Analysis Report on Fragile States Index

ITCS/DSBA – 6162: Knowledge Discovery in Databases
Instructor – Dr. Zbigniew W Ras

University of North Carolina at Charlotte

By – Group 10
Nagarjuna Challa (801318025), Praneeth Inturi (801257186), Mithra Buggaveeti (801310349) & Deeksha Reddy Ganta (801311836)

Contents

1. Project Description & Requirements

The main goal of the project is to extract categorization and action rules from the Financial Stability Index (FSI) 2010 dataset through pre-processing. The United States offers a yearly report detailing the collapse under the Financial Stability Index (FSI) concept. This list includes all sovereign governments that are UN members and have enough information to estimate their susceptibility to conflict or collapse.
Using Action-Rules, we determine when a nation should go from an alert to a warning state. This approach also highlights how crucial each dataset attribute is in producing the fragile state index for the nation.

Fragile State Index

The Fragile States Index, a yearly research published by the Fund for Peace and American Foreign Policy magazine, first appeared in 2005. (FSI). All sovereign states that are UN members and have enough information to determine their susceptibility to conflict or dissolution are ranked on this list. Taiwan, the Palestinian Territories, Northern Cyprus, Kosovo, and Western Sahara are not mentioned, despite the fact that they are recognized as sovereign by one or more other nations. The ranking is based on how many times each of the 12 criteria has been rated. Each indicator is given an overall grade from 0 to 120, with 0 being the weakest (most stable) indicator and 120 signifying the strongest (least stable).

Indicators

Conflict risk indices are used to evaluate a nation's current situation. The metrics offer a current snapshot that may be compared to other time series snapshots to determine whether or not things are becoming better or worse. The metrics for the Fragile States Index and the CAST system are listed below.

Security Apparatus (C1): It displays the scope of a nation's security risks. Examples include attacks, bombs, the mortality rate following assaults, psychological oppression, and others.

Factionalized Elites (C2): It shows how the state is divided along racial, social class, and ethnic lines.

Group Dissatisfaction (C3): It draws attention to the social and political differences present at numerous public gatherings.

Economy (E1): Indicates the devaluation of a country's currency. Unemployment rates, poverty levels, debt, business bankruptcies, and other factors can all play a role.

Economic Inequality (E2): Regardless of national representation, this pointer indicates economic imbalance.

Exodus of Humans and Brain Drain (E3): This represents the uprooting of people (the limited resettlement when they are not economically prosperous) and the state's grief over such developments.

Legitimacy of the State (P1): This shows the relationship between government and citizens and how accessible government is to citizens.

Public Services (P2): It reflects the existence of many projects that help people.

Human Rights (P3): This reflects the extent to which the country's basic freedoms and opportunities are respected and maintained.

Tensions in the demographics (S1): It focuses on issues governments have with their citizens, such as food security and access to shared assets.

Refugees and Internally Displaced Persons (S2): It reveals the stresses of nations due to their limited ability to resolve huge social and political networks.

External Intervention (X1): It shows state reactions to external events and the impact of those reactions.

2. Extended Features

This study proposes the discovery of new patterns and trends by classifying vulnerabilities within states. We extend this study by introducing seven additional factors that help in risk assessment and conflict detection. The following new features have been added:

Education (Mean Years of Schooling (Years)): Average years of education received by people aged 25 and over. Converted from education level using the official duration for each level. By 2030, substantially increase the number of young people with relevant skills, including technical and vocational skills, necessary for employment, decent work and entrepreneurship.

Unemployment: Percentage of working population aged 15 and over who are not employed or self-employed but are available for employment and are taking steps to seek dependent employment or self-employment. By 2030, achieve full and productive employment and decent work for all women and men, including for young people and persons with disabilities, and equal pay for work of equal value.

Health (Current Health Expenditure (% of GDP)): Spending on health goods and services expressed as a percentage of GDP; health care such as buildings, machinery, information technology, and stockpiling vaccines for emergencies or pandemics Investments are excluded, primarily promoting the recruitment, development, training and retention of health workers in developing countries, especially least developed countries and small island developing States.

Population Growth: The term "population growth" refers to the increase in the population of a country. This information relates to the rate of increase from the previous year to the current year. Rising population growth contributes to higher inflation, military spending, unemployment, and other factors, all of which contribute to increased vulnerability.

Corruption: The Corruption Perceptions Index (CPI) is an index that ranks countries "according to their perceived level of corruption in the public sector, as determined by expert assessments and public opinion polls." The CPI generally defines corruption as "abuse of delegated power for private gain." The index has been published annually since 1995 by the non-governmental organization Transparency International.

Human Development Index (HDI): A composite index that measures average performance in her three fundamental dimensions of human development – longevity and healthy living, knowledge and a decent standard of living.

See Technical Note 1 at the below URL: http://hdr.undp.org/sites/default/files/hdr2020_technical_notes.pdf for details on how the HDI is calculated.

3. Decision Attributes

According to the Fragile Index wiki page, the total score is calculated using 12 indicators and used to identify countries.
The ranges defined in the Wikipedia page are:
Alert (90.0 – 120.0)
Warning (60.0 – 89.9)
Stable (30.0 – 59.9)
Sustainable (0 – 29.9)
After adding six additional qualities, the scores were normalized as follows:
Total reduced from 180 to 120 (18 indications, each with a score of 1 to 10).
For example, a score of 60 out of 180 means 60*120/180 = 40 out of 120.

However, the 6 newly introduced indications did not match the 12 existing indicators, resulting in significantly lower numbers than predicted by the addition of the 6 additional indicators. Regions have been redefined to explain the differences. This range was derived by analyzing the original dataset and grouping countries in precarious conditions with the same number of countries.

The new ranges are:
Alert (>70.0)
Warning (45.0 – 69.9)
Stable (25.0 – 44.9)
Sustainable (0 – 25)
This classification was applied to both the original and expanded datasets.

4. Data Extraction

Values for these advanced features were extracted from various websites and added to the original FSI dataset. Data for 2010 are compiled and saved as an Excel spreadsheet for further processing and analysis.
Values for advanced features were collected from various websites listed below:
Education: https://hdr.undp.org/en/indicators/103006#
Unemployment: https://hdr.undp.org/en/indicators/140606#
Health: https://hdr.undp.org/en/indicators/137506
Population: https://databank.worldbank.org/
Corruption: https://www.transparency.org/en/cpi/2010
Human Development Index: https://hdr.undp.org/en/indicators/137506

5. Data Preprocessing

When retrieving data from multiple resources, there is inevitably also a large amount of unnecessary data, which can cause problems in performing the intended task. For this reason, preprocessing the data is very important. Preprocessing is a method of removing null values and some characters from a huge amount of data to solve pending problems. When the data are preprocessed, all noise data and discrepancies are removed before execution. You can also use data preprocessing to search for missing values and data. Therefore, it is important that your data is as clean and organized as possible before using it.

The characteristics required to maintain high quality data are:

- ➢ Accuracy
- ➢ Completeness
- ➢ Timeliness
- ➢ Consistency
- ➢ Validated
- ➢ Organized

All the above features can be gained by doing data preprocessing.

The data obtained from various websites was not clean, so some preprocessing was done before using it for classification. Below are the data cleansing steps.

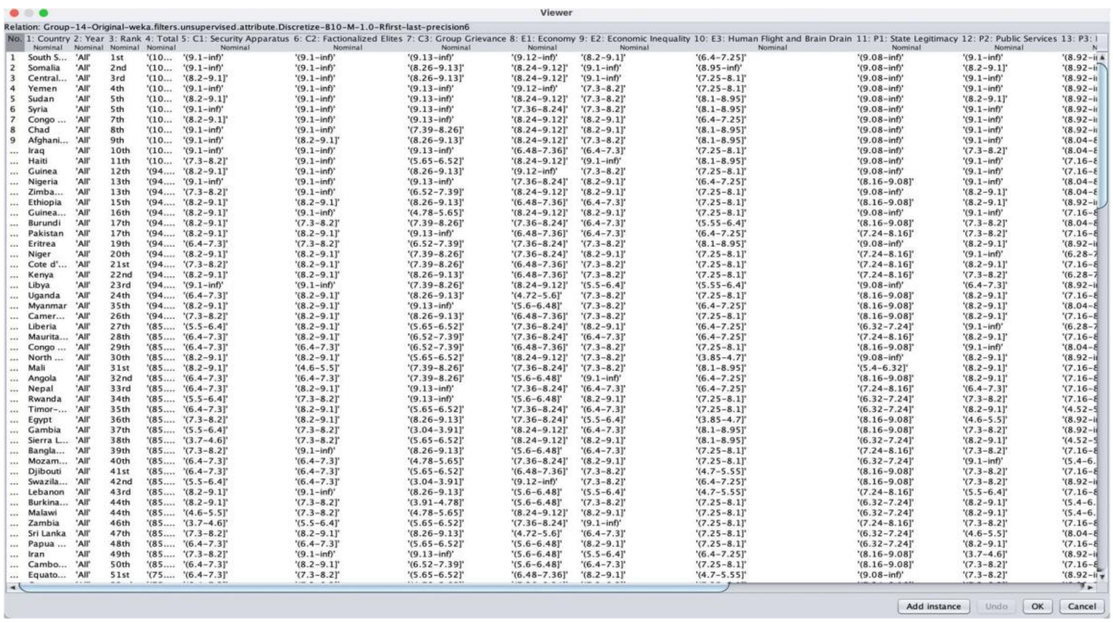


Figure 5.1: Representation of Preprocessed Data.

- ➢ Special characters such as !, % have been removed so that the WEKA tool can parse the sheet.
- ➢ Missing figures were filled in with average values.
- ➢ Rows with many empty values were excluded from the analysis.
- ➢ Numerical values for total decision variables have been replaced with nominal values based on the values in the following table.
- ➢ Round decimal values to 2 for better processing.
- ➢ Both the original Excel file and the expanded Excel file were converted to CSV and used as input to the WEKA software to filter the data.

6. Data Discretization using WEKA

Discretization is a method that attempts to reduce large amounts of data into smaller values so that the data can be easily managed.
Here we are trying to transform the real values into ordinal values or bins and the process is discretization of the data. I do this when using trees.

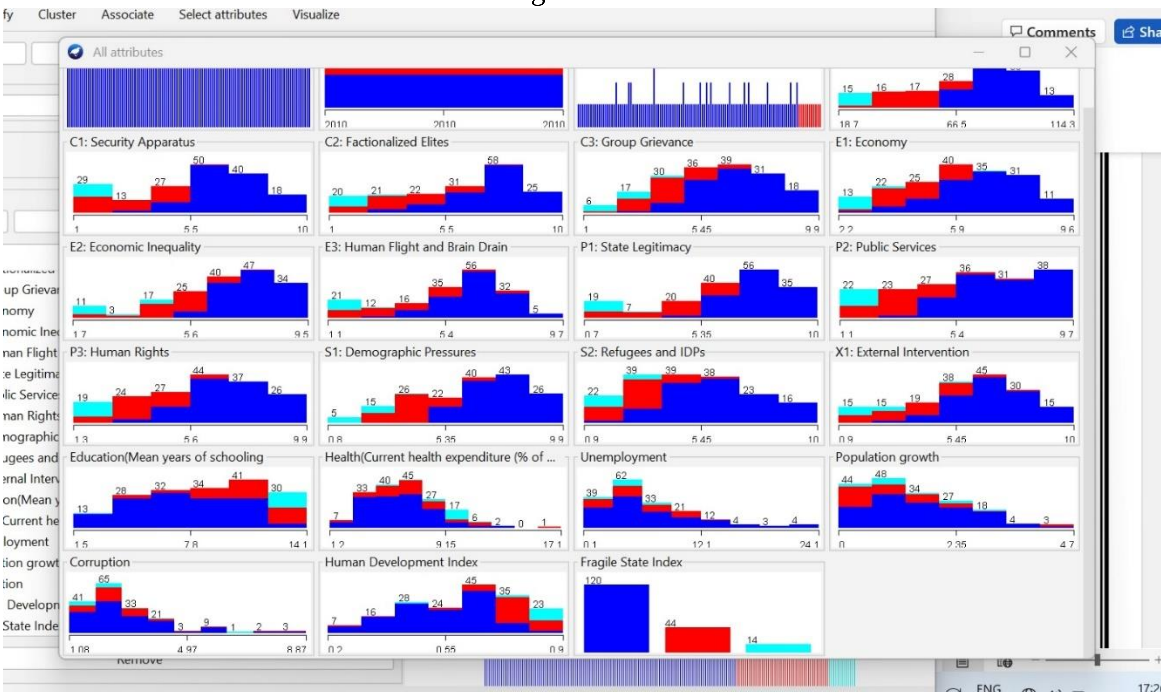


Figure 6.1: Visual Representation of Discretization.

