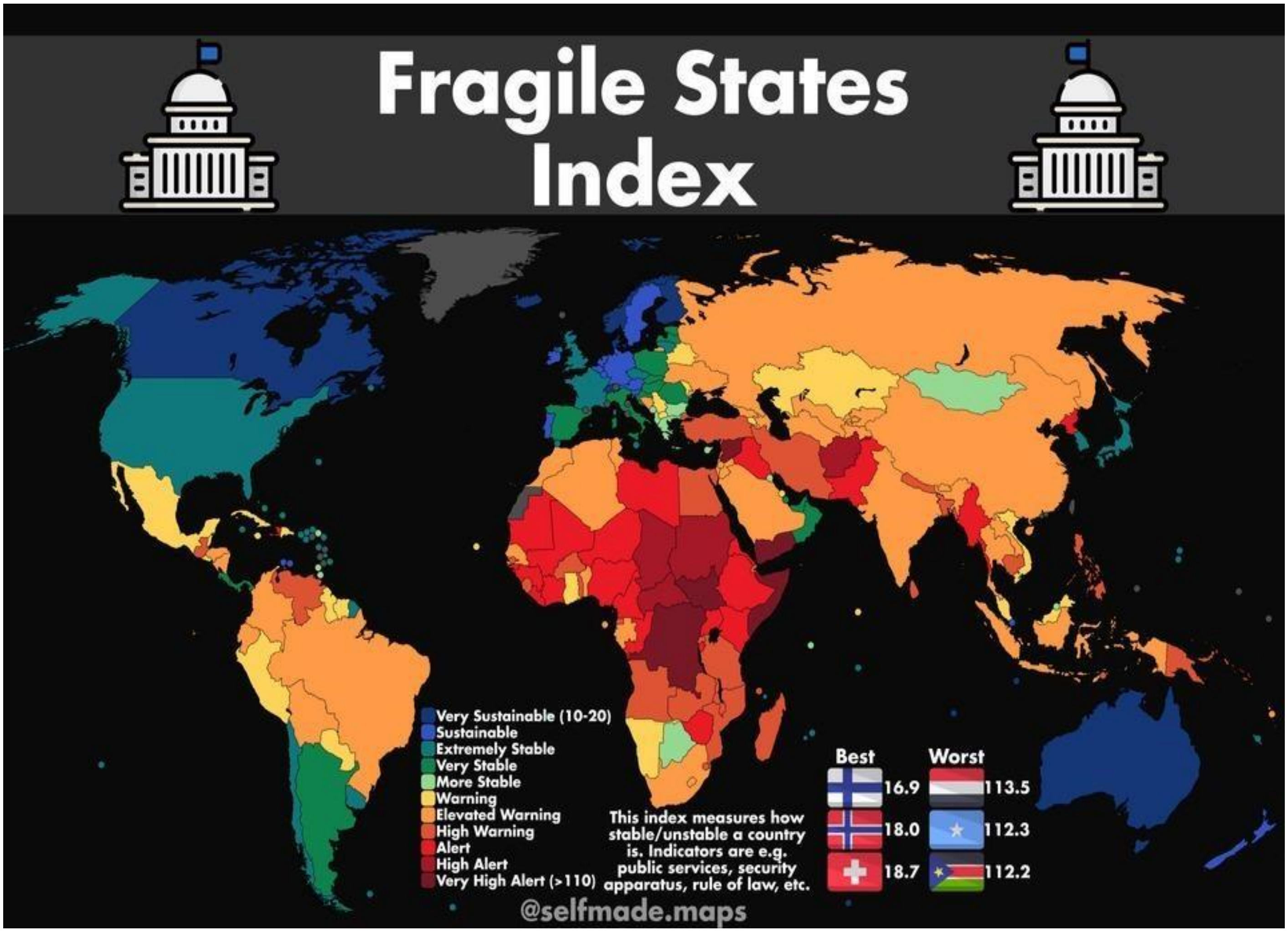


Analysis on Fragile States Data

Department of Computer Science
University of North Carolina Charlotte, NC, USA

Instructor
Dr. Zbigniew W. Ras

TEAM MEMBERS:

Shreeya Gupta (801136226)
Yash Kanodia (801136232)
Anushka Tibrewal (801134231)
Ridhi Shah (801151577)

Problem Statement:

The problem statement describes the Fragile State Index as the lower the index score the better. To get a lower Index score, we need to discretize our Decision feature(i.e `Total' column), which will then help us in finding the action rules. The action rules with the best support will give us the changes required in the classification feature to lower the index score, we are also required to add six new features in correlation with the data given. We will then analyze the results with and without adding the new features by running different classification algorithms on the data.

Introduction

In this project we were given Fragile State Index (FSI) dataset for all the countries which is available at https://fragilestatesindex.org/excel/ and we have extended the dataset by adding 6 new features namely Inflation rate, Birth rate, Death Rate, GDP Rank, Life Expectancy, Economic Growth. We extracted this data from different sources.

Using the values for these newly added features, and using the feature Total as our decision feature with discretization we replace numbers with concepts such as:

1. Alert
2. Warning
3. Stable
4. Sustainable

Using the above concepts we find the best classifier for our dataset with the help of software WEKA and using classification algorithms like Random Forest, Simple Logistic and K-star. After finding the best classifier, we find the action rules and from these rules we can get the action which needs to be performed to change the state of the country from Alert to Stable or Sustainable.

Dataset

The dataset for the project was obtained from https://fragilestatesindex.org/excel/ , we used the data for the year 2019. We downloaded the csv files of the features which correlated with our data from different source on the web and entered the new features to our dataset. The 6 new features are as follows:

1. Birth Rate : The ratio of live births in an area to the population of that area, expressed per 1000 population per area.
2. GDP Rank: It is the monetary market value of all final goods and services made within a country during a specific period.
3. Economic Growth : The economic growth for a country of one year, comparing one quarter of the country's gross domestic product to the previous quarter.
4. Death Rate: The ratio of live deaths in an area to the population of that area, expressed per 1000 population per area

5. Inflation Rate: The increase in consumer prices for the country in a year.
6. Life Expectancy : The average number of years to be lived by a group of people born in the same year, if mortality at each age remains constant in the future. Life expectancy at birth is also a measure of overall quality of life in a country and summarizes the mortality at all ages.

Motivation to select these features :
We searched for various features available on the Wikipedia, worldbank, and some other websites which provide us with different features which can fit with the previous features available in the Fragile State index 2019.
We performed different operations on six features, which we selected randomly and tried applying different classifiers to these features to check whether the accuracy increases or decreases based on the features selected.
After performing these operations, we got these six features:
1. Birth Rate
2. Death Rate
3. Gross Domestic Product (GDP)
4. Economic Growth
5. Inflation Rate
6. Life Expectancy

Adding these features to the previous data and applying different classifiers like random forest, Decision tree, simple Logistic Regression, Naïve Bayes, JRip and J48 on the data helped to improve the accuracy i.e the percentage of correctly classified class increases by a good margin and the percentage of incorrectly classified class decreases. After noting these accuracies, we performed operations to extract classification rules and action rules in this project. The extraction of classification and action rules are represented below.

DATA DISCRETIZATION (Using WEKA):

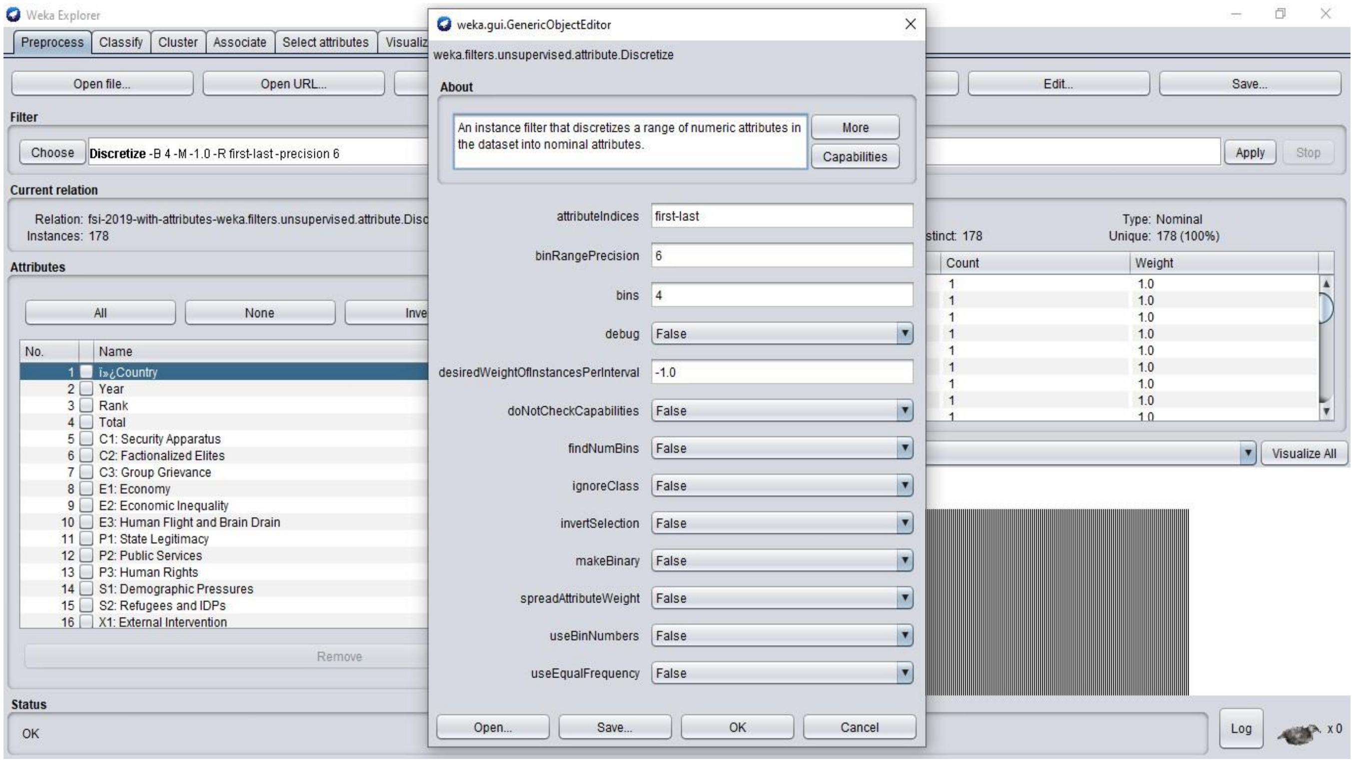
We used the WEKA tool for discretization as well as classification of our data.

Discretization alludes to the way toward changing over or parceling nonstop properties, highlights, or factors to discrete or ostensible qualities. The decision attribute “TOTAL” has continuous values throughout the dataset ranging from 0 to 120; we have used discretization to replace these numeric values with the following categories.

