UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Um Estudo de Caso Utilizando o Módulo de Combinação e Explicação do Rule System

Flávia Cristina Bernardini
Maria Carolina Monard

Nº 160

RELATÓRIOS TÉCNICOS

São Carlos - SP

Instituto de Ciências Matemáticas e de Computação

ISSN - 0103-2569

Um Estudo de Caso Utilizando o Módulo de Combinação e Explicação do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$

Flávia Cristina Bernardini
Maria Carolina Monard

Nº 160

RELATÓRIOS TÉCNICOS DO ICMC

São Carlos

Abril/2002

Um Estudo de Caso Utilizando o Módulo de Combinação e Explicação do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$*

Flávia Cristina Bernardini

Maria Carolina Monard

Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
Departamento de Ciências de Computação e Estatística
Laboratório de Inteligência Computacional
Caixa Postal 668, 13560-970 - São Carlos. SP, Brasil
e-mail: {fbernard, mcmonard}@icmc.sc.usp.br

Resumo

Encontra-se em desenvolvimento no LABIC — Laboratório de Inteligência Computacional — um projeto de grande porte denominado DISCOVER, que tem por objetivo fornecer um ambiente integrado para apoiar as etapas do processo de descoberta de conhecimento. No intuito de testar algumas idéias que poderão futuramente ser implementadas no ambiente DISCOVER, foi implementado um sistema computacional denominado $\mathcal{R}_{ule}\mathcal{S}_{ystem}$ na linguagem de programação lógica Prolog. O $\mathcal{R}_{ule}\mathcal{S}_{ystem}$ utiliza conjuntos de classificadores simbólicos induzidos por diferentes algoritmos de Aprendizado de Máquina e tem como objetivos: avaliar as regras de conhecimento que constituem esses classificadores, avaliar diversas formas de combinar esses classificadores bem como explicar a classificação de novos exemplos por esse conjunto, ou *ensemble*, de classificadores. Dentre os módulos do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$, está o Módulo de Combinação e Explicação — MCE —, responsável pela tarefa de explicação de *ensembles* e combinação de classificadores. Um dos procedimentos principais do MCE é o `rco/4`, que implementa o algoritmo $\mathcal{RCO}$. O $\mathcal{RCO}$ é um algoritmo de seleção de regras que utiliza como entrada um conjunto de hipóteses (classificadores), constituídas por regras e um subconjunto de dados do domínio. O objetivo deste trabalho é analisar as hipóteses construídas utilizando seis critérios diferentes do algoritmo $\mathcal{RCO}$, com as hipóteses induzidas pelo algoritmo de aprendizado de máquina $\mathcal{CN}2$. Os experimentos foram realizados utilizando um conjunto de dados do mundo real, relacionados ao processamento de sêmen diagnóstico.

Palavras-Chave: Aprendizado de Máquina, Combinação de Hipóteses, Análise de Regras.

Abril 2002

*Trabalho realizado com auxílio da FAPESP

Sumário

1 Introdução **1**

2 Combinação de Classificadores Simbólicos **2**

3 Estudo de Caso — Processamento de Sêmen Diagnóstico **3**

3.1 Descrição do Conjunto de Dados 7

3.2 Descrição do Experimento 7

3.3 Análise dos Resultados 11

4 Conclusão **13**

A Conjunto de Dados Utilizado pelo $\mathcal{RCO}$ **16**

B Classificadores Utilizados pelo $\mathcal{RCO}$ **19**

B.1 Classificador $\mathbf{h}_1$ 19

B.2 Classificador $\mathbf{h}_2$ 20

B.3 Classificador $\mathbf{h}_3$ 22

C Classificadores Construídos pelo $\mathcal{RCO}$ **23**

C.1 Classificador $\mathbf{h}_{\texttt{accR}}$ 24

C.2 Classificador $\mathbf{h}_{\texttt{covR}}$ 25

C.3 Classificador $\mathbf{h}_{\texttt{satR}}$ 26

C.4 Classificador $\mathbf{h}_{\texttt{sensR}}$ 27

C.5 Classificador $\mathbf{h}_{\texttt{specR}}$ 28

C.6 Classificador $\mathbf{h}_{\texttt{supR}}$ 29

Lista de Figuras

1 Ilustração do Algoritmo $\mathcal{RCO}$ 5

2 Ilustração do Experimento Realizado. 9

Lista de Tabelas

1 Sumário das Características do Conjunto de Dados *Proc-a-gmg-d* 7

2 Descrição dos Atributos do Conjunto de Dados *Proc-a-gmg-d* 8

3 Resultados Obtidos Utilizando o $\mathcal{CN}2$ 10

4 Medidas de Avaliação de Regras Utilizadas neste Trabalho 10

5 Resultados Obtidos Utilizando o $\mathcal{RCO}$ 11

6 Número de Exemplos Classificados por Classe pela Regra *Default* 12

7 Número de Regras Selecionadas por Classe 12

8 Regras Selecionadas pelo $\mathcal{RCO}$ 14

9 Motivos do Especialista para Considerar que uma Regra Não Faz Sentido 14

Lista de Algoritmos

1 $\mathcal{RCO}$ 4

[1] Este documento foi elaborado com o LaTeX, sua bibliografia é mantida com o auxílio da ferramenta BibView (Prati et al., 1999) e gerada automaticamente pelo BibTeX

1 Introdução

Encontra-se em desenvolvimento no LABIC[1] — Laboratório de Inteligência Computacional — um projeto de grande porte denominado DISCOVER, inicialmente proposto por Baranauskas e Batista (Baranauskas and Batista, 2000). O projeto DISCOVER tem como objetivo fornecer um ambiente integrado para apoiar as etapas do processo de descoberta de conhecimento, oferecendo funcionalidades voltadas para Aprendizado de Máquina (AM) (Batista, 1997; Caulkins, 2000; Milaré, 2000; Martins, 2001; Pila, 2000), Mineração de Dados[2](MD) (Batista, 2000; Félix, 1998; Horst, 1999; Lee, 2000; Nagai, 2000; Pugliesi, 2001; Baranauskas, 2001) e Mineração de Textos[3](MT) (Imamura, 2001).

As funcionalidades voltadas para AM consideram, entre outros, um formato padrão para as regras induzidas por algoritmos de AM simbólico, denominado $\mathcal{PBM}$ (Prati et al., 2001b; Prati et al., 2001a) bem como um formato padrão para os exemplos utilizados (Batista, 2001).

No intuito de testar algumas idéias que poderão futuramente ser implementadas no ambiente DISCOVER, foi implementado um sistema computacional utilizando a linguagem Prolog, denominado $\mathcal{R}_{ule}\mathcal{S}_{ystem}$ (Bernardini and Monard, 2002; Gomes and Monard, 2002b), com o objetivo específico de fazer a explicação de *ensembles* e combinação de classificadores e a análise de regras. O $\mathcal{R}_{ule}\mathcal{S}_{ystem}$ possui três módulos:

1. Módulo de Combinação e Explicação (MCE);
2. Módulo de Análise de Regras (MAR); e
3. Módulo Auxiliar (MA).

O MCE é o responsável pela tarefa de explicação de *ensembles* e combinação de classificadores. Nele, estão implementados cinco procedimentos principais relacionados com:

1. *Classificação de Exemplos* — `classifyExample/4`;
2. *Determinação da Matriz de Confusão* — `confusionMatrix/2`;
3. *Explicação de* Ensembles — `classifyExampleByEnsemble/6`;
4. *Combinação de Classificadores Simbólicos* — `rco/4` e `kFoldCrossValidation/4`.

Neste trabalho serão utilizadas as facilidades do MCE para combinar classificadores simbólicos através de vários critérios utilizados pelo algoritmo $\mathcal{RCO}$ — *Rule Combination* — implementados no procedimento `rco/4`.

O objetivo deste trabalho é analisar as hipóteses construídas utilizando seis critérios diferentes, implementados pelo procedimento `rco/4`, com as hipóteses induzidas pelo algoritmo de AM $\mathcal{CN}2$, utilizando um conjunto de dados do mundo real.

[1] http://labic.icmc.sc.usp.br
[2] Data Mining
[3] Text Mining

Este trabalho está organizado da seguinte forma: na Seção 2, é feita uma descrição sobre como é implementada a combinação de classificadores simbólicos no Módulo de Combinação e Explicação do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$; na Seção 3, é descrito o estudo de caso, os experimentos realizados, os resultados obtidos e é feita uma análise desses resultados; finalmente na Seção 4, é feita uma conclusão sobre o trabalho realizado.

2 Combinação de Classificadores Simbólicos

Os dois procedimentos relacionados com combinação de classificadores do MCE do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$ são:

1. `rco/4`; e
2. `kFoldCrossValidation/4`.

O procedimento `rco/4` implementa o algoritmo $\mathcal{RCO}$ — *Rule Combination* —, descrito resumidamente a seguir. Uma descrição mais detalhada pode ser encontrada em (Bernardini and Monard, 2002).

Algoritmo $\mathcal{RCO}$

Dado um conjunto de hipóteses $\mathcal{H} = \{\mathbf{h}_1, ..., \mathbf{h}_L\}$, um conjunto de exemplos de treinamento $S = \{(\mathbf{x}_i, y_i), i = 1, ..., N\}$ e uma medida de avaliação de regra $Measure$, o procedimento `rco/4` constrói uma nova hipótese como uma combinação das regras $R \in \mathbf{h}_l, l = 1, ..., L$, segundo a medida $Measure$. Os quatro argumentos do procedimento `rco/4` são:

\+ <**arg-1**> lista de identificadores de hipóteses;

\+ <**arg-2**> uma medida de avaliação de regra;

\- <**arg-3**> identificador da hipótese criada;

\- <**arg-4**> lista de exemplos não cobertos pela nova hipótese.

O Algoritmo 1 mostra como essa nova hipótese é construída. O algoritmo faz inicialmente uma cópia das regras de todas as hipóteses e cria o cabeçalho da hipótese a ser construída. Depois do cabeçalho criado, o algoritmo segue os seguintes passos:

1. Verifica se o conjunto de exemplos S está vazio. Se sim, a iteração pára;
2. Calcula $Measure$ para todas as regras $R \in \mathbf{h}_l, l = 1, ..., L$;
3. Retira a regra $Rule$, dentre as L hipóteses, com maior valor $Measure$[4];

[4]no caso de empate, ou seja, se houver mais de uma regra com o melhor valor $Measure$, o algoritmo considera todas as regras como sendo um conjunto de regras com o mesmo "melhor" valor de $Measure$ e seleciona aleatoriamente uma regra desse conjunto como sendo a regra $Rule$.