7. Data Classification using WEKA

Data classification is a method of classifying data into discrete groups. Classification examines, interprets, and organizes material from vast databases, whether structured or not. Classification analyzes, understands, and organizes information into a number of categories. It's easy to determine which category a new data item falls into and add it to that category.

The following are the three classification algorithms that were used in this project:

Bayes Net Classifier:

A classifier that uses Bayes' theorem to make strong and independent assumptions about the data is a Bayesian network classifier. The Bayes Net Classifier is an independent feature model because it leads to the assumption that the presence or absence of one class feature is independent of the presence or absence of another feature. Bayesian networks are best suited to the assumption that when an event occurs, one of several possible causes is likely to be one of the contributing factors.

Random Forest Classifier:

The Random Forest classifier is one of the simplest and easiest to use classifiers. Being a supervised learning algorithm, it builds and merges a series of decision trees to get an accurate and stable predictive model.
Using this there were two tasks performed:

- Classification
- Regression

For classification tasks the output will be the class selected from the maximum number of trees, whereas for regression tasks the output will be the average prediction for each tree.

Randomizable Filter Classifier:

This is the class used to run the classifier on the filtered data. The classifier is based on the training data, and the test instances are processed using filters, but no structural changes are made. This is one of the simplest variants of the filtered classifier that instantiates a random projection model. I am using randomizable filters as base classifiers, but each base classifier is seeded with a very different random number. The final prediction is therefore the average of the predictions produced by the individual base classifiers.

Classification using Original Dataset

We took 2010 data to perform the following experiment:

1. **Bayes Net Classifier:**

   Input: Classify -> Choose -> BayesNet -> Apply

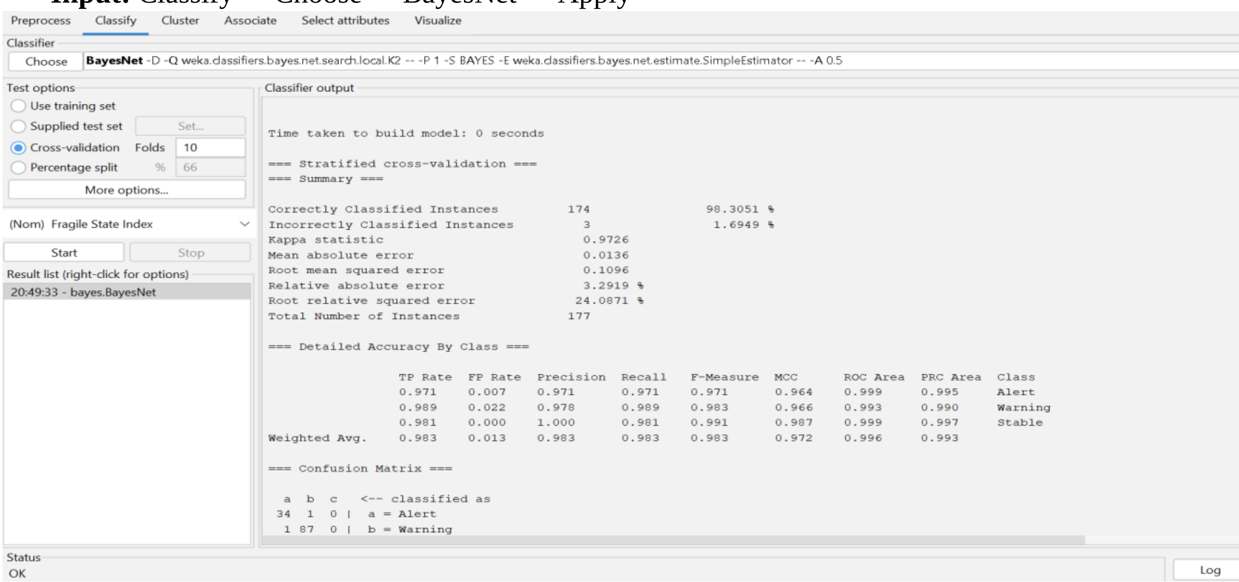


Figure 7.1: Applied Bayes Net Classifier on the Original Data.

Output:

=== Run information ===

Scheme: weka.classifiers.bayes.BayesNet -D weka.classifiers.bayes.net.search.local.K2 -- -P 1 -S BAYES -E weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5

Relation: Original

Instances: 177

Attributes: 17

Country
Year
Rank
Total
C1: Security Apparatus
C2: Factionalized Elites
C3: Group Grievance

E1: Economy
E2: Economic Inequality
E3: Human Flight and Brain Drain
P1: State Legitimacy
P2: Public Services
P3: Human Rights
S1: Demographic Pressures
S2: Refugees and IDPs
X1: External Intervention
Fragile State Index
Test mode: 10-fold cross-validation
=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=17 #classindex=16
Network structure (nodes followed by parents)
Country (177): Fragile State Index
Year (1): Fragile State Index
Rank (158): Fragile State Index
Total (3): Fragile State Index
C1: Security Apparatus (4): Fragile State Index
C2: Factionalized Elites (4): Fragile State Index
C3: Group Grievance (4): Fragile State Index
E1: Economy (3): Fragile State Index
E2: Economic Inequality (4): Fragile State Index
E3: Human Flight and Brain Drain (4): Fragile State Index
P1: State Legitimacy (4): Fragile State Index
P2: Public Services (4): Fragile State Index
P3: Human Rights (3): Fragile State Index
S1: Demographic Pressures (4): Fragile State Index
S2: Refugees and IDPs (4): Fragile State Index
X1: External Intervention (4): Fragile State Index
Fragile State Index (3):
LogScore Bayes: -3765.801905150933
LogScore BDeu: -11379.751359349042
LogScore MDL: -8967.399230854315
LogScore ENTROPY: -6097.224204142119
LogScore AIC: -7206.224204142119

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances 174 98.3051 %

Incorrectly Classified Instances 3 1.6949 %
Kappa statistic 0.9726
Mean absolute error 0.0136
Root mean squared error 01096
Relative absolute error 3.2919 %
Root relative squared error 24.0871 %
Total Number of Instances 177

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.971	0.007	0.971	0.971	0.971	0.964	0.999	0.995	Alert
	0.989	0.022	0.978	0.989	0.983	0.966	0.993	0.990	Warning
	0.981	0.000	1.000	0.981	0.991	0.987	0.999	0.997	Stable
Weighted Avg.	0.983	0.013	0.983	0.983	0.983	0.972	0.996	0.993	

=== Confusion Matrix ===

a b c <-- classified as
34 1 0 | a = Alert
1 87 0 | b = Warning
1 53 | c = Stable

2. **Random Forest Classifier:**
   Input: Classify -> Choose ->RandomForest -> Apply

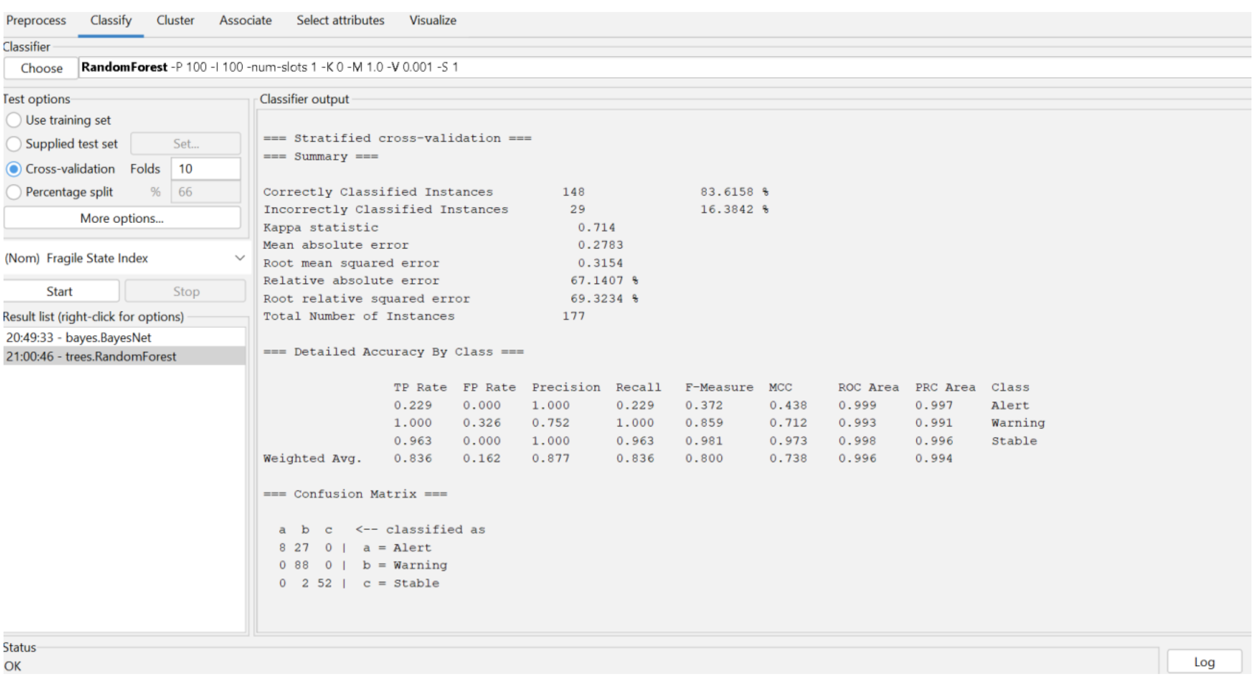


Figure 7.2: Applied Random Forest Classifier on Original Data.

```
=== Run information ===

Scheme:       weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1
Relation:     Original
Instances:    177
Attributes:   17
              Country
              Year
              Rank
              Total
              C1: Security Apparatus
              C2: Factionalized Elites
              C3: Group Grievance
              E1: Economy
              E2: Economic Inequality
              E3: Human Flight and Brain Drain
              P1: State Legitimacy
              P2: Public Services
              P3: Human Rights
              S1: Demographic Pressures
              S2: Refugees and IDPs
              X1: External Intervention
               Fragile State Index
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner

weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Time taken to build model: 0.05 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         148               83.6158 %
```

Incorrectly Classified Instances 29 16.3842 %
Kappa statistic 0.714
Mean absolute error 0.2783
Root mean squared error 0.3154
Relative absolute error 67.1407 %
Root relative squared error 69.3234 %
Total Number of Instances 177

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.229	0.000	1.000	0.229	0.372	0.438	0.999	0.997	Alert
	1.000	0.326	0.752	1.000	0.859	0.712	0.993	0.991	Warning
	0.963	0.000	1.000	0.963	0.981	0.973	0.998	0.996	Stable
Weighted Avg.	0.836	0.162	0.877	0.836	0.800	0.738	0.996	0.994	

=== Confusion Matrix ===

```
 a  b  c   <-- classified as
 8 27  0 |  a = Alert
 0 88  0 |  b = Warning
 0  2 52 |  c = Stable
```

3. Randomized Filter Classifier:

Input: Classify -> Choose ->RandomizableFilter-> Apply

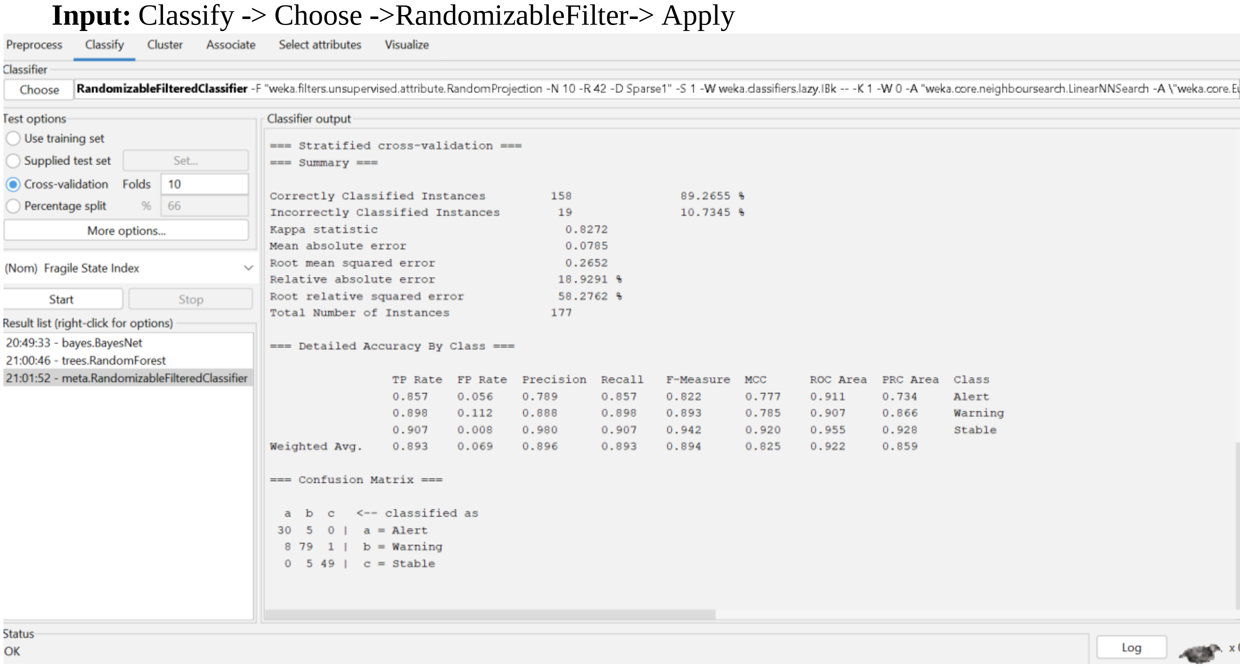


Figure 7.3: Applied Randomizable Filter Classifier on Original Data.

