Assignment 02 - jKlass

Intelligent Decision Support Systems

Beltran Fiz Roc Oliver Narcís Margall

January 20, 2010

Contents

1 Introduction

In very complex and unstructured domains, the Intelligent Decision Support Systems become very important tools for the expert, since allow to manage a quantity of information in a way that would be impossible to do manually. Inside this kind of systems, the classification tools are one of the most common, and, specifically, the clustering techniques. However, these techniques have problems when managing huge amount of variables and classes, because the interpretation of the generated classes becomes very complicate.

For this reason, in this asignment we want to generate an automatically conceptual interpretation of the classes generated by a clustering technique to help in the labor of the expert with a clearer vision of what is representing each class in order to understand quickly and easy what are the properties and characteristics of these data.

Main Objective:

In a more detailed view of point, our objective is to implement two alternative criteria proposed by the CCEC methodology. The way to do is following the implementation done by Esther Lozano in her Master Thesis ([25]).

2 Methodology CCEC

2.1 Introduction

In this chapter is introduced the methodology that represents the basis of this project, the *Methodology of conceptual characterization by embedded conditioning CCEC*, oriented to the automatic generation of conceptual descriptions of classifications that can support later decision-making and that was firstly formulated in [2]. Also, we have all the previous information that you need to make more undertandable this section in [2].

2.2 Methodology

The existence of a class hierarchy allows indexed address the problem of the interpretation of recursively descending into the dendrogram, this reduces each iteration to the interpretation of a binary partition and therefore the analysis to be presented below refers to the particular case of binary partitions simplifying the problem of finding distinctive classes. In a binary hierarchy necessarily $\mathcal{P}_{\xi+1}$ is *embedded* in $\mathcal{P}_{\xi}$, meaning that the two deriving new classes of one and only one of the two classes generated in the previous partition.

This property is exploited in the conceptual characterization methodology *CCEC* by considering the successive conditioning steps outlined following.

CCEC takes advantage of the existence of τ, and uses the property of any binary hierarchical structure that $\mathcal{P}_{\xi+1}$ has the same classes of $\mathcal{P}_{\xi}$ except one, which splits in two subclasses in $\mathcal{P}_{\xi+1}$. Binary hierarchical structure will be used by *CCEC* to discover particularities of the final classes step by step also in hierarchical way. The *CCEC* [18] allows generation of automatic interpretations of a given partition $\mathcal{P} \in \tau$.

1. Begin cutting the tree at highest level and we obtain $\xi = 2$ classes. And the partition formed by $\mathcal{P}_2 = \{C_t^{\xi}, C_q^{\xi}\}$ is created. Then, the root of the tree, C_0 has two childs: C_t^{ξ} and C_q^{ξ}.

2. Use *the boxplot based discretization* (*BbD*) presented in [10] and revised in [2], to find (total or partial) characteristic values for numerical variables [12].

3. Use *boxplot based induction rules (BbIR)*, to generate the knowledge Base and all the system or rules $\mathcal{R}(X_k, \mathcal{P}_{\xi})$ for both classes. Finally, built the global system of rules $\mathcal{R}(\mathcal{P}_{\xi})$.

4. Determine the concepts from $\mathcal{R}(X_k, \mathcal{P}_{\xi})$ to distinguish the two classes in $\mathcal{I}$ that is previous considerated. How to select the concept in each iteration is explained above.

5. Go down one level in the tree, taking advantage of the indexed hierarchy and see what class is opening. Then takes $\xi = \xi + 1$.

6. As said before, necessarily $\mathcal{P}_{\xi+1}$ is *embedded* in $\mathcal{P}_{\xi}$, meaning that the two new classes emerge from one and only one of two classes generated in the previous partition. So, in this way, we assume that the class C_t^{ξ} of $\mathcal{P}_{\xi}$ is the class that is subdivided (if is the other one, C_q^{ξ}, the process is the same), and then this class is divided into two new classes of $\mathcal{P}_{\xi+1}$:

$$C_i^{\xi+1} \text{ and } C_j^{\xi+1}$$

   Thus, we have that $C_q^{\xi+1} = C_q^{\xi}$ and $C_t^{\xi} = C_i^{\xi+1} \cup C_j^{\xi+1}$. Also it holds that $\mathcal{P}_{\xi+1} = \mathcal{P}_{\xi} \cup \{C_i^{\xi+1}, C_j^{\xi+1}\} \setminus \{C_t^{\xi}\}$.

   Then, we only need to separate $C_i^{\xi+1}$ from $C_j^{\xi+1}$. By repeating steps 2 to 4 we can distinguish $C_i^{\xi+1}$ from $C_j^{\xi+1}$ in $\mathcal{I} \setminus C_q^{\xi}$ (i.e. in C_t^{ξ}), thus we obtain the concepts $A_i^{*\xi+1}$ and $A_j^{*\xi+1}$ that allows to distinguish this two classes.

   Note: Identify between $\mathcal{P}_{\xi}$ and $\mathcal{P}_{\xi+1}$ which class is divided can be done from the cross table of two classes (contingency table).

7. Integrate the extracted knowledge of the iteration $\xi + 1$ with that of the iteration ξ, allows finally determine the concepts associated with the elements of $\mathcal{P}_{\xi+1}$. Since the common features of $C_i^{\xi+1}$ and $C_j^{\xi+1}$, which should allow to distinguish of $C_q^{\xi+1}$, are inherited from the class C_t^{ξ} and identified in the previous iteration, then, will be built the concepts

$$A_i^{\xi+1} = f_i(A_t^{\xi}, A_i^{*\xi+1})$$

$$A_j^{\xi+1} = f_j(A_t^{\xi}, A_j^{*\xi+1})$$

   that separe both classes with each other and also from $C_q^{\xi+1}$.

   Note: The possible forms of the function f_i and f_j depends on the hypotesis of Closed World Assumption. But, in fact, in our problem we use a unification form of $f = (f_i, f_j)$ and this form is the following:

$$f_i(A, A') = A \wedge A'$$

$$f_j(A, A') = A \wedge A'$$

   This is more explained in [2].

8. Go to step 5 until $\mathcal{P}_{\xi}$ coincides with the final partition that you want to interpret.

2.3 Knowledge Integration

In this section, we present 3 different ways of building the concepts $A_i^{*\xi+1}$ and $A_j^{*\xi+1}$ that is used in the methodology in step 4.

2.3.1 Best local concept and no Close-World Assumption

It consists of choosing among all the rules of $\mathcal{S}(\mathcal{P}^*_{\xi+1})$ that go to a same class, the one that has a higher relative coverage and not doing the CWA assumption. In this way, negation of the chosen concept is not used to define the concept of the complementary class, but for each class is used the certain concept with a higher relative coverage.

1. Restrict the search to the best rule of the knowledge base $\mathcal{S}(\mathcal{P}^*_{\xi+1})$ for the restricted partition $P^*_{\xi+1} = \{C_i^{\xi+1}, C_j^{\xi+1}\}$.