4. Retira da lista de exemplos S todos os exemplos que são cobertos pela regra *Rule*. Se nenhum exemplo for coberto pela regra *Rule*[5], a iteração pára[6];

5. Adiciona a regra *Rule* à nova hipótese.

Ao sair da iteração, segundo qualquer dos dois critérios de parada, o algoritmo constrói a regra *default*. A regra *default* possui *Corpo* vazio e sua *Cabeça* é definida pela classe com maior distribuição dos exemplos[7].

Deve ser observado que, além da hipótese construída, o usuário pode visualizar as regras do conjunto de hipóteses que não foram utilizadas pelo algoritmo para construir a nova hipótese. Após, o conjunto de hipóteses original é recuperado.

O algoritmo $\mathcal{RCO}$ pode ser visualizado na Figura 1, onde se pode observar que os dados de entrada para o algoritmo $\mathcal{RCO}$ consistem de L classificadores e um conjunto de exemplos de treinamento. Com esses dados de entrada, uma nova hipótese é construída e, com um conjunto de teste pode-se medir a precisão e/ou erro dessa hipótese.

3 Estudo de Caso — Processamento de Sêmen Diagnóstico

Na medicina, freqüentemente existem exames e processos importantes, mas apresentam um alto custo para sua realização. Em muitos casos, é desejável que os mesmos resultados, ou alguns resultados parciais, possam ser obtidos por métodos menos custosos. Um exemplo desse caso é o processamento de sêmen.

O estudo de caso sobre processamento de sêmen diagnóstico teve seu início em (Lee, 2000), onde é descrita a importância desse processamento no tratamento para a reprodução assistida. A explicação a seguir está baseada nos trabalhos de Huei Diana Lee (Lee and Monard, 2000b; Lee and Monard, 2000a; Esteves and F.C., 1998; Esteves et al., 2000; Esteves et al., 2001) e foi desenvolvida conjuntamente com Alan Keller Gomes (Gomes and Monard, 2002a). Esse exame permite:

1. Quantificar com precisão a qualidade do sêmen (processamento de sêmen diagnóstico);

2. Recuperar a maior quantidade possível de espermatozóides para a utilização na reprodução assistida (processamento de sêmen terapêutico).

[5]Se a regra *Rule* não cobre nenhum exemplo, isso significa que *Measure* atingiu seu mínimo (ou máximo) absoluto para todas as regras $R \in \mathbf{h}_l, l = 1, ..., L$ nesse passo da iteração. Por exemplo, no caso da medida ser cobertura (*Measure* =`covR`), o mínimo valor absoluto é 0(zero), e no caso da medida ser erro (*Measure* =`errR`), o máximo valor absoluto é 1.

[6]A iteração pára se somente havia uma regra para ser escolhida no passo 3. Em caso de empate, outras regras vão sendo selecionadas aleatoriamente do conjunto de regras até que o uma regra cubra algum exemplo ou que o conjunto de regras esteja vazio.

[7]Uma regra pertencente a uma hipótese geralmente é do tipo *if-then*, a qual pode ser reescrita como sendo *Corpo*→*Cabeça* ou $Body \rightarrow Head$.

Algoritmo 1 $\mathcal{RCO}$

Pré-condições: $\mathcal{H} = \mathbf{h}_l, l = 1, ..., L$: conjunto de hipóteses;
$S = \{(\mathbf{x}_i, y_i), i = 1, ..., N\}$: conjunto de exemplos;
$Measure$: medida de regra.

1: **procedure** rco($\mathcal{H}$,S,$Measure$)
2: Fazer cópia das regras das L hipóteses em $Rules$;
3: S':=S; {Cópia dos exemplos originais}
4: Criar o cabeçalho da hipótese $\mathbf{h}_{new}$;
5: **loop**
6: **if** vazio(S) or vazio($Rules$) **then**
7: **exit loop**; {Não há mais exemplos de treinamento ou não há mais regras em $Rules$ para processar}
8: **end if**
9: calculateMeasuresOverAllSetOfRules($Measure$);
10: $Rule$:=max($Rules, Measure$); {Encontrar a regra $Rule$ em $Rules$ com maior medida $Measure$}
11: S_{cov}:=cov(S,$Rule$); {Criar o conjunto de exemplos cobertos pela regra $Rule$}
12: **if** vazio(S_{cov}) **then**
13: **exit loop** {Nenhum exemplo foi coberto pela regra $Rule$}
14: **end if**
15: S:=S - S_{cov}; {Retirar de S os exemplos contidos em S_{cov}}
16: $\mathbf{h}_{new} := \mathbf{h}_{new} \bigcup \{Rule\}$; {Adicionar $Rule$ à hipótese $\mathbf{h}_{new}$}
17: $Rules := Rules - \{Rule\}$; {Retirar $Rule$ de $Rules$}
18: **end loop**
19: $defaultRule$:=calculate_default_rule(S'); {Calcular a regra *default*}
20: $\mathbf{h}_{new} := \mathbf{h}_{new} \bigcup \{defaultRule\}$; {adicioná-la a $\mathbf{h}_{new}$}
21: **return** $\mathbf{h}_{new}$

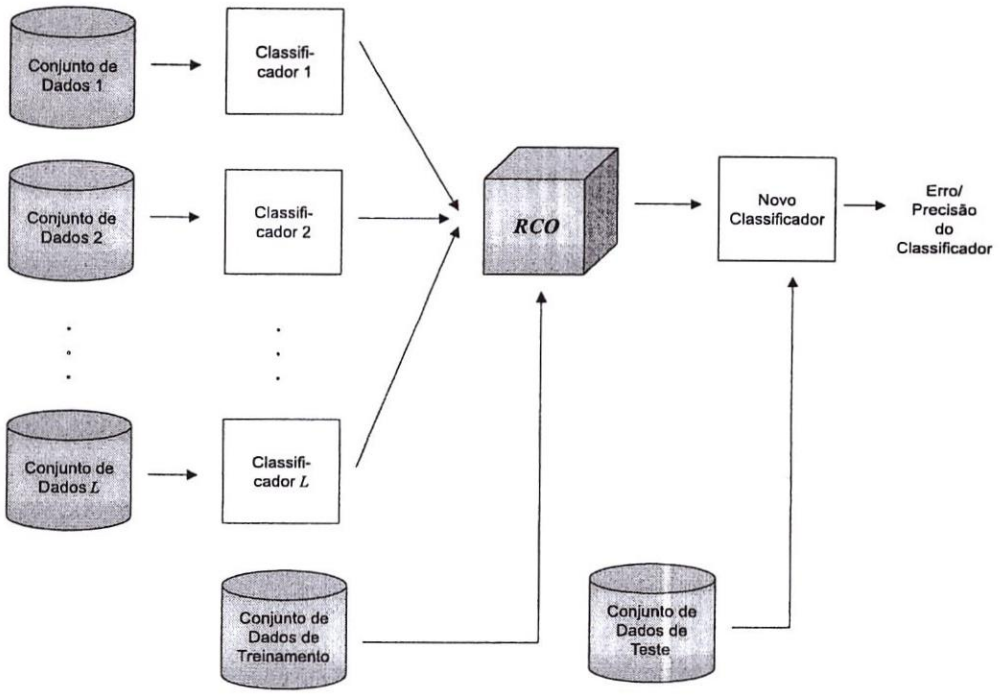

Figura 1: Ilustração do Algoritmo $\mathcal{RCO}$

A quantidade de espermatozóides recuperados pelo processamento de sêmen influencia na escolha da técnica que será utilizada no tratamento. São utilizadas três técnicas no tratamento para a reprodução assistida:

1. Inseminação Intra Uterina — IUI;
2. Fertilização In Vitro — FIV e
3. Injeção Intracitoplasmática do Espermatozóide no Oócito — ICSI.

Deve ser observado que o processamento de sêmen é bastante custoso. A realização do processamento de sêmen pode elevar o custo do exame em aproximadamente 80% do valor de um espermograma. Essa elevação de custo se deve principalmente a três fatores: necessidade de equipamentos especiais, mão de obra qualificada e tempo gasto para a realização do exame. Assim, um dos interesses do estudo desse tema é tentar predizer qual será a quantidade de espermatozóides recuperados pelo processamento de sêmen antes mesmo da realização desse exame, a partir de exames menos custosos, como o espermograma. Dessa forma, dependendo da qualidade da predição, o especialista poderia decidir por uma técnica sem a necessidade da realização do processamento de sêmen. Um outro interesse para o estudo desse tema é a extração e avaliação do conhecimento adquirido.

O processamento de sêmen, através do fornecimento de melhores condições, permite que o maior número de espermatozóides móveis (motilidade) seja recuperado. Os espermatozóides são classificados em graus A, B, C e D dependendo de sua motilidade:

- grau A: espermatozóides que apresentam o maior grau de motilidade;
- grau B: espermatozóides que apresentam um menor grau de motilidade;
- grau C: espermatozóides que se movem em círculos;
- grau D: espermatozóides que são imóveis.

Os graus de motilidade são atributos medidos tanto no espermograma (exame de baixo custo) quanto no processamento de sêmen (exame de alto custo). Os atributos medidos durante o processamento de sêmen são utilizados para a determinação das classes e os atributos medidos através do espermograma são fornecidos como atributos ao algoritmo de indução. Em outras palavras, os valores dos atributos de cada exemplo (paciente) são medidos utilizando um exame pouco custoso (espermograma, neste caso) enquanto que as classes desses exemplos são determinadas através de um outro exame mais custoso (processamento de sêmen, neste trabalho). A idéia é tentar verificar se é possível descobrir um relacionamento entre eles tal que, pelo menos para novos pacientes, a necessidade de realizar o exame mais custoso seja minimizada. Como já mencionado, este é um procedimento freqüentemente utilizado na área de medicina a fim de tentar diminuir, mas com segurança, o custo de um tratamento.

O especialista sugeriu duas possíveis opções para os valores considerados na classificação dos exemplos:

- A: considerar apenas a percentagem de espermatozóides classificados como de motilidade grau A no processamento de sêmen ou
- AB: considerar a percentagem de espermatozóides classificados como de motilidade grau A e grau B no processamento de sêmen.

Assim, foram definidas três classes, baseadas nas técnicas que são utilizadas no tratamento para a reprodução assistida:

- 1: $x < 1$ — IUI;
- 2: $1 \geq x < 5$ — FIV e
- 3: $x \geq 5$ — ICSI.

onde x representa milhões de espermatozóides por mililitro (ml).