Output:

=== Run information ===

```
Scheme:       weka.classifiers.meta.RandomizableFilteredClassifier -F
"weka.filters.unsupervised.attribute.RandomProjection -N 10 -R 42 -D Sparse1" -S 1 -W
weka.classifiers.lazy.IBk -- -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A
\"weka.core.EuclideanDistance -R first-last\""
Relation:     Original
Instances:    177
Attributes:   17
              Country
              Year
              Rank
              Total
              C1: Security Apparatus
              C2: Factionalized Elites
              C3: Group Grievance
              E1: Economy
              E2: Economic Inequality
              E3: Human Flight and Brain Drain
              P1: State Legitimacy
              P2: Public Services
              P3: Human Rights
              S1: Demographic Pressures
              S2: Refugees and IDPs
              X1: External Intervention
               Fragile State Index
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

RandomizableFilteredClassifier using weka.classifiers.lazy.IBk -K 1 -W 0 -A
"weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\"" on
data filtered through weka.filters.unsupervised.attribute.RandomProjection -N 10 -R -1375976779 -D
Sparse1

Filtered Header
@relation Original-weka.filters.supervised.attribute.NominalToBinary-
weka.filters.unsupervised.attribute.RandomProjection-N10-R-1375976779-DSparse1

@attribute K1 numeric
@attribute K2 numeric
@attribute K3 numeric
@attribute K4 numeric
@attribute K5 numeric
@attribute K6 numeric
@attribute K7 numeric
@attribute K8 numeric
@attribute K9 numeric
@attribute K10 numeric
@attribute ' Fragile State Index' {Alert,Warning,Stable}
```

@data

Classifier Model
IB1 instance-based classifier
using 1 nearest neighbor(s) for classification
Time taken to build model: 0.02 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances	158	89.2655 %
Incorrectly Classified Instances	19	10.7345 %
Kappa statistic	0.8272	
Mean absolute error	0.0785	
Root mean squared error	0.2652	
Relative absolute error	18.9291 %	
Root relative squared error	58.2762 %	
Total Number of Instances	177	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.857	0.056	0.789	0.857	0.822	0.777	0.911	0.734	Alert
	0.898	0.112	0.888	0.898	0.893	0.785	0.907	0.866	Warning
	0.907	0.008	0.980	0.907	0.942	0.920	0.955	0.928	Stable
Weighted Avg.	0.893	0.069	0.896	0.893	0.894	0.825	0.922	0.859	

=== Confusion Matrix ===

```
  a  b  c   <-- classified as
 30  5  0 |  a = Alert
  8 79  1 |  b = Warning
  0  5 49 |  c = Stable
```

Classification using Extended Dataset

We took 2010 data and added six new features to perform the following experiment:

1. **Bayes Net Classifier:**

   Input: Classify -> Choose -> BayesNet -> Apply

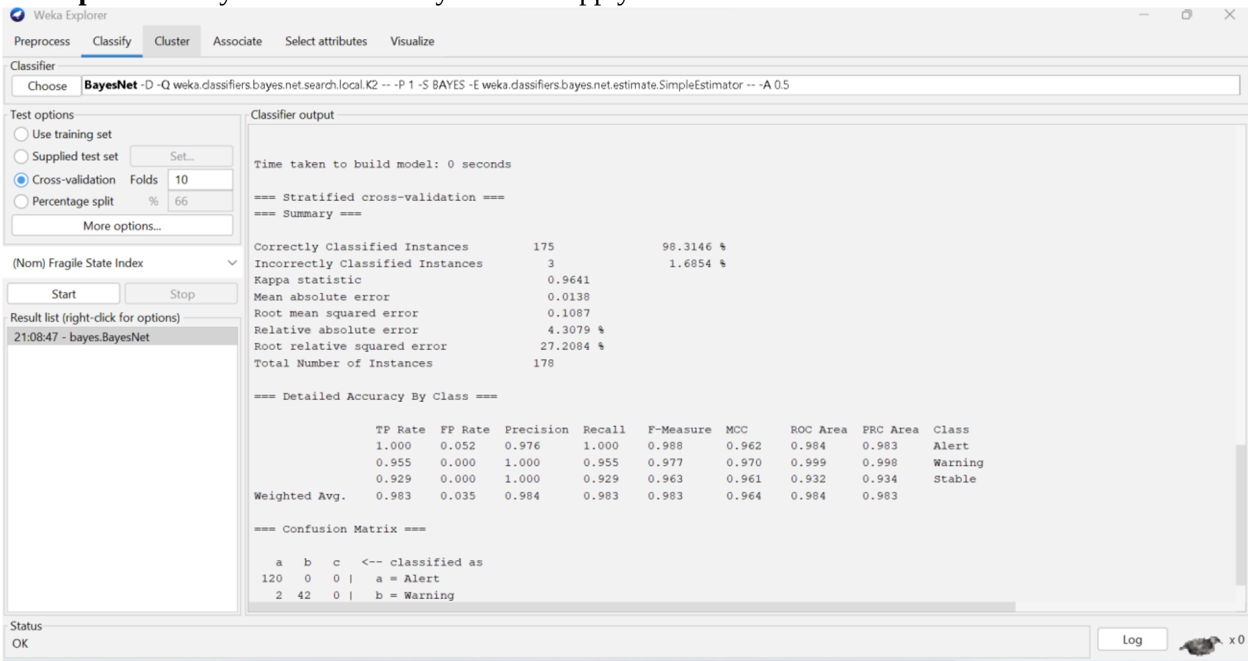


Figure 7.4: Applied Bayes Net Classifier on Extended Data.

Output:

== Run information ===

Scheme: weka.classifiers.bayes.BayesNet -D -Q weka.classifiers.bayes.net.search.local.K2 -- -P 1 -S BAYES -E weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5
Relation: Extended
Instances: 178
Attributes: 23
Country
Year
Rank
Total
C1: Security Apparatus
C2: Factionalized Elites
C3: Group Grievance
E1: Economy
E2: Economic Inequality
E3: Human Flight and Brain Drain
P1: State Legitimacy
P2: Public Services
P3: Human Rights
S1: Demographic Pressures
S2: Refugees and IDPs

X1: External Intervention
Education (Mean years of schooling
Health (Current health expenditure (% of GDP)
Unemployment
Population growth
Corruption
Human Development Index
Fragile State Index
Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=23 #classindex=22
Network structure (nodes followed by parents)
Country (177): Fragile State Index
Year (1): Fragile State Index
Rank (158): Fragile State Index
Total (3): Fragile State Index
C1: Security Apparatus (4): Fragile State Index
C2: Factionalized Elites (3): Fragile State Index
C3: Group Grievance (3): Fragile State Index
E1: Economy (3): Fragile State Index
E2: Economic Inequality (4): Fragile State Index
E3: Human Flight and Brain Drain (3): Fragile State Index
P1: State Legitimacy (4): Fragile State Index
P2: Public Services (3): Fragile State Index
P3: Human Rights (4): Fragile State Index
S1: Demographic Pressures (4): Fragile State Index
S2: Refugees and IDPs (3): Fragile State Index
X1: External Intervention (3): Fragile State Index
Education (Mean years of schooling) (3): Fragile State Index
Health (Current health expenditure (% of GDP) (2): Fragile State Index
Unemployment (1): Fragile State Index
Population growth (2): Fragile State Index
Corruption (1): Fragile State Index
Human Development Index (2): Fragile State Index
Fragile State Index (3):
LogScore Bayes: -3830.276205696485
LogScore BDeu: -11423.429032919707
LogScore MDL: -9021.998215446984
LogScore ENTROPY: -6148.6992368100155
LogScore AIC: -7257.6992368100155

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances	175	98.3146 %
Incorrectly Classified Instances	3	1.6854 %
Kappa statistic	0.9641	
Mean absolute error	0.0138	
Root mean squared error	0.1087	
Relative absolute error	4.3079 %	
Root relative squared error	27.2084 %	
Total Number of Instances	178	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1.000	0.052	0.976	1.000	0.988	0.962	0.984	0.983	Alert
	0.955	0.000	1.000	0.955	0.977	0.970	0.999	0.998	Warning
	0.929	0.000	1.000	0.929	0.963	0.961	0.932	0.934	Stable
Weighted Avg.	0.983	0.035	0.984	0.983	0.983	0.964	0.984	0.983	

=== Confusion Matrix ===

```
  a   b   c   <-- classified as
  120   0   0 |   a = Alert
42   0 |   b = Warning
 0  13 |   c = Stable
```

2. Random Forest Classifier:

Input: Classify -> Choose ->RandomForest -> Apply

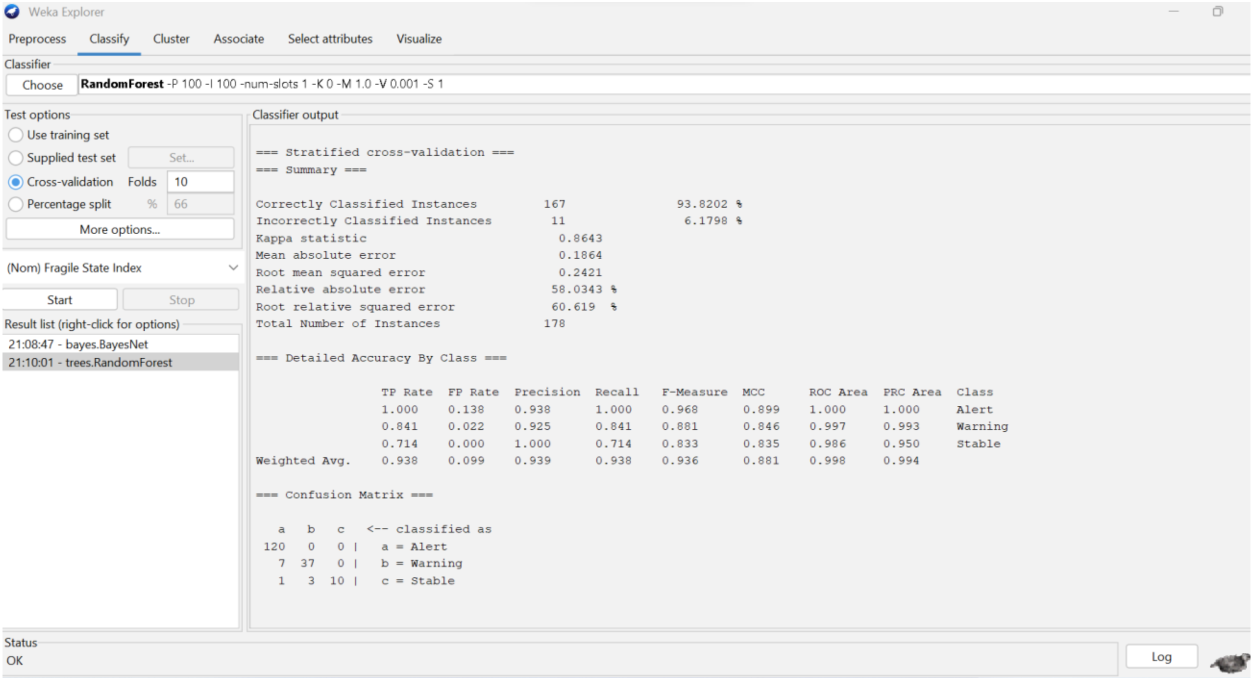


Figure 7.5: Applied Random Forest Classifier on Extended Data.