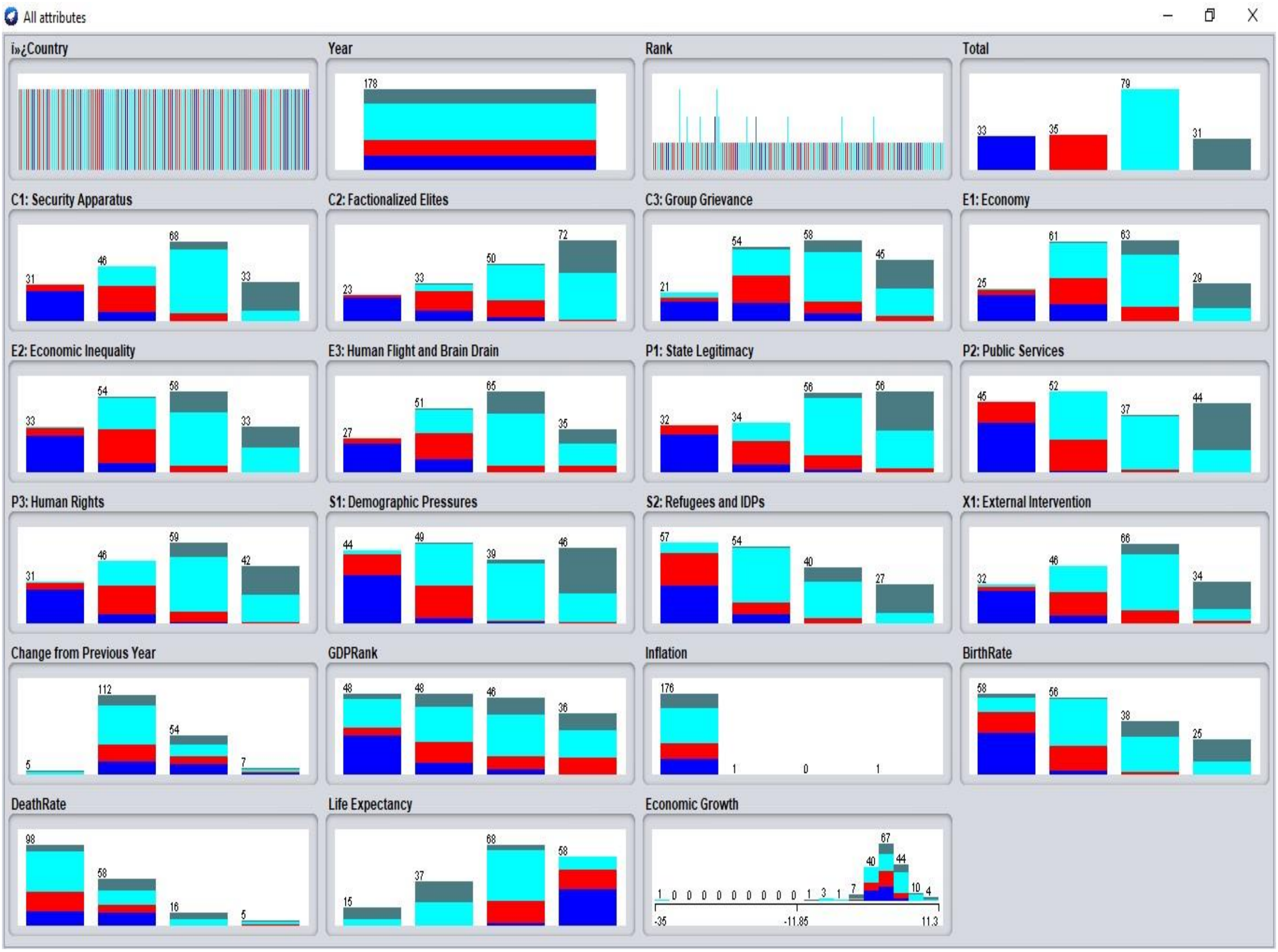
1. Alert: 90 - 120
2. Warning: 60 - 90
3. Stable: 30 - 60
4. Sustainable: 0 - 30

Source: https://en.wikipedia.org/wiki/Fragile_States_Index

We also had to discretize other attributes from numeric values to nominal values to perform effective classification.



Discretized data



DATA CLASSIFICATION (Using WEKA)

Data Classification is the way toward sorting out information by applicable classes with the goal that it might be utilized and ensured all the more proficiently. Following classification algorithms are used for data classification:

1. Naive Bayes: Naive Bayes is a classification algorithm. Traditionally it assumes that the input values are nominal, although it numerical inputs are supported by assuming a distribution. It is a basic strategy for building classifiers models that allocate class names to issue occasions, spoke to as vectors of highlight esteems, where the class marks are drawn from some limited set.

2. Random Forest: Random forests is the learning strategy for arrangement, relapse and different errands that works by building a large number of choice trees at preparing time and yielding the class that is the method of the classes (order) or mean expectation (regression) of the individual trees.

3. Decision Table: Decision Table classifier calculation is utilized to sum up the dataset by utilizing a decision table containing a similar number of properties as that of the first dataset. It is a short visual portrayal for indicating which activities to perform contingent upon given conditions.

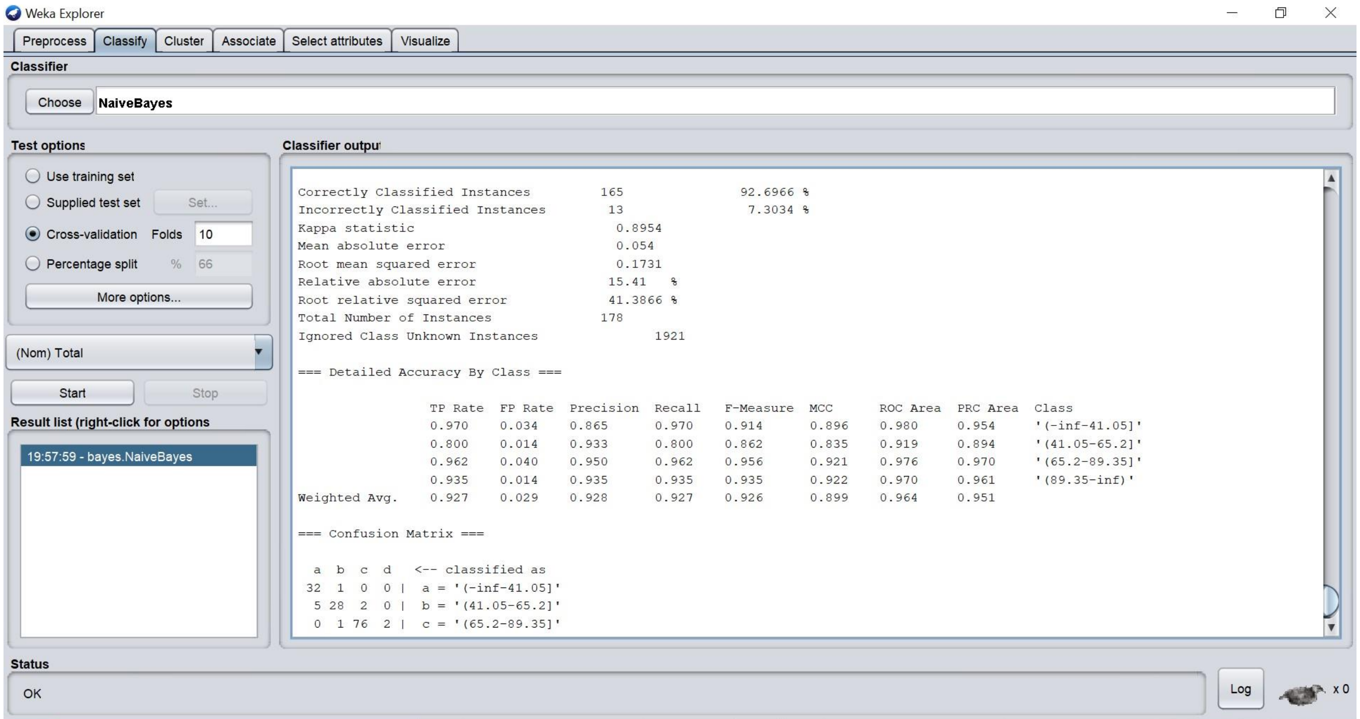
4. JRip: It depends on affiliation rules with diminished error pruning (REP), extremely normal and powerful system found in decision tree calculations. At each phase of improvement, the pruning administrator picked is the one that yields the best decrease of mistake on the pruning set.

   Class JRip java.lang.Object
   weka.classifiers.AbstractClassifier weka.classifiers.rules.JRip

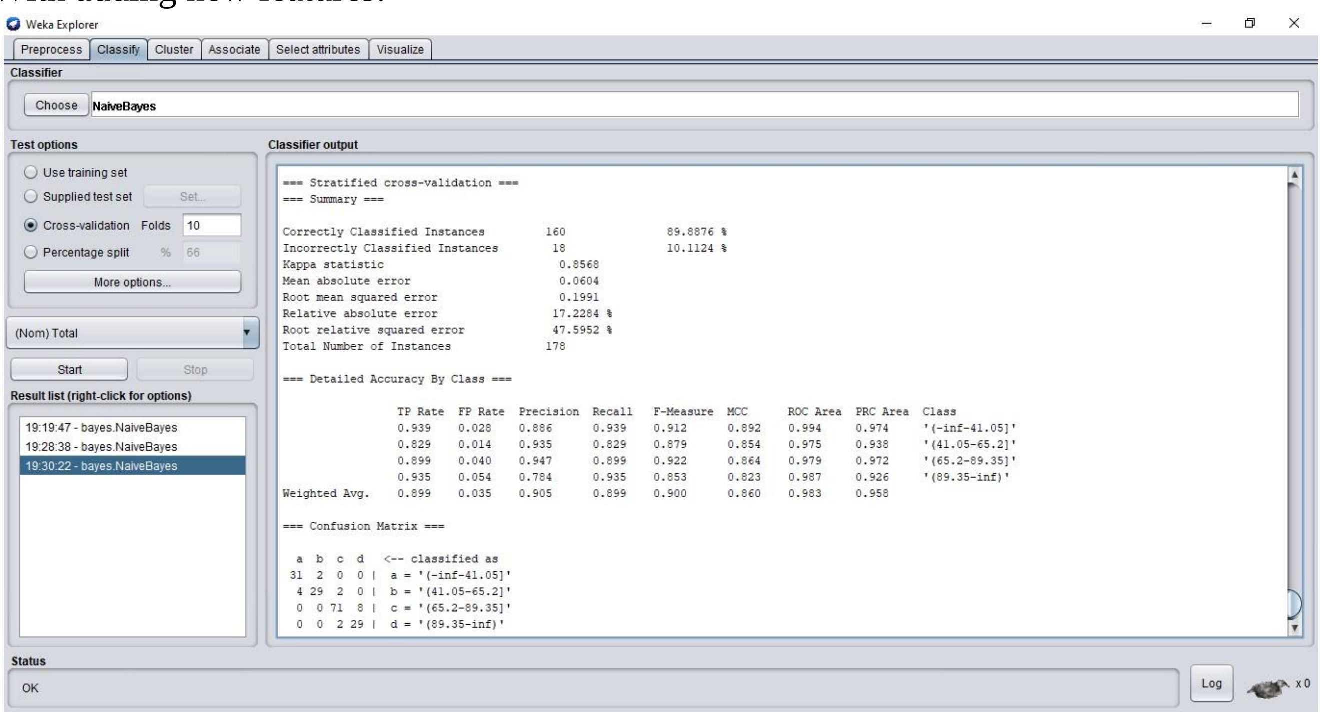
5. J48: The C4.5 calculation for building decision trees is executed in Weka as a classifier called J48. Envision that you have a dataset with a rundown of indicators or free factors and a rundown of targets or ward factors. Decision tree J48 is the usage of calculation Iterative Dichotomiser 3.

