SIF8015 Logic

Exercise 4

Applied Logic: Data Mining and Knowledge Discovery

Task 1

Consider the decision table $\mathcal{A} = (U, A \cup \{d\})$ given below, where Walk is the decision attribute.

	Age	Sex	LEMS	Walk
Smith	16-30	Male	50	Yes
Jones	16-30	Male	0	No
Parker	31-45	Male	1-25	No
Hanson	31-45	Male	1-25	Yes
Moore	46-60	Female	26-49	No
Fields	16-30	Female	26-49	Yes
Starr	46-60	Female	26-49	No

(a) For each object, compute its equivalence class with respect to the condition attributes.

(b) For each object, compute its value for the generalized decision attribute. Is the table consistent?

(c) Compute the entries of the discernibility matrix with respect to the condition attributes. Also compute the corresponding discernibility matrix computed modulo the decision attribute.

(d) Compute the Boolean discernibility function that expresses how all decision classes can be discerned from each other, and state its prime implicants. How do you interpret these?

(e) Compute the Boolean discernibility function derived from Ms. Fields that expresses how the decision class to which she belongs can be discerned from the other decision classes, and state its prime implicants. What are the decision rules that these define?

(f) Consider the following decision rule:

$$\text{if (Age} = \text{16-30) then (Walk} = \text{No)}$$

What is the accuracy and the coverage of this decision rule in the context of $\mathcal{A}$?

Task 2

In this exercise you will use the ROSETTA software system to perform a small KDD experiment.

GUI Preliminaries

For this exercise, the following ROSETTA GUI details are worth noting:

- An information system can be read into a new ROSETTA project by selecting **Open...** from the main **File** menu, and will be placed immediately below the root of the **Structures** node in the project tree.
- Branches in the project tree can be expanded or collapsed by left-clicking on the "⊞" or "⊟" symbols next to the icons.
- Right-clicking on an icon in the project tree brings up a pop-up menu for that object. In the following, the symbol "▶" will be used to denote menu navigation. Right-clicking on many other GUI items (e.g., row or column headers in data views) will often bring up pop-up menus, too.
- Left-clicking twice on an icon in the project tree can be used as a shortcut for viewing that object in detail.
- Grayed columns in views of decision systems indicate that the corresponding attributes are "masked away" and subsequently ignored by the ROSETTA kernel in any analysis steps. Missing values are indicated by the string "**Undefined**".
- Rules can be sorted directly in their views by right-clicking the column to sort by.
- To rename an object, first left-click once on its icon to select it. Then left-click once more on the icon's label. The icon's label is edited directly in place.
- To view progress messages and warnings, select **Messages** from the main **View** menu. (A project has to be present for this menu option to be present.)

Background

Amateur botanist and computer science student Daffy Dill has during a field trip picked 150 flowers for his herbarium. Daffy knows that the flowers all belong to the iris family of flowers, but he isn't quite sure to which subspecies of the iris family each flower belongs. Daffy therefore decides to consult with expert botanist Orry Kidd. Orry then tells Daffy the correct classification of each flower: Iris setosa, iris versicolor or iris virginica.

Daffy decides to write a small computer program that can help him classify iris flowers when he comes across them in the future, but isn't quite sure how he should implement the actual decision-making logic since Orry didn't disclose exactly how she arrived at her conclusions. But Daffy suspects that the length and width of the sepals and the petals of the flowers have something to do with it, since he could see that Orry examined these. Daffy therefore measures these features, and collects the measurements of each flower in a table together with the true classification of the flower as determined by Orry.

Daffy has installed the ROSETTA system on his PC, and decides to use this to induce minimal if-then rules that he can subsequently implement in the computer program he plans to write. By examining the rules that ROSETTA computes, he also hopes to learn something about which mechanisms Orry used when she classified the flowers. However, Daffy also needs to verify somehow that the rules he obtains are able to classify new flowers well, i.e., he needs to apply the rules to classify some new flowers and make an estimate of how well they do.

With your help, Daffy therefore decides to do the following small experiment:

1. Randomly split the set of 150 flowers into two disjoint sets: A training set from which he will induce minimal if-then rules, and a test set that he will use to verify how good the rules are at classifying new cases. Daffy decides to let 2/3 of the data belong to the training set, and 1/3 of the data to belong to the test set.
2. Discretize the training set, i.e., determine "cuts" that define intervals for each of the condition attributes.
3. Discretize the test set using the cuts that were found from the training set.
4. Induce minimal if-then rules from the discretized training set.
5. Apply the induced rules to the discretized test set to assess how accurate the rules are in predicting the correct iris subclass of new and unseen flowers in the iris family.

ROSETTA Steps

The iris data table can be found in the **Samples/Iris** folder where ROSETTA is installed. Start ROSETTA, and load the table[1]. Name the table *Iris*, and view the data and familiarize yourself with it.

(a) *Iris*▶**Other**▶**Split in two...**

```
FACTOR = 0.666;                 SEED = 1;

                                APPEND = T;
```

Name the largest of the resulting two tables *Training* and the other one *Test*.

(b) *Training*▶**Discretize**▶**Boolean reasoning algorithm (RSES)...**

```
MODE = Save;                    MASK = T;

FILENAME=c:/temp/cuts.txt
```

Name the resulting discretized table *Training discretized*.

(c) *Test*▶**Discretize**▶**From file with cuts (RSES)...**

```
MODE = Load;                    MASK = T;

FILENAME=c:/temp/cuts.txt
```

Name the resulting discretized table *Test discretized*.

(d) *Training discretized*▶**Reduce**▶**Johnson algorithm...**

[1] The table is in ordinary ROSETTA table import format. If prompted by the system to identify the format, simply indicate this.

```
DISCERNIBILITY = Object;           BRT = F;

SELECTION = All;                   IDG = F;

MODULO.DECISION = T;               PRECOMPUTE = F;

                                   APPROXIMATE = F;
```

Name the resulting set of reducts (i.e., prime implicants) as *Johnson reducts* and the resulting set of rules as *Johnson rules*.

(e) Examine *Johnson rules*. Are you able to arrive at some intuitive understanding as to how the flowers are classified? Comment on how you think the rules reflect the decision-making logic of an expert.

(f) *Test discretized*▶**Classify...**

```
CLASSIFIER = StandardVoter;        FRACTION = 0.0;

RULES = Johnson rules;             IDG = F;

FALLBACK = F;                      SPECIFIC = F;

MULTIPLE = Best;                   VOTING = Support;

LOG = F;                           NORMALIZATION = Firing;

CALIBRATION = F;

ROC = F;
```

Name the resulting confusion matrix *Johnson result*.

(g) Examine *Johnson result*. How well did the induced rules perform?

(h) Repeat the steps above, but use a different seed to the random number generator in step (a) than you did the first time.

Comment on any differences in performance estimates that occur. Briefly, discuss how you can obtain reliable performance estimates.

(i) Give at least two examples of situations where classification accuracy is not a sensible performance measure to use.

You are encouraged to experiment with ROSETTA and try different algorithms and parameters instead of the ones suggested above. You may also try to process some of the other data sets in the **Samples** folder.

NOTE: In Task 2 you are REQUIRED to document that you actually used Rosetta. Simply answering the questions is not enough.