Output:

=== Run information ===

Scheme: weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Relation: Extended

Instances: 178

Attributes: 23

Country
Year
Rank
Total
C1: Security Apparatus
C2: Factionalized Elites
C3: Group Grievance
E1: Economy
E2: Economic Inequality
E3: Human Flight and Brain Drain
P1: State Legitimacy
P2: Public Services
P3: Human Rights
S1: Demographic Pressures
S2: Refugees and IDPs
X1: External Intervention

```
              Education (Mean years of schooling
              Health (Current health expenditure (% of GDP)
              Unemployment
              Population growth
              Corruption
              Human Development Index
              Fragile State Index
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner

weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        167               93.8202 %
Incorrectly Classified Instances       11                6.1798 %
Kappa statistic                     0.8643
Mean absolute error                   0.1864
Root mean squared error               0.2421
Relative absolute error              58.0343 %
Root relative squared error          60.619 %
Total Number of Instances            178

=== Detailed Accuracy By Class ===

              TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area
Class
              1.000    0.138    0.938      1.000    0.968      0.899    1.000     1.000     Alert
              0.841    0.022    0.925      0.841    0.881      0.846    0.997     0.993     Warning
              0.714    0.000    1.000      0.714    0.833      0.835    0.986     0.950     Stable
Weighted Avg.    0.938    0.099    0.939      0.938    0.936      0.881    0.998     0.994

=== Confusion Matrix ===

  a   b   c   <-- classified as
   0   0 |   a = Alert
  37   0 |   b = Warning
              1    3  10 |   c = Stable
```

3. Randomizable Filter Classifier:

Input: Classify -> Choose ->RandomizableFilter-> Apply

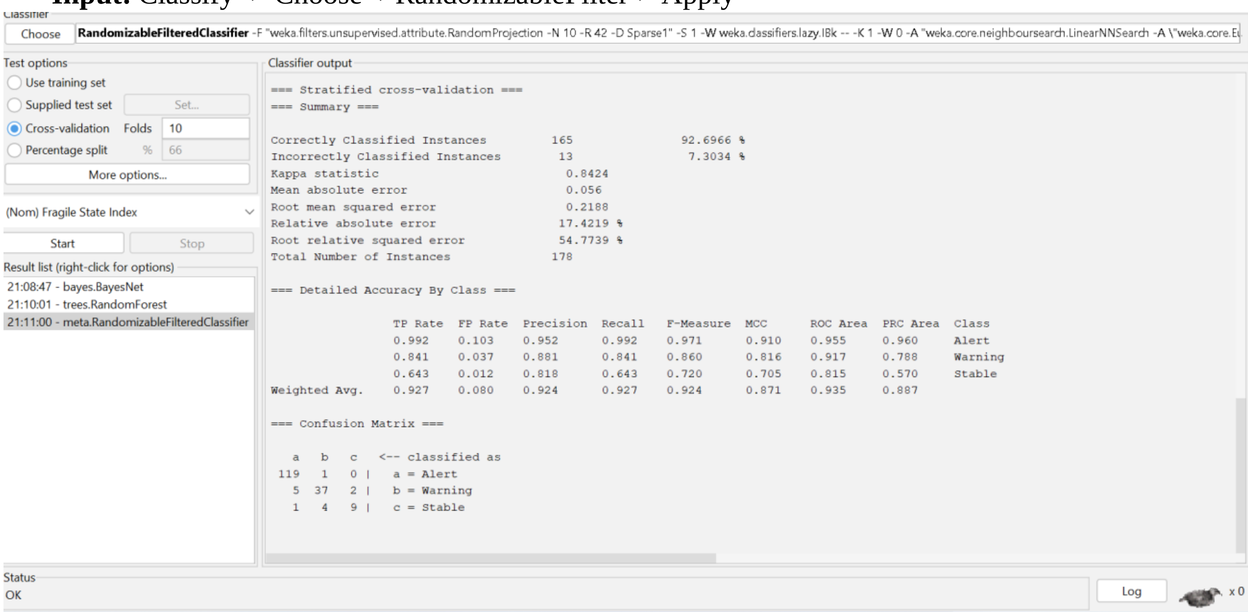


Figure 7.6: Applied Randomizable Filter Classifier on Extended Data.

Output:

=== Run information ===

Scheme: weka.classifiers.meta.RandomizableFilteredClassifier -F "weka.filters.unsupervised.attribute.RandomProjection -N 10 -R 42 -D Sparse1" -S 1 -W weka.classifiers.lazy.IBk -- -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""
Relation: Extended
Instances: 178
Attributes: 23
Country
Year
Rank
Total
C1: Security Apparatus
C2: Factionalized Elites
C3: Group Grievance
E1: Economy
E2: Economic Inequality
E3: Human Flight and Brain Drain
P1: State Legitimacy
P2: Public Services
P3: Human Rights
S1: Demographic Pressures
S2: Refugees and IDPs
X1: External Intervention
Education(Mean years of schooling
Health(Current health expenditure (% of GDP)
Unemployment

Population growth
Corruption
Human Development Index
Fragile State Index
Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

RandomizableFilteredClassifier using weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\"" on data filtered through weka.filters.unsupervised.attribute.RandomProjection -N 10 -R -1634313657 -D Sparse1

Filtered Header
@relation Extended-weka.filters.supervised.attribute.NominalToBinary-weka.filters.unsupervised.attribute.RandomProjection-N10-R-1634313657-DSparse1

@attribute K1 numeric
@attribute K2 numeric
@attribute K3 numeric
@attribute K4 numeric
@attribute K5 numeric
@attribute K6 numeric
@attribute K7 numeric
@attribute K8 numeric
@attribute K9 numeric
@attribute K10 numeric
@attribute 'Fragile State Index' {Alert,Warning,Stable}

@data

Classifier Model
IB1 instance-based classifier
using 1 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances	165	92.6966 %
Incorrectly Classified Instances	13	7.3034 %
Kappa statistic	0.8424	
Mean absolute error	0.056	
Root mean squared error	0.2188	
Relative absolute error	17.4219 %	
Root relative squared error	54.7739 %	
Total Number of Instances	178	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.992	0.103	0.952	0.992	0.971	0.910	0.955	0.960	Alert
	0.841	0.037	0.881	0.841	0.860	0.816	0.917	0.788	Warning
	0.643	0.012	0.818	0.643	0.720	0.705	0.815	0.570	Stable
Weighted Avg.	0.927	0.080	0.924	0.927	0.924	0.871	0.935	0.887	

=== Confusion Matrix ===

a	b	c	<-- classified as
119	1	0	a = Alert
5	37	2	b = Warning
1	4	9	c = Stable

8. Generation of Action Rules using Lisp Miner

Action Rules:

Action rules apply to databases where features are classified as flexible or stable. A flexible property is required to reclassify a set of objects into a new decision class. Use the Lisp Miner program to create action rules for the Fragile State Index 2010 dataset.

Lisp Miner:

The LISP Miner System is an academic data mining software program developed at the University of Economics in Prague. This is a project focused on extracting various kinds of association rules from categorical data. LISP miners use multiple data mining processes to create different kinds of associations between the left and right sides of a rule. This project uses Ac4ft miner data mining technology to extract action rules. Ac4ft-Miner identifies rules outlining actions to be taken to improve a defined condition. This is done by analyzing the relationships between the data provided as input.

Here's how Lisp Miner action rules are detected:

LISP miners use different GUHA processes to mine different kinds of knowledge patterns. The system consists of 10 different data mining methods, 4 of which are based on his original GUHA method ASSOC, the rest occurred during the development of the system.

The 4ft miner process looks for knowledge patterns that can be viewed as his 4ft association rules of this type.

$\varphi \approx \psi / \gamma$

Where φ (preceding), ψ (continuous), and γ (conditional) are cedants, and $\approx$ is a quantifier applied to the subset of samples that satisfy the condition.

Working with LISP miners is more difficult than working with other data mining systems because they come as a set of executables that users must call. Each mining technique in the LISP miner program uses different processing engines, including:

1. LM Admin:

The LMAdmin module is used first. The main purpose of this module is to connect to the data examined in the metadata base. The concept of metadata enables storage and reuse of task inputs and analysis results. Databases are used to store both data and metadata. This is a mandatory step that must be completed before running any analysis.

2. LM Data Source:

This module contains multiple data transformation and preparation techniques that can be used to select features for specific data mining operations, generate derived attributes, or discretize numeric attributes.

3. Data processing (Tasks):

This is a task module that analyzes data and creates tasks using related xxxTask modules.

4. Data Interpretation (Results):

The Results module is responsible for displaying and evaluating results. When the associated xxxResult module is executed, this is used to display, sort or select the created rules.

Properties:

We have classified our attributes into three categories:

- Stable Attributes
- Flexible Attributes
- Decision Attributes

We choose the following stable properties:

- **Antecedent Stable Part:** We distribute all the consistent ascribes to this community.
- **Antecedent Variable Part:** This collection contains all the adaptable ascribes.
- **Succedent Variable Part:** For this set, we assigned the following decision variable:
  - Attribute Type = Nominal
  - Coefficient Type = One Category

Quantifiers:

- a(BASE0 Before: 2
- a(BASE) After: 2

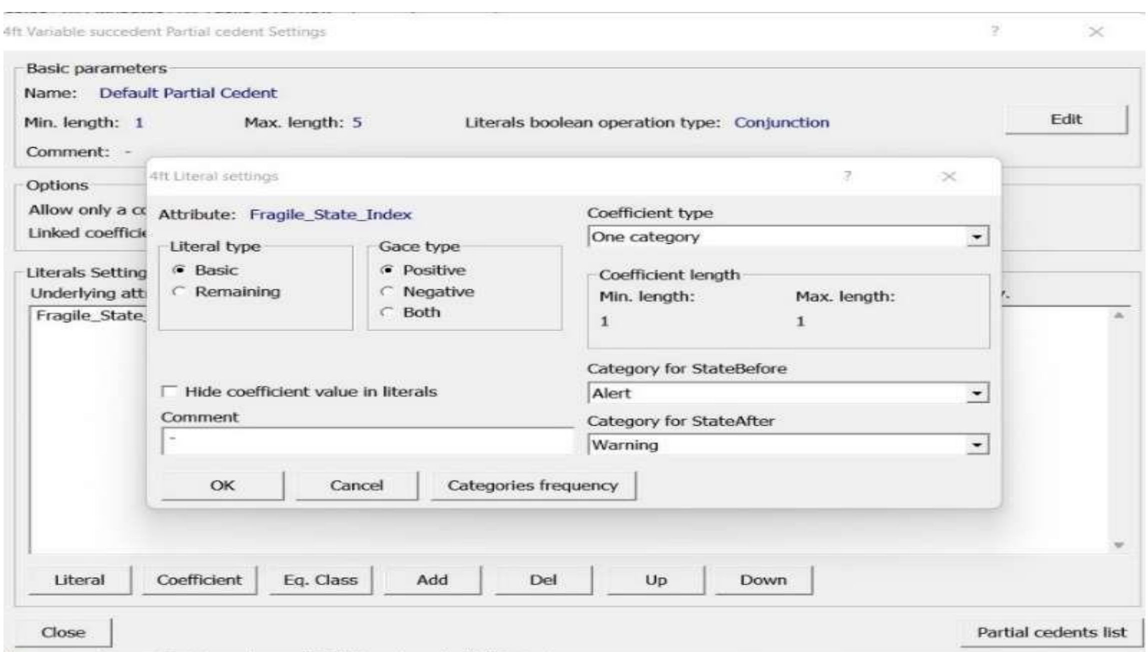
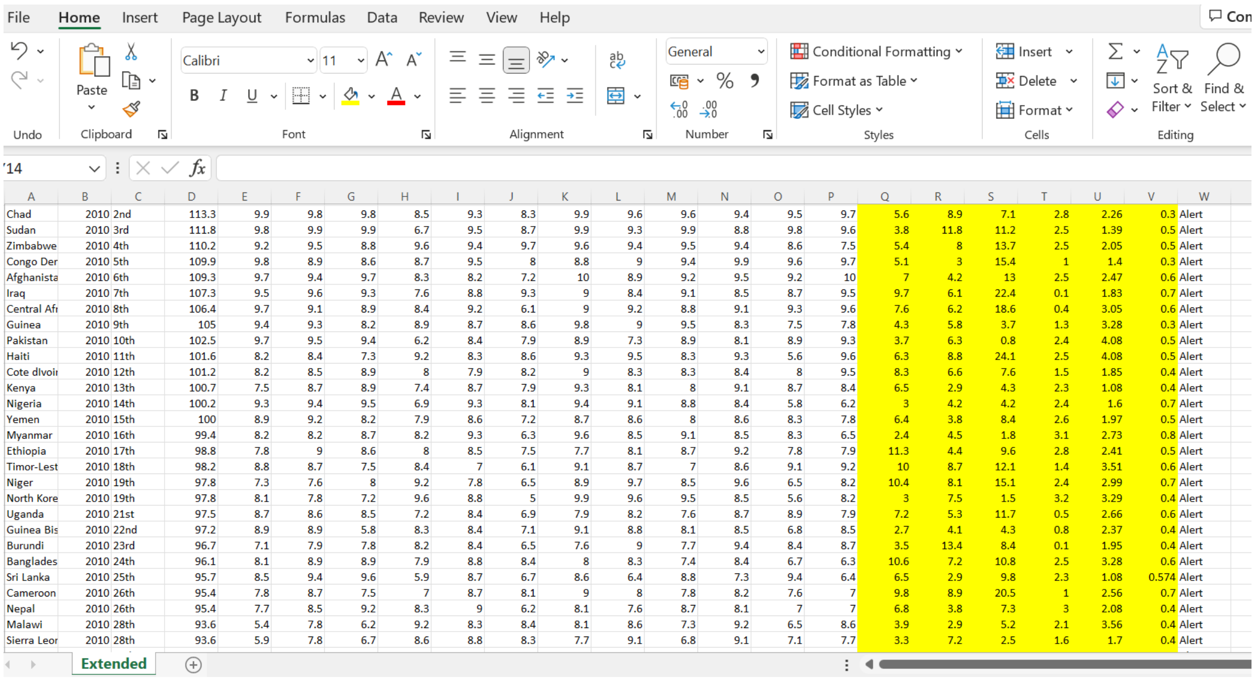


Figure 8.1: Changing the coefficient and Category for Succedent Variable.