Os casos para a composição desses conjunto de dados foram extraídos de casos reais do Centro de Referência em Infertilidade Masculina — Androfert[8] —, em Campinas, SP, fornecidos pelo Dr. Sandro C. Esteves. Em (Lee, 2000), foram calculados para cada caso os valores A e AB, gerando dois conjuntos de dados, *Proc-a* e *Proc-ab*, respectivamente, cada um com 231 exemplos. Diversos experimentos foram conduzidos considerando-se esses dois conjuntos separadamente. Com o decorrer do trabalho, foi necessário criar novos atributos como combinação de atributos originais, criando, assim, diferentes conjuntos de dados (Lee and Monard, 2000b). Um desses conjuntos

[8] `http://www.androfert.com.br/`

de dados que mostrou um bom resultado com relação à precisão dos classificadores induzidos foi o denominado *Proc-a-gmg-d*, o qual considera somente os exemplos em que o processamento de sêmen é diagnóstico.

Posteriormente, ao conjunto de exemplos *Proc-a* do trabalho inicial foram acrescentados novos casos. Tomando desse conjunto de exemplos maior somente os exemplos em que o processamento de sêmen é diagnóstico, foi obtido um segundo conjunto de exemplos *Proc-a-gmg-d*, semelhante ao construído com os exemplos *Proc-a* do trabalho inicial, mas incluindo os novos casos, o qual é utilizado neste trabalho. Deve ser ressaltado que todo o trabalho de coleta, preparação e limpeza dos dados foi realizada por Huei Diana Lee.

3.1 Descrição do Conjunto de Dados

A Tabela 1 sumariza as características do conjunto de dados utilizado neste estudo. Ela mostra o número de exemplos (#Exemplos), número e porcentagem de exemplos com valores duplicados (que aparecem mais de uma vez) ou conflitantes (que possuem o mesmo atributo-valor mas têm diferentes classificações), número de atributos (# Atributos) contínuos e nominais, distribuição de classes, o erro majoritário e se o conjunto de dados tem ao menos um valor desconhecido[9]. Já a Tabela 2 descreve os atributos, exceto o atributo classe. do conjunto de dados utilizado — *Proc-a-gmg-d*.

# Exemplos	# Duplicados ou conflitantes (%)	# Atributos (cont.,nom.)	Classe	Classe %	Erro Majoritário	Valores Desconhecidos
240	1 (0.42%)	21 (10,11)	1	42.92%	57.08%	S
			2	22.08%	no valor 1	
			3	35.00%		

Tabela 1: Sumário das Características do Conjunto de Dados *Proc-a-gmg-d*

3.2 Descrição do Experimento

O experimento conduzido tem por objetivo gerar um novo classificador simbólico utilizando vários classificadores simbólicos previamente induzidos por algum algoritmo de aprendizado de máquina, através de diferentes critérios para eleger as regras que farão parte desse novo classificador. Logo após, verificar o erro cometido pelo classificador assim construído com os erros cometidos pelos classificadores individuais, bem como verificar o conjunto de regras que constituem esse novo classificador.

A construção do novo classificador foi realizada utilizando as facilidades implementadas no Módulo de Combinação e Explicação do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$ (Bernardini and Monard, 2002). Uma ilustração do experimento realizado pode ser visualizada na Figura 2.

Os classificadores simbólicos iniciais foram induzidos utilizando o algoritmo de AM $\mathcal{CN}2$. Inicialmente, foi induzido um único classificador com o $\mathcal{CN}2$, utilizando todo o conjunto de exemplos S disponível, ou seja, os 240 exemplos, e foi medido o erro aparente desse classificador. Após, foi

[9]Estas informações foram obtidas utilizando a ferramenta *info* da $\mathcal{MLC}$++.

Número da Feature	Nome da Feature	Descrição	#Valores Distintos		
			possíveis	atuais	tipo
#0	idade	Idade do paciente	—	26	contínuo
#1	hora	Período da coleta da amostra de sêmen 1: manhã; 2: tarde	2	3	discreto
#2	processamento	Processamento do sêmen realizado após esta quantidade de minutos após a coleta	—	12	contínuo
#3	tempo-abs	Tempo de abstinência	—	12	contínuo
#4	volume	Volume de sêmen coletado	—	55	contínuo
#5	cor	Cor do sêmen coletado 1: branco-opalescente (normal); 2: amarelo-opalescente; 3: translúcido	3	4	discreto
#6	odor	Odor do sêmen coletado 1: característico; 2: forte; 3: urina	3	3	discreto
#7	pH	pH do sêmen coletado	—	13	contínuo
#8	viscosidade	Viscosidade do sêmen coletado 1: normal; 2: aumentada	2	2	discreto
#9	liquefacao	Liquefação do sêmen coletado 1: completa; 2: imconpleta	2	2	discreto
#10	concentracao	Concentração de espermatozóides por ml coletado	—	198	contínuo
#11	concentracao-total	Concentração total de espermatozóides	—	219	contínuo
const #12	motilidade	% de espermatozóides móveis	—	74	contínuo
#13	class-A	Classificação da motilidade - Grau A	—	66	contínuo
#14	class-B	Grau B	—	53	contínuo
#15	class-C	Grau C	—	44	contínuo
#16	class-D	Grau D	—	76	contínuo
#17	vitalidade	% de espermatozóides vivos	—	59	contínuo
#18	num-leu	Número de leucócitos	—	66	contínuo
#19	Kruger	Morfologia Estrita de Kruger - % de formas normais	—	28	contínuo
#20	HP	Teste Hipo-osmótico - % de inchados	—	54	contínuo

Tabela 2: Descrição dos Atributos do Conjunto de Dados *Proc-a-gmg-d*

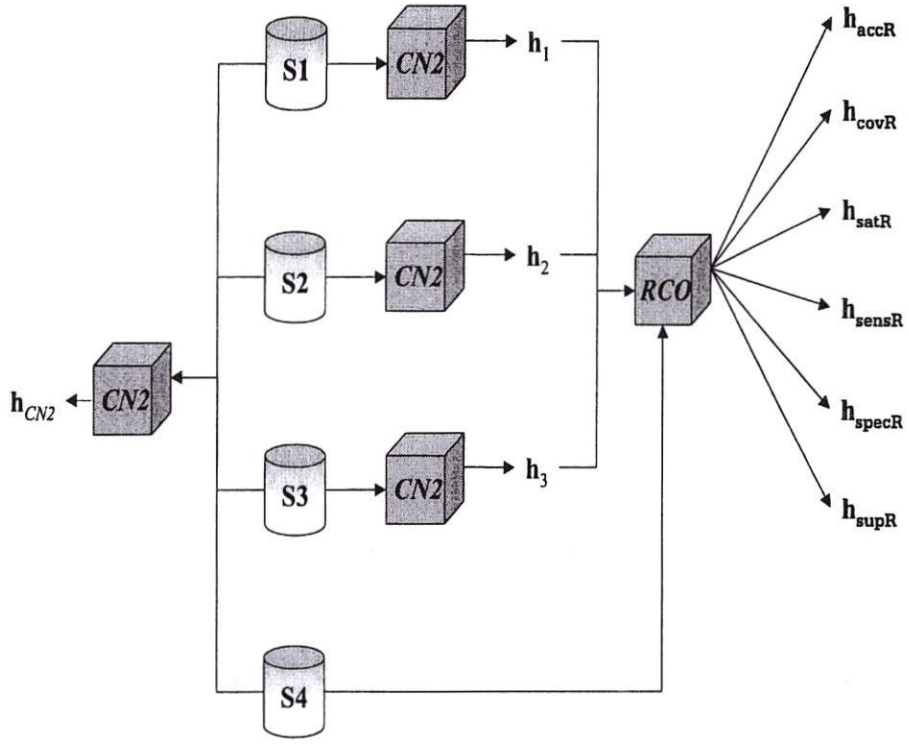

Figura 2: Ilustração do Experimento Realizado.

utilizada a técnica de 10-fold cross-validation para estimar o erro verdadeiro e o desvio padrão desse classificador, bem como o erro de classificação em cada classe.

Num passo seguinte, o conjunto total de exemplos S, com 240 exemplos, foi dividido aleatoriamente em quatro subconjuntos disjuntos S_1, S_2, S_3 e S_4 de exemplos com o mesmo número de exemplos em cada um deles — 60 exemplos.

Utilizando cada um dos subconjuntos de exemplos S_1, S_2 e S_3. foram induzidos três classificadores $\mathbf{h}_1$, $\mathbf{h}_2$ e $\mathbf{h}_3$, respectivamente, com o indutor $\mathcal{CN}2$. Para esses três classificadores foi também calculado o erro aparente, a estimativa do erro verdadeiro e desvio padrão, bem como a estimativa do erro e desvio padrão em cada classe através de 10-fold cross validation. Os resultados obtidos são encontrados na Tabela 3.

Conj. de Dados → Classificador	Erro Aparente	Erro Majoritário	Erro nas Classes ($e \pm dp$ (%))	Erro do Classificador ($e \pm dp$ (%))	# Regras Induzidas
$S \rightarrow \mathbf{h}_{\mathcal{CN}2}$	12.90%	57.08%	1 - 14.96±3.59 2 - 82.58±5.65 3 - 44.93±5.74	40.83±2.83	33
$S1 \rightarrow \mathbf{h}_1$	3.30%	57.08%	1 - 33.33±10.54 2 - 80.00±13.33 3 - 20.67±7.11	41.66±5.13	12
$S2 \rightarrow \mathbf{h}_2$	1.70%	57.08%	1 - 8.50±4.60 2 - 60.00±16.33 3 - 40.83±12.94	36.66±7.78	13
$S3 \rightarrow \mathbf{h}_3$	0.00%	57.08%	1 - 20.83±7.38 2 - 80.00±13.33 3 - 18.33±8.03	41.68±5.69	16

Tabela 3: Resultados Obtidos Utilizando o $\mathcal{CN}2$

Com as hipóteses induzidas $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$[10] e o conjunto de exemplos S_4[11], foram construídos seis classificadores utilizando o algoritmo $\mathcal{RCO}$ do Módulo de Combinação e Explicação do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$. Cada um desses seis classificadores foi construído utilizando um critério (medida) diferente de avaliação de regra para eleger a melhor regra em cada iteração. As medidas de avaliação de regra utilizadas neste trabalho[12] constam na Tabela 4. As hipóteses construídas pelo $\mathcal{RCO}$ são denominadas $\mathbf{h}_{\texttt{accR}}$, $\mathbf{h}_{\texttt{covR}}$, $\mathbf{h}_{\texttt{satR}}$, $\mathbf{h}_{\texttt{sensR}}$, $\mathbf{h}_{\texttt{specR}}$ e $\mathbf{h}_{\texttt{supR}}$.

accR:	Precisão da regra
covR:	Cobertura da regra
satR:	Satisfação da regra
sensR:	Sensitividade da regra
specR:	Especificidade da regra
supR:	Suporte da regra

Tabela 4: Medidas de Avaliação de Regras Utilizadas neste Trabalho

Finalmente, foi calculado o erro para cada um dos seis classificadores construídos por $\mathcal{RCO}$ utilizando os subconjuntos de exemplos S_1, S_2 e S_3. Ainda que neste experimento os exemplos

[10] Uma listagem das hipóteses $\mathbf{h}_1$,$\mathbf{h}_2$ e $\mathbf{h}_3$ pode ser visualizada no Apêndice B.

