

**AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY (AUST)**  
**141 & 142, Love Road, Tejgaon Industrial Area, Dhaka-1208.**



Department of Computer Science and Engineering  
Program: Bachelor of Science in Computer Science and Engineering

Course No: 4142 Course Title: Data Warehousing and Mining Lab

## **Assignment 2**

Date of Submission: 25/5/2024

**Submitted by,**

Name: Md Rafiu Alam Rafi

Id: 20200204051

Section: A

## Task 1: Create a Custom Dataset Which Will Have 5 Attributes: 2 Numeric, 2 Nominal & 1 Class (3 Class Values)

In the notepad, I created a file where the dataset name “Student\_Performance” is defined by @relation and the attributes are defined by @attribute. There are 5 attributes and those are age, gender, exam\_score, part\_time\_job, class\_performance.

## Task 2: Create 20 Instances of That Dataset Which Should Have Some Missing Values inside Any 2 Attributes + Make 10 Instances of 1st Class Value, 6 Instances of 2nd Class Value & Rest of the Instances Should be of 3rd Class Value

The attributes are different types and the data are written respectively.

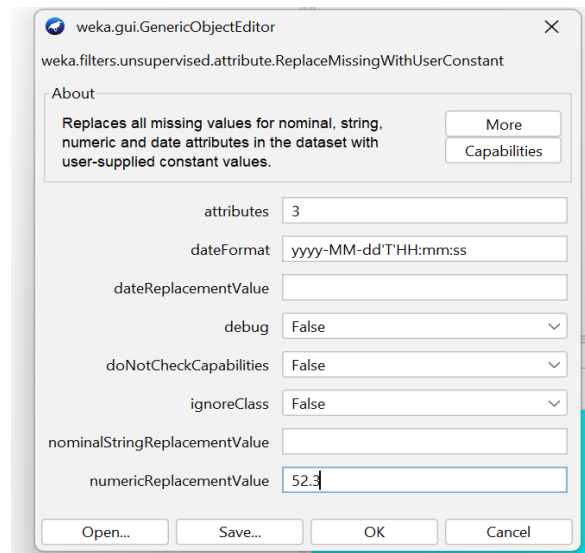
```
@relation Student_Performance

@attribute age numeric
@attribute gender {male, female}
@attribute exam_score numeric
@attribute part_time_job {yes, no}
@attribute class_performance {poor, average, excellent}

@data
21, male, 65.5, yes, poor
25, female, 60.8, no, poor
21, female, 63.2, no, poor
26, female, 55.4, no, poor
21, female, 58.5, no, poor
23, male, 50.0, yes, poor
24, female, 59.0, no, poor
20, male, 57.3, yes, poor
22, male, 56.1, no, poor
19, female, 61.2, no, poor
19, female, 72.3, yes, average
20, female, ?, no, average
?, female, 68.7, yes, average
24, male, 70.9, no, average
22, female, 73.9, no, average
23, male, ?, yes, average
22, female, 79.8, no, excellent
19, male, 90.0, yes, excellent
20, female, 82.7, no, excellent
24, male, 77.3, yes, excellent
```

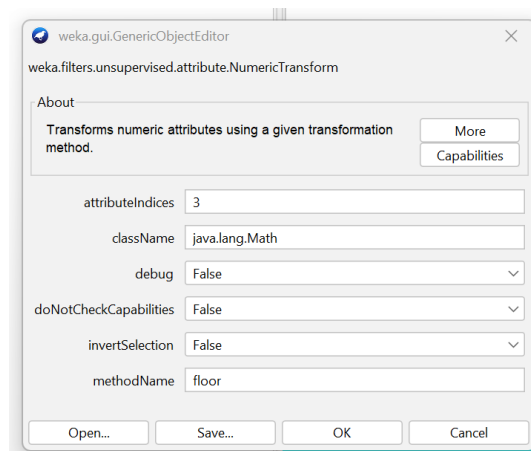
### Task 3: Using Preprocessing Tab, Fill-Out Those Missing Values using Your Preferred Values

For filling the missing values, I went to Filters -> unsupervised -> “ReplaceMissingWithUserConstant”. Then I put value (3) for rating attribute and value (52.3) to fill the missing values.