2. Consider for each class $C_i^{\xi+1}$ and $C_j^{\xi+1}$ of $\mathcal{P}^*_{\xi+1}$ a *subsystem* of rules that satisfies the rules pointed to a same class:

$$\mathcal{S}_{C_i}(\mathcal{P}^*_{\xi+1}) = \{r^k_{s,c} : C = C_i \ \wedge \ r^k_{s,c} \in \mathcal{S}(\mathcal{P}^*_{\xi+1})\} \text{ where } \mathcal{S}_{C_i}(\mathcal{P}^*_{\xi+1}) \subseteq \mathcal{S}(\mathcal{P}^*_{\xi+1})$$

$$\mathcal{S}_{C_j}(\mathcal{P}^*_{\xi+1}) = \{r^k_{s,c} : C = C_j \ \wedge \ r^k_{s,c} \in \mathcal{S}(\mathcal{P}^*_{\xi+1})\} \text{ where } \mathcal{S}_{C_j}(\mathcal{P}^*_{\xi+1}) \subseteq \mathcal{S}(\mathcal{P}^*_{\xi+1})$$

   Where $C_i \neq C_j$ and $C_i, C_j \in \mathcal{P}^*_{\xi+1}$.

3. Choose the concept linked to the rule of higher relative coverage of $\mathcal{S}_{C_i}(\mathcal{P}^*_{\xi+1})$ and the one of $\mathcal{S}_{C_j}(\mathcal{P}^*_{\xi+1})$.

   Determine k_i, s_i, c_i such as the concept "$X_{k_i} \in I_{s_i}^{k_i,\xi+1}$" has $p_{s_i c_i} = 1$ and the relative coverage of the rule $r^{k_i}_{s_i,c_i}$ is maximum at $\mathcal{S}_{C_i}(\mathcal{P}^*_{\xi+1})$. If there's more than one rule tied, consider all of them, let $r_i = \{r^{k_i,l}_{s_i,c_i}\}_{l\geqslant 1}$ that rules.

   Determine also, k_j, s_j, c_j such as the concept "$X_{k_j} \in I_{s_j}^{k_j,\xi+1}$" has $p_{s_j c_j} = 1$ and the relative coverage of the rule $r^{k_j}_{s_j,c_j}$ is maximum at $\mathcal{S}_{C_j}(\mathcal{P}^*_{\xi+1})$. If there's more than one rule tied, consider all of them, let $r_j = \{r^{k_j,l}_{s_j,c_j}\}_{l\geqslant 1}$ that rules.

   With this information, let

$$\mathcal{K}_i := \{k \mid \exists\ l \geq 1 \text{ such that } r^{k,l}_{s,c} \in r_i\}$$

$$\mathcal{K}_j := \{k \mid \exists\ l \geq 1 \text{ such that } r^{k,l}_{s,c} \in r_j\}$$

and let

$$\mathcal{K} := \mathcal{K}_i \cup \mathcal{K}_j$$

4. Determine concepts $A_i^{*\xi+1}$ and $A_j^{*\xi+1}$ induced for $C_i^{\xi+1}$ and $C_j^{\xi+1}$ respectively. Let $A^{\xi+1,k}$ = "$X_k \in I_s^{k,\xi+1}$", $k \in \mathcal{K}$. Then, all of them are considered in the construction of the concept and it is done in a different way depending on if it is a totally or partially characterizing variable.

   (a) If X_k, $\forall k \in \mathcal{K}$ are totally characterizing (it generates rules $p_{sc} = 1$ and $CovR = 100\%$). It is built in the following way:

$$A_i^{*\xi+1} = \bigwedge_{\forall\ k \in \mathcal{K}_i} A^{\xi+1,k} \tag{1}$$

$$A_j^{*\xi+1} = \bigwedge_{\forall\ k \in \mathcal{K}_j} A^{\xi+1,k} \tag{2}$$

   (b) If X_k, $\forall k \in \mathcal{K}$ are partially characterizing (it generates rules $p_{sc} = 1$ and $CovR < 100\%$). It is built in the following way:

$$A_i^{*\xi+1} = \bigvee_{\forall\ k \in \mathcal{K}_i} A^{\xi+1,k} \tag{3}$$

$$A_j^{*\xi+1} = \bigvee_{\forall\ k \in \mathcal{K}_j} A^{\xi+1,k} \tag{4}$$

   (c) If X_k, $k \in \mathcal{K}_i$ and X_k, $k \in \mathcal{K}_j$ have different behaviour we need to distinguish two cases depending on if it is a totally or partially characterizing each variable.

   - If X_k, $\forall k \in \mathcal{K}_i$ are partially characterizing and X_k, $\forall k \in \mathcal{K}_j$ are totally characterizing. It is built in the following way:

$$A_i^{*\xi+1} = \bigvee_{\forall\ k \in \mathcal{K}_i} A^{\xi+1,k} \tag{5}$$

$$A_j^{*\xi+1} = \bigwedge_{\forall\ k \in \mathcal{K}_j} A^{\xi+1,k} \tag{6}$$

- If X_k, $\forall k \in \mathcal{K}_i$ are totally characterizing and X_k, $\forall k \in \mathcal{K}_j$ are partially characterizing. It is built in the following way:

$$A_i^{*\xi+1} = \bigwedge_{\forall\ k \in \mathcal{K}_i} A^{\xi+1,k} \tag{7}$$

$$A_j^{*\xi+1} = \bigvee_{\forall\ k \in \mathcal{K}_j} A^{\xi+1,k} \tag{8}$$

2.3.2 Best local concept and Close-World Assumption

It consists of choosing among all the rules of $\mathcal{S}(\mathcal{P}_{\xi+1}^{*})$ that go to a same class the one that has a higher relative coverage and doing a strong *Close-World* assumption *CWA*.

In this way, for each class, negation of the chosen concept is used to define the other class in logical disjunction ($\vee$) with the certain concept obtained by the maximum relative coverage.

1. Restrict the search to the best rules of the knowledge base $\mathcal{S}(\mathcal{P}_{\xi+1}^{*})$ for the restricted partition $P_{\xi+1}^{*} = \{C_i^{\xi+1}, C_j^{\xi+1}\}$.

2. Consider for each class $C_i^{\xi+1}$ and $C_j^{\xi+1}$ of $\mathcal{P}_{\xi+1}^{*}$ a subsystem of rules that satisfies the rules pointed to a same class:

$$\mathcal{S}_{C_i}(\mathcal{P}_{\xi+1}^{*}) \subseteq \mathcal{S}(\mathcal{P}_{\xi+1}^{*}) \text{ and } \mathcal{S}_{C_j}(\mathcal{P}_{\xi+1}^{*}) \subseteq \mathcal{S}(\mathcal{P}_{\xi+1}^{*})$$

   Where $C_i \neq C_j$ and $C_i, C_j \in \mathcal{P}_{\xi+1}^{*}$.

3. Choose the concept linked to the rule with a higher relative coverage of $\mathcal{S}_{C_i}(\mathcal{P}_{\xi+1}^{*})$ and the one with $\mathcal{S}_{C_j}(\mathcal{P}_{\xi+1}^{*})$.

   Determine k_i, s_i, c_i such as the concept "$X_{k_i} \in I_{s_i}^{k_i,\xi+1}$" has $p_{s_i c_i} = 1$ and the relative coverage of the rule $r_{s_i,c_i}^{k_i}$ is maximum at $\mathcal{S}_{C_i}(\mathcal{P}_{\xi+1}^{*})$. If there's more than one rule tied, consider all of them, let $r_i = \{r_{s_i,c_i}^{k_i,l}\}_{l \geqslant 1}$ that rules.