[11] Uma listagem dos exemplos pertencentes a S_4 pode ser visualizada no Apêndice A.

[12] Uma explicação detalhada sobre essas medidas encontra-se em (Gomes, 2001) e (Prati et al., 2001a)

Classificador	Erro Aparente em S_4	Erro Majoritário	Erro nas Classes em $S_1 \bigcup S_2 \bigcup S_3$	Erro do Classificador em $S_1 \bigcup S_2 \bigcup S_3$	# Regras Selecionadas	# Exemplos Não Cobertos
$\mathbf{h}_{\mathbf{accR}}$	36.67%	57.08%	1 - 1.30% 2 - 100.00% 3 - 31.82%	32.78%	12	26 - 43.33%
$\mathbf{h}_{\mathbf{covR}}$	36.67%	57.08%	1 - 7.79% 2 - 100.00% 3 - 33.33%	36.11%	7	21 - 35.00%
$\mathbf{h}_{\mathbf{satR}}$	33.33%	57.08%	1 - 3.90% 2 - 97.30% 3 - 27.27%	31.67%	14	21 - 35.00%
$\mathbf{h}_{\mathbf{sensR}}$	38.33%	57.08%	1 - 2.60% 2 - 100.00% 3 - 45.46%	38.33%	6	24 - 40.00%
$\mathbf{h}_{\mathbf{specR}}$	53.33%	57.08%	1 - 0.00% 2 - 100.00% 3 - 95.45%	55.56%	5	49 - 81.66%
$\mathbf{h}_{\mathbf{supR}}$	38.33%	57.08%	1 - 0.00% 2 - 100.00% 3 - 51.52%	40.56%	6	25 - 41.67%

Tabela 5: Resultados Obtidos Utilizando o $\mathcal{RCO}$

contidos em S_1, S_2 e S_3 não foram utilizados por $\mathcal{RCO}$ para construir os classificadores, eles foram utilizados para induzir as hipóteses $\mathbf{h}_1$, $\mathbf{h}_2$ e $\mathbf{h}_3$, utilizadas pelo $\mathcal{RCO}$. O ideal seria estimar o erro cometido pelos classificadores construídos por $\mathcal{RCO}$ em um conjunto de exemplos independente. Neste trabalho utilizamos a primeira opção devido ao número limitado de exemplos disponíveis. Assim, deve ser ressaltado que o erro por nós estimado utilizando os exemplos contidos nos subconjuntos S_1, S_2 e S_3 tende a ser otimista. Foi medido o erro aparente com o subconjunto de exemplos S_4.

Para construir um novo classificador, o algoritmo $\mathcal{RCO}$ pára em duas condições durante a iteração que executa, como mostra o Algoritmo 1 na página 4 :

1. Não há mais exemplos a serem cobertos;
2. As regras restantes atingiram seu mínimo (ou máximo) absoluto, relacionado à medida sendo considerada.

Na Tabela 5, são mostrados os resultados obtidos, onde a última coluna mostra o número de exemplos restantes na construção do novo classificador. Ou seja, em todos os casos o segundo critério de parada foi utilizado pelo algoritmo $\mathcal{RCO}$. Isso significa que esses exemplos foram classificados pela regra *default* da hipótese construída, a qual prediz a classe 1. Os seis classificadores construídos pelo algoritmo $\mathcal{RCO}$ podem ser visualizados no Apêndice C.

3.3 Análise dos Resultados

Quanto ao número de regras selecionadas pelo $\mathcal{RCO}$ para a construção das hipóteses, pode ser observado que esse número é geralmente menor que o número de regras que constituem cada um

dos classificadores $\mathbf{h}_1$, $\mathbf{h}_2$ e $\mathbf{h}_3$. Entretanto, este resultado é esperado já que, para este conjunto de exemplos, o segundo critério de parada foi sempre utilizado por $\mathcal{RCO}$.

Com relação ao erro dos classificadores construídos por $\mathcal{RCO}$ (erro aparente em S_4 e erro do classificador em $S_1 \bigcup S_2 \bigcup S_3$ na Tabela 5), todos eles são muito mais altos que o erro das hipóteses induzidas por $\mathcal{CN}2$ — Tabela 3. Ainda que o erro em $S_1 \bigcup S_2 \bigcup S_3$ seja semelhante ao erro obtido pelas hipóteses induzidas por $\mathcal{CN}2$ utilizando 10-fold cross-validation, essa comparação não é válida, pois o erro em $S_1 \bigcup S_2 \bigcup S_3$ é muito otimista, enquanto que o erro obtido utilizando 10-fold cross validation é uma boa estimativa de erro verdadeiro. Porém, deve ser observado que o objetivo de utilizar $\mathcal{RCO}$ com as medidas consideradas não é, necessariamente, o de construir um bom classificador no sentido de classificar relativamente bem qualquer exemplo. Na realidade, o classificador construído por $\mathcal{RCO}$ é aquele que possui regras selecionadas segundo um critério (medida) utilizado. Dessa forma, é interessante observar qual o comportamento dessas melhores regras.

Para verificar o comportamento das regras que constituem as hipóteses construídas pelo $\mathcal{RCO}$, devem ser analisados o número de regras de cada hipótese e os exemplos cobertos pela regra *default*. Esses dados são mostrados nas Tabelas 6 e 7. Na Tabela 6 consta, para cada hipótese construída pelo $\mathcal{RCO}$, o número de exemplos do conjunto S_4 classificados pela regra *default*, e na Tabela 7 consta o número de regras selecionadas por $\mathcal{RCO}$ para cada classe do domínio. Como pode ser observado, a classe mais crítica é a classe 2. O conjunto de exemplos S_4 possui 16 exemplos de classe 2. Assim, quando não há nenhuma regra que prediz essa classe, como no caso das hipóteses $\mathbf{h}_{\texttt{accR}}$, $\mathbf{h}_{\texttt{sensR}}$, $\mathbf{h}_{\texttt{specR}}$ e $\mathbf{h}_{\texttt{supR}}$, então os 16 exemplos são cobertos pela regra *default*. Como essa regra prediz a classe 1, todos esses exemplos são classificados erroneamente, incrementando o erro dos classificadores construídos pelo $\mathcal{RCO}$. Pode-se assim concluir que não há "melhores" regras para predizer a classe 2 utilizando esses quatro critérios.

Classe	$\mathbf{h}_{\texttt{accR}}$	$\mathbf{h}_{\texttt{covR}}$	$\mathbf{h}_{\texttt{satR}}$	$\mathbf{h}_{\texttt{sensR}}$	$\mathbf{h}_{\texttt{specR}}$	$\mathbf{h}_{\texttt{supR}}$	# Exemplos em S_4
1	6	6	6	6	17	6	26
2	16	13	13	16	16	16	16
3	4	2	2	2	16	3	18
Total	26	21	21	24	49	25	60

Tabela 6: Número de Exemplos Classificados por Classe pela Regra *Default*

Classe	$\mathbf{h}_{\texttt{accR}}$	$\mathbf{h}_{\texttt{covR}}$	$\mathbf{h}_{\texttt{satR}}$	$\mathbf{h}_{\texttt{sensR}}$	$\mathbf{h}_{\texttt{specR}}$	$\mathbf{h}_{\texttt{supR}}$	Distribuição dos Exemplos
1	7	2	7	2	3	3	42.92%
2	0	1	1	0	0	0	22.08%
3	4	3	5	3	1	2	35.00%
1 (*default*)	1	1	1	1	1	1	-
Total	12	7	14	6	5	6	-

Tabela 7: Número de Regras Selecionadas por Classe

Já para a classe 3, observa-se em geral uma pequena quantidade de exemplos cobertos pela regra *default*; no entanto, as regras selecionadas que classificam exemplos como sendo pertencentes a essa classe não os cobrem corretamente.

Fazendo uma análise da hipótese $\mathbf{h}_{\texttt{specR}}$, a qual apresentou uma alta taxa de erro, deve ser

lembrado que o critério specR é uma medida de avaliação de regra do número (relativo) de exemplos da classe prevista pela *Cabeça* cobertos pela regra. É definida pela propriedade de o *Corpo* ser falso dado que a *Cabeça* é falsa (Prati et al., 2001a). Isso implica que, se a *Cabeça* da regra não coincide com a classe do exemplo então ela não cobre o exemplo, o que é equivalente a dizer que uma regra com alta especificidade, quando cobre um exemplo, ela o faz corretamente. Se for observada a Tabela 6, há somente 11 exemplos não cobertos pela regra *default*, dos quais 2(dois) exemplos são da classe 2 e 9(nove) exemplos são da classe 1. Como os exemplos restantes são classificados pela regra *default* e a regra *default* classifica os exemplos como sendo pertencentes da classe 1, o erro cometido por esse classificador foi alto para estes exemplos. Para a classe 2, o erro cometido foi de 100%, e o mesmo não aconteceu para a classe 3 porque existe uma regra na hipótese que cobre corretamente 2 exemplos pertencentes a essa classe (Tabela 5). Deve ser observado que o erro para a classe 1 foi de 0% porque os exemplos cobertos pelas regras da hipótese cobrem corretamente os exemplos da classe 1, e o restante dos exemplos é coberto (corretamente) pela regra *default*.

Quanto às regras selecionadas pelo $\mathcal{RCO}$ para os 6(seis) critérios utilizados, observa-se, no Apêndice C, que foram selecionadas um total de 17 regras diferentes que participam dos classificadores construídos pelo $\mathcal{RCO}$. Nas hipóteses construídas, essas 17 regras aparecem em ordens diferentes nas hipóteses ou aparecem em algumas hipóteses e não aparecem em outras. Esse fato foi observado quando se deu a comparação das hipóteses construídas com algumas hipóteses analisadas pelo especialista Dr. Sandro C. Esteves. A Tabela 8 mostra o corpo e a cabeça (classe) das regras pertencentes às hipóteses construídas pelo $\mathcal{RCO}$, na sintaxe padrão Prolog, o número de vezes que a regra foi selecionada utilizando esses seis critérios, e a opinião do especialista em relação á regra (**N** se não faz sentido, **OK** se a regra faz sentido e **?** se a regra não foi analisada pelo especialista). Quando a regra foi rotulada por **N** na opinião do especialista, a tabela também contém o índice do motivo que o levou a dizer que a regra não faz sentido. Os motivos do especialista e seus respectivos índices constam na Tabela 9.