9. LISP Miner Screenshots

Step 1: The initial stage is to get a dataset, which is subsequently modified by adding 6 new attributes.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Chad	2010	2nd	113.3	9.9	9.8	9.8	8.5	9.3	8.3	9.9	9.6	9.6	9.4	9.5	9.7	5.6	8.9	7.1	2.8	2.26	0.3	Alert
Sudan	2010	3rd	111.8	9.8	9.9	9.9	6.7	9.5	8.7	9.9	9.3	9.9	8.8	9.8	9.6	3.8	11.8	11.2	2.5	1.39	0.5	Alert
Zimbabwe	2010	4th	110.2	9.2	9.5	8.8	9.6	9.4	9.7	9.6	9.4	9.5	9.4	8.6	7.5	5.4	8	13.7	2.5	2.05	0.5	Alert
Congo Der	2010	5th	109.9	9.8	8.9	8.6	8.7	9.5	8	8.8	9	9.4	9.9	9.6	9.7	5.1	3	15.4	1	1.4	0.3	Alert
Afghanista	2010	6th	109.3	9.7	9.4	9.7	8.3	8.2	7.2	10	8.9	9.2	9.5	9.2	10	7	4.2	13	2.5	2.47	0.6	Alert
Iraq	2010	7th	107.3	9.5	9.6	9.3	7.6	8.8	9.3	9	8.4	9.1	8.5	8.7	9.5	9.7	6.1	22.4	0.1	1.83	0.7	Alert
Central Afı	2010	8th	106.4	9.7	9.1	8.9	8.4	9.2	6.1	9	9.2	8.8	9.1	9.3	9.6	7.6	6.2	18.6	0.4	3.05	0.6	Alert
Guinea	2010	9th	105	9.4	9.3	8.2	8.9	8.7	8.6	9.8	9	9.5	8.3	7.5	7.8	4.3	5.8	3.7	1.3	3.28	0.3	Alert
Pakistan	2010	10th	102.5	9.7	9.5	9.4	6.2	8.4	7.9	8.9	7.3	8.9	8.1	8.9	9.3	3.7	6.3	0.8	2.4	4.08	0.5	Alert
Haiti	2010	11th	101.6	8.2	8.4	7.3	9.2	8.3	8.6	9.3	9.5	8.3	9.3	5.6	9.6	6.3	8.8	24.1	2.5	4.08	0.5	Alert
Cote dIvoiı	2010	12th	101.2	8.2	8.5	8.9	8	7.9	8.2	9	8.3	8.3	8.4	8	9.5	8.3	6.6	7.6	1.5	1.85	0.4	Alert
Kenya	2010	13th	100.7	7.5	8.7	8.9	7.4	8.7	7.9	9.3	8.1	8	9.1	8.7	8.4	6.5	2.9	4.3	2.3	1.08	0.4	Alert
Nigeria	2010	14th	100.2	9.3	9.4	9.5	6.9	9.3	8.1	9.4	9.1	8.8	8.4	5.8	6.2	3	4.2	4.2	2.4	1.6	0.7	Alert
Yemen	2010	15th	100	8.9	9.2	8.2	7.9	8.6	7.2	8.7	8.6	8	8.6	8.3	7.8	6.4	3.8	8.4	2.6	1.97	0.5	Alert
Myanmar	2010	16th	99.4	8.2	8.2	8.7	8.2	9.3	6.3	9.6	8.5	9.1	8.5	8.3	6.5	2.4	4.5	1.8	3.1	2.73	0.8	Alert
Ethiopia	2010	17th	98.8	7.8	9	8.6	8	8.5	7.5	7.7	8.1	8.7	9.2	7.8	7.9	11.3	4.4	9.6	2.8	2.41	0.5	Alert
Timor-Lest	2010	18th	98.2	8.8	8.7	7.5	8.4	7	6.1	9.1	8.7	7	8.6	9.1	9.2	10	8.7	12.1	1.4	3.51	0.6	Alert
Niger	2010	19th	97.8	7.3	7.6	8	9.2	7.8	6.5	8.9	9.7	8.5	9.6	6.5	8.2	10.4	8.1	15.1	2.4	2.99	0.7	Alert
North Kore	2010	19th	97.8	8.1	7.8	7.2	9.6	8.8	5	9.9	9.6	9.5	8.5	5.6	8.2	3	7.5	1.5	3.2	3.29	0.4	Alert
Uganda	2010	21st	97.5	8.7	8.6	8.5	7.2	8.4	6.9	7.9	8.2	7.6	8.7	8.9	7.9	7.2	5.3	11.7	0.5	2.66	0.6	Alert
Guinea Bis	2010	22nd	97.2	8.9	8.9	5.8	8.3	8.4	7.1	9.1	8.8	8.1	8.5	6.8	8.5	2.7	4.1	4.3	0.8	2.37	0.4	Alert
Burundi	2010	23rd	96.7	7.1	7.9	7.8	8.2	8.4	6.5	7.6	9	7.7	9.4	8.4	8.7	3.5	13.4	8.4	0.1	1.95	0.4	Alert
Banglades	2010	24th	96.1	8.1	8.9	8.9	7.9	8.8	8.4	8	8.3	7.4	8.4	6.7	6.3	10.6	7.2	10.8	2.5	3.28	0.6	Alert
Sri Lanka	2010	25th	95.7	8.5	9.4	9.6	5.9	8.7	6.7	8.6	6.4	8.8	7.3	9.4	6.4	6.5	2.9	9.8	2.3	1.08	0.574	Alert
Cameroon	2010	26th	95.4	7.8	8.7	7.5	7	8.7	8.1	9	8	7.8	8.2	7.6	7	9.8	8.9	20.5	1	2.56	0.7	Alert
Nepal	2010	26th	95.4	7.7	8.5	9.2	8.3	9	6.2	8.1	7.6	8.7	8.1	7	7	6.8	3.8	7.3	3	2.08	0.4	Alert
Malawi	2010	28th	93.6	5.4	7.8	6.2	9.2	8.3	8.4	8.1	8.6	7.3	9.2	6.5	8.6	3.9	2.9	5.2	2.1	3.56	0.4	Alert
Sierra Leor	2010	28th	93.6	5.9	7.8	6.7	8.6	8.8	8.3	7.7	9.1	6.8	9.1	7.1	7.7	3.3	7.2	2.5	1.6	1.7	0.4	Alert

Figure 9.1: Extended Dataset after adding additional six features.

Step 2: Import the dataset into LISP Miner program by clicking the "New from TXT" button. When importing data via text file, the file must be of the text or csv format that processes all 178 rows, one for each country.

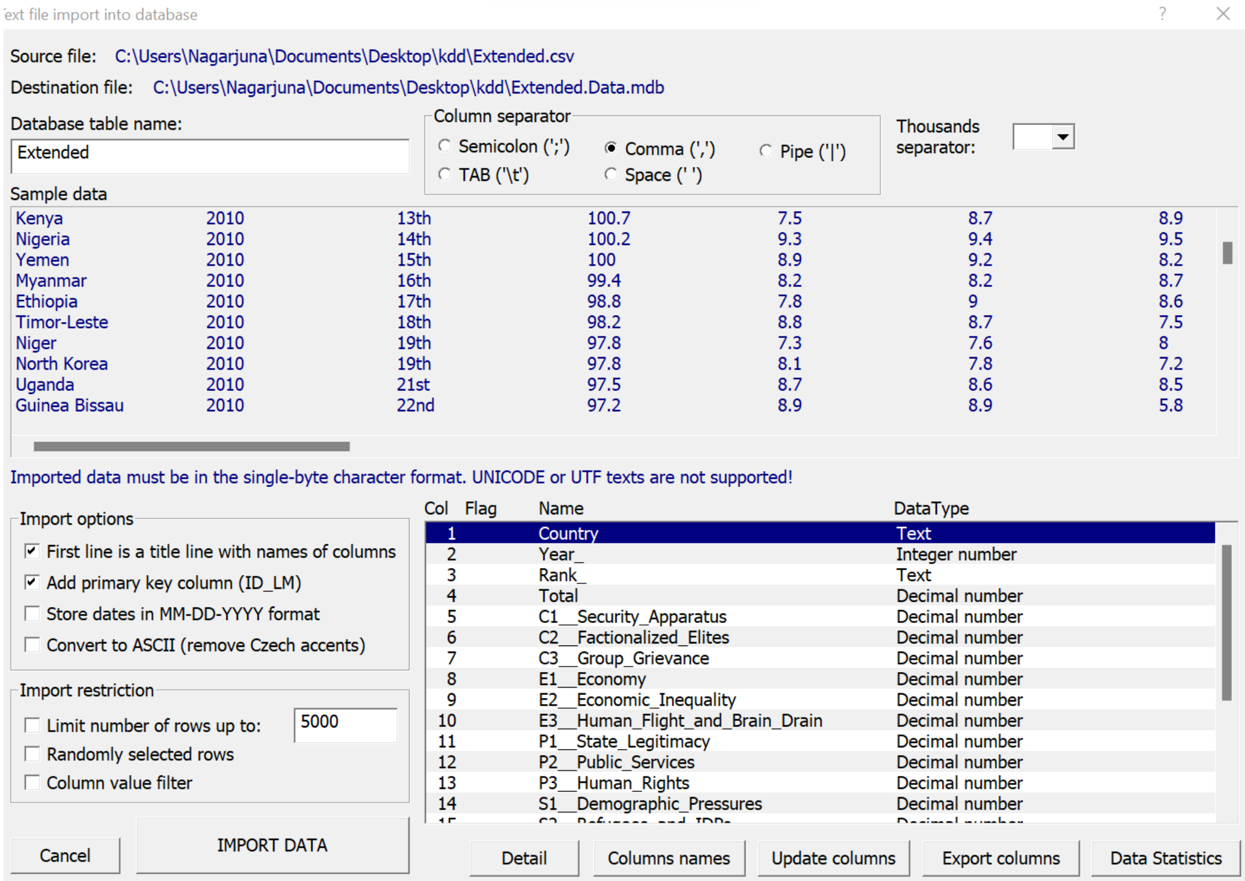


Figure 9.2: Extended data of 2010 imported in LISP Miner to perform further analysis.

Step 3: After the file is loaded, it can be viewed in the LISP Miner software in the table as the dataset is converted to a database in the previous step.

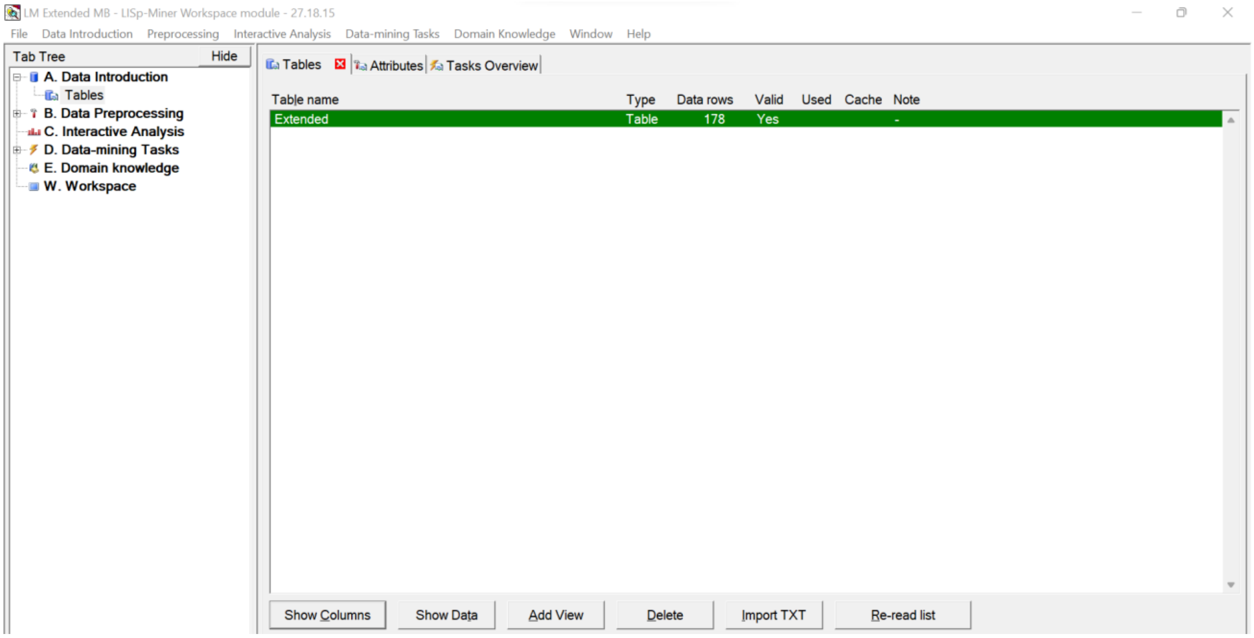


Figure 9.3: Dataset is converted to a database .mdb file.

