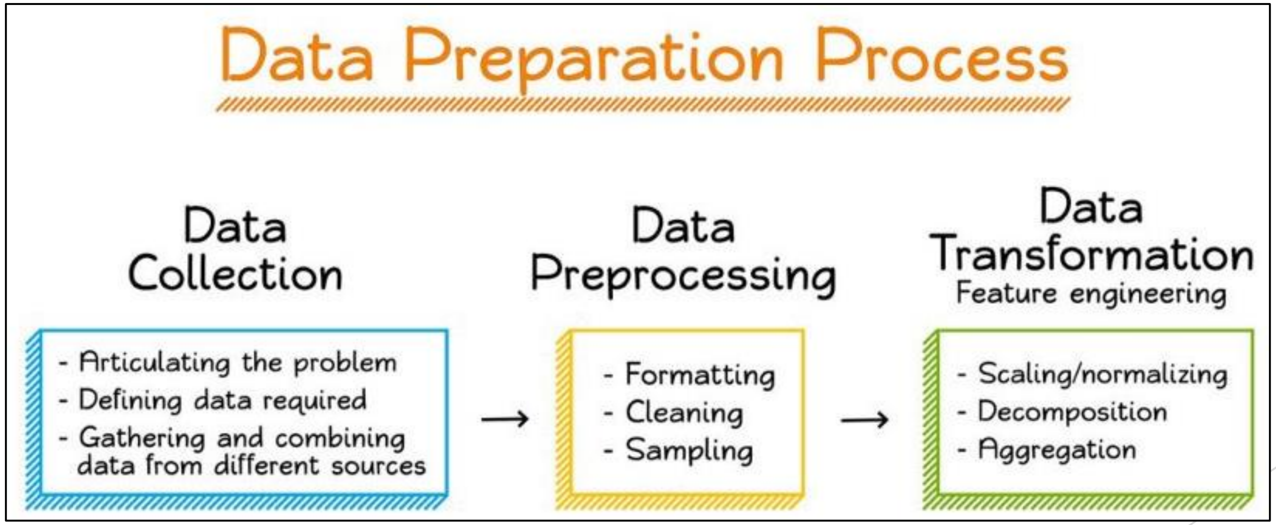
Certification Program in Business Analytics & Optimisation From IITD

Assignment 1
Data Cleaning and Data Handling

From:
Shivesh Kumar Sareen
shivesh72@gmail.com

Objective: To demonstrate Data preprocessing steps like data cleaning and data handling

- To begin with building of machine learning model for data analysis, its important to do preprocessing of the data.
- All the raw data which is available as input, might not be completely relevant for analysis. It needs to be preprocessed in order to check missing details, noisy data, and other inconsistencies before executing it to the algorithm.
- Data preprocessing ensures that data which is being used for building model is correct in all respects and free from any errors.



Steps for Performing Preprocessing operation on the dataset

Step 1: Importing Libraries : Certain libraries like Numpy, Pandas and Matplotlib needs to be imported to carry out data pre-processing.

Step 2: Loading the dataset : Dataset which is a (.csv) file needs to be loaded in Python.

Step 3: Defining the variables : Input and output parameters need to be defined.

Step 4: Handing of Missing Data: *There are 2 approaches, one is the replace null value with the mean of other values for that variable. Other is, to remove that dataset. Second approach is generally avoided in case of small datasets.*

Step 5: Encoding categorical data : There are 2 types namely, ONE HOT ENCODING and Label Encoding.

Step 6: Splitting the dataset : To develop a machine learning model, dataset needs to be divided into training and test data. Bigger the dataset, more is the training data, better is the accuracy of the model.

Step 7: Feature Scaling: In dataset, often the range of input data varies. Like age will be a 2 digit number and salary 6 digit number. In order to normalize the same, feature scaling is used.

Sample code where Data Preprocessing has been done

```
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Libraries are imported in Python

```
# Importing the dataset
dataset = pd.read_csv('Data.csv')
```

Datasets which are to be studied are imported in Python

```
# Defining the variables
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
print(X)
print(y)
```

```
# Taking care of missing data
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(missing_values=np.nan, strategy='mean')
imputer.fit(X[:, 1:3])
X[:, 1:3] = imputer.transform(X[:, 1:3])
print(X)
```

```
# Encoding categorical data
# Encoding the Independent Variable
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(), [0])], remainder='passthrough')
X = np.array(ct.fit_transform(X))
print(X)
# Encoding the Dependent Variable

from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
y = le.fit_transform(y)
print(y)
```

```
# Splitting the dataset into the Training set and Test set
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 = 1)
print(X_train)
print(X_test)
print(y_train)
print(y_test)
```

```
# Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train[:, 3:] = sc.fit_transform(X_train[:, 3:])
X_test[:, 3:] = sc.transform(X_test[:, 3:])
print(X_train)
print(X_test)
```

Output of Data Preprocessing : The data shown below is further analyzed based on different ML models

```
[0.0   1.0   -0.19159184384578545      -1.0781259408412425]
[1.0   0.0   -0.014117293757057777     -0.07013167641635372]
[0.0   0.0   0.566708506533324          0.633562432710455]
[0.0   1.0   -0.30453019390224867      -0.30786617274297867]
[0.0   1.0   -1.9018011447007988       -1.420463615551582]
[0.0   0.0   1.1475343068237058         1.232653363453549]
[1.0   0.0   1.4379472069688968         1.5749910381638885]
[0.0   0.0   -0.7401495441200351       -0.5646194287757332]
[1.0   0.0   -1.4661817944830124       -0.9069571034860727]
[0.0   0.0   -0.44973664397484414       0.2056403393225306]
```

After One Hot Encoding, ‘COUNTRY’ column has been split into 2 columns based on (n-1) principle

Range of the values of age and salary has been standardized using Feature Scaling

THANK YOU