1)Naive Bayes:

Without adding new features-

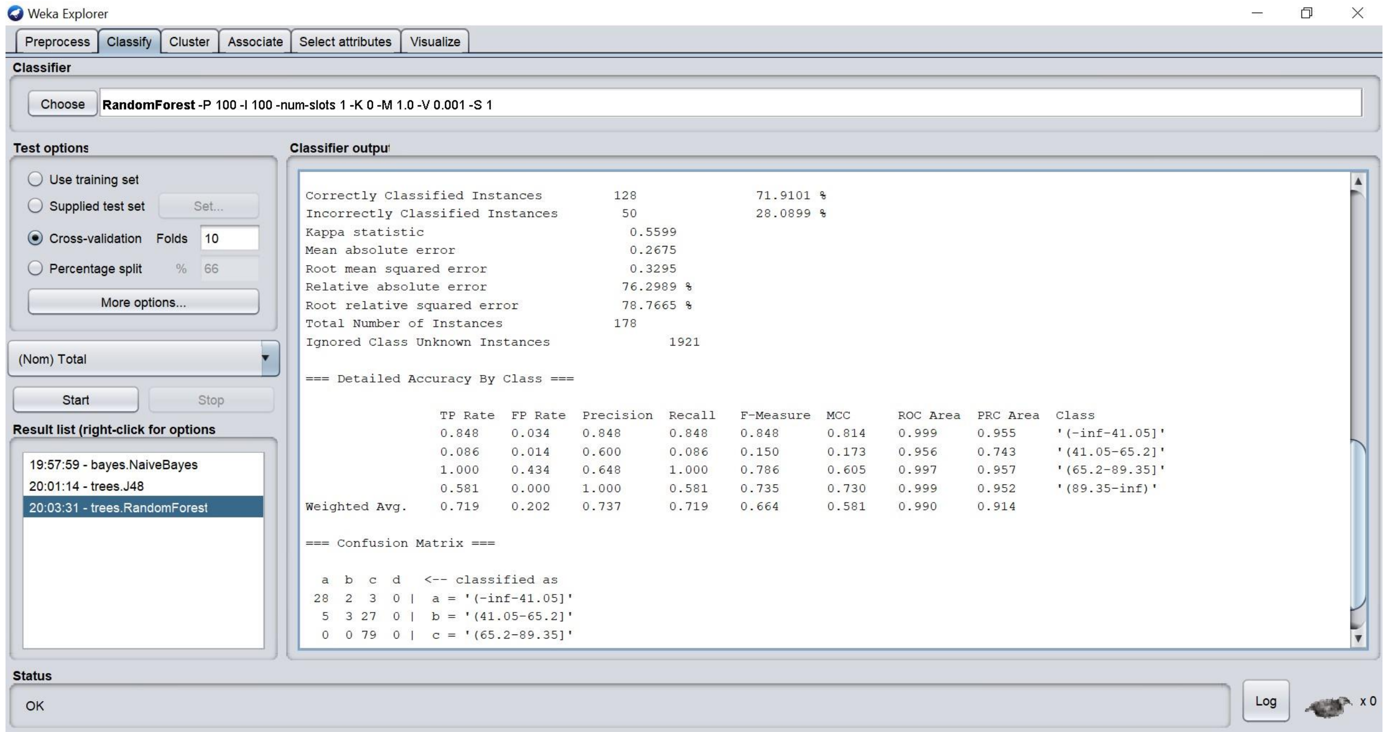


With adding new features:

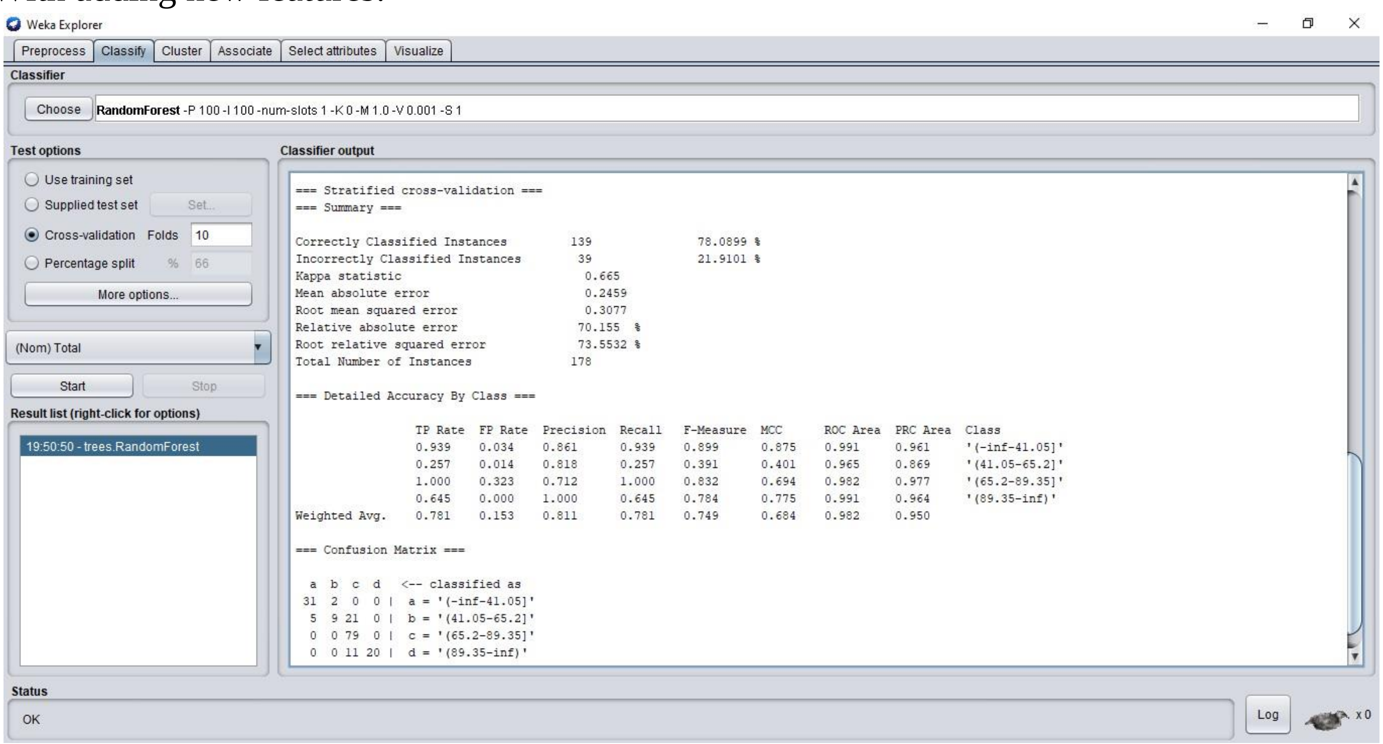


2)Random Forest:

Without adding new features:

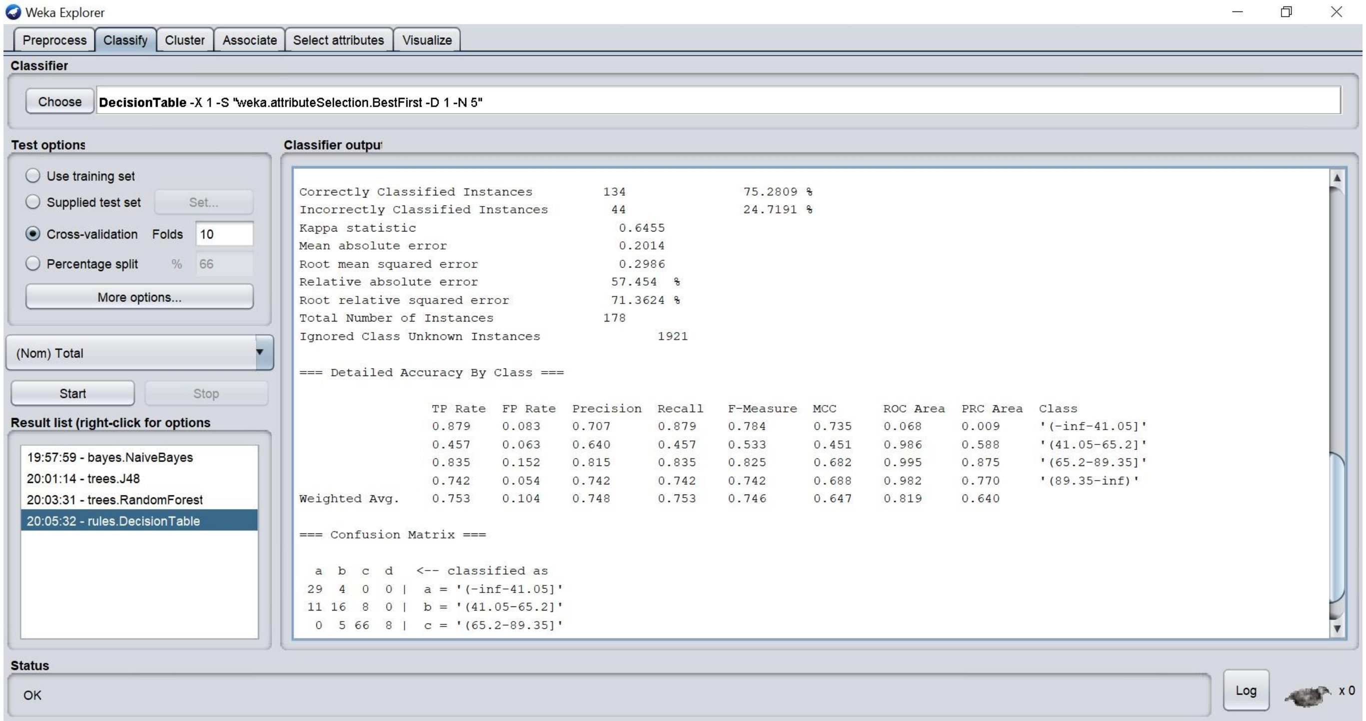


With adding new features:

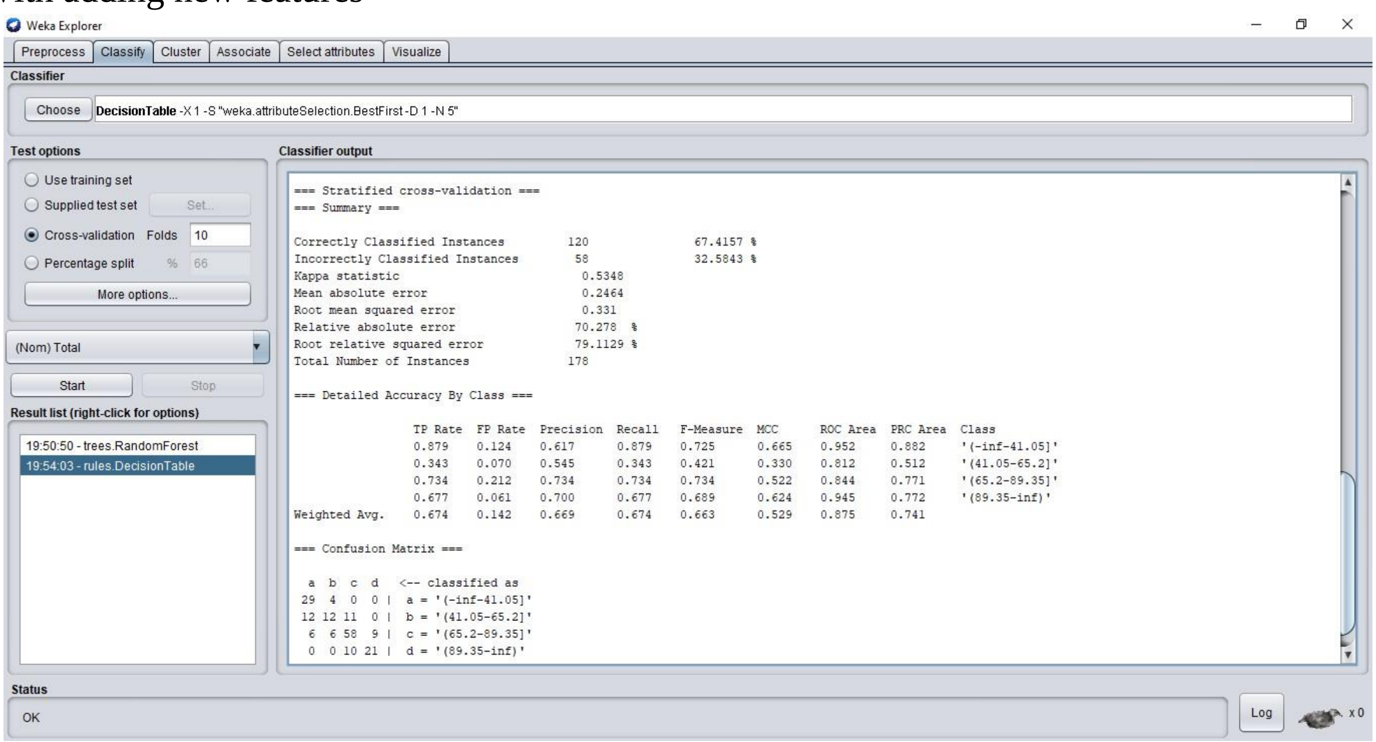


3)Decision Table:

Without adding new features:

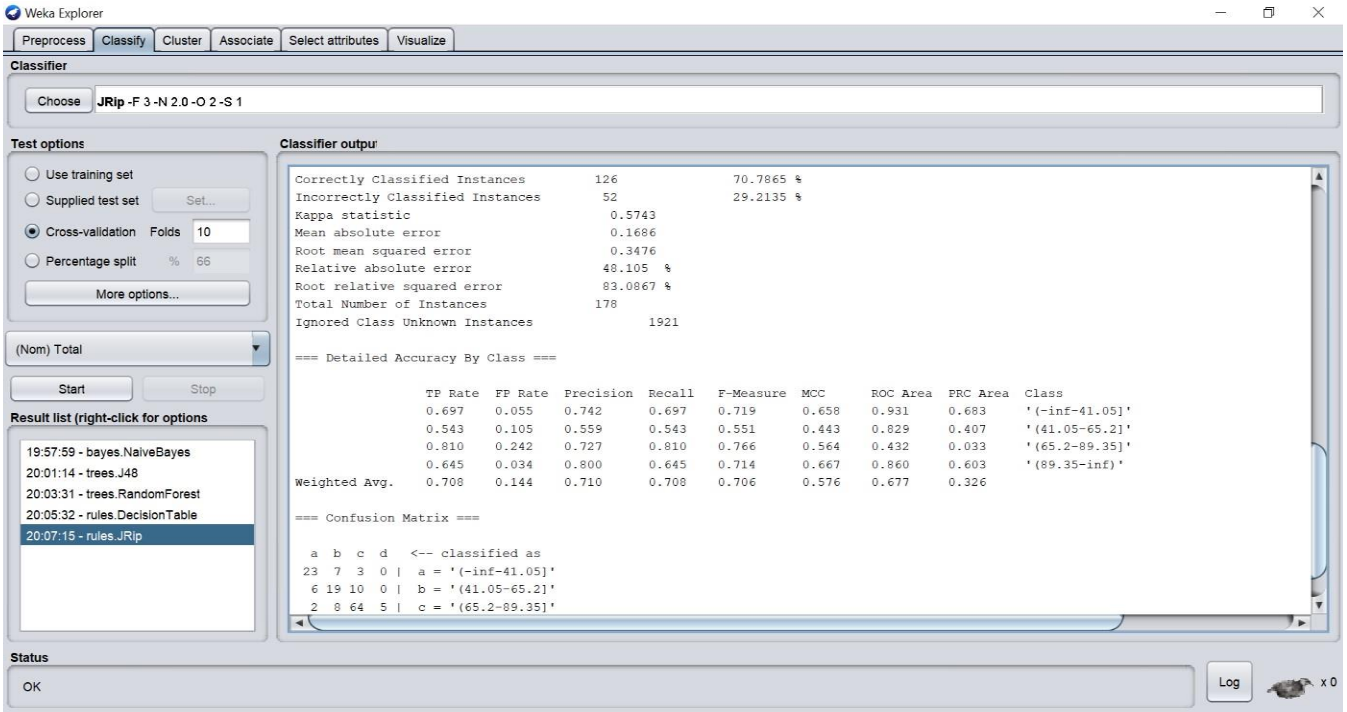


With adding new features-

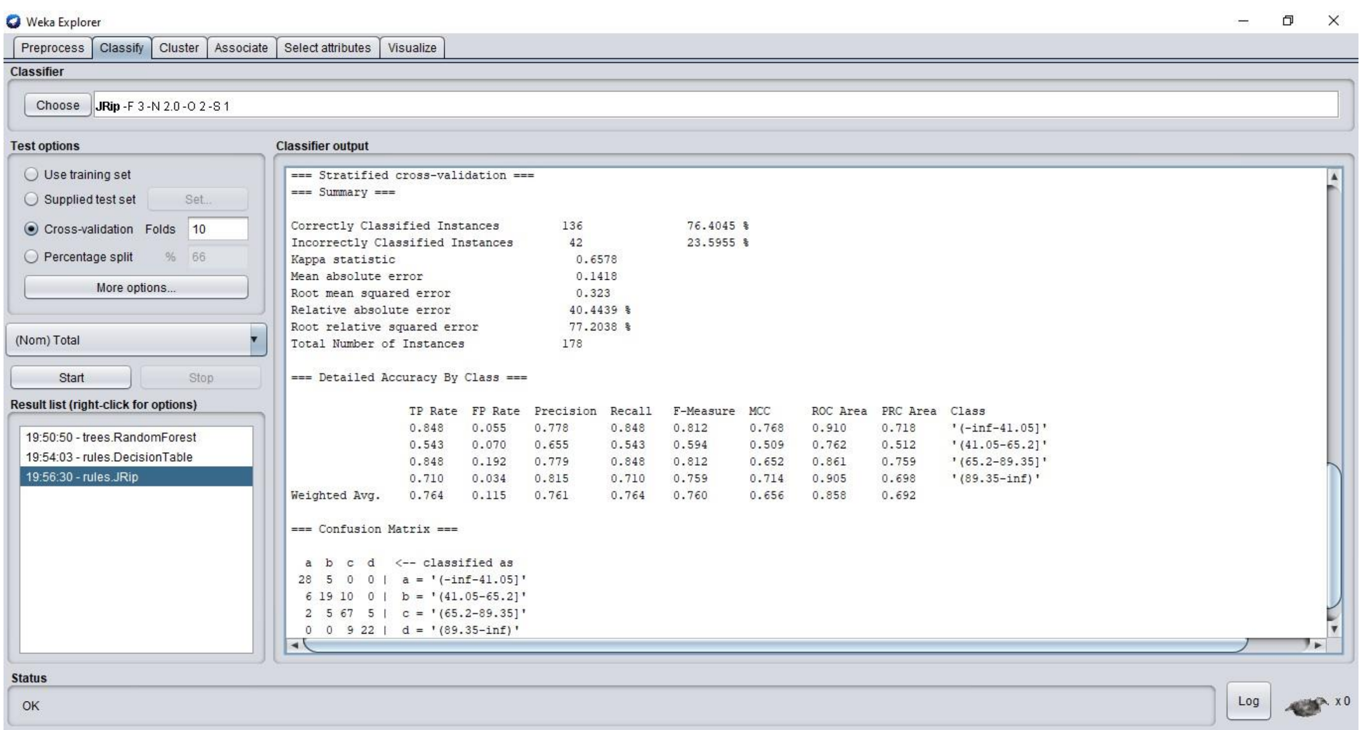


4)JRip:

Without adding new features:

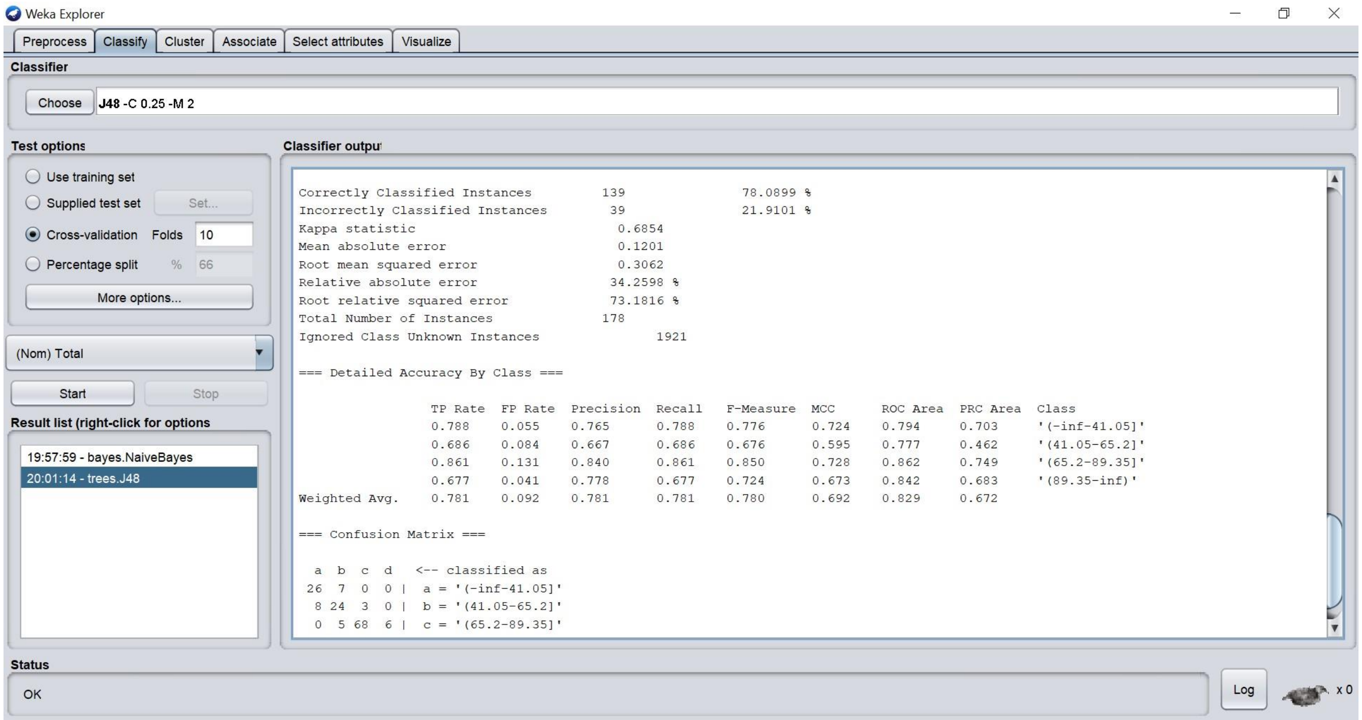


With adding new features:

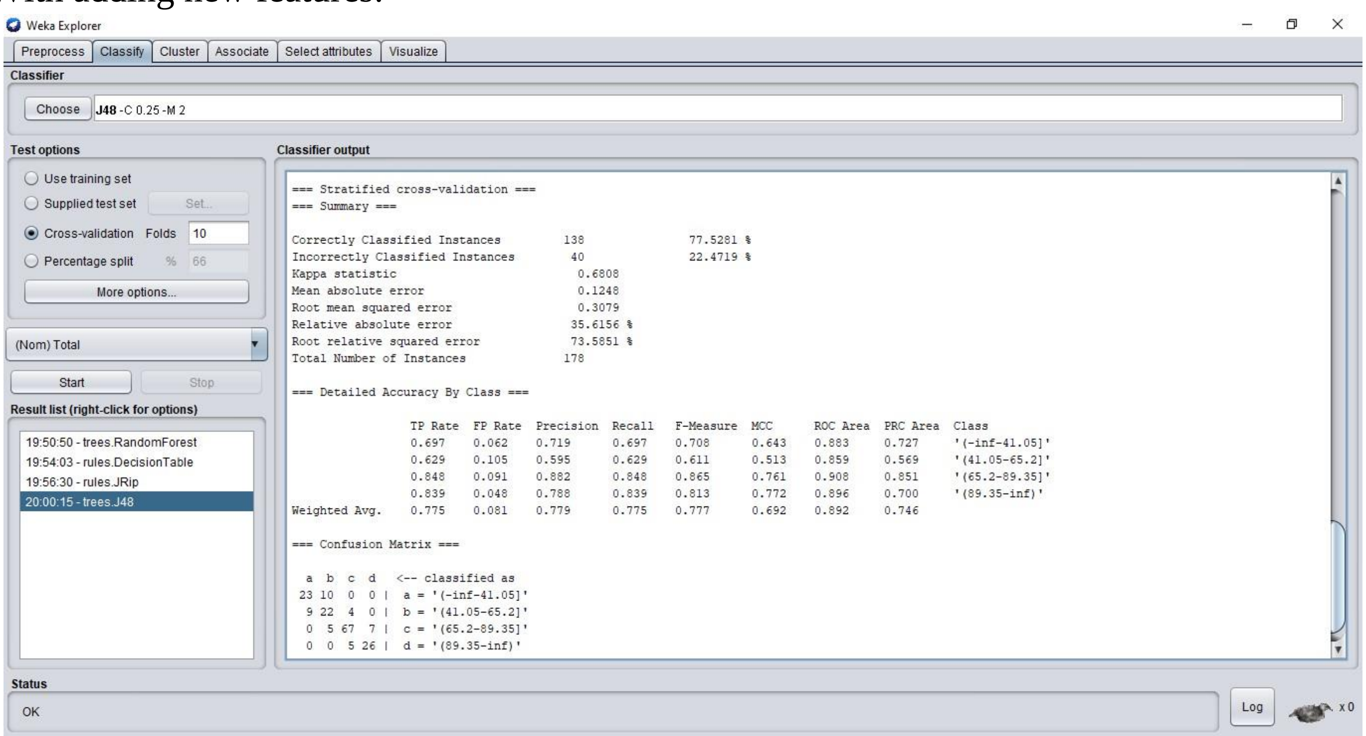


5)J48:

Without adding new features:



With adding new features:



RESULTS

Classifier Used on the data	Before Adding Attributes		After Adding Attributes	
	Correctly Classified	Incorrectly Classified	Correctly Classified	Incorrectly Classified
Bayes Net	92%	8%	91%	9%
Decision Table	75%	25%	67.5%	32.5%
J48	78%	22%	77.5%	22.5%
JRip	71%	29%	76.4%	23.6%
Naïve Bayes	93%	7%	89.9%	10.1%
Random Forest	72%	28%	78%	22%

When these classifiers are applied on the data (without new features), the accuracy which is represented by the correctly classified and incorrectly classified classes was around 85% and when the new features were added in the data the accuracy got decrease by 4-5% in 4 classifiers and in 2 classifiers it got improved.