   Determine also, k_j, s_j, c_j such as the concept "$X_{k_j} \in I_{s_j}^{k_j,\xi+1}$" has $p_{s_j c_j} = 1$ and the relative coverage of the rule $r_{s_j,c_j}^{k_j}$ is maximum at $\mathcal{S}_{C_j}(\mathcal{P}_{\xi+1}^{*})$. If there's more than one rule tied, consider all of them, let $r_j = \{r_{s_j,c_j}^{k_j,l}\}_{l \geqslant 1}$ that rules.

   With this information, let

$$\mathcal{K}_i := \{k \ \mid \ \exists\ l \geq 1 \text{ such that } r_{s,c}^{k,l} \in r_i\}$$

$$\mathcal{K}_j := \{k \ \mid \ \exists\ l \geq 1 \text{ such that } r_{s,c}^{k,l} \in r_j\}$$

and

$$\mathcal{K} := \mathcal{K}_i \cup \mathcal{K}_j$$

4. Do a strong hypothesis of *Close-World Assumption* to describe the similar class depending on the complementary concept.

5. Determine concepts $A_i^{*\xi+1}$ and $A_j^{*\xi+1}$ induced for $C_i^{\xi+1}$ and $C_j^{\xi+1}$ respectively. Let $A^{\xi+1,k}$ = "$X_k \in I_s^{k,\xi+1}$", $k \in \mathcal{K}$. Then, all of them are considered in the construction of the concept and it is done in a different way depending on if it is a totally or partially characterizing variable.

   First of all, let,

$$B_i^{\wedge} := \bigwedge_{\forall\ k \in \mathcal{K}_i} A^{\xi+1,k} \text{ and } B_i^{\vee} := \bigvee_{\forall\ k \in \mathcal{K}_i} A^{\xi+1,k}$$
$$B_j^{\wedge} := \bigwedge_{\forall\ k \in \mathcal{K}_j} A_j^{\xi+1,k} \text{ and } B_j^{\vee} := \bigvee_{\forall\ k \in \mathcal{K}_j} A^{\xi+1,k}$$

   (a) If X_k, $\forall k \in \mathcal{K}$ are totally characterizing (it generates rules $p_{sc} = 1$ and $CovR = 100\%$). It is built in the following way:

$$A_i^{*\xi+1} = B_i^{\wedge} \vee \neg B_j^{\wedge} \tag{9}$$
$$A_j^{*\xi+1} = B_j^{\wedge} \vee \neg B_i^{\wedge} \tag{10}$$

   (b) If X_k, $\forall k \in \mathcal{K}$ are partially characterizing (it generates rules $p_{sc} = 1$ and $CovR < 100\%$). It is built in the following way:

$$A_i^{*\xi+1} = B_i^{\vee} \vee \neg B_j^{\vee} \tag{11}$$
$$A_j^{*\xi+1} = B_j^{\vee} \vee \neg B_i^{\vee} \tag{12}$$

   (c) If X_k, $k \in \mathcal{K}_i$ and X_k, $k \in \mathcal{K}_j$ have different behaviour we need to distinguish two cases depending on if it is a totally or partially characterizing each variable.

   - If X_k, $\forall k \in \mathcal{K}_i$ are partially characterizing and X_k, $\forall k \in \mathcal{K}_j$ are totally characterizing. It is built in the following way:

$$A_i^{*\xi+1} = B_i^{\vee} \vee \neg B_j^{\wedge} \tag{13}$$
$$A_j^{*\xi+1} = B_j^{\wedge} \vee \neg B_i^{\vee} \tag{14}$$

- If X_k, $\forall k \in \mathcal{K}_i$ are totally characterizing and X_k, $\forall k \in \mathcal{K}_j$ are partially characterizing. It is built in the following way:

$$A_i^{*\xi+1} = B_i^{\wedge} \vee \neg B_j^{\vee} \tag{15}$$

$$A_j^{*\xi+1} = B_j^{\vee} \vee \neg B_i^{\wedge} \tag{16}$$

2.3.3 Best Global concept and Close-World Assumption

It consists of choosing among all the rules of $\mathcal{S}(\mathcal{P}^*_{\xi+1})$ the rules that have a higher relative coverage. Take this rules doing strong *Close-World Assumption* and use the negation for conceptualizate the complementari class.

1. Restrict the search to the best rules of the knowledge base $\mathcal{S}(\mathcal{P}^*_{\xi+1})$ for the restricted partition $P^*_{\xi+1} = \{C_i^{\xi+1}, C_j^{\xi+1}\}$.

2. Choose the concept linked to the rule with a higher relative coverage of $\mathcal{S}(\mathcal{P}^*_{\xi+1})$. The role of the rule determine which class is characterized.

   Determine c, k, s such as the concept "$X_k \in I_s^{k,\xi+1}$" has $p_{sc} = 1$, $C \in \mathcal{P}^*_{\xi+1}$, and the relative coverage of the rule $r^k_{s,c}$ is maximum in the knowledge base $\mathcal{S}_C(\mathcal{P}^*_{\xi+1})$.

   If there's more than one rules tied, take all of them. Let the set of all this rules, $r = \{r^{k,l}_{s,c}\}_{l \geqslant 1}$. Therefore, we can define $\mathcal{K}_i$ and $\mathcal{K}_j$:

   $$\mathcal{K}_i := \{k \mid \exists\, l \geqslant 1 \text{ such that the consequent of } r^{k,l}_{s,c} \in r \text{ is } C_i^{\xi+1}\}$$

   $$\mathcal{K}_j := \{k \mid \exists\, l \geqslant 1 \text{ such that the consequent of } r^{k,l}_{s,c} \in r \text{ is } C_j^{\xi+1}\}$$

   It's possible that $\mathcal{K}_i = \emptyset$ or $\mathcal{K}_j = \emptyset$.

3. Choose the consequent of the set of rules $r = \{r^{k,l}_{s,c}\}_{l \geqslant 1}$.

   (a) If $\#\mathcal{K}_i \geqslant \#\mathcal{K}_j$, then take the rules $r^{k,l}_{s,c}$ such that $l \in \mathcal{K}_i$ and

   $$\mathcal{K} := \mathcal{K}_i$$

   (b) If $\#\mathcal{K}_i < \#\mathcal{K}_j$, then take the rules $r^{k,l}_{s,c}$ such that $l \in \mathcal{K}_j$ and

   $$\mathcal{K} := \mathcal{K}_j$$

4. Do a strong hypothesis of *Close-World Assumption* to describe the similar class depending on the complementary concept.

5. Determine concepts $A_i^{*\xi+1}$ and $A_j^{*\xi+1}$ induced for $C_i^{\xi+1}$ and $C_j^{\xi+1}$ respectively. Let $A^{\xi+1,k}$ = "$X_k \in I_s^{k,\xi+1}$" where $k \in \mathcal{K}$, then all of them are considered in the construction of the concept and it is done in a different way depending on if it is a totally or partially characterizing variable.