Step 4: The entire database attributes can be seen in the columns sub section of Data Introduction inthe Lisp Miner tool. Now, select all the columns except ID and click on create attribute.

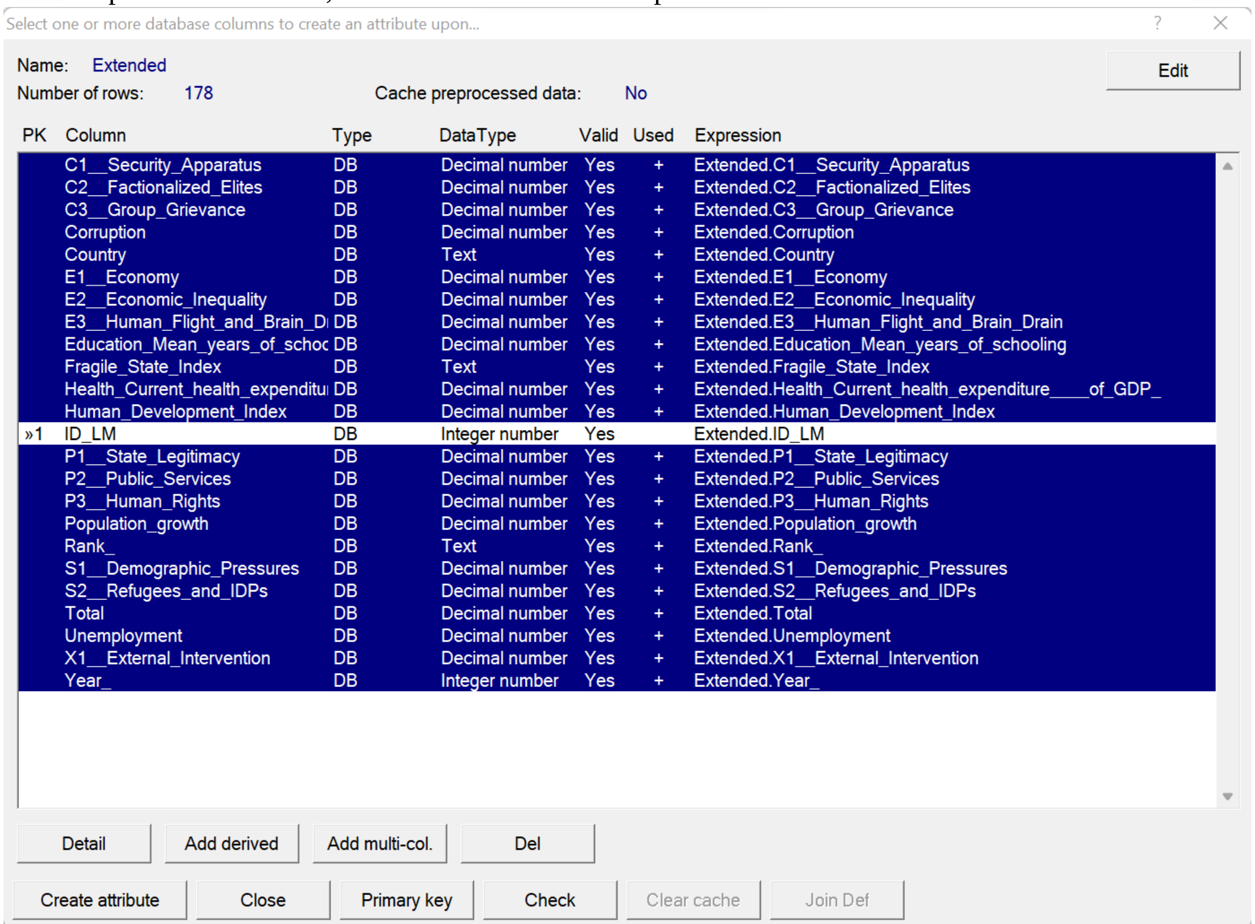


Figure 9.4: Selecting Attributes except system generated ID.

Now, Change 'f'-bound for interval boundaries in Float values from 0.01 to 0.001.

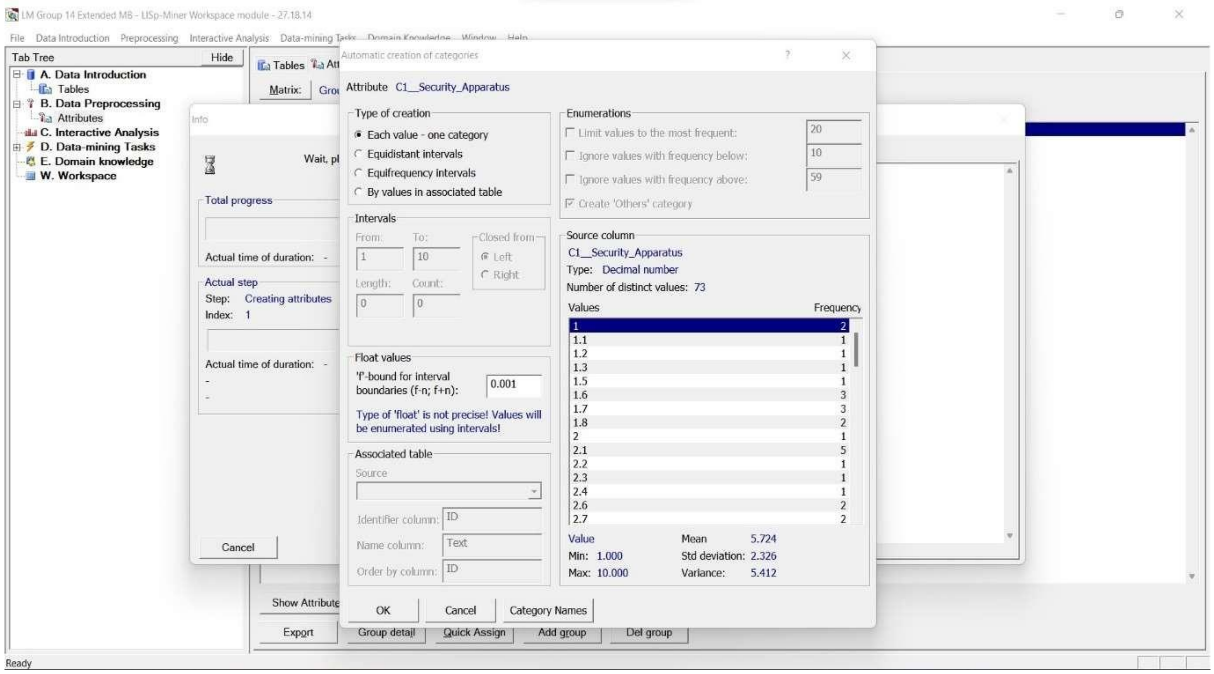


Figure 9.5: Changing the float value to 0.001 from 0.01.

All the attributes are now added except ID.

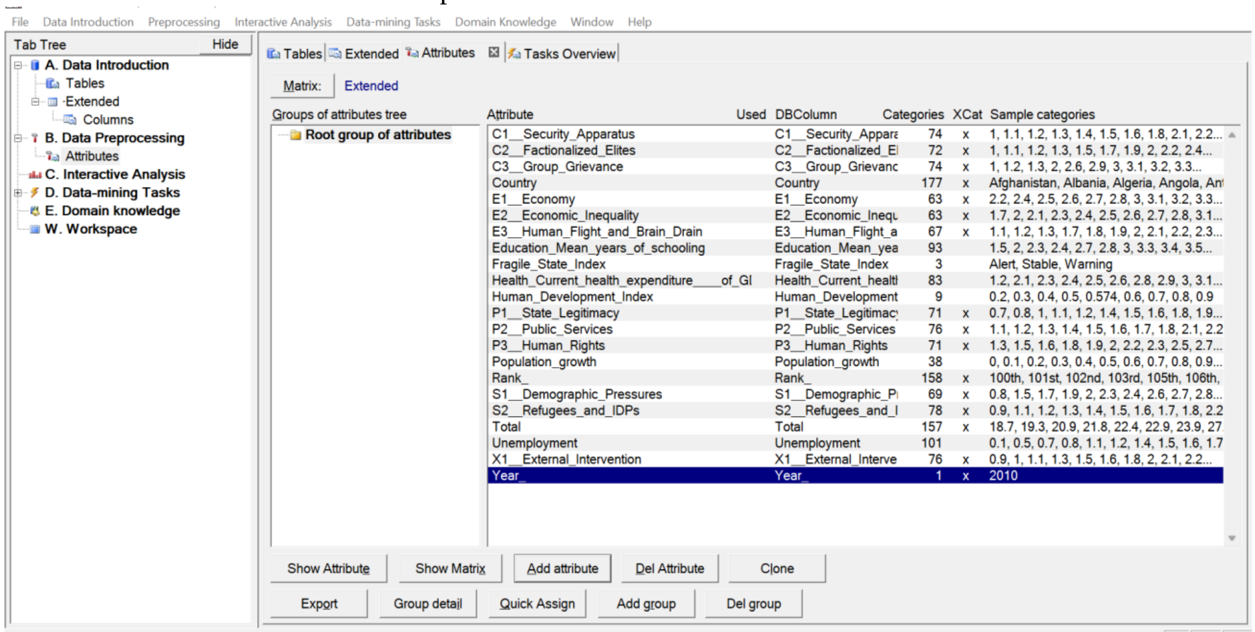


Figure 9.6: Selected Attributes.

Step 5: Click on Tasks Overview and add any task.

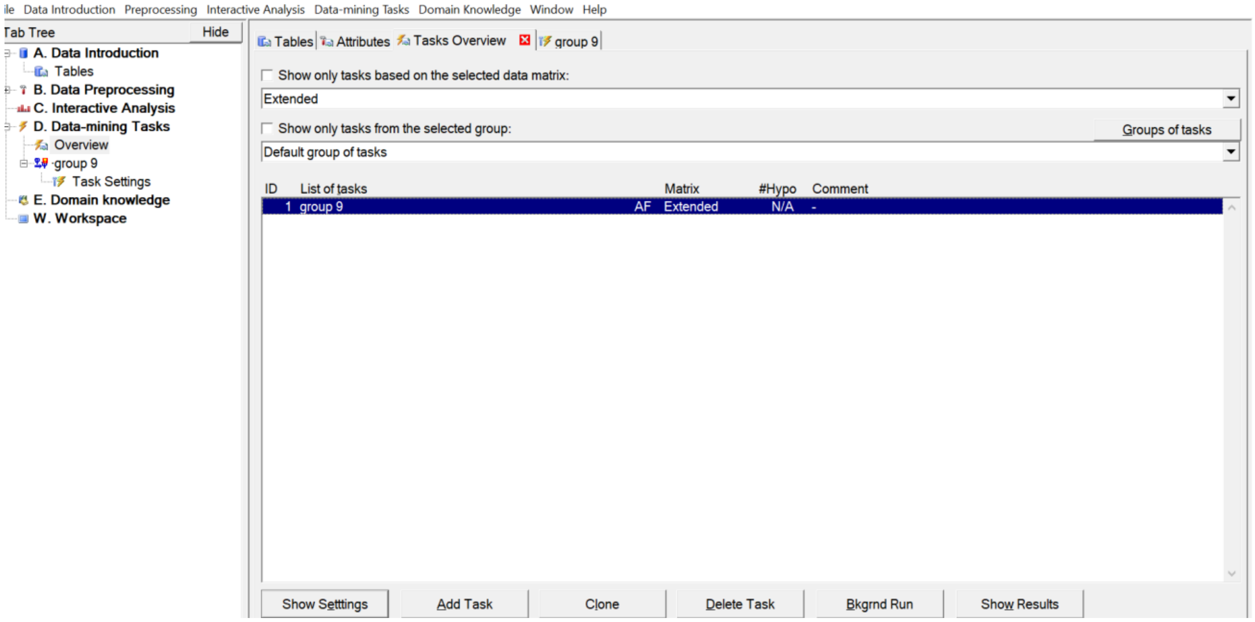


Figure 9.7: Adding the task in task overview.

Step 6: After adding the required attributes from the dataset, a new task is initialized. To perform Action rules tasks, we select Ac4ft-Miner which is an action rules miner. Now, select Ac4ft-Miner in Tasks Tab.

