

**Experiment No.1 Title:** Data Pre-processing

# Batch: B4 Roll No.: 16010420117 Experiment No.:1

**Aim**: Data pre-processing by applying data normalization and data discretization

**Resources needed:** Any RDBMS, Java

# Theory:

Data processing techniques, when applied before mining, can substantially improve the overall quality of the patterns mined and/or the time required for the actual mining.

Different kinds of pre-processing tasks are performed on the data before applying mining techniques. Data reduction can reduce data size by, for instance, aggregating, eliminating redundant features, or clustering. *Data transformations* (e.g., normalization) may be applied, where data are scaled to fall within a smaller range like 0.0 to 1.0. This can improve the accuracy and efficiency of mining algorithms involving distance measurements. These techniques are not mutually exclusive; they may work together. Normalization, data discretization, and concept hierarchy generation are forms of data transformation.

# Data Reduction:

Data reduction techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data. That is, mining on the reduced data set should be more efficient yet produce the same (or almost the same) analytical results. Data reduction strategies include dimensionality reduction, numerosity reduction, and data compression.

Dimensionality reduction is the process of reducing the number of random variables or attributes under consideration. Dimensionality reduction methods include wavelet transforms and principal components analysis, which transform or project the original data onto a smaller space. Attribute subset selection is a method of dimensionality reduction in which irrelevant, weakly relevant, or redundant attributes or dimensions are detected and removed.

Numerosity reduction techniques replace the original data volume by alternative, smaller forms of data representation. These techniques may be parametric or nonparametric. For parametric methods, a model is used to estimate the data, so that typically only the data parameters need to be stored, instead of the actual data. (Outliers may also be stored.) Regression and log-linear models are examples. Nonparametric methods for storing reduced representations of the data include histograms, clustering, sampling, and data cube aggregation.

# Data Normalization:

The measurement unit used can affect the data analysis. For example, changing measurement units from meters to inches for *height*, or from kilograms to pounds for *weight*, may lead to very different results. In general, expressing an attribute in smaller units will lead to a larger range for that attribute, and thus tend to give such an attribute greater effect or “weight.” To

help avoid dependence on the choice of measurement units, the data should be *normalized* or *standardized*. This involves transforming the data to fall within a smaller or common range such as [-1, 1] or [0.0, 1.0].

In ***z*-score normalization** (or *zero-mean normalization*), the values for an attribute, *A*, are normalized based on the mean (i.e., average) and standard deviation of *A*. A value, *vi* , of *A* is normalized to *v*’i by computing ,



where σ*A* and 𝐴̅ are the mean and standard deviation, respectively, of attribute *A*. This method of normalization is useful when the actual minimum and maximum of attribute *A* are unknown, or when there are outliers that dominate the min-max normalization. **z-score normalization.** Suppose that the mean and standard deviation of the values for the attribute *income* are $54,000 and $16,000, respectively. With z-score normalization, a value of

$73,600 for *income* is transformed to,



# Data Discretization:

In data descretization, the raw values of a numeric attribute (e.g., *age*) are replaced by interval labels (e.g., 0–10, 11–20, etc.) or conceptual labels (e.g., *youth, adult*, *senior*). The labels, in turn, can be recursively organized into higher-level concepts, resulting in a *concept hierarchy* for the numeric attribute.

# Binning:

Binning is a top-down splitting technique based on a specified number of bins. These methods are also used as discretization methods for data reduction and concept hierarchy generation. For example, attribute values can be discretized by applying equal-width or equal-frequency binning, and then replacing each bin value by the bin mean or median, as in *smoothing by bin means* or *smoothing by bin medians*, respectively. These techniques can be applied recursively to the resulting partitions to generate concept hierarchies. Binning does not use class information and is therefore an unsupervised discretization technique. It is sensitive to the user-specified number of bins, as well as the presence of outliers. Example is shown in figure 1.