   (a) If X_k, $\forall k \in \mathcal{K}$ are totally characterizing (it generates rules $p_{sc} = 1$ and $CovR = 100\%$). It is built in the following way:

   - If 3a is fulfilled, then

$$A_i^{*\xi+1} = \bigwedge_{k \in \mathcal{K}} A^{\xi+1,k} \tag{17}$$

$$A_j^{*\xi+1} = \neg A_i^{*\xi+1} \tag{18}$$

   - If 3b is fulfilled, then

$$A_j^{*\xi+1} = \bigwedge_{k \in \mathcal{K}} A^{\xi+1,k} \tag{19}$$

$$A_i^{*\xi+1} = \neg A_j^{*\xi+1} \tag{20}$$

   (b) If X_k, $\forall k \in \mathcal{K}$ are partially characterizing (it generates rules $p_{sc} = 1$ and $CovR < 100\%$). It is built in the following way:

   - If 3a is fulfilled, then

$$A_i^{*\xi+1} = \bigvee_{k \in \mathcal{K}} A^{\xi+1,k} \tag{21}$$

$$A_j^{*\xi+1} = \neg A_i^{*\xi+1} \tag{22}$$

   - If 3b is fulfilled, then

$$A_j^{*\xi+1} = \bigvee_{k \in \mathcal{K}} A^{\xi+1,k} \tag{23}$$

$$A_i^{*\xi+1} = \neg A_j^{*\xi+1} \tag{24}$$

3 KLASS

3.1 The framework project description

This work is within the frame of SISPD course, which has the objective of developing hybrid support methodologies to the Knowledge Discovery and Data Mining in unstructured domains [15] to solve decision support problems, mainly in medical and environmental domains. This project was started in 1995 with the idea of combining Statistical techniques with the ones from Artificial Intelligence to overcome the limitations of classical techniques I the different steps of the analysis of these kinds of domains [11], [10].

The first proposal constitutes the Karina Gibert's degree thesis [6] and PhD thesis [8] that resulted in the formaulation of the $m$ethodology of classification based on rules and a first version of the computer system that implements it, called *KLASS* [8] and that has been used in different real applications [13, 14, 17, 16, 20, 19].

All the methodologies developed within the framework project are integrated in a master tool, that nowadays is $java.KLASS$ and that joins tolls of very different nature, offering the necessary interface to communicate the different modules and transfer the necessary information in each moment of the analysis.

3.2 KLASS' Chronology

- Feb. 1991 **K**LASS v0. Karina Gibert's dissertation. "KLASS. Study of an assistance system for statistical treatment of large databases". It classifies data matrixes of heterogeneous data with mixed distance. [7]
- Nov. 1994 **K**LASS v1. Karina Gibert's thesis. "Use of symbolic information in the automatization of the statistic treatment of unstructured domains". It is an extension of **K**LASS v0. It includes classification based in rules. [9]
- Jul. 1996 **K**LASS v1.1. PFC Xavier Castillejo. It incorporates to an independent windows interface with a system that enables the use of KLASS from a SUN and from a PC to users that don't know Lisp and UNIX. Let's call **x**cn.KLASS to the Lisp kernel of this new version and **x**cn.i to the C interface. [3]
- Oct. 1997 **j**j.KLASS. PFC Juan Jos? Marquez and Juan Carlos Mart?n. It incorporates to the **K**LASS.v1 version new options for the treatment of missing values, the possibility of working with weighted objects and implements a non-parametric test for the comparison of classifications [26].
- Set. 1999 **K**LASS v1.2. PFC Xavier Tubau (β version). It incorporates to the **x**cn.KLASS version the comparison of classifications module of **j**j.KLASS,the Ralambondrainy's mixed metric [28] [29]and prepares the formulation of three

more for their later implementation. Let's call **x**t.KLASS to the Lips kernel of this new version and X to the associated C interface. [31]

- 1999-2000 **K**LASS+ v1. PFC S?lvia Bayona. Definitive fusion of versions **x**t.KLASS and **jj**.KLASS. It incorporates a new module of data descriptive analysis, as well as the resultant classes, reorienting KLASS to a more general proposal and less specialized. Let's call **s**bh.KLASS to the Lisp kernel of this new version and **s**bh.i to the associated C interface. [1]

- 2000-2002 **K**LASS+ v2. PFC Josep Oliveras. It incorporates to **sbh.KLASS** the pending mixed metrics (Gower [23] [21] [22], Gowda-Diday [5] [4] and Ichino-Yaguchi [24]). Let's call **joc.KLASS** to this new version. [27]

- 2000-2003 **j**r.KLASS+. Jorge Rodas's thesis. Integrates **KLASS+ v.2** and **Columbus**, that is later presented. [30]

- 2000-2003 Anna Salvador and Fernando V?zquez research. Developement of **CIADEC**, that is later presented. [32]

- 2002-2003 **J**ava-KLASS v0. PFC M? del Mar Colillas. Java version of the descriptive analysis module and integration with **CIADEC** and **Columbus**.
  - 2003-2005 **J**ava-KLASS v0.22. In collaboration with Mar Colillas. Extension of the descriptive analysis module and introduction of tools for data management *(definition of orders in the informs, possibility of simultaneously different matrixes of objects in the system, change of active matrix).*
  - 2005-2006 **J**ava-KLASS v1.0. In collaboration with Mar Colillas. It includes the reading and visualization of isolated dendograms, as well as the generation of partitions from them.

- 2006-2007 **J**ava-KLASS v2.0. PFC Jose Ignacio Mateos. Extension of **J**ava-KLASS with a module of calculation of distances for different types of data matrixes, including the ones that combine qualitative and quantitative information, treatment of missing values and creation of submatrixes.

- 2006-2007 **J**ava-KLASS v3.0. PFC Roberto Tuda. It includes a module of automatic classification by hierarchical methods, using all the distances implemented in v2.0 and an option for studying aggregations of objects step by step. The option of selecting the default work directory is created. The option of adding and recording weighted objects is included.

- 2006-2007 **J**ava-KLASS v4.0. PFC Laia Riera Guerra. Introduction, management and evaluation of Knowledge Bases. Extension of **J**ava-KLASS with a module of transformation of variables that allows discretizations, recodifications and arithmetic calculations with numerical variables. Finally, this version includes the definition of submatrixes through logical filters over the objects, the edition of metainformation of the matrix variables, elimination of variables and importation of files in .dat standard format.

- 2007 **J**ava-KLASS v5.0. PFC Andreu Raya. It includes the embedded classification, classification based on rules and functionalities for division of the database and for management of classification trees *(or dendograms)* associated to the different data matrixes.

- 2007 **J**ava-KLASS v6.0. Alejandro Garc?a's thesis. Exogenous classification based on rules. Internationalization and location to three languages (Catalan, English, Spanish). Matrix merger.

- 2008 **J**ava-KLASS v6.4. master thesis of Alfons Bosch Sansa, Patricia Garc?a Gim?nez, Ismael Sayyad Hernando. Boxplot-based discretization, Boxplot-based Induccion rules.

- 2008. Alejandra Perez's thesis. Characterization by embedded conditionings, methodology that induces automatically associated concepts to the discovered classes.

- 2008. Gustavo Rodriguez's thesis. Classification based on rules by states that allows analysis of dinamical systems.

- 2009. Esther Lozano thesis. Inducted Concepts From Embbeded Classes For Automatic Interpretation In Hierarchical Clustering.