EXTRACTING ACTION RULES USING LISP MINER

Introduction

The LISP-Miner system is an academic data mining software tool developed at the University of Economics, Prague1, which is focused on mining various types of association rules from categorical data. LISP-Miner offers a greater variety of different types of relations between the left-hand and right-hand sides of a rule using various data mining procedures implemented by LISP-miner. For the purpose of this project, we are using Ac4ft-Miner data mining procedure for extracting action rules. Ac4ft-Miner finds rules that express which actions should be performed to improve the defined state. It achieves it by examining the dependencies among the data given as an input.

The process of action rule discovery used by Lisp Miner:

LISP-Miner implements various GUHA procedures that mine for different types of knowledge patterns. This system consists of 10 different data mining procedures where 4ft-Miner procedure is based on the original GUHA procedure ASSOC while other procedures have been designed during the development of the system.

The 4ft-Miner procedure mines for knowledge patterns, that can be understood as 4ft association rules of the form

$$\varphi \approx \psi/\gamma$$

where φ (antecedent), ψ (succedent) and γ (condition) are cedents and $\approx$ is a quantifier which is evaluated on the subset of examples, that satisfy the condition γ.

The work with the LISP-Miner is not so straightforward as with other data mining systems because LISPMiner is distributed as a number of executables that must be invoked by the user. Each mining procedure in LISP-Miner tool uses different processing modules as:

1. Module 1 - LMAdmin: The first module to be used is the LMAdmin. The primary aim of this module is to attach the meta-data base to the analyzed data. The concept of meta-data allows to store and reuse inputs for the tasks as well the results obtained during analysis. Both data and meta-data are stored using a database. This is a necessary step that must precede any analysis.

2. Module 2 – LMDataSource: This module implements a variety of data transformation and data preprocessing methods that can be used to select attributes for the specific data mining task, create derived attributes, or discretize numeric attributes.

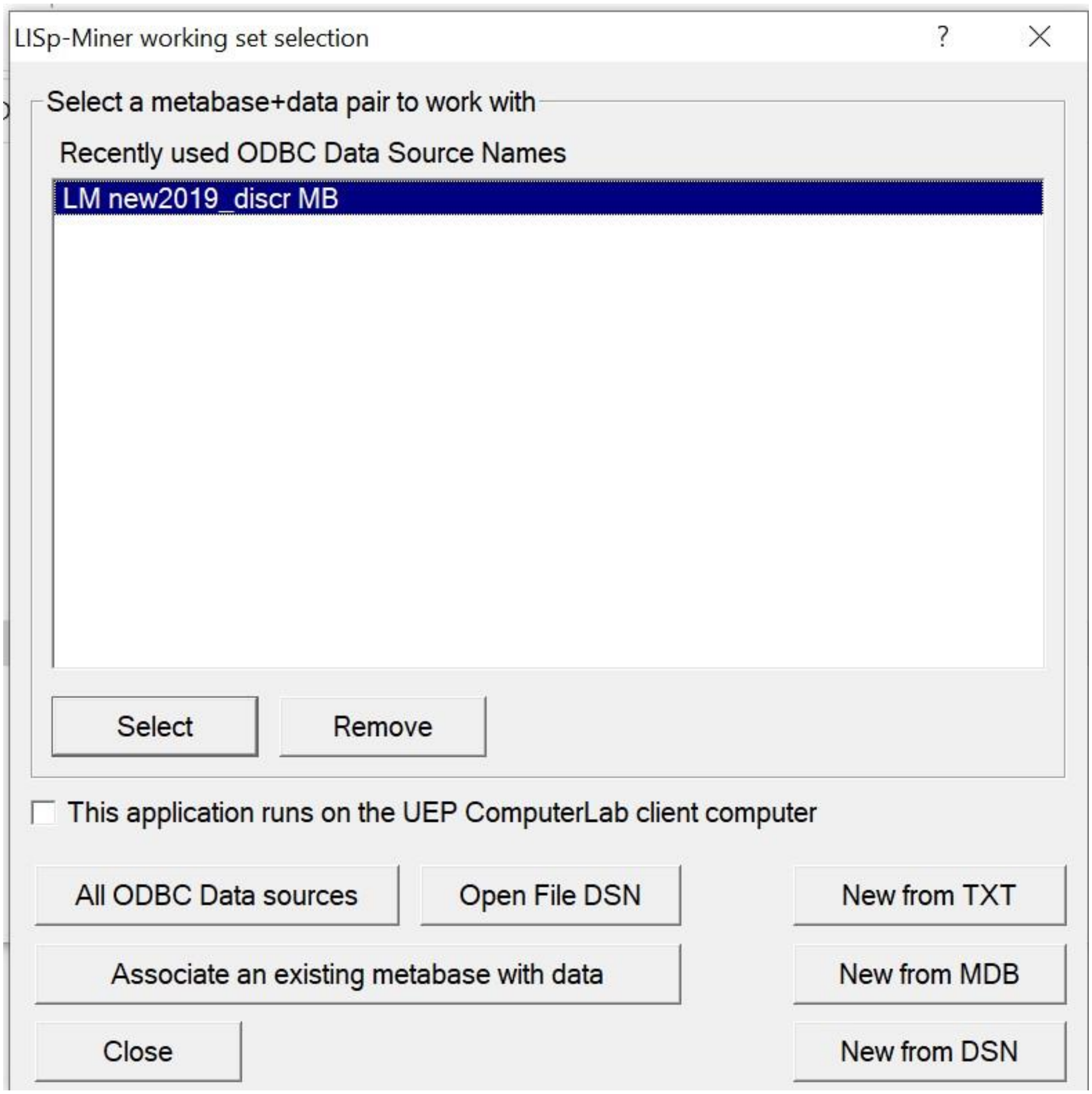
3. Module 3 - Data processing module (Task): This is task module which is used to analyze the data, and creates the task using respective xxxTask module.

4. Module 4 - Data interpretation module (Result): This is Result module which is used to display and evaluate the results. This is used to display, sort or select the resulting rules generated by running the corresponding xxxResult module.

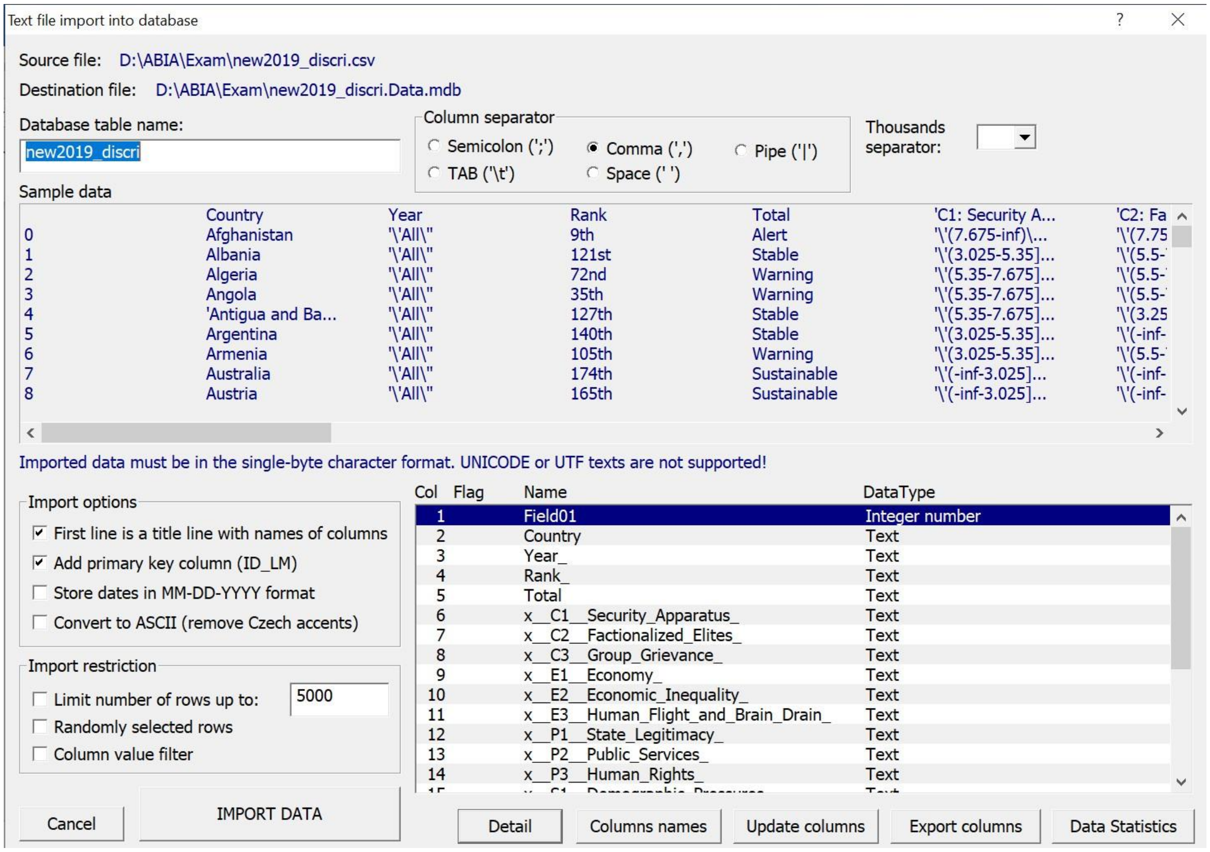
Steps to extract action rules using LISP miner :

Step 1 : First step is to acquire dataset and manipulate the data by adding 6 new features.

Step 2 : Import dataset into LISP miner tool using ‘New from TXT’ tab. For importing data through text file, file should be of type text or csv.

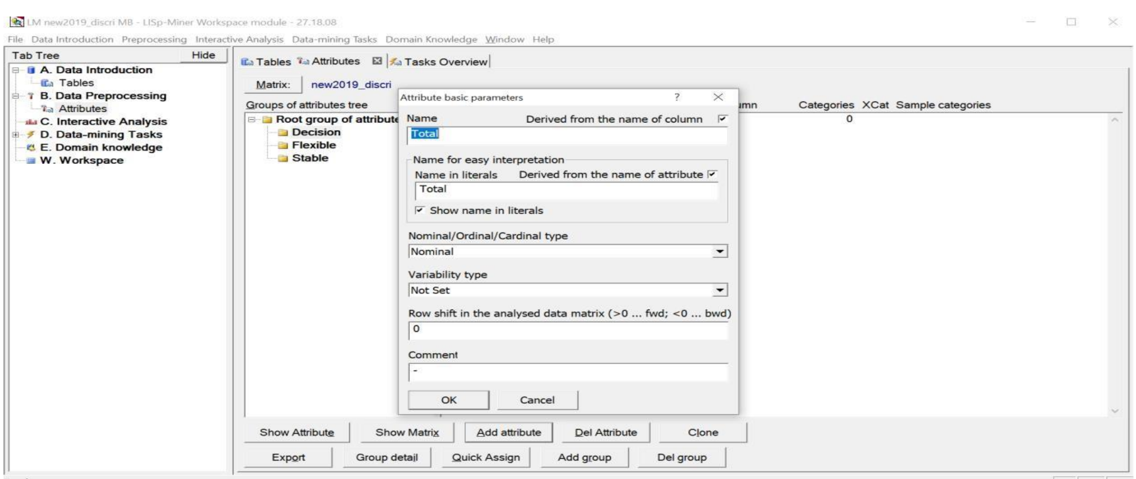


Step 3 : After uploading dataset to miner tool, import data from CSV file into database. At this stage, LISP miner displays all the attribute names as mentioned in dataset file.