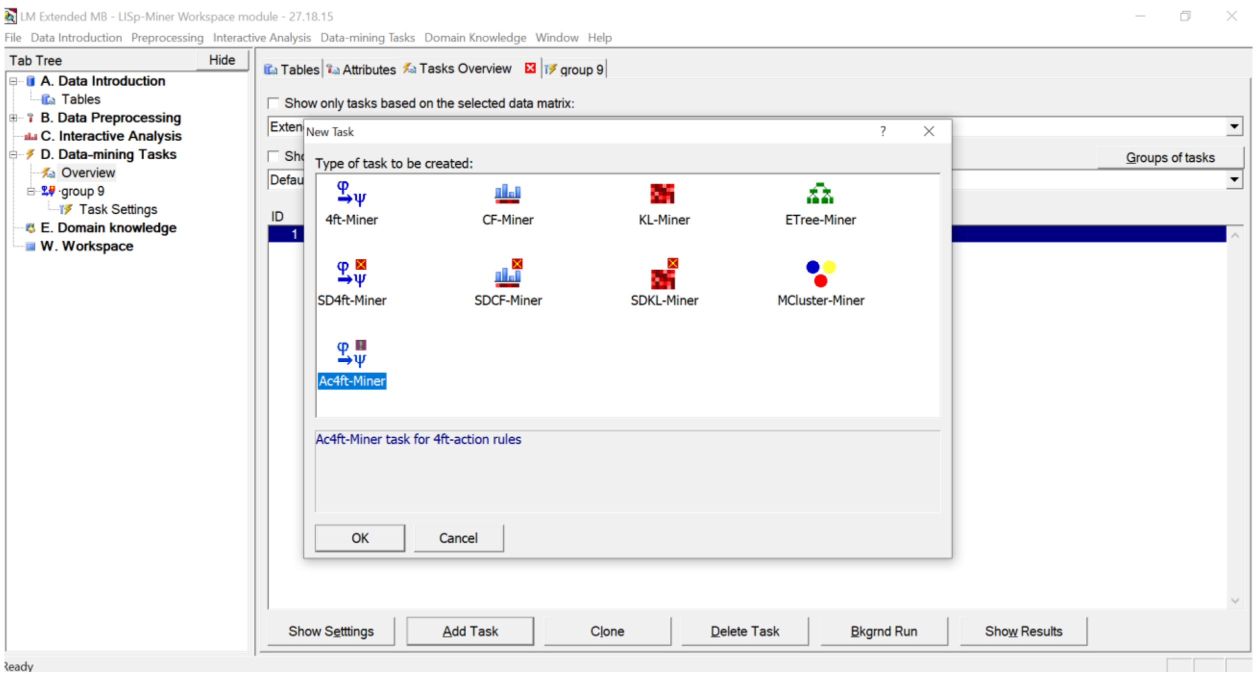


Figure 9.8: Selecting Ac4ft-Miner task to generate action rules.

Step 7: Selected Country, Rank, year attribute in the Antecedent stable part, Education_Mean_years_of_schooling and Unemployment in the Antecedent variable part. Selected Fragile State Index in the Succedent Variable part with the one category.

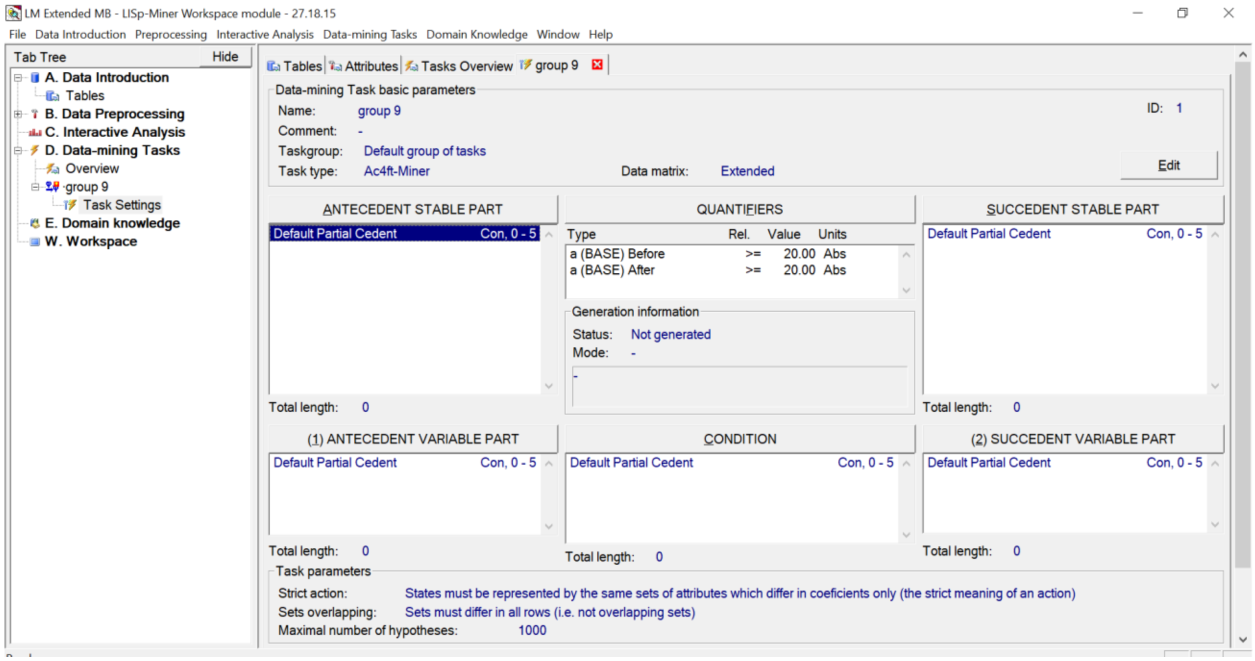


Figure 9.9: Parameter Settings.

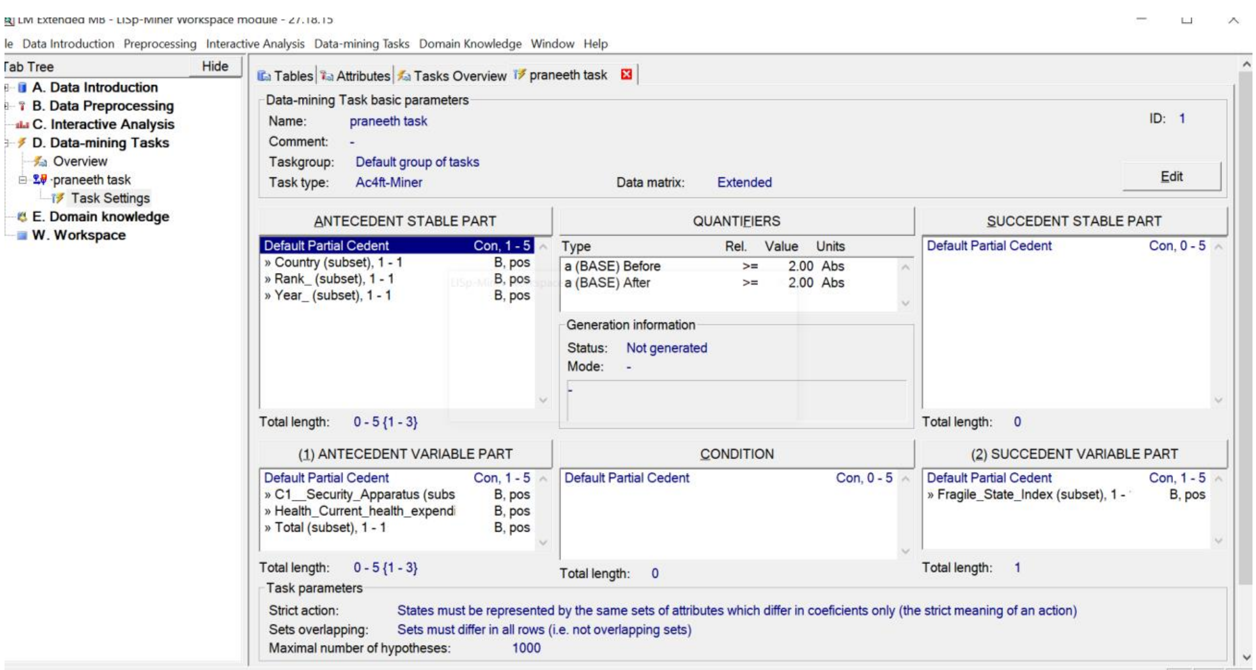


Figure 9.10: Added the attributes in Antecedent Stable and Variable Parts and Succedent Variable.

The coefficient type is one category, and the statebefore and stateafter attributes are set to Alert to Warning, implying that action rules will be developed for countries in the alert stage to move into thewarning stage. A threshold of 2 in before and after value in the Quantifiers.

Step 8: Click Run.

Output:

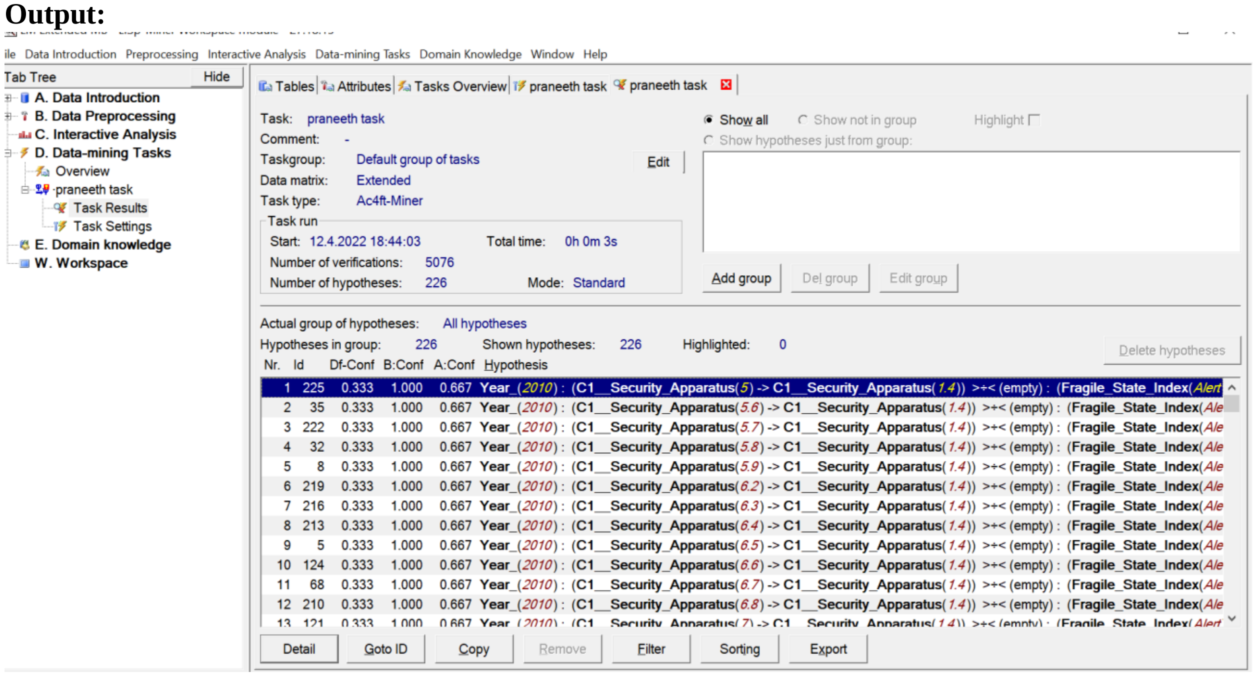


Figure 9.11: Task output which contains action rules.