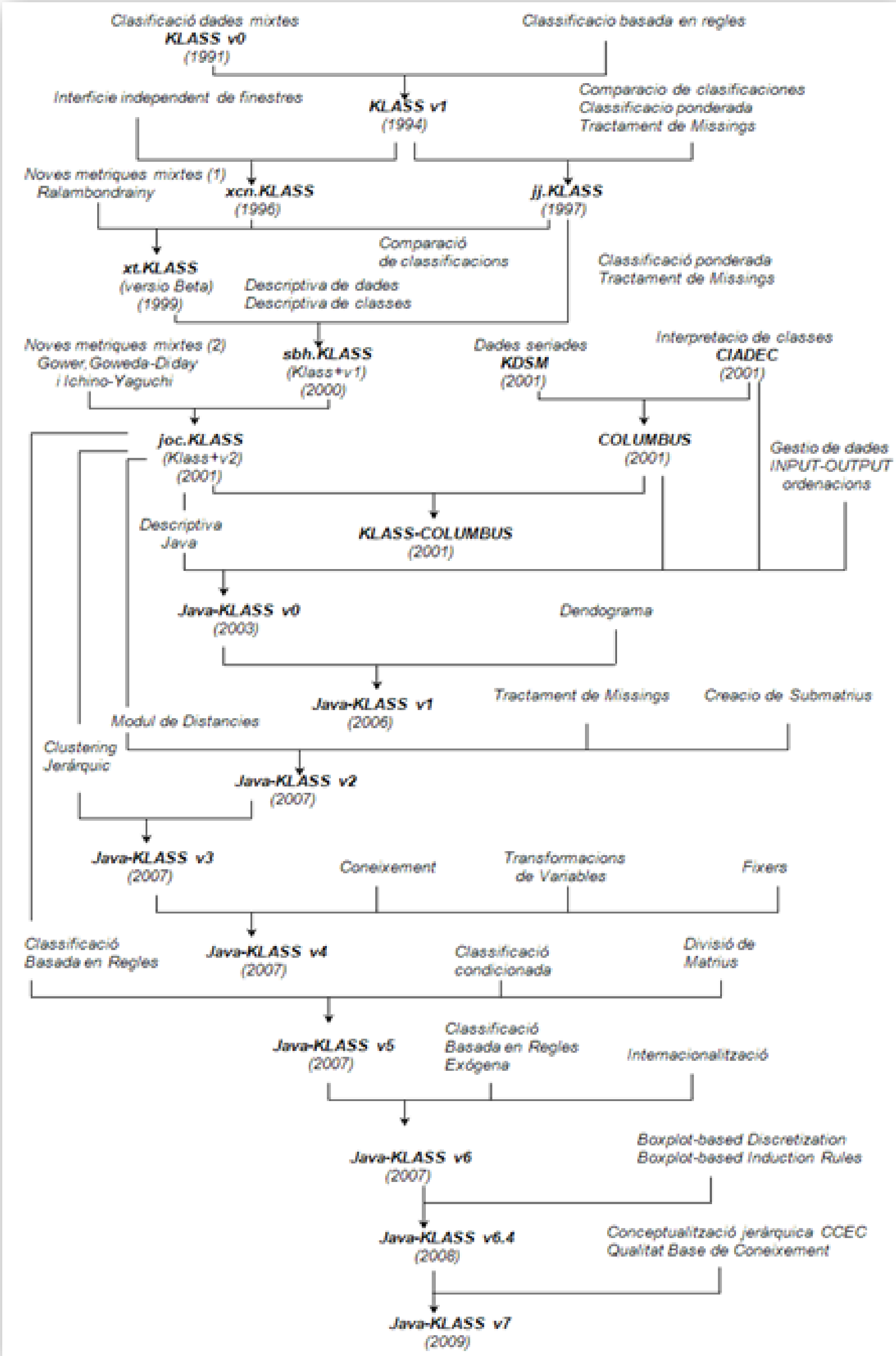

Figure 3.1: KLASS' Chronology

3.3 KLASS' Structure

In order to understand the implementation details of the next section, is necessary to introduce how KLASS is structured from this point of view.

KLASS is a Java application that has been implemented following a structure of layers to separate the graphic interface part from the main methods and data objects. KLASS consists of the following packages:

- **jklass.ui:** This package contains the classes related to the graphical user interface. KLASS has a windows interface, and each of these windows is implemented in a specific class. If a new window wants to be added, a new class must be created for the description of the window, and it must be declared in the corresponding menu. These classes can only call to methods from the kernel part to execute the actions, so the data is protected of this *external* layer.
- **jklass.nucli:** This package is the kernel part of KLASS. It contains all the methods that actually execute the actions asked for the user.
- **jklass.util:** This package contains classes for the management of the system options, configuration parameters and calls to the operative system.

4 Implementation

This section explains the software elements modified in order to accomplish the assignment requirements. Mainly:

- Implementation of *Best Local and Closed-World Assumption* method
- Implementaton of *Best Global and Closed-World Assumption* method

Also, it shows the extra amount of work done; highlighting:

- Fix of *Best Local and No Closed-World Assumption Simple* method
- Fix of *Best Global and No Closed-World Assumption* method

General considerations:

- The removal of class methods is done by commenting them, never deleting.
- Every code change or addition is identified by the doclet `@author Grup SISPD QT 2009-2010`.

4.1 Class Diagram

The main two classes involved in the implementation of *Best Local and Closed-World Assumption* and *Best Global and Closed-World Assumption* methods are the following:

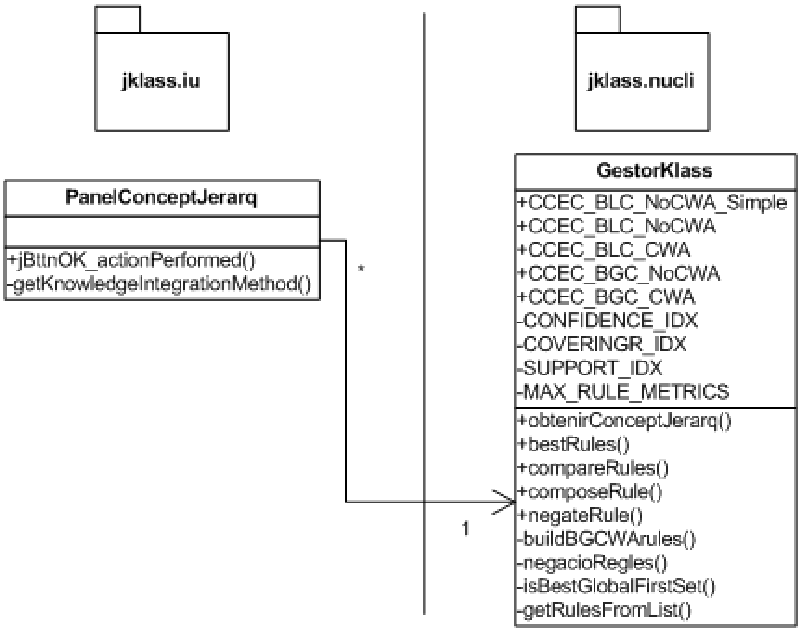

Figure 4.1: Class Diagram

4.2 Methods Explanation

In the `jklass.iu.PanelConceptJerarq.java` class, in order to add the two new knowledge integration methods, it has been modified the following methods:

jBttnOK_actionPerformed: Method that starts the interpretation process.

getKnowledgeIntegrationMethod: Identifies the knowledge integration method selected using the constants defined in `jklass.nucli.GestorKlass`.