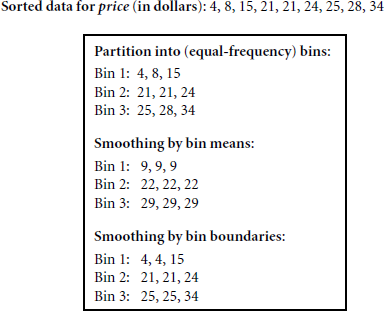




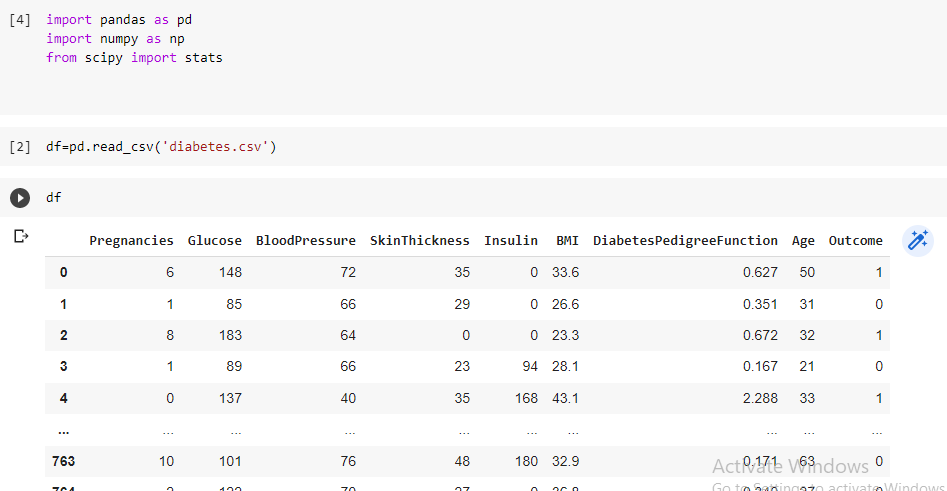
Fig 1. Example of binning Other methods of Discretization are,

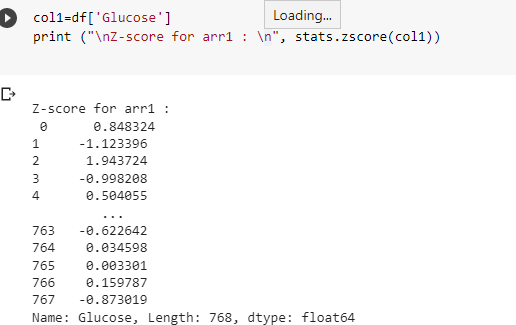
* Histogram Analysis
* Discretization by Cluster, Decision Tree, Correlation Analyses

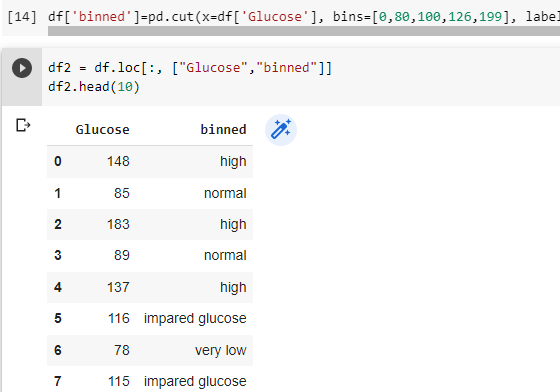
**Procedure / Approach /Algorithm / Activity Diagram:**

1. Identify attribute suitable for normalization and discretization
2. Apply Z- score normalization on your dataset.
3. Apply discretization using Binning technique

# Results: (Program printout with output / Document printout as per the format)







**Questions:**

1. Explain with example Min-Max normalization technique.

Ans: Min Max is a data normalising method similar to Z-score, decimal scaling, and standard deviation normalisation. The data is helped to be normalised. The data will be scaled between 0 and 1. We can more easily interpret the data thanks to this standardisation.

For instance, it's a little bit more perplexing if I ask you to tell me the difference between 200 and 1000 than it is if I ask you to tell me the difference between 0.2 and 1.