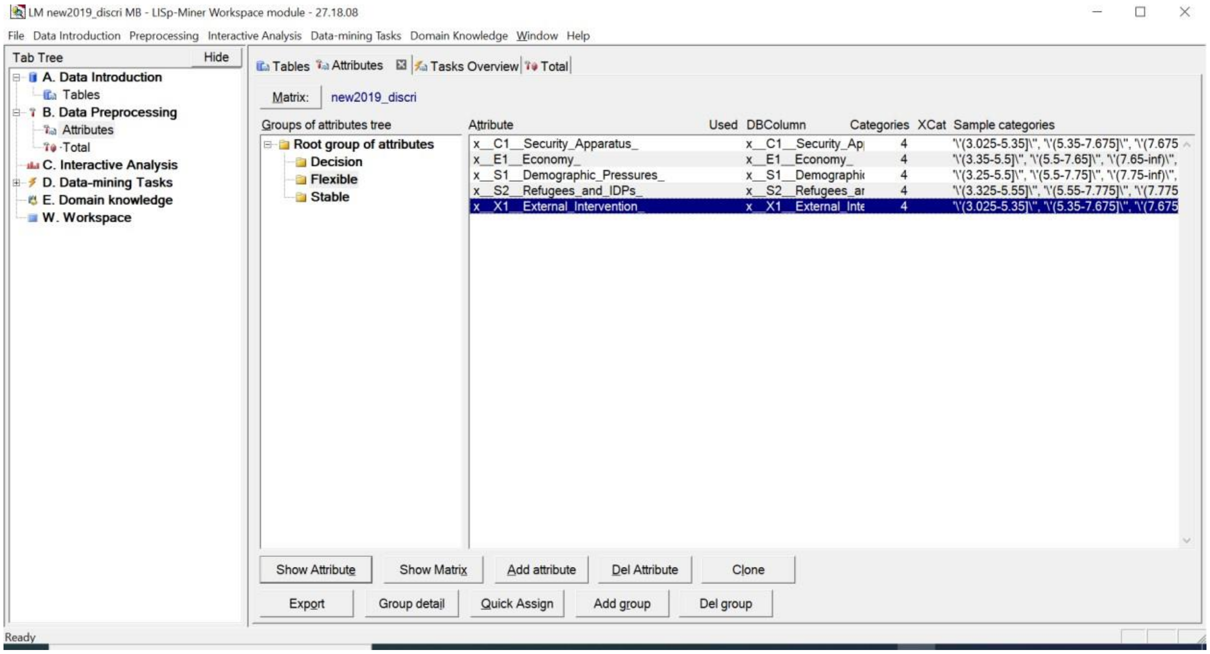


Step 4 : Select attributes from given list to extract rules from uploaded database. These attributes can be used to decide stable attributes, variable attributes and decision attributes.

Select Decision attribute: Selecting Total attribute in the Decision Group

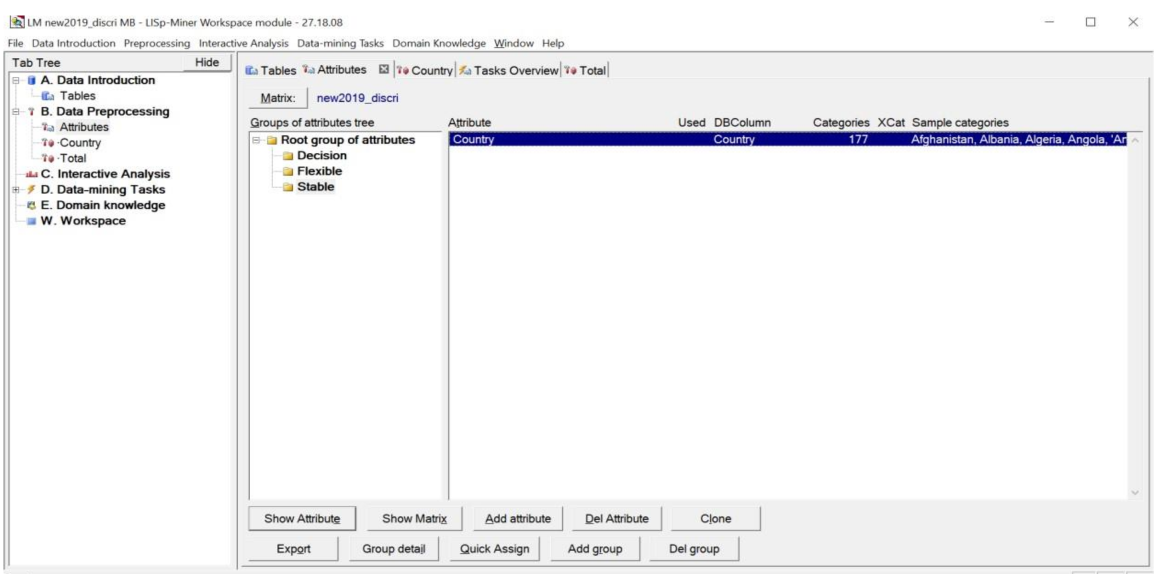


Step 5: Select Flexible attribute: Selecting Security_Apparatus, Economy, Demographic_Pressure, Refugees_and_IDPs, External_Intervention attribute in the Flexible Group

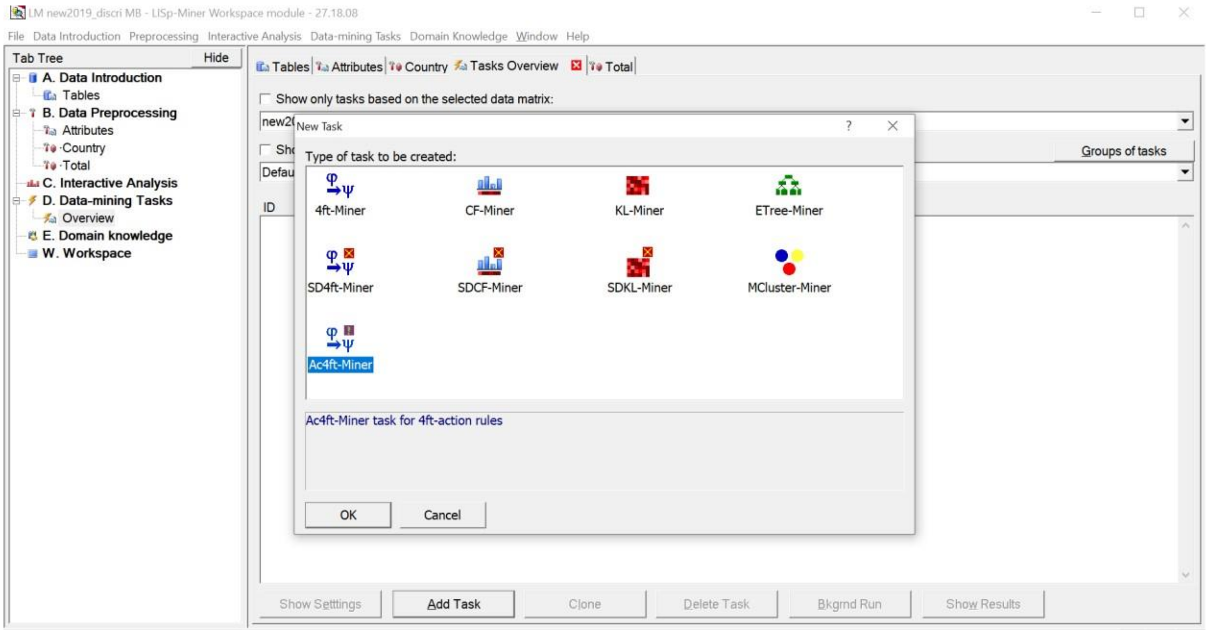


Step 6:

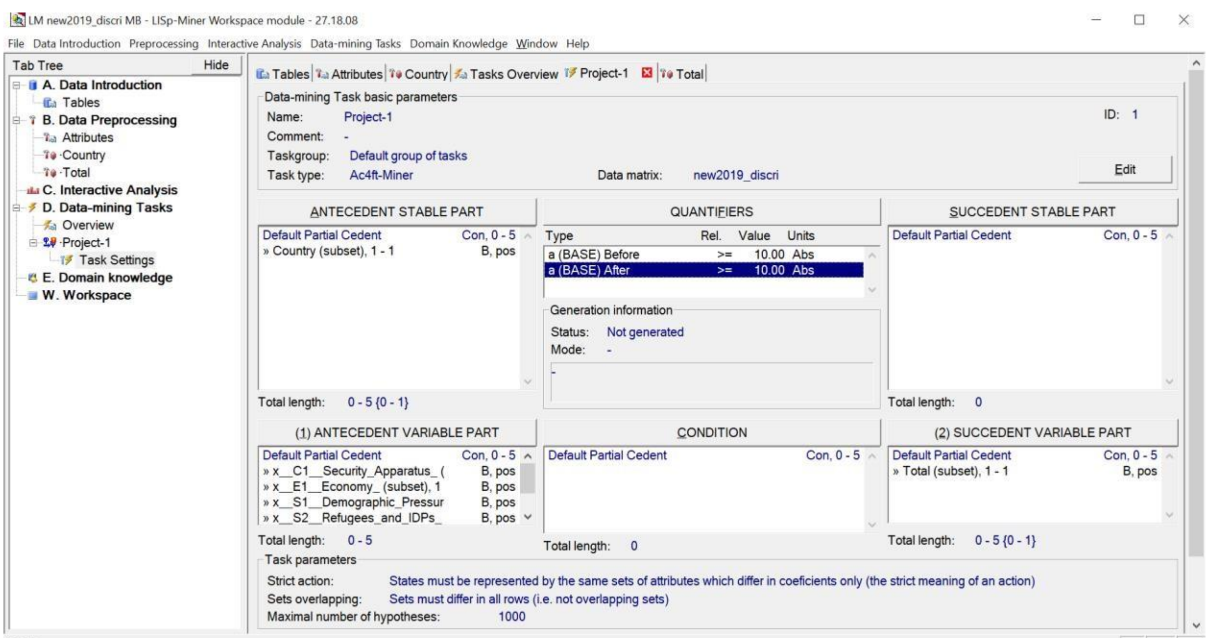
Selecting Stable attribute: Selecting Country attribute in Stable group



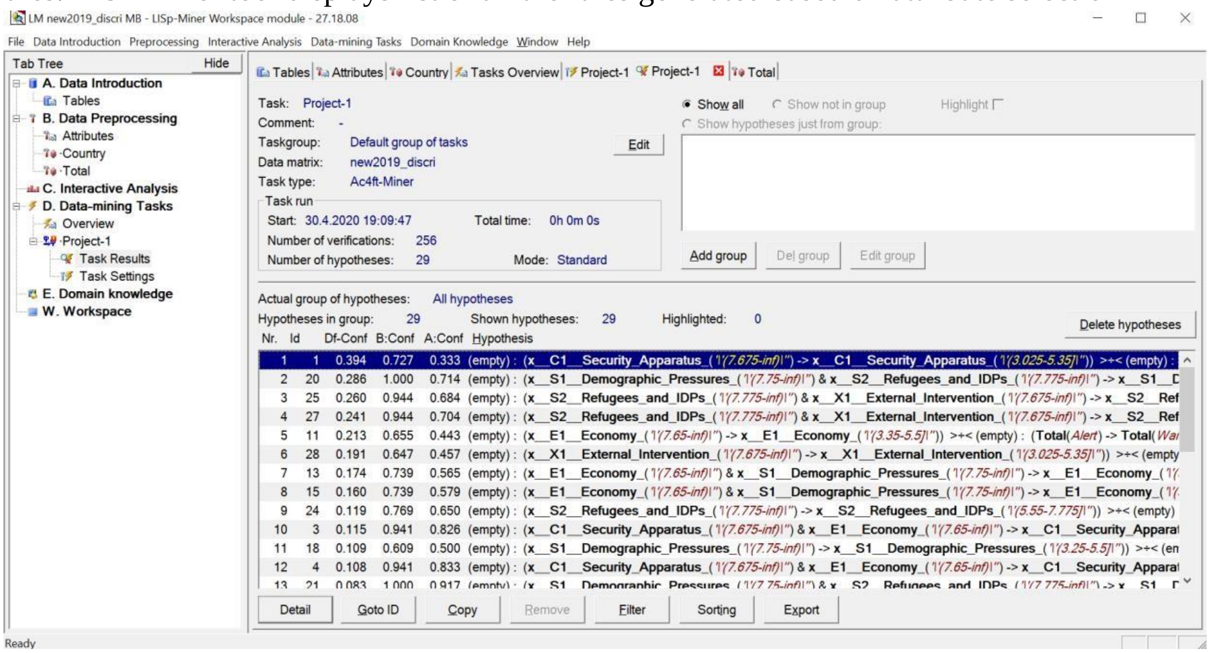
Step 7 : Create new task using Ac4ft-Miner data mining procedure for extracting rules.