An example of the PanelConceptJerarq panel:

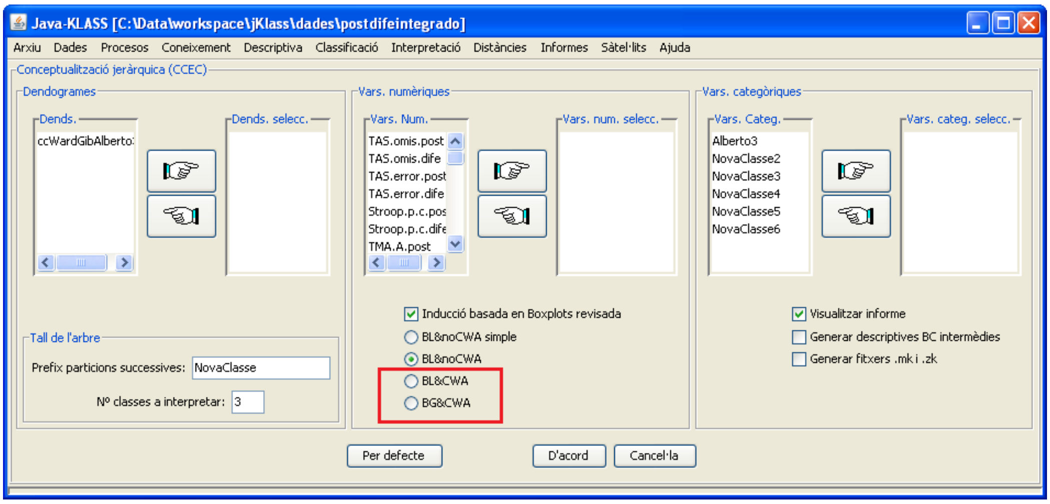

Figure 4.2: CCEC frame

In the `jklass.nucli.GestorKlass.java` class, have been implemented the following methods:

obtenirConceptJerarq: Method that realizes the hierarchical conceptualization. This consists of doing cuts in the dendogram, integrating the resulting classification in the data matrix and selecting the best rules from the whole set of inducted rules. Every new iteration makes the cut with a higher number of classes until reach the number given by the user.

bestRules: Method that, according to the knowledge integration criteria chosen by the user, returns the two best rules sets of the given knowledge base (one for each new generated classes at the current step).

buildBGCWArules: Method that creates the two resulting rules of the Best Global and Closed-World Assumtion method.

isBestGlobalFirstSet: Method that indicates whether the best global set of rules is the first one.

negacioRegles: Method that negates the premises of each of a given rules set.

compareRules: Method that, given two rules belonging to the same class, applies the selected knowledge integration criteria and returns the best one.

composeRule: Method that composes a set of rules with the specified logic operation.

getRulesFromList: Method that transforms a set of rule names to a set of rule objects.

negateRule: Method that, for a given rule, generates a new one with the same conclusion and the negation of the premises.

4.2.1 Deliverables

The only modified deliverable that comes with the *jKlass* software is the *User Manual*. The modification is done in order to update the CCEC interpretation chapter to the new available knowledge integration methods.

4.3 Extra: improvements and issue

4.3.1 Summarize

To fulfill the assignment presented in this document, we make an important effort, not only to understand the existing code but to enhance it. These lead to correcting existing code bugs and improving the software quality properties. In a summary, we have done the following extra work:

- **Issues:** it has been detected and solved 4 issues.
- **Improvements:** it has been done 6 necessary improvements.

4.3.2 Detail

For each of the issues or improvements done in this assignment, we provide rellevant information to describe it. That is:

- *Name:* an identifier of the issue/improvement.
- *Type:* indicates that is whether an issue or an improvement.
- *Description:* an explanation about the element.
- *Localization:* the class and/or the method that is modified in order to solve it.

Issues

Name	ID01 - Method CCEC BLnCWA Simple
Type	Issue
Description	The method CCEC Best Local simple No Closed-World Assumption did not work. So we modified it in order to correctly operate.
Localization	`jklass.nucli.GestorKlass.bestRules`

Name	ID02 - Operators at BLnCWA
Type	Issue
Description	The method CCEC Best Local No Closed-World Assumption did the composition between elements in a wrong way (it did not deal with the whole premises of a rule, just with each subpart).
Localization	`jklass.nucli.GestorKlass.bestRules`

Name	ID03 - Handling of a non expected state at CCEC BLnCWA
Type	Issue
Description	The main documentation of the Best Local and No-Closed World Assumption knowledge integration method does not deals with a situation explained at **??**. For dealing with this case we modified the BLnCWA method accordingly.
Localization	`jklass.nucli.GestorKlass.bestRules`

Name	ID04 - Use of wrong default path in the hierarchical interpretation process
Type	Issue
Description	During the hierarchical interpretation process, intermediate files were created in a default directory (dades/resultats).
Localization	`jKlass.iu.GestorKlass.obtenirConceptJerarq`

Improvements

Name	ID05 - WinVista and Mac OS
Type	Improvement
Description	The program was not able to be executed in Windows Vista and Mac OS operating systems.
Localization	`jklass.util.SO`

Name	ID06 - Refactoring SO.java
Type	Improvement
Description	Code modification in order to improve the readability, ease the comprehension and the maintainability (i.e: adding constants as the definition of the operating system).
Localization	`jklass.util.SO`

Name	ID07 - CCEC Methods refactoring
Type	Improvement
Description	Code modification in order to improve the readability, ease the comprehension and the maintainability (i.e: adding constants to indexing arrays, modifying switchs commands, . . .).
Localization	`jklass.nucli.GestorKlass.bestRules`

Name	ID08 - Latex operators display modifications
Type	Improvement
Description	The Latex document obtained at the end of the interpretation process didn't use the logic notation for conjunction and disjunction operators. The prior version use & and || for conjunction a disjunction operators. The current one displays them as $\wedge$ and $\vee$, correspondingly.
Localization	`jKlass.nucli.AND.escriureNormalLatex` and `jKlass.nucli.OR.escriureNormalLatex`

Name	ID09 - Latex relationship symbols modifications
Type	Improvement
Description	The Latex obtained document at the end of the interpretation process didn't use the correct relationship symbols. The prior version use >= and <=. The current one displays them as $\geq$ and $\leq$, correspondingly.
Localization	`jKlass.nucli.EsMajorIgual.escriureNormalLatex` and `jKlass.nucli.EsMenorIgual.escriureNormalLatex`

Name	ID10 - Modification of generated class rules names
Type	Improvement
Description	In order to making easy to check whether the generated rule names for a given class is correct or not, we named it as an understandable composition of intermediate knowledge base and rules involved. Taking the example of generated describing rule to 34 class in the Best Local and No-Closed World Assumption integration method used in the [25] case study. This rules is composed by the conjunction of rules $r14$ and $r94$ of the intermediate knowledge base BC0 and also by the conjunction of rules $r24$ and $r58$ of $BC1$. The previous given rule name was: $r1.BC0.r24 - r58$, only taking into account the last rules. So, we set the name to a more correctly describing string such as: $r1.BC0.BC0.r14 - r94BC1.r24 - r58$.
Localization	`jKlass.nucli.GestorKlass`

5 Validation within a Case Study

We have manually calculated the methods implemented within a case study to validate our methods used. once the manual calculation is done we can then compare it to the printed output of KLASS and verify they match. We will be using the same case study Esther [25] presented.