## Min Max normalization formula

|  |
| --- |
| ****marks**** |
| 8 |
| 10 |
| 15 |
| 20 |

**Min:**

The minimum value of the given attribute. Here Min is **8**

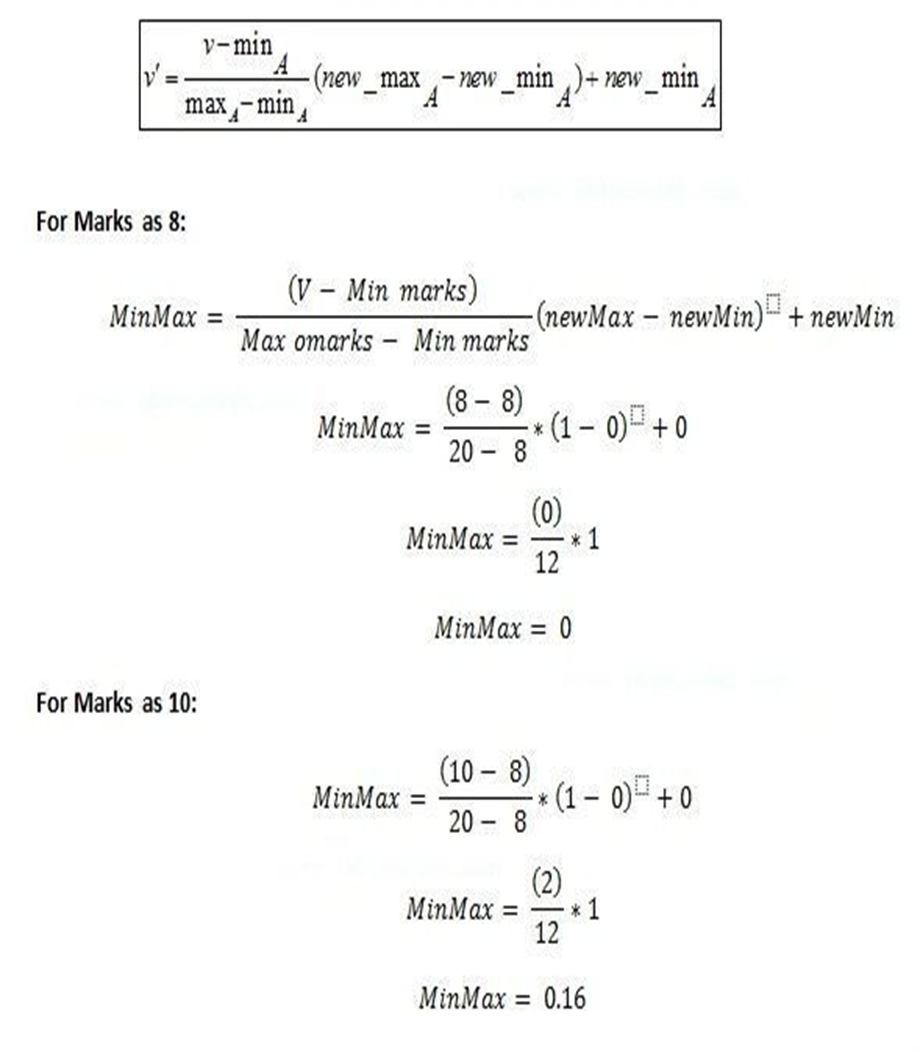
**Max:**

The maximum value of the given attribute. Here Max is **20**

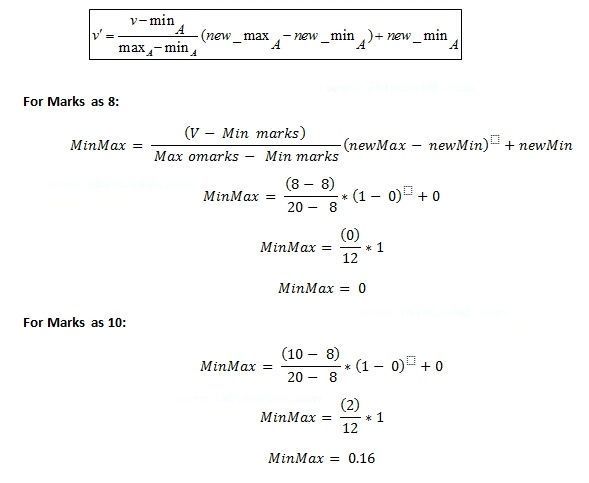
**V:** V is the respective value of the attribute. For example here V1=8, V2=10, V3=15, and V4=20