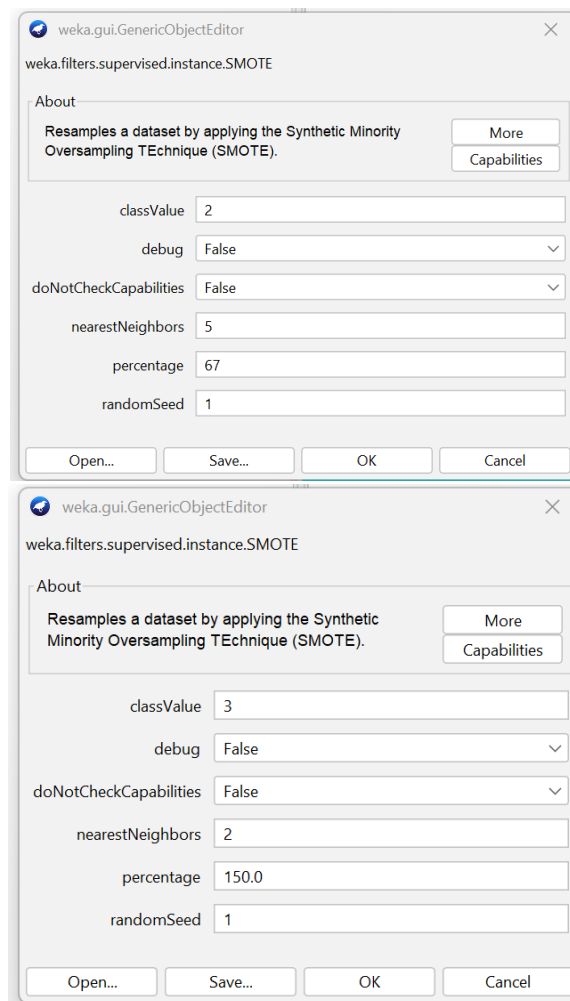
### Task 4: Convert Any 1 Real Attribute's Values from Float to Integers (which is less than or equal to the original value)

For converting from float to integer, I went to Filters -> unsupervised -> “NumericTransformation”. Then I put value (3) for rating attribute and “floor” to make the float values transfer into integer values.



## Task 5: Fix the Class Imbalance Problem for the 2nd and 3rd Class by Making the Number of Instances for 2nd Class and 3rd Class Equal as the Number of Instances for 1st Class (10)

Here, I went to Filters -> supervised -> "SMOTE". As input, ClassValue = 2 for the class "avarage", nearNeighbours = 5 and percentage = 67% to make the "poor" class value equal with the "avarage" where there are 10 instances. Similarly, 150% was used for "excellent" class values as there was 4 values of that class.



## Task 6: Apply Any Classification Algorithm on the Modified Dataset (Use 5-Fold Cross Validation)

Here, I went to Classify -> Choose -> DecisionStump and input 5 for Cross Validation folds.

The screenshot shows the Weka Explorer interface with the 'Classify' tab selected. The 'DecisionStump' classifier is chosen, and 'Cross-validation' is set to 5 folds. The 'Result list' shows '23.0746 - trees.DecisionStump'. The 'Classifier output' pane displays the following information:

**Decision Stump**

**Classifications**

```
exam_score <= 75.0 : poor
exam_score > 75.0 : excellent
exam_score is missing : poor
```

**Class distributions**

```
exam_score <= 75.0
poor    average excellent
0.5     0.5     0.0
exam_score > 75.0
poor    average excellent
0.0     0.0     1.0
exam_score is missing
poor    average excellent
0.3333333333333333 0.3333333333333333 0.3333333333333333
```

**Time taken to build model: 0 seconds**

**Stratified cross-validation Summary**

Metric	Value
Correctly Classified Instances	20
Incorrectly Classified Instances	10
Kappa statistic	0.5
Mean absolute error	0.2222
Root mean squared error	0.3333
Relative absolute error	50 %
Root relative squared error	70.7107 %
Total Number of Instances	30

**Detailed Accuracy By Class**

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	FRC Area	Class
	1.000	0.500	0.500	1.000	0.667	0.500	0.750	0.500	poor
	0.000	0.000	?	0.000	?	?	0.750	0.500	average
	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	excellent
Weighted Avg.	0.667	0.167	?	0.667	?	?	0.833	0.667	

**Confusion Matrix**

```
a b c <- classified as
10 0 0 | a = poor
10 0 0 | b = average
0 0 10 | c = excellent
```

Finally, we got this output.