5.1 Case Study

The Case Study used in order to show KLASS in action is a sample of patients with traumatic brain injuries (TBI patients) who followed a neurorehabilitation treatment between April and September of 2006 in the Institut Guttmann Hospital de Neurorehabilitacio.

Some facts about the sample data used in our case study :

- Mean age at time of injury 28,36 years (SD=12 years).
- Mean time from injury 3,02 months (SD=1,38 months).
- Initial clinical severity:
  - severe (GCS between 3 and 8): 34 patients (71,43
  - moderate (GCS between 9 and 12): 13 patients (28,57
- Assessed at admission and discharge:
  - 38 variables (Language, Attention, Memory and Learning, Executive Functions)
- Measures of particular patient improvement:
  - Differences between pre- and post-treatment scores

5.2 Validation

In this section I will be displaying the steps taken to validate the method, along with the results. I will simply be describing at each iteration the variables selected by our method, and how they are merged together. In the slides which will be presented at a later date, we will display a visual guide (slide by slide) that will allow newcomers to understand the knowledge integration methods in more detail.

There are two knowledge integration methods that require validation, they are :

- Best Local and Closed-World Assumption method.
- Best Global and Closed-World Assumption method.

The selection of the rules from the Knowledge Bases is actually similar for both methods, as we are interested in selecting the one with maximal trust and relative cover. In our testing we merely dealt with the first five iterations (ie: five cuts of the original class). Here are the rules chosen at each step :

5.2.1 First Step:

Selected from Knowledge Base 0 (named in our code BC0).

Qualitative Table				
Rule	**Consequent**	**Trust**	**Support**	**Relative cover**
r14	C0	1	0.3404	1
r15	C1	1	0.6383	0.9677
r94	C0	1	0.3404	1
r95	C1	1	0.6383	0.9677

So the selected rules are :

Result of first step			
Class	**Rules**	**Trust**	**Relative cover**
C0	r14 , r94	1	1
C1	r15 , r95	1	0.9677

And the initial class split is :[$\rightarrow$ C0, C1]

5.2.2 Second Step:

Selected from Knowledge Base 1 (named in our code BC1).

Qualitative Table				
Rule	**Consequent**	**Trust**	**Support**	**Relative cover**
r24	34	1	0.0625	1
r57	graves13	1	0.9375	1
r58	34	1	0.0625	1

So the selected rules are :

Result of second step			
Class	**Rules**	**Trust**	**Relative cover**
34	r24 , r58	1	1
Graves13	r57	1	1

And the resulting class split is :[C0 → 34, Graves13]

5.2.3 Third Step:

Selected from Knowledge Base 2 (named in our code BC2).

Qualitative Table				
Rule	**Consequent**	**Trust**	**Support**	**Relative cover**
r16	graves12	1	0.3333	1
r19	graves12	1	0.3333	1
r22	graves12	1	0.3333	1
r25	graves12	1	0.3333	1
r87	graves12	1	0.3333	1
r88	graves9	1	0.6667	1
r89	graves9	1	0.6667	1

So the selected rules are :

Result of third step			
Class	**Rules**	**Trust**	**Relative cover**
Graves9	r88, r89	1	1
Graves12	r16, r19, r22, r25, r87	1	1

And the resulting class split is :[Graves13 → Graves9, Graves12]

5.2.4 Fourth Step:

Selected from Knowledge Base 2 (named in our code BC3).

Qualitative Table				
Rule	**Consequent**	**Trust**	**Support**	**Relative cover**
r44	valorables14	1	0.5161	1
r46	Inhibidos13	1	0.4516	0.9333
r70	Inhibidos13	1	0.4516	0.9333

So the selected rules are :

Result of fourth step			
Class	**Rules**	**Trust**	**Relative cover**
Inhibidos13	r46, r70	1	0.9333
valorables14	r44	1	1

And the resulting class split is :[C1 $\rightarrow$ inhibidos13, valorables14]

5.2.5 Fifth Step:

Selected from Knowledge Base 2 (named in our code BC3).

Qualitative Table				
Rule	**Consequent**	**Trust**	**Support**	**Relative cover**
r0	Inhibidos11	1	0.6	1
r3	Inhibidos11	1	0.6	1
r5	Inhibidos12	1	0.3333	0.8333
r9	Inhibidos12	1	0.3333	0.8333
r13	Inhibidos12	1	0.3333	0.8333

So the selected rules are :

Result of fifth step			
Class	**Rules**	**Trust**	**Relative cover**
Inhibidos11	r0, r3	1	0.9333
Inhibidos12	r5, r9, r13	1	1

And the resulting class split is :[inhibidos 13 $\rightarrow$ inhibidos11, inhibidos12]

5.3 Best Local and Closed-World Assumption Validation

Now that all the rules at every step have been selected we can build the final solution.Here are the final results :

- 34 :
$$(r14 \wedge r94) \vee \neg(r15 \vee r95) \wedge (r24 \wedge r58) \vee \neg(r57)$$
- graves 9 :
$$(r14 \wedge r94) \vee \neg(r15 \vee r95)\wedge$$
$$(r57) \vee \neg(r24 \wedge r58)\wedge$$
$$(r88 \wedge r89) \vee \neg(r16 \wedge r19 \wedge r22 \wedge r25 \wedge r87)$$
- graves 12 :
$$(r14 \wedge r94) \vee \neg(r15 \vee r95)\wedge$$
$$(r57) \vee \neg(r24 \wedge r58)\wedge$$
$$(r16 \wedge r19 \wedge r22 \wedge r25 \wedge r87) \vee \neg(r88 \wedge r89)$$
- inhibidos 11 :
$$(r15 \wedge r95) \vee \neg(r14 \wedge r94)\wedge$$
$$(r46 \wedge r70) \vee \neg(r44)\wedge$$
$$(r0 \wedge r3) \vee \neg(r2 \wedge r5 \wedge r9 \wedge r13)$$
- inhibidos 12 :
$$(r15 \wedge r95) \vee \neg(r14 \wedge r94)\wedge$$
$$(r46 \wedge r70) \vee \neg(r44)\wedge$$
$$(r2 \wedge r5 \wedge r9 \wedge r13) \vee \neg(r0 \wedge r3)$$
- valorables 14 :
$$(r15 \wedge r95) \vee \neg(r14 \wedge r94)\wedge$$
$$(r44) \vee \neg(r46 \wedge r70)$$

5.4 Best Global and Closed-World Assumption Validation

And here are the final results for this method :

- 34 :
$$(r14 \wedge r94) \wedge (r24 \wedge r58)$$

- graves 9 :
$$(r14 \wedge r94) \wedge \neg(r24 \wedge r58) \wedge \neg(r16 \wedge r19 \wedge r22 \wedge r25 \wedge r87)$$

- graves 12 :
$$(r14 \wedge r94) \wedge \neg(r24 \wedge r58) \wedge (r16 \wedge r19 \wedge r22 \wedge r25 \wedge r87)$$

- inhibidos 11 :
$$\neg(r14 \wedge r94) \wedge (r44) \wedge (r0 \wedge r3)$$

- inhibidos 12 :
$$\neg(r14 \wedge r94) \wedge (r44) \wedge \neg(r0 \wedge r3)$$

- valorables 14 :
$$\neg(r14 \wedge r94) \wedge (r44)$$

6 Conclusions

Our goals were to implement two new methods within the KLASS software, and ensure they were correctly implemented. This was fully achieved and was done in a consistent manner without changing any of the core features of the program.