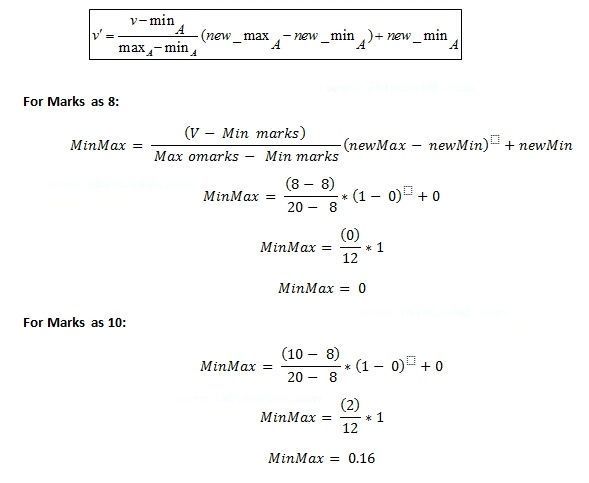
**newMax:**

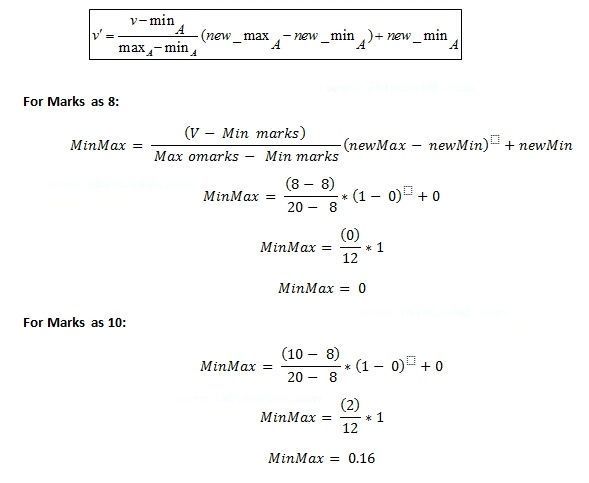
1

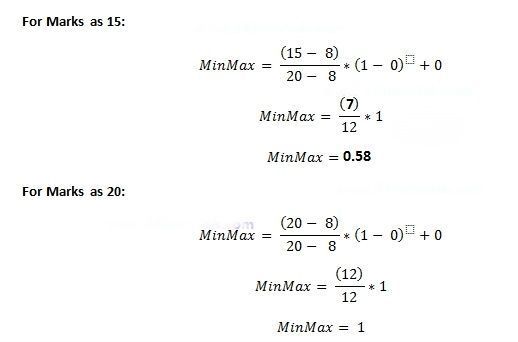


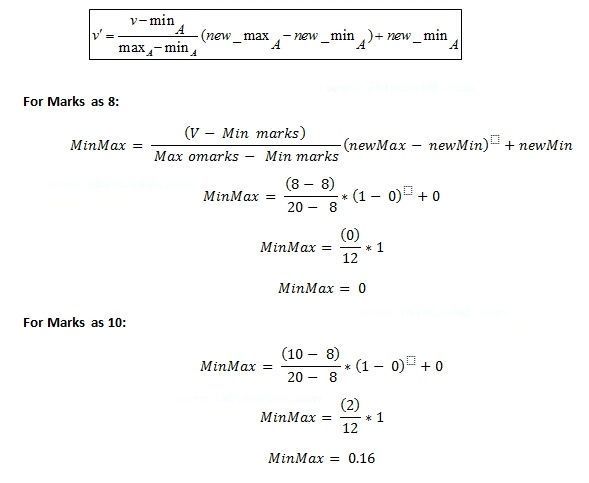
**newMin:**

**0**



****

****

****

# Outcomes: CO1:

**Conclusion: Understood and Implemented data preprocessing successfully.**



**Grade: AA / AB / BB / BC / CC / CD /DD**

Signature of faculty in-charge with date

# References:

Books/ Journals/ Websites:

* 1. Han, Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann 3nd Edition