|  |  |
| --- | --- |
| **TITLE** | Data Wrangling II |
| **PROBLEM STATEMENT/ DEFINITION** | 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 at least 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.  Reason and document your approach properly. |
| **OBJECTIVE** | 1. To do data cleansing on the given dataset.  2. To find outliers and deal with them  3. Data transformations for scaling/removing nonlinear relationship to linear/ to decrease skewness. |
| **S/W PACKAGES AND HARDWARE APPARATUS USED** | S/W- Jupyter Notebook/ Weka/ Python  OS-LINUX 64 bit OS  H/W: Core 2 DUO/i3/i5/i7 64-bit processor |
| **REFERENCES** | 1. CHIRAG SHAH, “A HANDS-ON INTRODUCTION TO DATA SCIENCE”,ISBN 978-1-108-47244-9  2. Wes McKinney and the Pandas Development Team, “Pandas: powerful Python data analysis toolkit”  3. https://pandas.pydata.org/ |
| **STEPS** | **Refer to student activity flow chart if found necessary by subject teacher and relevant to the subject manual.**  **Describe steps only.** |
| **INSTRUCTIONS FOR WRITING JOURNAL** | 1. Title 2. Problem statement 3. Learning objective 4. Learning outcome 5. Theory (includes methods, libraries and functions, 6. Analysis (as per assignment), 7. Conclusion. |

1. Title: Data Wrangling II

2. 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 at least 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.

Reason and document your approach properly.

3. Learning objective:

1. To do data cleansing on the given dataset.

2. To find outliers and deal with them

3. To apply transformations for scaling/removing nonlinear relationship to linear/ to decrease skewness.

4. Learning outcome:

After performing this assignment students will be able to:

* Find missing values and inconsistencies in the dataset and apply data cleansing on it.
* Find outliers in the dataset and deal with them.
* Apply data transformations such as scaling, removing nonlinear relationships, decrease skewness and convert to normal distribution.

5. Theory:

Libraries: Pythhon libraries Pandas, matplotlib, sklearn,seaborn

Pandas: **Pandas** is a Python library. **Pandas** is used to analyze data. Pandas allows importing data from various file formats such as comma-separated values, JSON, SQL database tables or queries, and Microsoft Excel.

Matplotlib: **Matplotlib** is a comprehensive library for creating static, animated, and interactive visualizations in Python.

Sklearn: Scikit-learn (Sklearn) is the most**useful and robust library for machine learning in Python**. It provides a selection of efficient tools for machine learning, preprocessing and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

Seaborn: Seaborn is an open-source Python library built on top of matplotlib. It is used **for data visualization and exploratory data analysis**. Seaborn works easily with dataframes and the Pandas library. Graphs can help us find data trends that are useful in any machine learning or forecasting project.

Methods and functions:

Pandas.read\_csv(): to load csv file into Pandas dataframe.

Isnull(): to check presence of null values in the dataset.

skew(): The DataFrame class of pandas has a **method skew() that computes the skewness of the data present in a given axis of the DataFrame object**. Skewness is computed for each row or each column of the data present in the DataFrame object.

distplot(): The seaborn. distplot() function is used to plot the distplot. The distplot represents the **univariate distribution of data** i.e. data distribution of a variable against the density distribution.

Dealing with Outliers:

Outlier is an observation in a given dataset that lies far from the rest of the observations.

Techniques of detecting outliers are:

* Boxplots
* Z-score
* Inter Quantile Range(IQR)

Methods of treating the outliers:

1. Z-score:is also called a standard score. This value/score helps to understand that how far is the data point from the mean. And after setting up a threshold value one can utilize z score values of data points to define the outliers. Now to define an outlier threshold value is chosen which is generally 3.0. As 99.7% of the data points lie between +/- 3 standard deviation (using Gaussian Distribution approach). Data values above the threshold value are to be dealt with.

2. IQR: Inter Quartile Range approach to finding the outliers is the most commonly used and most trusted approach used in the research field.

Q1 **=** np.percentile(df[column\_name], 25, interpolation **=** 'midpoint')

Q3 **=** np.percentile(df[column\_name], 75, interpolation **=** 'midpoint')

IQR **=** Q3 **-** Q1

To define the outlier base value is defined above and below datasets normal range namely Upper and Lower bounds, define the upper and the lower bound (1.5\*IQR value is considered) :

*upper = Q3 +1.5\*IQR*

*lower = Q1 – 1.5\*IQR*

Deal with the values below lower and above upper.

**Data Scaling:**

**MinMaxScaler(feature\_range = (0, 1))** will transform each value in the column proportionally within the range [0,1] . Use this as the first scaler choice to transform a feature, as it will preserve the shape of the dataset (no distortion).

**StandardScaler()** will transform each value in the column to range about the mean 0 and standard deviation 1, ie, each value will be normalised by subtracting the mean and dividing by standard deviation. Use StandardScaler if you know the data distribution is normal.

If there are outliers, use **RobustScaler()**. Alternatively you could remove the outliers and use either of the above 2 scalers (choice depends on whether data is normally distributed)

### Using The min-max feature scaling:

The min-max approach (often called normalization) rescales the feature to a hard and fast range of [0,1] by subtracting the minimum value of the feature then dividing by the range.



Standardization doesn’t have any fixed minimum or maximum value. Here, the values of all the columns are scaled in such a way that they all have a mean equal to 0 and standard deviation equal to 1. This scaling technique works well with outliers. Thus, this technique is preferred if outliers are present in the dataset.

6. Analysis:

Observe % of missing values and outliers in the dataset. Observe skewness or plot density distribution of each column and see if it normal or skewed. After data cleansing and applying data transformation plot density distribution to see the change.

7. Conclusion:

Using python libraries Pandas, matplotlib, sklearn data is cleaned after finding missing values and inconsistancies in the dataset, outliers are dealt with and data transformations for scaling are applied, skewed columns are converted to normal distribution.