During the process of implementation, we also upgraded and fixed some parts of the code that were working incorrectly, adding an extra layer of features to our project. The specific changes were previously mentioned and described.

We can safely say that the project was a success ,even though the program has a steep learning curve,we had limited time to dedicate to this project and finally the regrettable departure of one of our colleges which left us with a gap to fill in terms of planning and coding.

7 Future Work

We believe KLASS as a piece of software, still requires a good polish in the code in terms of consistency. Much of the comments for the code is misplaced and/or offers little help.

The use of 3 different languages (English , Spanish and Catalan) within the code causes a certain confusion in terms of the naming of variables, data structures and inner methods. This, from a software development point of view, is definitely something that requires revision.

Finally, there are still several methods within the CCEC methodology that need to be implemented, namely :

- Best Local concept and partial closed world assumption
- Best Local-Global and closed world assumption

References

[1] S. Bayona. Descriptiva de dades y de classes. PFC Facultat d'Informàtica, UPC, 2000.

[2] Alejandra Pérez Bonilla. Metodología de caracterización conceptual por condicionamientos sucesivos. una aplicación a sistemas medioambientales. Master's thesis, Statistics and Operative Research Department, UPC, April 2009.

[3] X. Castillejo. Un entorn de treball per a Klass. PFC Facultat d'Informàtica, UPC, 1996.

[4] E. Diday and K.C. Gowda. Symbolic clustering using a new dissimilarity measure. 24(6):567–578, 1991.

[5] E. Diday and K.C. Gowda. Symbolic clustering using a new similaritiy measure. In *IEEE Trans. on systems, man.;and cib.*, volume 22, pages 368–378, 1992.

[6] K. Gibert. Klass. Estudi d'un sistema d'ajuda al tractament estadístic de grans bases de dades. Master's thesis, UPC, 1991.

[7] K. Gibert. Klass. estudi d'un sistema d'ajuda al tractament estadístic de grans bases de dades. Master's thesis, UPC, 1991.

[8] K. Gibert. *L'us de la Informació Simbòlica en l'Automatització del Tractament Estadístic de Dominis Poc Estructurats*. In the statistics and operations research phd. thesis., Universitat Politecnica de Catalunya, Barcelona, Spain, 1994.

[9] K. Gibert. *Ph. D. Thesis*. Eio dep., UPC, Barcelona, Spain, 1994.

[10] K. Gibert. *Tendencia de la Minería de Datos en España*, chapter :Técnicas híbridas de Inteligencia Artificial y Estadística para el descubrimiento de conocimiento y la minería de datos, pages 119–130. Thompson Ed., 2004.

[11] K. Gibert and T. Aluja. A computational technique for comparing classifications and its relationship with knowledge discovery. In *International Seminar on New Techniques and Technologies for Statistics*, pages 193–198, Italy, 1998.

[12] K. Gibert, T. Aluja, and U. Cortés. Knowledge Discovery with Clustering Based on Rules. Interpreting results. In *Principles of Data Mining and Knowledge Discovery*, volume 1510 of *LNAI*, pages 83–92. Springer, 1998.

[13] K. Gibert and U Cortés. Combining a knowledge based system with a clustering method for an inductive construction of models. In *Proc. 4th Int Work. on AI and Stats.*, 1993. Florida, USA.

[14] K. Gibert and U Cortés. On the uses of the expert knowledge for automatic biasing of a clustering method. In *ITI 93. Proceedings of the International Conference on Information Technology Interfaces*, pages 219–224, Croatia, 1993. issn 1330-1012.

[15] K. Gibert and U. Cortés. *Combining a knowledge-based system and a clustering method for a construction of models in ill-structured domains*, volume 89 of *LNS*, pages 351–360. Springer-Verlag., 1994.

[16] K. Gibert and U. Cortés. Clustering based on rules and knowledge discovery in ill-structured domains. *Computación y Sistemas.*, 1(4):213–227, 1998.

[17] K. Gibert, M. Hernández, and U. Cortés. Classification based on rules: an application to Astronomy. In U. Tokio. Japón, editor, *5th. IFCS*, pages 69–72, 1996.

[18] K. Gibert and A. Pérez-Bonilla. Ventajas de la estructura jerárquica del clustering en la interpretación automática de clasificaciones. In *III Spanish Workshop on Data Mining and Learning*, pages 67–76, Granada, 2005.

[19] K. Gibert and I. Roda. Identifying characteristic situations in wastewater treatment plants. In *Workshop in Binding Environmental Sciences and Artificial Intelligence*, volume 1, pages 1–9, 2000.

[20] K. Gibert and Z. Sonicki. Classification based on rules and thyroids dysfunctions. *Applied Stochastic Models in Business and Industry*, 15(4):319–324, 1999.

[21] J.C. Gower. Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53:315–328, 1966.

[22] J.C. Gower. A comparison of some methods of cluster analysis. *Biometrics*, 23(4):623–37, 1967.

[23] J.C. Gower. A General coefficient of similarity and some of its properties. *Biometrics*, 27:857–874, 1971.

[24] M. Ichino and H. Yaguchi. Generalized Minkowski metrics for mixed features. *Trans. IEICE Japó*, J72-A(2):398–405, 1989. (en japonès).

[25] Esther Lozano. Ainducted concepts from embbeded classes for automatic interpretation in hierarchical clustering. Master's thesis, Master Thesis. Universitat Politècnica de Catalunya, 2009.

[26] J. Márquez and J.C. Martín. La clasificación automática en las ciencias de la salud. PFC, 1997. Facultat de Matemàtiques i Estadística, UPC.

[27] Oliveras Castellà, Josep. Dades heterogènies amb classificació automàtica. Implementació i comparativa de mètriques mixtes. PFC.

[28] H.A. Ralambondrainy. A conceptual version of K-means algorithm. *Pattern Recognition Letters.*, 16:1147–1157, 1995.

[29] Henri Ralambondrainy. *A clustering method for nominal data and mixture of numerical and nominal data. Clasification and Related Methods of Data Analysis.* H.H.Bock, Elsevier Science Publishers, B.V. (North-Holland), 1988.

[30] Jorge Enrique Rodas Osollo. *Knowledge Discovery in repeated and very short serial measures with a blocking factor.* Programa de doctorado: Inteligencia artificial, Universitat Politecnica de Catalunya, 2003.

[31] X. Tubau. Sobre el comportament de les mètriques mixtes en algorismes de Clustering. PFC.

[32] F. Vázquez and K Gibert. Robustness of class prediction depending on reference partition in ill-structured domains. In *XVIII Iberoamerican Conference on Artificial Intelligence, Workshop de Minería de Datos y Aprendizaje (IBERAMIA2002)*, pages 13–22, Sevilla, 2002.