Sourav Kotkar
TE-3 (N3)
31378

Performance Date: 21/01/2022
Submission Date: 21/01/2022

Page No.:
Date:

DSBDAL

Group-A Assignment-2

* **Title :-**

Data Wrangling II

* **Problem Statement :-**

Create an "Academic Performance" dataset of students and perform the following operations using Python.

1. Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies, use any of the suitable techniques to deal with them.
2. Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal with them.
3. Apply data transformations on atleast one of the variables. The purpose of this transformation should be one of the following reasons: to change the scale for better understanding of the variable, to convert a non-linear relation into a linear one, or to decrease the skewness and convert the distribution into a normal distribution.

* **Learning Objectives :-**

1. To implement data preprocessing techniques on raw data.
2. To understand how to deal with missing values and inconsistencies.
3. To understand outliers and to deal with them using suitable techniques.

* **Learning Outcomes :-**

One should be able to :-

1. Preprocess data and convert raw data into proper usable format.
2. Deal with missing data as well as outliers using proper techniques.

* H/W and S/W Requirements:-
8GB RAM laptop, Windows 10, Jupyter notebook, Python 3.9

* Theory:-

⇒ Data Wrangling:-
It is sometimes referred to as data munging. It is the process of transforming and mapping data from one "raw" data form into another format with the intent of making it more appropriate and valuable for a variety of downstream purposes such as analytics. The goal of data wrangling is to assure quality and useful data. Data analysis typically spend the majority of their time in the process of data wrangling compared to the actual analysis of the data.

⇒ Missing Data:-
Missing data can occur when no information is provided for one or more items or for a whole unit. Missing data is a very big problem in a real-life scenarios. Missing data can also refer to as NA (Not Available) values in pandas. In dataframe sometimes many datasets simply arrive with missing data, either because it exists and was not collected or it never existed.
In Pandas, missing data is represented by two values:
1) None: None is a Python singleton object that is often used for missing data in Python code.
2) NaN: NaN (an acronym for Not a Number), is a special floating-point value recognized by all systems that use the standard IEEE floating-point representation.
Pandas treat None and NaN as essentially interchangeable for indicating missing or null values.

⇒ Outliers :-

Outliers are extreme values that deviate from other observations on data, they may indicate a variability in a measurement, experimental errors or a novelty. In other words, an outlier is an observation that diverges from an overall pattern on a sample.

Most common causes of outliers on a dataset :-

- Data entry errors (human errors)
- Measurement errors (instrument errors)
- Experimental errors (data extraction or experimental planning/executing errors)
- Intentional (dummy outliers made to test detection methods)
- Data processing errors (data manipulation or data set unintended mutations)
- Sampling errors (extracting or mixing data from wrong or various sources)
- Natural (not an error, novelties in data)

Outlier detection techniques :-

1) Z-score or Extreme Value Analysis (parametric)
2) Inter Quartile Range (IQR)
3) Linear Regression Models (PCA, LMS)
4) Proximity Based Models (non-parametric)

⇒ Interquartile Range (IQR) :-

IQR is used to measure variability by dividing a data set into quartiles. The data is sorted in ascending order and split into 4 equal parts. Q1, Q2, Q3 called first, second and third quartiles are the values which seperate the 4 equal parts.

- Q1 represents the 25th percentile of the data.
- Q2 represents the 50th percentile of the data.
- Q3 represents the 75th percentile of the data.

If a dataset has 2n/2n+1 data points, then
Q1 = median of the dataset
Q2 = median of n smallest data points
Q3 = median of n largest data points

IQR is the range between the first and the third quartiles namely Q1 and Q3: IQR = Q3 - Q1. The data points which fall below Q1 - 1.5 IQR and above Q3 + 1.5 IQR are outliers.

⇒ Skewness :-

Skewness refers to a distortion or asymmetry that deviates from the symmetrical bell curve, or normal distribution, in a set of data. If the curve is shifted to the left or to the right, it is said to be skewed. Skewness can be quantified as a representation of the extent to which a given distribution varies from a normal distribution. A normal distribution has a skew of zero.

- If skewness is less than -1 or greater than 1, the distribution is highly skewed.
- If skewness is between -1 and -0.5 or between 0.5 and 1, the distribution is moderately skewed.
- If skewness is between -0.5 and 0.5, the distribution is approximately symmetric.

⇒ Data Transformation Techniques :-

1) Reciprocal Transformation
2) Square root Transformation
3) Logarithmic Transformation
4) Exponential Transformation
5) Box-Cox Transformation

For this assignment, I have used 3 techniques:-

1) Reciprocal Transformation :-
This transformation will inverse the values of the column selected.
Syntax -

```
df[`math score reciprocal'] = 1/df[`math score']
```

2) Square Root Transformation :-
This transformation will take the square root of the column selected.
Syntax -

```
df[`math score sqrt'] = np.sqrt(df[`math score'])
```

3) Logarithmic Transformation :-
This transformation will convert the value to its log value.
Syntax -

```
df[`math score log'] = np.log(df[`math score'])
```

⇒ Methods and Functions Used :-

1) read_csv() - It is used to read a csv file into Dataframe.

2) head() - It returns first n rows for the object based on position.

3) tail() - It returns last n rows for the object based on position.

4) describe() - It gives descriptive statistics of the dataset.

5) isnull() - It returns a boolean same-sized object indicating if the values are NA.

6) fillna() - It is used to fill NA/NAN values using the specified method.

7) dtypes - It returns a series with the data type of each column.

8) astype() - It is used to cast a pandas object to a specified datatype.

9) quantile() - It returns values at the given quantile over requested axis.

10) skew() - It returns an unbiased skew over requested axis.

⇒ Packages / Libraries used :-

1) Pandas :-
It is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

2) Numpy :-
It is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

3) Matplotlib :-
It is a plotting library for the Python programming language and its numerical mathematics extension Numpy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt or GTK.

4) Scipy :-
It is a free and open-source Python library used for scientific computing and technical computing. Scipy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal & image processing, ODE solvers and other tasks common in science and engineering.

⇒ Plots used :-

1) Box plot :-
It shows the distribution of quantitative data in a way that facilitates comparisons between variables or across levels of a categorical variable. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution, except for points that are determined to be "outliers" using a method that is a function of the inter-quartile range.

2) KDE plot :-
A kernel density estimate (KDE) plot is a method for visualizing the distribution of observations in a dataset, analogous to a histogram. KDE represents the data using a continuous probability density curve in one or more dimensions.

3) Prob plot :-
It generates a probability plot of sample data against the quantiles of a specified theoretical distribution (the normal distribution by default). Probplot optimally calculates a best-fit line for the data & plots the results using Matplotlib or a given plot function.

* Observations :-

- The dataset which is used is the Students Performance dataset. It consists of the marks secured by the students in various subjects.
- There are missing values in 'math score', 'reading score' and 'writing score' columns which are replaced with the mean values.
- Outliers are present in 'math score' column which is removed using interquartile range method.
- Reciprocal, logarithmic and square root data transformations are performed on the dataset to reduce skewness of the data values.

* Conclusion :-

Through this assignment, we learnt and implemented how to handle data inconsistencies & missing values, removing outliers and performed data transformations to reduce skewness of data.