Observa-se, na Tabela 8, que, das regras analisadas pelo especialista, metade das regras faz sentido para o especialista, e metade não faz sentido.

As 3(três) primeiras regras da Tabela 8 foram selecionadas pelos critérios accR, satR e specR como as primeiras regras com melhor valor nessas medidas. No entanto, para o especialista, elas não fazem sentido, o que mostra que essas medidas devem ser melhor analisadas para utilizá-las como critério de seleção de regras para a construção de uma hipótese, utilizando o $\mathcal{RCO}$.

4 Conclusão

A técnica de combinação de classificadores simbólicos proposta pelo algoritmo $\mathcal{RCO}$ em (Bernardini and Monard, 2002) foi pensada para ser utilizada em grandes bases de dados, particionando o conjunto de exemplos disponível em subconjuntos que sejam manipuláveis por algoritmos de AM simbólico, considerando que o conjunto de exemplos todo não é manipulável por um único algoritmo de AM. Assim, a idéia é construir diversos classificadores utilizando subconjuntos desse grande conjunto de exemplos, tal que esses subconjuntos são manipuláveis por algoritmos de AM simbólico. Logo após, a proposta é construir um único classificador simbólico utilizando as "melhores" regras do conjunto de classificadores induzidos inicialmente.

Corpo	Cabeça (Classe)	# Vezes Selecionada	Opinião do Especialista	Motivo
greater(idade,36.5),greater(class_B,29.5)	1	3	N	1
greater(tempo_abs,1.5),less(class_C,65), less(vitalidade,53.5)	1	3	N	3
greater(volume,4.1),greater(pH,8.25)	3	3	N	4
less(processamento,32.5),less(volume,4.2), less(concentracao_total,27.95)	1	3	OK	-
greater(pH,7.75),less(pH,8.25), less(concentracao_total,72.7),less(motilidade,12.5)	1	3	OK	-
greater(class_C,34.5),less(kruger,1.5)	1	2	OK	-
greater(idade,29),less(tempo_abs,6.5), less(hP,58.5)	1	2	?	-
less(idade,48.5),less(concentracao,33.65), greater(class_B,7),less(hP,75.5)	1	5	N	1 e 5
greater(vitalidade,70.5),greater(kruger,11.5), greater(hP,60)	3	2	OK	-
greater(concentracao_total,221.25),less(class_B,28.5), less(class_C,70.5)	3	5	OK	-
greater(concentracao,24),greater(motilidade,28.5)	3	2	OK	-
greater(volume,1.35),less(num_leu,1.28),less(hP,69)	1	3	N	-
greater(motilidade,19.5),less(num_leu,0.68)	3	1	N	7
less(pH,8.25),greater(concentracao,63.5), greater(motilidade,0.5)	3	4	OK	Observar Motivo 6
less(class_A,11.5)	2	1	N	1
greater(class_B,25.5),greater(kruger,3.5)	2	1	OK	
greater(idade,31.5),less(processamento,35), greater(concentracao,20.35),less(class_A,21.5)	3	1	N	1

Tabela 8: Regras Selecionadas pelo $\mathcal{RCO}$

Motivos do Especialista	
1:	Quanto maior o valor dos atributos class_A ou class_B, maior a chance da regra predizer as classes 2 ou 3
2:	Quanto maior o valor dos atributos class_C ou class_D, menor a chance da regra predizer a classe 3
3:	Quanto menor o valor dos atributos class_C ou class_D, menor a chance da regra predizer a classe 1
4:	Especialista não utilizaria os atributos da regra
5:	Valor de corte muito alto para o atributo hP
6:	Valor de corte baixo para o atributo motilidade
7:	Quanto menor o valor do atributo num_leu, menor a chance da regra predizer a classe 3

Tabela 9: Motivos do Especialista para Considerar que uma Regra Não Faz Sentido

O estudo de caso descrito neste trabalho mostrou o uso da técnica proposta pelo $\mathcal{RCO}$ para domínios nos quais o conjunto de exemplos disponível é pequeno. Como esperado, foi observado que as hipóteses construídas pelo $\mathcal{RCO}$ possuem um número bem menor de regras em relação ao número total de regras dos classificadores que fazem parte dos dados de entrada do $\mathcal{RCO}$.

Nesta primeira implementação do $\mathcal{RCO}$ foram consideradas as "melhores" regras levando em conta seis medidas de regras, mas não o número de exemplos envolvidos no cálculo dessas medidas. Por exemplo, no caso da medida ser considerada `accR`, se duas (ou mais) regras R_u e R_q têm a mesma medida de `accR`, pelo critério atual implementado em $\mathcal{RCO}$, essas duas regras ficam empatadas. Entretanto, é fácil ver que neste caso seria melhor escolher a regra que cobre um número maior de exemplos, e não escolher aleatoriamente a "melhor" regra somente analisando o valor da medida.

Na realidade, a idéia de considerar o número de exemplos envolvidos no cálculo das medidas de regras poderia ser utilizado não somente em caso de empate. Por exemplo, ainda considerando a medida `accR`, supondo que para a regra R_u, o valor de `accR` seja 1, e para a regra R_q, o valor de `accR` seja 0.98; segundo o critério atual de $\mathcal{RCO}$, a regra R_u é melhor que a regra R_q. Entretanto, se a regra R_u cobre somente um exemplo e a regra R_q cobre 200 exemplos (portanto, 4 deles são cobertos erroneamente), o critério de escolha da melhor regra poderia ser revisto.

Como pode ser observado pelos resultados apresentados na Tabela 5, o erro dos classificadores construídos utilizando a versão atual do $\mathcal{RCO}$ deixa a desejar. O que sim é possível apresentar ao especialista é um conjunto de regras que são as melhores regras encontradas nos classificadores utilizados como dados de entrada do $\mathcal{RCO}$.

Na Tabela 8, é possível observar que as regras selecionadas pelo $\mathcal{RCO}$ também não são as melhores regras no ponto de vista do especialista. Este fato reforça a nossa idéia de continuar investigando a revisão dos critérios implementados no $\mathcal{RCO}$ para seleção das melhores regras, conforme dito anteriormente.

Deve-se notar a importância de analisar os classificadores simbólicos internamente, e não considerá-los como caixas pretas. Para analisar os resultados obtidos, foram analisadas as regras dos classificadores construídos pelo $\mathcal{RCO}$, olhando para o número de regras selecionadas para cada classe e o corpo das regras das hipóteses construídas pelo $\mathcal{RCO}$, por parte do especialista, para analisar a qualidade das regras selecionadas.

