DATA PREPROCESSING

AIM: To understand and perform the basic data pre-processing operations on a given dataset.

THEORY: Data pre-processing is a data mining technique that performs a series of operations to transform the raw data in a useful and efficient format. These transformations are applied to the data before feeding it to algorithm. Whenever data is gathered from different sources, it is collected in raw format, which is not feasible for analysis. Pre-processing converts raw format into readable format (graphs, documents, etc.), so that it can be interpreted by computers and utilized by employees throughout an organization.

STEPS INVOLVED IN DATA PRE-PROCESSING:

1. DATA CLEANING: The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

(a) Missing Data: This situation arises when some data is missing in the data. It can be handled in various ways. Some of them are:

- **i.** Ignore the tuples: This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.
- **ii.** Fill the Missing values: There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

(b) Noisy Data: Noisy data is a meaningless data that cannot be interpreted by machines. It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways:

i. Binning Method: This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

ii. Regression: Here data can be made smooth by fitting it to a regression function. The regression used may be linear (having one independent variable) or multiple (having multiple independent variables).

iii. Clustering: This approach groups the similar data in a cluster. The outliers may be undetected or it will fall outside the clusters.

2. DATA TRANSFORMATION: This step is taken in order to transform the data in appropriate forms suitable for mining process. This involves following ways:

(a) Normalization: It is done in order to scale the data values in a specified range (-1.0 to 1.0 or 0.0 to 1.0)

(b) Attribute Selection: In this strategy, new attributes are constructed from the given set of attributes to help the mining process.

(c) Discretization: This is done to replace the raw values of numeric attribute by interval levels or conceptual levels.

(d) Concept Hierarchy Generation: Here attributes are converted from lower level to higher level in hierarchy. For Example-The attribute "city" can be converted to "country."

3. DATA REDUCTION: Since data mining is a technique that is used to handle huge amount of data. While working with huge volume of data, analysis became harder in such cases. In order to get rid of this, we use data reduction technique. It aims to increase the storage efficiency and reduce data storage and analysis costs. The various steps to data reduction are:

(a) Data Cube Aggregation: Aggregation operation is applied to data for the construction of the data cube.

(b) Attribute Subset Selection: The highly relevant attributes should be used, rest all can be discarded. For performing attribute selection, one can use level of significance and p- value of the attribute. The attribute having p-value greater than significance level can be discarded.

(c) Numerosity Reduction: This enables to store the model of data instead of whole data, for example: Regression Models.

(d) Dimensionality Reduction: This reduces the size of data by encoding mechanisms. It can be lossy or lossless. If after reconstruction from compressed data, original data can be retrieved, such reduction are called lossless reduction else it is called lossy reduction. The two effective methods of dimensionality reduction are: Wavelet transforms and PCA (Principal Component Analysis).

DATA CLEANING USING WEKA

Some of the common filters available in Weka for data pre-processing are:

1. Replace Missing Values: Replaces all missing values for nominal and numeric attributes in a datasetwith the modes and means from the training data.

2. Remove With Values: Filters instances according to the value of an attribute.

3. Inter Quartile Range: A filter for detecting outliers and extreme values based on interquartile ranges.

4. Discretize: An instance filter that discretizes a range of numeric attributes in the dataset into nominal attributes.

5. Normalize: Normalizes all numeric values in the given dataset (apart from the class attribute, if set).

6. Nominal To Binary: Converts all nominal attributes into binary numeric attributes.

7. Numeric To Nominal: A filter for turning numeric attributes into nominal ones.

8. Remove: A filter that removes a range of attributes from the dataset.

9. Remove By Name: Removes attributes based on a regular expression matched against their names butwill not remove the class attribute.

10. Rename Attribute: This filter is used for renaming attributes.

IMPLEMENTATION:

1. DATA CLEANING USING WEKA

1.1. Missing Values: These values are handled in two ways

(a) Replace missing values:

Steps:

- Load the `soybean.arff` dataset
- Select the `plant-stand` attribute (Note the % of missing values and save a screenshot of the selected attributes sub window)
- In the filters sub window click on Choose button and select filters→unsupervised→attributes→ReplaceMissingValues→Apply
- Again, select the `plant-stand` attribute (Note the % of missing values and save a screenshot of the selected attributes sub window)
- Save the processed dataset with a new name in a folder of your own.

(b) Remove instances with missing values:

Steps:

- Load the `soybean.arff` dataset

- Select the `plant-stand` attribute (Note the % of missing values and save a screenshot of the selected attributes subwindow)
- In the filters subwindow click on Choose button and select filters→ unsupervised→instances→RemoveWithValues
- Before Applying the filter, click on the box with the selected filter. A pop-up window appears. Assign the following values in the window:

  `attributeIndex = 2`(index is 2 for attribute plant-stand)
  `invertSelection = True`
  `matchMissingValues = True`

- Click on Apply, after setting the above values.
- Again, select the `plant-stand` attribute (Note the % of missing values and save a screenshot of the selected attributes subwindow)
- Save the processed dataset with a new name in a folder of your own.

1.2. Outlier Data:

(a) <u>Identify outlier and extreme values:</u>

<u>Steps:</u>

- Load the `cpu.arff` dataset
- Save screenshot of the Attributes subwindow.
- In the filters subwindow click on Choose button and select filters→ unsupervised→attributes→InterQuartileRange→Apply
- Save screenshots of the newly generated attributes: Outlier and ExtremeValue from the selected attribute subwindow.
- Save the processed dataset with a new name in a folder of your own.

(b) <u>Removing outliers:</u>

<u>Steps:</u>

- Load the processed dataset from 2.2.(a)
- Save screenshots of the attributes: Outlier and ExtremeValue from the selected attribute subwindow.
- In the filters subwindow click on Choose button and select filters→ unsupervised→instances→RemoveWithValues
- Before Applying the filter, click on the box with the selected filter. A pop-up window appears. Assign the following values in the window:

  `attributeIndex = 9` (index is 9 for attribute `Outlier`)
  `nominalIndices = last`

- Click on Apply, after setting the above values.
- Save screenshots of the attributes: Outlier and ExtremeValue from the selected attribute subwindow.
- Save the processed dataset with a new name in a folder of your own.

2. DATA TRANSFORMATION USING WEKA

2.1. Discretization:

Steps:

- Load the `credit-g.arff` dataset
- Save screenshot of the attribute: `age` from the `Selected Attribute` subwindow.
- In the filters subwindow click on Choose button and select filters→ unsupervised→attributes→Discretize
- Before Applying the filter, click on the box with the selected filter. A pop-up window appears. Assign the following values in the window:

  `attributeIndex = 13` (index is 13 for attribute `age`)
  `binRangePrecision = 2`
  `bins = 3`
- Click on Apply, after setting the above values.
- Save screenshots of the attribute `age` from the `Selected Attribute` subwindow.
- Save the processed dataset with a new name in a folder of your own.

2.2. Normalization

Steps:

- Load the `iris.arff` dataset
- Click on the Edit button and save a screenshot of the data values.
- In the Attributes subwindow, select the attribute `sepallength`.
- Save screenshot of the attribute: `sepallength` from the `Selected Attribute` subwindow.
- In the filters subwindow click on Choose button and select filters→ unsupervised→attributes→Normalize→Apply.
- Click on the Edit button and save a screenshot of the data values.
- Save screenshots of the attribute `sepallength` from the `Selected Attribute` subwindow.
- Save the processed dataset with a new name in a folder of your own.