Step 8: Generate tasks: Selected Country attribute in the Antecedent stable part, Security_Apparatus, Economy, Demographic_Pressure, Refugees_and_IDPs, External_Intervention in the Antecedent variable part. Selected Total in the Succedent Variable part with the one category constraint having min value 1 and max value 2. A threshold of 10 in before and after value in the Quantifiers.

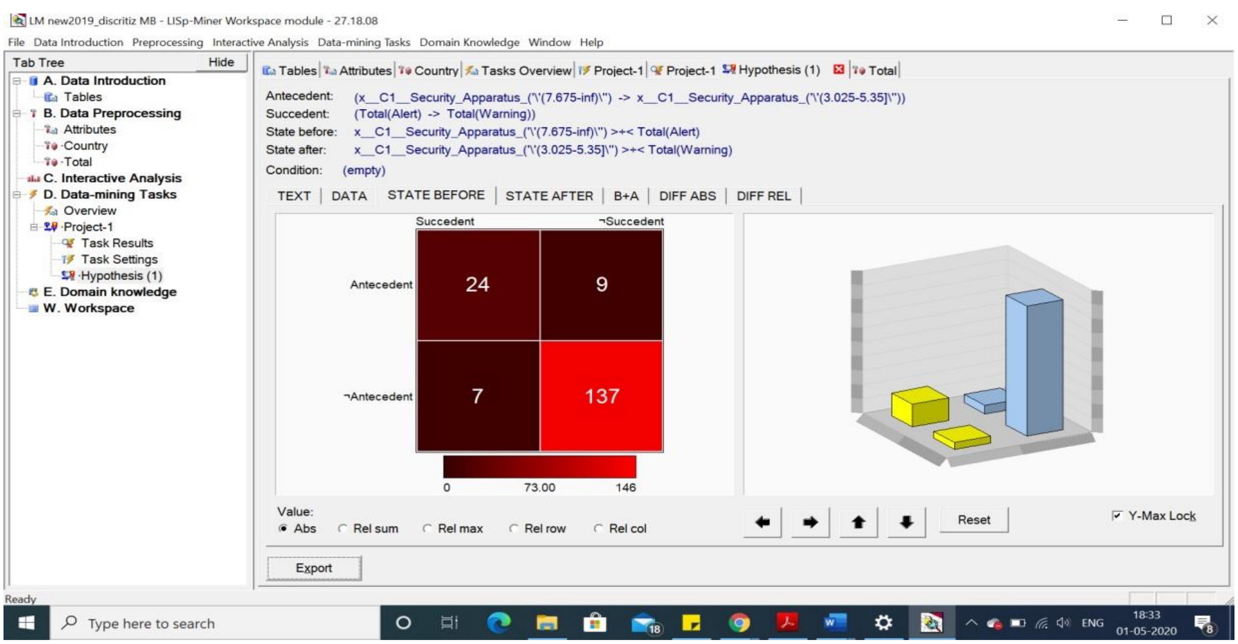


Step 9 : Once stable, variable & decision attributes, conditions, etc. are selected , run the query to generate rules. LISP miner tool displays list of all the rules generated based on attribute selection

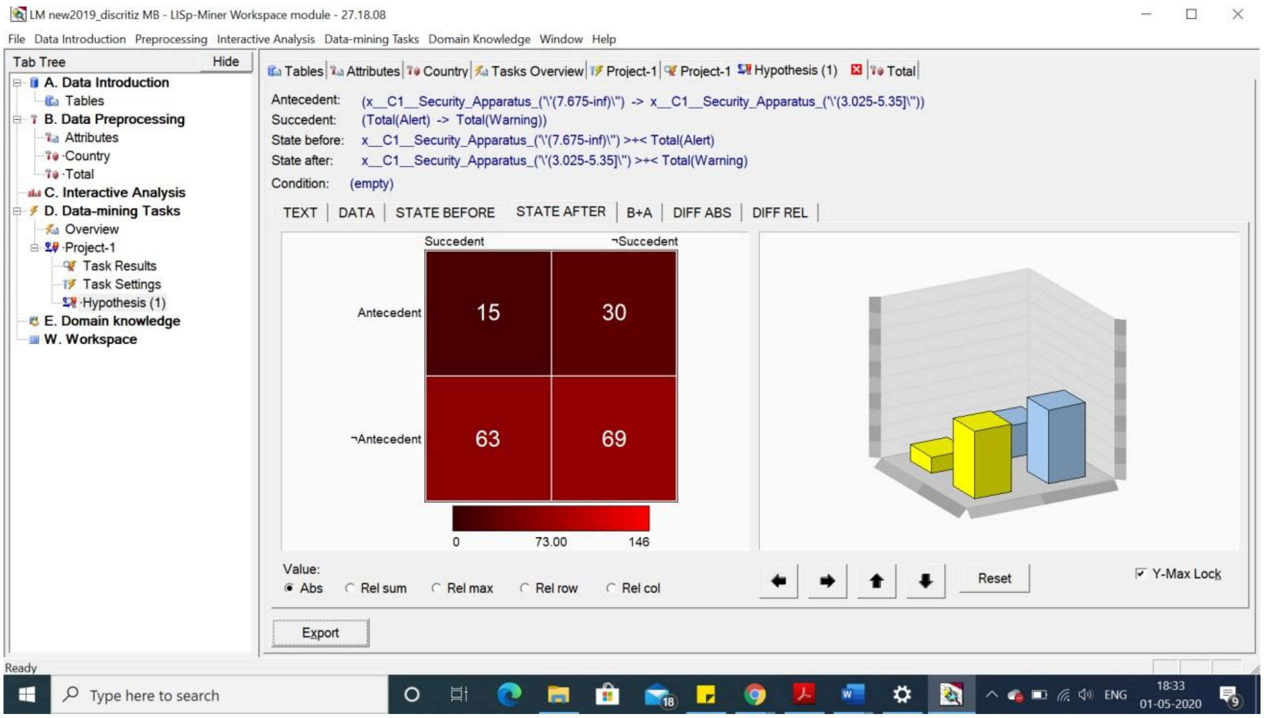


Visualization:

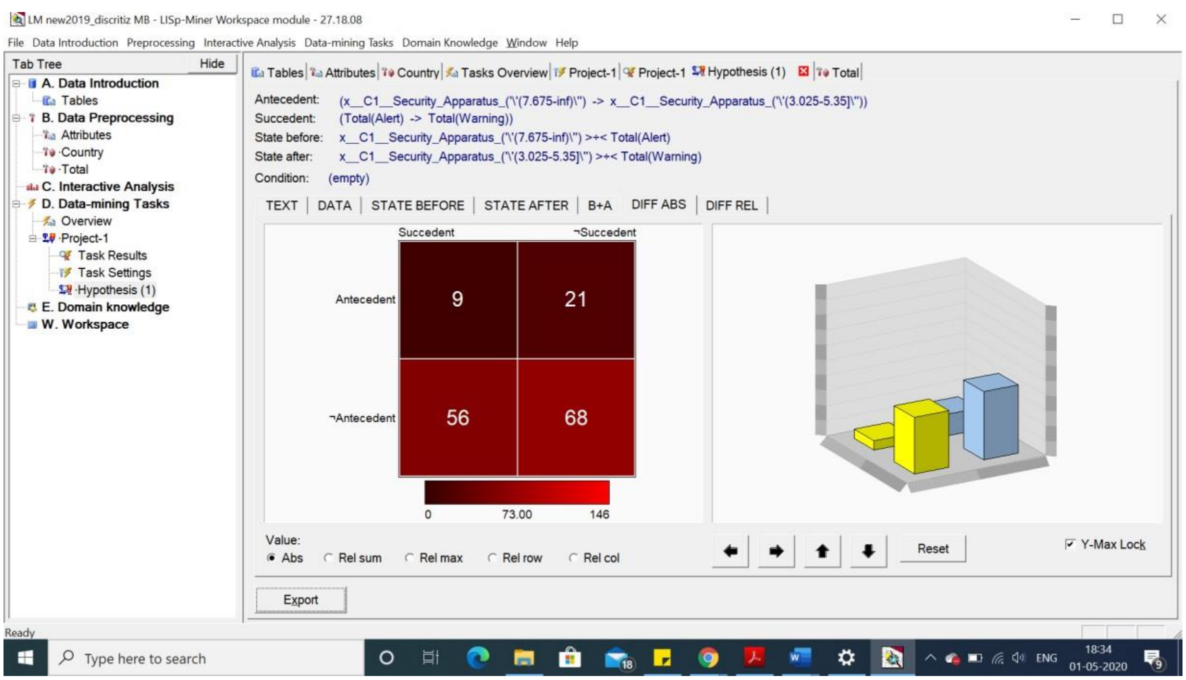
State Before:



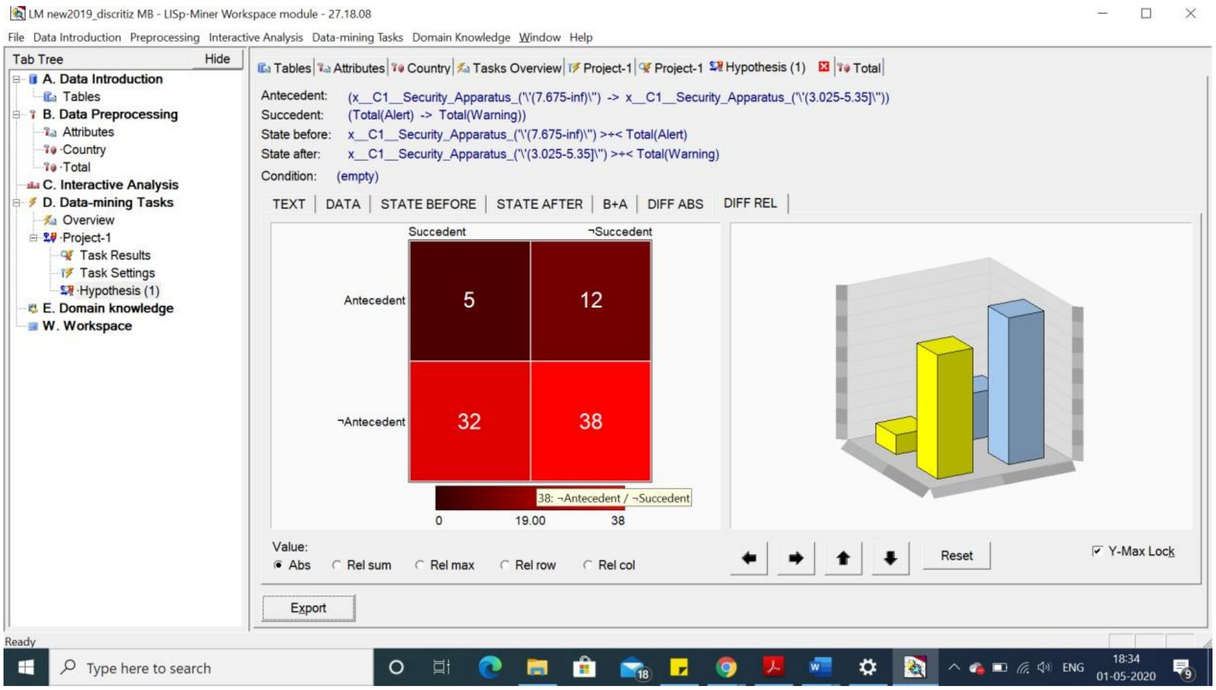
State After:



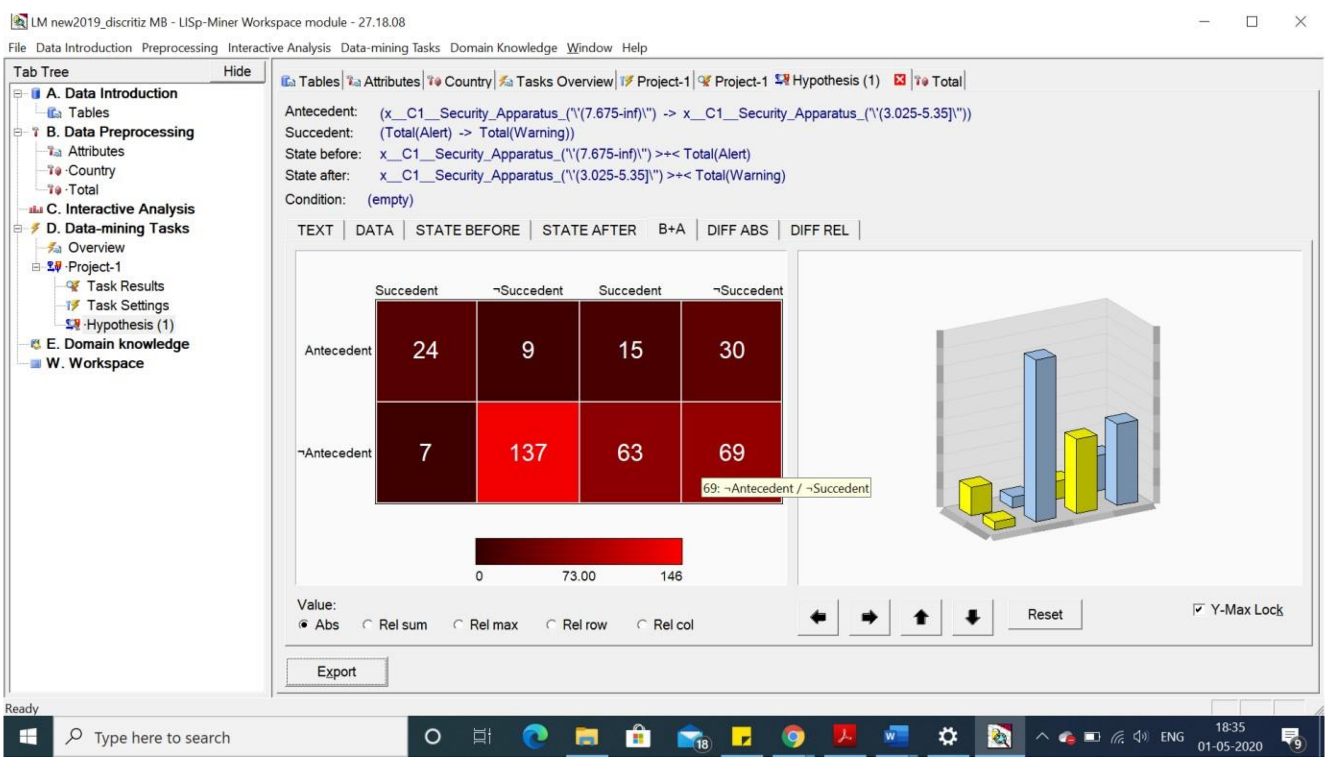
Diff-ABS:



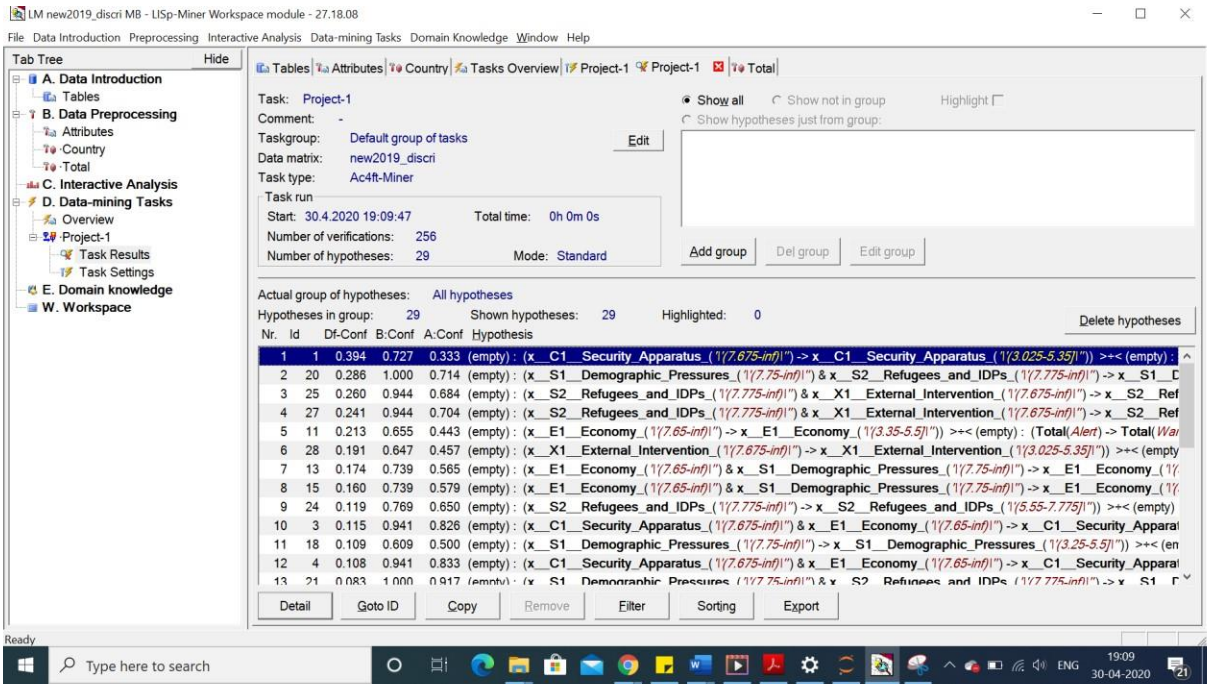
Diff-REL:


LM new2019_discritiz MB - LISp-Miner Workspace module - 27.18.08
Antecedent: (x__C1__Security_Apparatus_('\'(7.675-inf)\'') -> x__C1__Security_Apparatus_('\'(3.025-5.35]\''))
Succedent: (Total(Alert) -> Total(Warning))
State before: x__C1__Security_Apparatus_('\'(7.675-inf)\'') >+< Total(Alert)
State after: x__C1__Security_Apparatus_('\'(3.025-5.35]\'') >+< Total(Warning)
Condition: (empty)
TEXT
DATA
STATE BEFORE
STATE AFTER
B+A
DIFF ABS
DIFF REL
Succedent
¬Succedent
Antecedent
¬Antecedent
5
12
32
38
38: ¬Antecedent / ¬Succedent
0
19.00
38
Value:
Abs
Rel sum
Rel max
Rel row
Rel col
Reset
Y-Max Lock
Export

B+A:


LM new2019_discritiz MB - LISp-Miner Workspace module - 27.18.08
Antecedent: (x__C1__Security_Apparatus_('\'(7.675-inf)\'') -> x__C1__Security_Apparatus_('\'(3.025-5.35]\''))
Succedent: (Total(Alert) -> Total(Warning))
State before: x__C1__Security_Apparatus_('\'(7.675-inf)\'') >+< Total(Alert)
State after: x__C1__Security_Apparatus_('\'(3.025-5.35]\'') >+< Total(Warning)
Condition: (empty)
TEXT
DATA
STATE BEFORE
STATE AFTER
B+A
DIFF ABS
DIFF REL
Succedent
¬Succedent
Succedent
¬Succedent
Antecedent
¬Antecedent
24
9
15
30
7
137
63
69
69: ¬Antecedent / ¬Succedent
0
73.00
146
Value:
Abs
Rel sum
Rel max
Rel row
Rel col
Reset
Y-Max Lock
Export

All Hypothesis:



INFERENCES FROM RULES

(x__C1__Security_Apparatus_('\'(7.675-inf)\'') -> x__C1__Security_Apparatus_('\'(3.025-5.35]\'')) >÷< (Total(Alert) -> Total(Warning))

(x__C1__Security_Apparatus_('\'(7.675-inf)\'') -> x__C1__Security_Apparatus_('\'(5.35-7.675]\'')) >÷< (Total(Alert) -> Total(Warning))

(x__E1__Economy_('\'(7.65-inf)\'') -> x__E1__Economy_('\'(3.35-5.5]\'')) >÷< (Total(Alert) -> Total(Warning))

- A shift from a factor of 8 plus to 3 or 8 plus to 5 from security Apparatus would transform a country from an alert to a Warning state. It means higher security is slightly increased, resulting in a less fragile state.
- As the economy decreases from 8 plus to 3 the fragility result of country also decreases and hence transforming the country from an alert to a warning state

(x__S2__Refugees_and_IDPs_('\'(7.775-inf)\'') & x__X1__External_Intervention_('\'(7.675-inf)\'') -> x__S2__Refugees_and_IDPs_('\'(3.325-5.55]\'') & x__X1__External_Intervention_('\'(3.025-5.35]\'')) >÷< (Total(Alert) -> Total(Warning)) (x__S2__Refugees_and_IDPs_('\'(7.775-inf)\'') & x__X1__External_Intervention_('\'(7.675-inf)\'') -> x__S2__Refugees_and_IDPs_('\'(5.55-7.775]\'') & x__X1__External_Intervention_('\'(5.35-7.675]\'')) >÷< (Total(Alert) -> Total(Warning))

(x__X1__External_Intervention_('\'(7.675-inf)\'') -> x__X1__External_Intervention_('\'(3.025-5.35]\'')) >÷< (Total(Alert) -> Total(Warning))

(x__X1__External_Intervention_('\'(7.675-inf)\'') -> x__X1__External_Intervention_('\'(5.35-7.675]\'')) >÷< (Total(Alert) -> Total(Warning)) 0.63

- Refugees and internally displaced persons, a higher influx of refugees and IDPs will result in fragility and a decrease in this rating will result in less fragility.
- A significant shift from a factor of 8 plus to 3 from external interventions would transform a country from an alert to a Warning state.

All the other rules are included in other document named – Rules Extracted

CONCLUSION

The data with additional features was collected through the web from different sources and information was pre-processed, discrete and grouped using WEKA for classification.

Activity rules using Lisp Miner were produced and investigated. In this way, recommendations for intervention can be used as a tool to assess a nation's condition and to take critical corrective steps to improve a nation's state. We have discovered and studied various action rules from the dataset out and have action rules explained in this report.

REFERENCES:

[1] https://tradingeconomics.com/country-list/inflation-rate
[2] https://en.wikipedia.org/wiki/List_of_countries_by_GDP_(nominal)
[3] https://www.kaggle.com/kerneler/starter-countries-database-sqlite-bdd8e6f2-0/data
[4] https://ourworldindata.org/
[5] https://data.worldbank.org
[6] http://hdr.undp.org/en/data
[7] http://weka.sourceforge.net/doc.dev/weka/classifiers/bayes/BayesNet.html
[8] http://weka.sourceforge.net/doc.dev/weka/classifiers/rules/JRip.html [9] http://fsi.fundforpeace.org/