A Conjunto de Dados Utilizado pelo $\mathcal{RCO}$

Segue uma listagem completa do conjunto de dados S_4,na Sintaxe Padrão Prolog (Gomes et al., 2002) do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$, utilizado para realizar os experimentos.

```
ex(0,[39,manha,20,2,4.9,branco,caracteristico,8,normal,
   completa,14.9,73.01,0,53,12,35,53,80,0.08,2,63,um]).
ex(1,[26,manha,30,3,4.7,translucido,caracteristico,8,normal,
   completa,0.8,3.76,0,53,7,40,53,60,0.02,?,50,um]).
ex(2,[32,manha,20,2,4.7,branco,caracteristico,8.5,aumentada,
   incompleta,56,263.2,4,69,12,15,73,72,0.03,15,63,tres]).
ex(3,[38,manha,30,3,3,amarelo,caracteristico,8,aumentada,
   incompleta,69.3,207.9,0,56,8,36,56,70,1.24,11,74,dois]).
ex(4,[34,manha,40,2,2.8,branco,caracteristico,8,normal,
   completa,46,128.8,6,26,9,59,32,58,0,0.5,55,um]).
ex(5,[32,manha,30,3,1.7,branco,caracteristico,8,normal,
   completa,19.7,33.5,0,51,18,31,51,67,5.17,?,65,dois]).
ex(6,[52,tarde,20,3,2.8,branco,caracteristico,8,normal,
   incompleta,68.5,191.8,9,65,8,18,74,79,0.42,3,80,tres]).
ex(7,[22,tarde,49,3,2.9,branco,caracteristico,8,aumentada,
   incompleta,29.7,86.1,0,49.3,20,30.7,49.3,13,0,6,69,um]).
ex(8,[40,manha,50,8,5.7,branco,caracteristico,8.5,normal,
   completa,89.8,511.8,0,55,15,30,55,84,0,7,81,tres]).
ex(9,[27,manha,25,3,3.4,branco,caracteristico,8,normal,
   incompleta,2.1,7.14,10,15,5,70,25,65,0,?,60,um]).
ex(10,[?,tarde,30,4,2.5,branco,caracteristico,8,normal,
   incompleta,201.5,503.8,19,58,8,15,77,72,0.04,4,83,tres]).
ex(11,[32,manha,30,3,2.4,amarelo,caracteristico,8,normal,
   incompleta,13.2,31.7,2,72,5,21,74,68,28.2,9,46,um]).
ex(12,[38,manha,20,3,1.8,amarelo,caracteristico,8,aumentada,
   incompleta,27.5,49.5,13,21,21,45,34,57,0,20,48,um]).
ex(13,[32,tarde,30,5,2.8,branco,caracteristico,8,aumentada,
   incompleta,182.8,511.84,0,36,8,56,36,77,0,8,81,tres]).
ex(14,[35,tarde,25,2,4.9,branco,caracteristico,8,aumentada,
   incompleta,56.4,276.4,0,43,18,39,43,68,0,6,61,dois]).
ex(15,[34,manha,35,3,6,translucido,caracteristico,8,normal,
   completa,14.8,88.8,6,65,8,21,71,80,0,1,50,dois]).
ex(16,[34,manha,20,3,4.6,branco,caracteristico,8,normal,
   completa,18.3,84.2,25,18,22,35,43,70,1.16,3,65,dois]).
ex(17,[35,tarde,20,4,1.9,branco,caracteristico,8.5,normal,
   incompleta,65,123.5,1,67,11,21,68,70,0,6,66,um]).
ex(18,[27,manha,20,3,1.2,branco,caracteristico,8,normal,
   incompleta,115,138,18,36,25,21,54,67,0,6,80,tres]).
ex(19,[44,manha,20,4,3.2,amarelo,caracteristico,8,normal,
   completa,63,201.6,6,21,27,46,27,65,10.3,9,78,dois]).
```

```
ex(20,[37,manha,20,3,2.8,branco,caracteristico,8,normal,
   completa,130,364,10,47,29,14,57,78,0.09,2,76,dois]).
ex(21,[37,tarde,30,3,3.2,branco,caracteristico,8,normal,
   completa,122.9,393.3,32,51,9,8,83,75,0.21,2,72,dois]).
ex(22,[28,tarde,45,3,3.4,branco,caracteristico,8,aumentada,
   completa,89.5,304.3,24,37,18,21,61,75,76.5,13,72,dois]).
ex(23,[27,tarde,25,4,3.9,branco,caracteristico,8,normal,
   completa,76.3,297.6,0,67,7,26,67,88,0.89,11,84,um]).
ex(24,[29,manha,30,3,3.8,translucido,caracteristico,7.5,normal,
   incompleta,11.4,43.32,24,48,8,20,72,80,0.14,4,64,um]).
ex(25,[39,tarde,20,3,3.8,translucido,caracteristico,8,normal,
   incompleta,43.9,166.8,15,62,5,18,77,78,0,1,76,tres]).
ex(26,[29,tarde,20,3,1.8,branco,caracteristico,8,aumentada,
   completa,101.1,182,42,31,0,27,73,73,0,4,67,tres]).
ex(27,[36,manha,60,3,1.9,branco,caracteristico,8.5,aumentada,
   completa,25.6,48.6,4,21,10,65,25,25,0.2,4,57,um]).
ex(28,[34,manha,20,3,2.6,branco,caracteristico,8.5,normal,
   completa,161.3,419.4,49,15,8,28,64,76,0.87,2,84,tres]).
ex(29,[30,manha,60,3,0.4,branco,caracteristico,8,aumentada,
   incompleta,1.4,0.56,0,0,38,62,0,31,0.29,3,44,um]).
ex(30,[32,manha,30,2,2.8,branco,caracteristico,8,aumentada,
   completa,7.4,20.7,11,11,5,73,22,19,0,4,28,um]).
ex(31,[38,tarde,20,2,4.4,branco,caracteristico,8,normal,
   completa,22,96.8,40,19,11,30,59,72,0.57,0,66,dois]).
ex(32,[43,tarde,30,3,2.8,branco,caracteristico,8.5,normal,
   incompleta,40,112,61,11,17,11,72,83,0,15,?,tres]).
ex(33,[36,manha,25,5,5,amarelo,caracteristico,8,normal,
   incompleta,197.8,989,58,9,6,27,67,73,0,15,79,tres]).
ex(34,[31,manha,20,3,2.2,branco,caracteristico,8,normal,
   incompleta,120.5,265.1,15,20,30,35,35,70,0.22,10,85,dois]).
ex(35,[25,tarde,40,4,2,branco,caracteristico,8,normal,
   incompleta,24.3,48.6,67,2,5,26,69,82,0,5,?,dois]).
ex(36,[39,tarde,20,3,4,amarelo,caracteristico,8,normal,
   completa,23.8,95.2,1,5,22,72,6,60,0,10,78,um]).
ex(37,[29,manha,20,9,2.2,branco,caracteristico,8,normal,
   incompleta,18,39.6,0,7,15,78,7,26,2.15,5,28,um]).
ex(38,[45,manha,20,10,3.8,branco,caracteristico,7.5,normal,
   completa,99.8,379.2,62,10,5,23,72,78,0,10,99,tres]).
ex(39,[34,manha,60,4,2,branco,caracteristico,8.8,aumentada,
   incompleta,56.3,112.6,2,18,31,49,20,67,0,0,67,dois]).
ex(40,[36,tarde,20,8,0.9,branco,caracteristico,8,aumentada,
   incompleta,78,70.2,40,26,10,24,66,80,0,6,78,dois]).
ex(41,[32,manha,40,3,3.1,branco,caracteristico,8,normal,
   incompleta,43.8,135.8,15,15,39,31,30,58,0,10,77,um]).
ex(42,[26,tarde,30,2,2.7,branco,caracteristico,8,normal,
   incompleta,92.3,249.2,0,23,30,47,23,80,0,5,75,um]).
```

```
ex(43,[36,manha,45,7,3.3,branco,caracteristico,8,normal,
   incompleta,9.25,30.5,3,24,22,51,27,49,0,0,46,um]).
ex(44,[34,manha,20,3,3.2,branco,caracteristico,?,normal,
   completa,7.6,24.3,3,22,21,54,25,45,0.63,0,39,um]).
ex(45,[40,manha,20,3,0.9,branco,caracteristico,8,normal,
   completa,688.8,601.9,16,9,19,56,25,68,0,18,82,tres]).
ex(46,[30,tarde,20,3,3.7,branco,caracteristico,8,aumentada,
   incompleta,0.92,3.4,8,11,12,69,19,?,0.03,0,?,um]).
ex(47,[35,tarde,20,3,6.3,branco,caracteristico,8,normal,
   completa,16.5,104,30,17,23,76,47,?,4.63,7,74,um]).
ex(48,[40,tarde,20,3,4.5,translucido,caracteristico,8,normal,
   completa,4,18,3,14,36,47,17,?,0,3,64,um]).
ex(49,[40,tarde,20,5,3,branco,caracteristico,7.5,normal,
   completa,88.5,265.5,46,15,16,23,61,77,0,8,87,tres]).
ex(50,[30,manha,20,3,1.3,branco,caracteristico,8,aumentada,
   completa,73.4,95.4,0,5,27,68,5,45,0,3,59,dois]).
ex(51,[31,manha,30,2,2.8,amarelo,forte,8,normal,
   incompleta,83.5,233.8,0,12,33,55,12,71,0,7,76,dois]).
ex(52,[32,tarde,20,3,4,branco,caracteristico,8,normal,
   incompleta,85,340,0,20,28,52,20,64,0.29,1.5,77,tres]).
ex(53,[28,tarde,40,3,3.7,amarelo,caracteristico,8,aumentada,
   completa,69.3,256.4,37,18,10,35,55,83,0.06,4,78,tres]).
ex(54,[34,manha,40,1,1.4,branco,caracteristico,8.8,normal,
   completa,242,338.8,0,3,25,72,3,72,0,4,77,um]).
ex(55,[29,manha,30,4,2.5,amarelo,forte,8,aumentada,
   incompleta,32.3,80.8,0,0,17,83,0,22,1.22,8,20,um]).
ex(56,[36,manha,20,5,3.4,branco,caracteristico,8.3,normal,
   incompleta,28.8,97.9,0,14,38,48,14,56,0.38,2,64,um]).
ex(57,[29,manha,50,4,3.5,translucido,caracteristico,8.3,normal,
   completa,3.5,12.3,45,20,14,21,65,84,0.38,0,76,tres]).
ex(58,[38,tarde,20,5,3.5,branco,caracteristico,7.5,normal,
   incompleta,13.8,48.3,23,10,5,62,33,39,0,0,48,um]).
ex(59,[37,tarde,20,4,3,branco,caracteristico,8,normal,
   completa,50,150,56,18,13,13,74,88,0,5,84,tres]).

% Features
feature(0, idade, numeric).
feature(1, hora, nominal(manha,tarde)).
feature(2, processamento, numeric).
feature(3, tempo_abs, numeric).
feature(4, volume, numeric).
feature(5, cor, nominal(branco,amarelo,translucido)).
feature(6, odor, nominal(caracteristico,forte,urina)).
feature(7, pH, numeric).
feature(8, viscosidade, nominal(normal,aumentada)).
feature(9, liquefacao, nominal(completa,incomleta)).
```

```
feature(10, concentracao, numeric).
feature(11, concentracao_total, numeric).
feature(12, motilidade, numeric).
feature(13, class_A, numeric).
feature(14, class_B, numeric).
feature(15, class_C, numeric).
feature(16, class_D, numeric).
feature(17, vitalidade, numeric).
feature(18, num_leu, numeric).
feature(19, kruger, numeric).
feature(20, hP, numeric).
feature(21, class, nominal(um,dois,tres)).

% Class Feature
classFeature(class).
```

B Classificadores Utilizados pelo $\mathcal{RCO}$

Os três classificadores simbólicos $\mathbf{h}_1$, $\mathbf{h}_2$ e $\mathbf{h}_3$ utilizados no experimento, induzidos pelo algoritmo de AM $\mathcal{CN}2$ e escritos em linguagem Prolog, encontram-se a seguir. Deve ser observado que a matriz de contingência (freqüências) das regras que constituem cada um desses classificadores foi calculada utilizando todo o conjunto de exemplos disponível, ou seja, 240 exemplos.

B.1 Classificador $\mathbf{h}_1$

```
inducer(id_170,'cn2'). inputFile(id_170,'-').
dAte(id_170,day('Tue','Jan',8,2002),time(17:6:35)).
evaluatedAs(id_170,'UNORDERED'). dataFile(id_170,'proc0.data').
nameFile(id_170,'proc.names').

rule(id_170,1,
  [greater(idade,29),less(tempo_abs,6.5),less(hP,58.5)],
  class(um),
  [0.178,0.017,0.548,0.257,230.000],[0.300,0.300,0.400,0.000,10.000]).
rule(id_170,2,
  [greater(pH,7.75),less(pH,8.25),less(concentracao_total,72.7),
  less(motilidade,12.5)],
  class(um),
  [0.150,0.026,0.556,0.269,234.000],[0.833,0.000,0.167,0.000,6.000]).
rule(id_170,3,
  [greater(idade,25.5),less(idade,26.5)],
  class(um),
  [0.017,0.008,0.559,0.416,238.000],[0.000,1.000,0.000,0.000,2.000]).
```

```
rule(id_170,4,
  [greater(concentracao,83.5),less(concentracao,120.9)],
  class(dois),
  [0.038,0.050,0.729,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_170,5,
  [less(pH,7.7),greater(class_A,20.5)],
  class(dois),
  [0.017,0.030,0.744,0.209,234.000],[0.000,0.167,0.833,0.000,6.000]).
rule(id_170,6,
  [greater(idade,26.5),less(idade,39.5),equal(cor,amarelo),
  greater(class_A,12.5)],
  class(dois),
  [0.021,0.051,0.722,0.205,234.000],[0.000,0.000,1.000,0.000,6.000]).
rule(id_170,7,
  [greater(processamento,15),less(tempo_abs,2.5),greater(class_A,64)],
  class(dois),
  [0.013,0.008,0.771,0.208,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_170,8,
  [greater(motilidade,19.5),less(num_leu,0.68)],
  class(tres),
  [0.155,0.075,0.573,0.197,239.000],[0.000,0.000,1.000,0.000,1.000]).
rule(id_170,9,
  [greater(concentracao,29.9),less(motilidade,12.5),less(class_B,8.5)],
  class(tres),
  [0.050,0.029,0.621,0.300,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_170,10,
  [less(tempo_abs,3.5),greater(concentracao_total,251.35),
  greater(motilidade,1.5)],
  class(tres),
  [0.083,0.017,0.633,0.267,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_170,11,
  [greater(motilidade,12.5),less(class_A,19)],
  class(tres),
  [0.096,0.063,0.588,0.254,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_170,12,
  [],
  class(tres),
  [],[]).
```