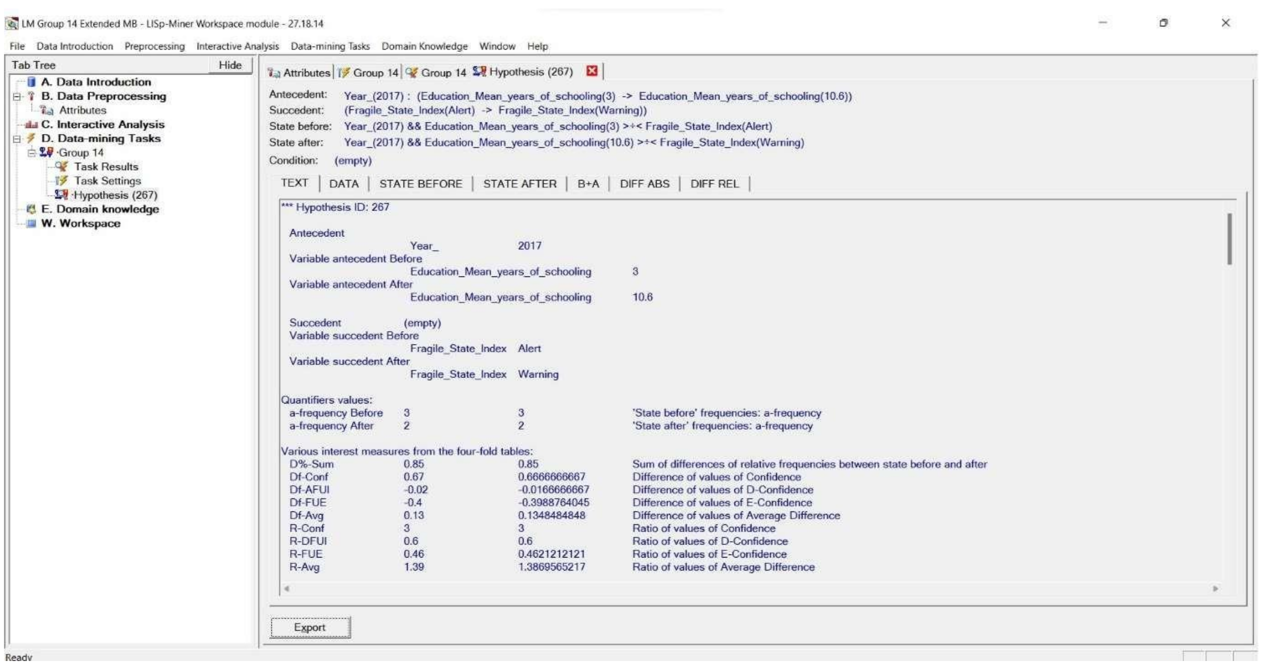


Figure 9.12: Task Hypothesis.

Visualization:

Action Rules-State Before:

Figure 9.13: Hypothesis for State Before.

Action Rules-State After:

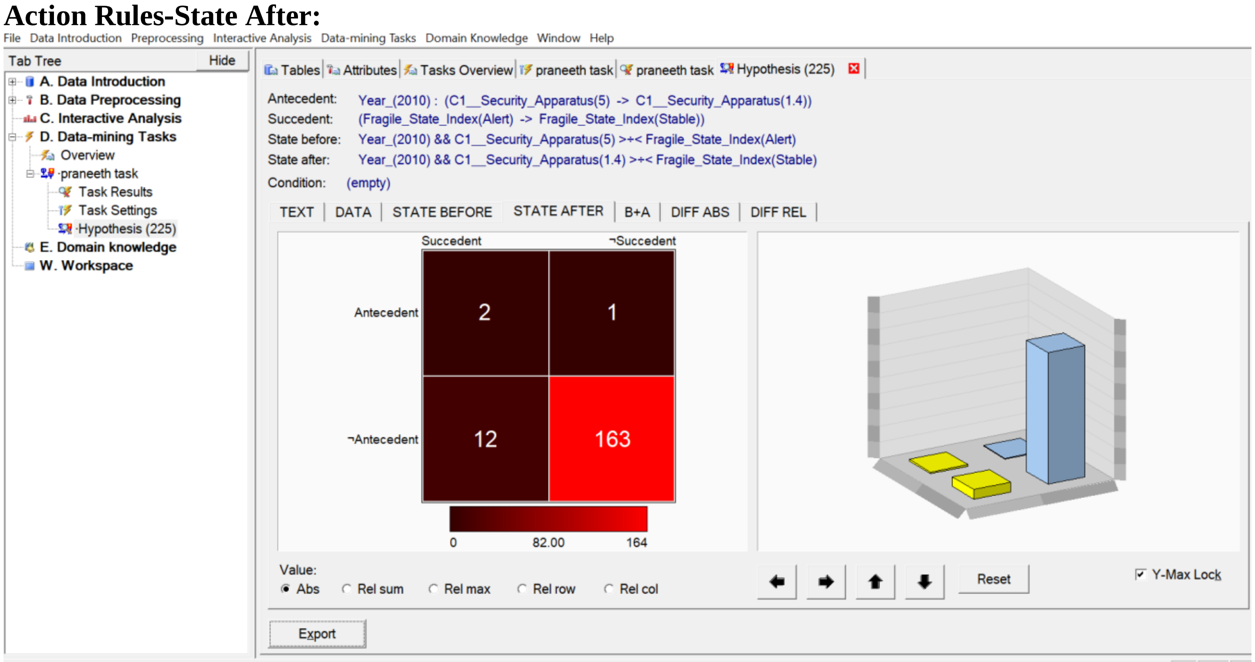


Figure 9.14: Hypothesis for State After.

Action Rules – B+A:

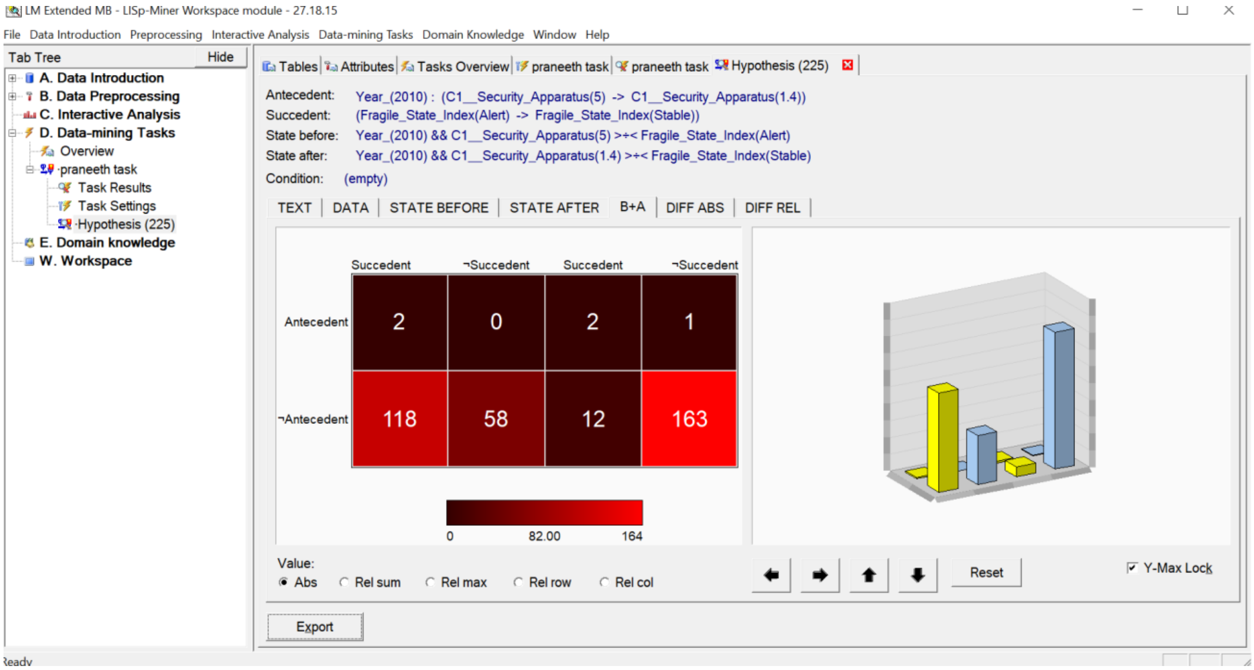


Figure 9.15: Hypothesis for B+A.

Action Rules – DIFF ABS:

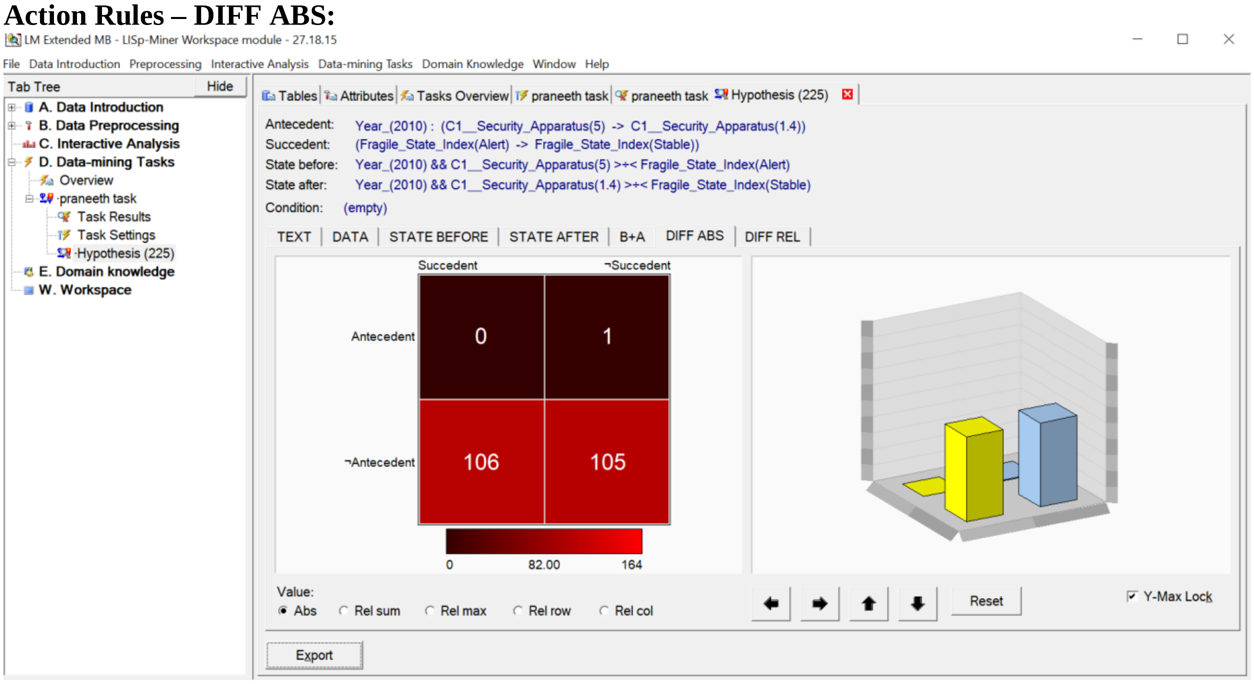


Figure 9.16: Hypothesis for DIFF ABS.

Action Rules – DIFF REL:

Figure 9.17: Hypothesis for DIFF REL.

10. Fragile Attributes Used

1. **Education (Mean Years of Schooling (Years)):** The average number of years of education for people aged 25 and over converted from the education level by the official retirement age for each level. By 2030, increase the number of skilled young people, including the technical and vocational skills needed for employment, good jobs and entrepreneurship.
2. **Unemployment:** Percentage of the workforce aged 15 and over who are not in paid employment or self-employed, but who are available and are taking steps to find paid employment or self-employment. By 2030, achieve full and productive employment and decent work for all women and men, including for young people and persons with disabilities, and equal pay for equal work.

11. Action Rules and Inferences

Stable Attributes: Country, Rank and Year
Flexible attributes: Education_Mean_years_of_schooling and Unemployment
Decision Attribute: Fragile State Index
Action Rules: (Boundary values)
1. (Education_Mean_years_of_schooling(3) -> Education_Mean_years_of_schooling(10.6)) >÷< (Fragile_State_Index(Alert) -> Fragile_State_Index(Warning))
2. (Education_Mean_years_of_schooling(8) -> Education_Mean_years_of_schooling(10.6)) >÷< (Fragile_State_Index(Alert) -> Fragile_State_Index(Warning))
3. (Unemployment (1.6) -> Unemployment (4.2)) >÷< (Fragile_State_Index(Alert) -> Fragile_State_Index(Warning))
4. (Unemployment (1.8) -> Unemployment (4.2)) >÷< (Fragile_State_Index(Alert) -> Fragile_State_Index(Warning))

The extracted rules are included in the text file below.

Group 10.txt

12. Conclusion

Added six new features to the current FSI dataset for 2010. Each affects the assessment of a country's Fragile States Index. Data were preprocessed using cleansing and normalization methods. The data were then discretized and classified using the WEKA program. Classification rules were found in WEKA. Then I used Lisp Miner to create action rules. The developed rules of conduct can be seen as preventive measures that a country can take to improve its position. A screenshot of the entire process is attached throughout. I have also attached all the results to a ZIP file for review.

13. References

https://fragilestatesindex.org/wp-content/uploads/data/fsi-2010.xlsx
https://en.wikipedia.org/wiki/Fragile_States_Index
https://lispminer.vse.cz/demonstration/index.html
https://weka.sourceforge.io/doc.dev/weka/classifiers/trees/RandomForest.html
http://weka.sourceforge.net/doc.dev/weka/classifiers/bayes/NaiveBayes.html
https://weka.sourceforge.io/doc.dev/weka/classifiers/meta/RandomizableFilteredClassifier. html
https://hdr.undp.org/en/indicators/103006#
https://hdr.undp.org/en/indicators/140606#
https://hdr.undp.org/en/indicators/137506
https://databank.worldbank.org/
https://www.transparency.org/en/cpi/2010
https://hdr.undp.org/en/indicators/137506