B.2 Classificador h_2

```
inducer(id_153,'cn2'). inputFile(id_153,'-').
dAte(id_153,day('Tue','Jan',8,2002),time(17 : 5 : 19)).
evaluatedAs(id_153,'UNORDERED'). dataFile(id_153,'proc1.data').
```

```
nameFile(id_153,'proc.names').

rule(id_153,1,
  [greater(volume,1.35),less(num_leu,1.28),less(hP,69)],
  class(um),
  [0.284,0.108,0.461,0.147,232.000],[0.375,0.500,0.125,0.000,8.000]).
rule(id_153,2,
  [less(processamento,32.5),less(volume,4.2),less(concentracao_total,27.95)],
  class(um),
  [0.088,0.017,0.554,0.342,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_153,3,
  [greater(concentracao,46.75),less(concentracao,111.3),greater(class_C,48)],
  class(um),
  [0.042,0.033,0.538,0.388,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_153,4,
  [less(concentracao,33.15),greater(hP,88.5)],
  class(um),
  [0.004,0.009,0.560,0.427,232.000],[0.375,0.375,0.250,0.000,8.000]).
rule(id_153,5,
  [less(volume,1.55),equal(viscosidade,normal),less(num_leu,1.97)],
  class(dois),
  [0.021,0.021,0.757,0.201,239.000],[0.000,0.000,1.000,0.000,1.000]).
rule(id_153,6,
  [greater(class_B,25.5),greater(kruger,3.5)],
  class(dois),
  [0.039,0.034,0.742,0.185,233.000],[0.000,0.000,0.857,0.143,7.000]).
rule(id_153,7,
  [greater(idade,26.5),less(idade,27.5)],
  class(dois),
  [0.017,0.021,0.756,0.206,238.000],[0.000,1.000,0.000,0.000,2.000]).
rule(id_153,8,
  [less(processamento,22.5),greater(volume,4.75)],
  class(dois),
  [0.008,0.058,0.721,0.213,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_153,9,
  [greater(concentracao_total,221.25),less(class_B,28.5),less(class_C,70.5)],
  class(tres),
  [0.167,0.058,0.592,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_153,10,
  [greater(concentracao,24),greater(motilidade,28.5)],
  class(tres),
  [0.113,0.013,0.638,0.238,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_153,11,
  [greater(vitalidade,70.5),greater(kruger,11.5),greater(hP,60)],
  class(tres),
  [0.104,0.014,0.615,0.267,221.000],[0.105,0.105,0.789,0.000,19.000]).
```

```
rule(id_153,12,
  [greater(volume,4.1),greater(pH,8.25)],
  class(tres),
  [0.021,0.000,0.645,0.333,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_153,13,
  [],
  class(tres),
  [],[]).
```

B.3 Classificador h_3

```
inducer(id_162,'cn2'). inputFile(id_162,'-').
dAte(id_162,day('Tue','Jan',8,2002),time(17 : 6 : 27)).
evaluatedAs(id_162,'UNORDERED'). dataFile(id_162,'proc2.data').
nameFile(id_162,'proc.names').

rule(id_162,1,
  [less(idade,48.5),less(concentracao,33.65),greater(class_B,7),less(hP,75.5)],
  class(um),
  [0.248,0.052,0.513,0.187,230.000],[0.300,0.000,0.700,0.000,10.000]).
rule(id_162,2,
  [greater(tempo_abs,1.5),less(class_C,65),less(vitalidade,53.5)],
  class(um),
  [0.094,0.021,0.562,0.322,233.000],[0.571,0.143,0.000,0.286,7.000]).
rule(id_162,3,
  [less(processamento,35),less(motilidade,7.5),less(hP,62.5)],
  class(um),
  [0.159,0.013,0.556,0.272,232.000],[0.125,0.250,0.375,0.250,8.000]).
rule(id_162,4,
  [greater(class_C,34.5),less(kruger,1.5)],
  class(um),
  [0.069,0.017,0.562,0.352,233.000],[0.429,0.000,0.286,0.286,7.000]).
rule(id_162,5,
  [greater(idade,36.5),greater(class_B,29.5)],
  class(um),
  [0.008,0.004,0.563,0.424,238.000],[0.000,0.000,1.000,0.000,2.000]).
rule(id_162,6,
  [greater(hP,79.5),less(hP,83.5)],
  class(dois),
  [0.034,0.060,0.728,0.177,232.000],[0.500,0.500,0.000,0.000,8.000]).
rule(id_162,7,
  [greater(idade,34.5),less(idade,42),less(volume,2.35),less(motilidade,11.5)],
  class(dois),
  [0.025,0.055,0.723,0.197,238.000],[0.000,0.000,1.000,0.000,2.000]).
```

```
rule(id_162,8,
  [less(class_A,11.5)],
  class(dois),
  [0.025,0.113,0.667,0.196,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_162,9,
  [equal(cor,translucido),less(kruger,1.25)],
  class(dois),
  [0.009,0.013,0.760,0.218,229.000],[0.000,0.091,0.818,0.091,11.000]).
rule(id_162,10,
  [greater(idade,49),greater(processamento,22.5)],
  class(dois),
  [0.004,0.000,0.777,0.218,238.000],[0.000,0.500,0.500,0.000,2.000]).
rule(id_162,11,
  [less(pH,8.25),greater(concentracao,63.5),greater(motilidade,0.5)],
  class(tres),
  [0.145,0.068,0.577,0.209,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_162,12,
  [less(idade,44.5),greater(concentracao_total,189.98),greater(num_leu,0.11)],
  class(tres),
  [0.076,0.046,0.608,0.270,237.000],[0.000,0.000,0.333,0.667,3.000]).
rule(id_162,13,
  [greater(idade,31.5),less(processamento,35),greater(concentracao,20.35),
  less(class_A,21.5)],
  class(tres),
  [0.084,0.046,0.609,0.261,238.000],[0.000,0.000,0.000,1.000,2.000]).
rule(id_162,14,
  [greater(idade,35.5),greater(processamento,38)],
  class(tres),
  [0.021,0.008,0.647,0.324,238.000],[0.000,0.000,0.000,1.000,2.000]).
rule(id_162,15,
  [greater(processamento,25),greater(hP,84.5)],
  class(tres),
  [0.026,0.000,0.642,0.332,232.000],[0.125,0.375,0.500,0.000,8.000]).
rule(id_162,16,
  [],
  class(tres),
  [],[]).
```

C Classificadores Construídos pelo $\mathcal{RCO}$

a matriz de contingência (freqüências) das regras que constituem os classificadores construídos pelo $\mathcal{RCO}$ foi calculada utilizando todo o conjunto de dados disponível para os experimentos, ou seja, 240 exemplos.

Os 6(seis) classificadores simbólicos construídos pelo $\mathcal{RCO}$ estão escritos em linguagem Prolog. Seus identificadores são criados a partir do critério de seleção de regra. Antes do cabeçalho de cada hipótese, consta uma lista com os identificadores dos exemplos não cobertos pela hipótese. Os exemplos podem ser visualizados na íntegra no Apêndice A.

C.1 Classificador h_{accR}

```
%Exemplos não cobertos: [3,5,6,14,15,16,17,18,19,20,21,
%22,23,25,31,34,35,36,39,40,41,42,50,51,54,57]

inducer(id_accR211,rco).
inputFile(id_accR211,'-').
dAte(id_accR211,day('-','Mar',15,2002),time(16 : 7 : 2)).
evaluatedAs(id_accR211,'UNORDERED').
dataFile(id_accR211,'-').

rule(id_accR211,1,
  [greater(idade,36.5),greater(class_B,29.5)],
  class(um),
  [0.008,0.004,0.563,0.424,238.000],[0.000,0.000,1.000,0.000,2.000]).
rule(id_accR211,2,
  [greater(tempo_abs,1.5),less(class_C,65),less(vitalidade,53.5)],
  class(um),
  [0.094,0.021,0.562,0.322,233.000],[0.571,0.143,0.000,0.286,7.000]).
rule(id_accR211,3,
  [greater(volume,4.1),greater(pH,8.25)],
  class(tres),
  [0.021,0.000,0.645,0.333,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_accR211,4,
  [less(processamento,32.5),less(volume,4.2),less(concentracao_total,27.95)],
  class(um),
  [0.088,0.017,0.554,0.342,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_accR211,5,
  [greater(pH,7.75),less(pH,8.25),less(concentracao_total,72.7),less(motilidade,12.5)],
  class(um),
  [0.150,0.026,0.556,0.269,234.000],[0.833,0.000,0.167,0.000,6.000]).
rule(id_accR211,6,
  [greater(class_C,34.5),less(kruger,1.5)],
  class(um),
  [0.069,0.017,0.562,0.352,233.000],[0.429,0.000,0.286,0.286,7.000]).
rule(id_accR211,7,
  [greater(idade,29),less(tempo_abs,6.5),less(hP,58.5)],
  class(um),
  [0.178,0.017,0.548,0.257,230.000],[0.300,0.300,0.400,0.000,10.000]).
```

```
rule(id_accR211,8,
  [less(idade,48.5),less(concentracao,33.65),greater(class_B,7),less(hP,75.5)],
  class(um),
  [0.248,0.052,0.513,0.187,230.000],[0.300,0.000,0.700,0.000,10.000]).
rule(id_accR211,9,
  [greater(vitalidade,70.5),greater(kruger,11.5),greater(hP,60)],
  class(tres),
  [0.104,0.014,0.615,0.267,221.000],[0.105,0.105,0.789,0.000,19.000]).
rule(id_accR211,10,
  [greater(concentracao_total,221.25),less(class_B,28.5),less(class_C,70.5)],
  class(tres),
  [0.167,0.058,0.592,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_accR211,11,
  [greater(concentracao,24),greater(motilidade,28.5)],
  class(tres),
  [0.113,0.013,0.638,0.238,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_accR211,12,
  [],
  class(um),
  [],[]).
```

C.2 Classificador h_{covR}

```
%Exemplos não cobertos: [3,5,11,14,15,16,19,20,21,22,23,
%25,31,34,36,39,40,41,42,51,54]

inducer(id_covR211,rco).
inputFile(id_covR211,'-').
dAte(id_covR211,day('-','Mar',15,2002),time(16 : 7 : 25)).
evaluatedAs(id_covR211,'UNORDERED').
dataFile(id_covR211,'-').

rule(id_covR211,1,
  [greater(volume,1.35),less(num_leu,1.28),less(hP,69)],
  class(um),
  [0.284,0.108,0.461,0.147,232.000],[0.375,0.500,0.125,0.000,8.000]).
rule(id_covR211,2,
  [less(idade,48.5),less(concentracao,33.65),greater(class_B,7),less(hP,75.5)],
  class(um),
  [0.248,0.052,0.513,0.187,230.000],[0.300,0.000,0.700,0.000,10.000]).
rule(id_covR211,3,
  [greater(concentracao_total,221.25),less(class_B,28.5),less(class_C,70.5)],
  class(tres),
  [0.167,0.058,0.592,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
```

```
rule(id_covR211,4,
  [greater(motilidade,19.5),less(num_leu,0.68)],
  class(tres),
  [0.155,0.075,0.573,0.197,239.000],[0.000,0.000,1.000,0.000,1.000]).
rule(id_covR211,5,
  [less(pH,8.25),greater(concentracao,63.5),greater(motilidade,0.5)],
  class(tres),
  [0.145,0.068,0.577,0.209,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_covR211,6,
  [less(class_A,11.5)],
  class(dois),
  [0.025,0.113,0.667,0.196,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_covR211,7,
  [],
  class(um),
  [],[]).
```

C.3 Classificador h_{satR}

```
%Exemplos não cobertos: [3,5,14,15,16,17,20,21,22,23,
%25,31,35,36,39,40,41,42,50,54,57]

inducer(id_satR211,rco).
inputFile(id_satR211,'-').
dAte(id_satR211,day('-','Mar',15,2002),time(16 : 7 : 43)).
evaluatedAs(id_satR211,'UNORDERED').
dataFile(id_satR211,'-').

rule(id_satR211,1,
  [greater(volume,4.1),greater(pH,8.25)],
  class(tres),
  [0.021,0.000,0.645,0.333,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_satR211,2,
  [less(processamento,32.5),less(volume,4.2),less(concentracao_total,27.95)],
  class(um),
  [0.088,0.017,0.554,0.342,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_satR211,3,
  [greater(idade,36.5),greater(class_B,29.5)],
  class(um),
  [0.008,0.004,0.563,0.424,238.000],[0.000,0.000,1.000,0.000,2.000]).
rule(id_satR211,4,
  [greater(tempo_abs,1.5),less(class_C,65),less(vitalidade,53.5)],
  class(um),
  [0.094,0.021,0.562,0.322,233.000],[0.571,0.143,0.000,0.286,7.000]).
```

```
rule(id_satR211,5,
  [greater(pH,7.75),less(pH,8.25),less(concentracao_total,72.7),less(motilidade,12.5)],
  class(um),
  [0.150,0.026,0.556,0.269,234.000],[0.833,0.000,0.167,0.000,6.000]).
rule(id_satR211,6,
  [greater(class_C,34.5),less(kruger,1.5)],
  class(um),
  [0.069,0.017,0.562,0.352,233.000],[0.429,0.000,0.286,0.286,7.000]).
rule(id_satR211,7,
  [greater(vitalidade,70.5),greater(kruger,11.5),greater(hP,60)],
  class(tres),
  [0.104,0.014,0.615,0.267,221.000],[0.105,0.105,0.789,0.000,19.000]).
rule(id_satR211,8,
  [greater(concentracao_total,221.25),less(class_B,28.5),less(class_C,70.5)],
  class(tres),
  [0.167,0.058,0.592,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_satR211,9,
  [greater(concentracao,24),greater(motilidade,28.5)],
  class(tres),
  [0.113,0.013,0.638,0.238,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_satR211,10,
  [greater(idade,29),less(tempo_abs,6.5),less(hP,58.5)],
  class(um),
  [0.178,0.017,0.548,0.257,230.000],[0.300,0.300,0.400,0.000,10.000]).
rule(id_satR211,11,
  [less(pH,8.25),greater(concentracao,63.5),greater(motilidade,0.5)],
  class(tres),
  [0.145,0.068,0.577,0.209,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_satR211,12,
  [greater(class_B,25.5),greater(kruger,3.5)],
  class(dois),
  [0.039,0.034,0.742,0.185,233.000],[0.000,0.000,0.857,0.143,7.000]).
rule(id_satR211,13,
  [less(idade,48.5),less(concentracao,33.65),greater(class_B,7),less(hP,75.5)],
  class(um),
  [0.248,0.052,0.513,0.187,230.000],[0.300,0.000,0.700,0.000,10.000]).
rule(id_satR211,14,
  [],
  class(um),
  [],[]).
```

C.4 Classificador h_{sensR}

```
%Exemplos não cobertos: [3,5,11,14,15,16,19,20,21,22,23,
%25,31,34,35,36,39,40,41,42,50,51,54,57]

inducer(id_sensR211,rco).
inputFile(id_sensR211,'-').
dAte(id_sensR211,day('-','Mar',15,2002),time(16 : 8 : 22)).
evaluatedAs(id_sensR211,'UNORDERED').
dataFile(id_sensR211,'-').

rule(id_sensR211,1,
  [greater(volume,1.35),less(num_leu,1.28),less(hP,69)],
  class(um),
  [0.284,0.108,0.461,0.147,232.000],[0.375,0.500,0.125,0.000,8.000]).
rule(id_sensR211,2,
  [greater(concentracao_total,221.25),less(class_B,28.5),less(class_C,70.5)],
  class(tres),
  [0.167,0.058,0.592,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_sensR211,3,
  [less(idade,48.5),less(concentracao,33.65),greater(class_B,7),less(hP,75.5)],
  class(um),
  [0.248,0.052,0.513,0.187,230.000],[0.300,0.000,0.700,0.000,10.000]).
rule(id_sensR211,4,
  [less(pH,8.25),greater(concentracao,63.5),greater(motilidade,0.5)],
  class(tres),
  [0.145,0.068,0.577,0.209,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_sensR211,5,
  [greater(idade,31.5),less(processamento,35),greater(concentracao,20.35),less(class_A,
  class(tres),
  [0.084,0.046,0.609,0.261,238.000],[0.000,0.000,0.000,1.000,2.000]).
rule(id_sensR211,6,
  [],
  class(um),
  [],[]).
```

C.5 Classificador h_{specR}

```
%Exemplos não cobertos: [0,1,3,4,5,6,10,11,12,13,14,15,16,
%17,18,19,20,21,22,23,24,25,26,27,28,31,32,33,34,35,36,37,
%38,39,40,41,42,45,47,49,50,51,52,53,54,55,56,57,59]

inducer(id_specR211,rco).
inputFile(id_specR211,'-').
dAte(id_specR211,day('-','Mar',15,2002),time(16 : 8 : 45)).
evaluatedAs(id_specR211,'UNORDERED').
```

```
dataFile(id_specR211,'-').

rule(id_specR211,1,
  [greater(volume,4.1),greater(pH,8.25)],
  class(tres),
  [0.021,0.000,0.645,0.333,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_specR211,2,
  [greater(idade,36.5),greater(class_B,29.5)],
  class(um),
  [0.008,0.004,0.563,0.424,238.000],[0.000,0.000,1.000,0.000,2.000]).
rule(id_specR211,3,
  [greater(tempo_abs,1.5),less(class_C,65),less(vitalidade,53.5)],
  class(um),
  [0.094,0.021,0.562,0.322,233.000],[0.571,0.143,0.000,0.286,7.000]).
rule(id_specR211,4,
  [less(processamento,32.5),less(volume,4.2),less(concentracao_total,27.95)],
  class(um),
  [0.088,0.017,0.554,0.342,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_specR211,5,
  [],
  class(um),
  [],[]).
```

C.6 Classificador h_{supR}

```
%Exemplos não cobertos: [3,5,14,15,16,19,20,21,22,23,
%25,31,32,34,35,36,39,40,41,42,50,51,54,57,59]

inducer(id_supR211,rco).
inputFile(id_supR211,'-').
dAte(id_supR211,day('-','Mar',15,2002),time(16 : 9 : 10)).
evaluatedAs(id_supR211,'UNORDERED').
dataFile(id_supR211,'-').

rule(id_supR211,1,
  [greater(volume,1.35),less(num_leu,1.28),less(hP,69)],
  class(um),
  [0.284,0.108,0.461,0.147,232.000],[0.375,0.500,0.125,0.000,8.000]).
rule(id_supR211,2,
  [less(idade,48.5),less(concentracao,33.65),greater(class_B,7),less(hP,75.5)],
  class(um),
  [0.248,0.052,0.513,0.187,230.000],[0.300,0.000,0.700,0.000,10.000]).
rule(id_supR211,3,
  [greater(concentracao_total,221.25),less(class_B,28.5),less(class_C,70.5)],
```

```
  class(tres),
  [0.167,0.058,0.592,0.183,240.000],[0.000,0.000,0.000,0.000,0.000]).
rule(id_supR211,4,
  [greater(pH,7.75),less(pH,8.25),less(concentracao_total,72.7),less(motilidade,12.5)],
  class(um),
  [0.150,0.026,0.556,0.269,234.000],[0.833,0.000,0.167,0.000,6.000]).
rule(id_supR211,5,
  [less(pH,8.25),greater(concentracao,63.5),greater(motilidade,0.5)],
  class(tres),
  [0.145,0.068,0.577,0.209,234.000],[0.000,0.000,0.833,0.167,6.000]).
rule(id_supR211,6,
  [],
  class(um),
  [],[]).
```

Referências

Baranauskas, J. A. (2001). Extração automática de conhecimento por múltiplos indutores. Tese de Doutorado, ICMC-USP. http://www.teses.usp.br/teses/disponiveis/55/55134/tde-08102001-112806/.

Baranauskas, J. A. and Batista, G. E. A. P. A. (2000). O projeto DISCOVER: Idéias iniciais (comunicação pessoal).

Batista, G. E. A. P. A. (1997). Um ambiente de avaliação de algoritmos de aprendizado de máquina utilizando exemplos. Dissertação de Mestrado, ICMC-USP.

Batista, G. E. A. P. A. (2000). Pré-processamento de dados em aprendizado de máquina supervisionado. Monografia do Exame de Qualificação de Doutorado, ICMC-USP.

Batista, G. E. A. P. A. (2001). Sintaxe padrão do arquivo de exemplos do projeto DISCOVER. http://www.icmc.sc.usp.br/ gbatista/SintaxePadraoFinal.htm.

Bernardini, F. C. and Monard, M. C. (2002). Um módulo de explicação de *ensembles* e combinação de regras. Technical report, ICMC-USP. Em elaboração.

Caulkins, C. W. (2000). Aquisição de conhecimento utilizando aprendizado de máquina relacional. Dissertação de Mestrado, ICMC-USP.

Esteves, S. and F.C., B. (1998). Sperm processing for assisted reproductive technology (art): effects of pentoxifyline on recovery of motile sperm in asthenozoospermic men. *FertilSteril*, 70:5196.

Esteves, S., Lee, H., Monard, M., and Lopes, L. (2001). Inteligência artificial aplicada à andrologia: um estudo de caso do processamento de sêmen diagnóstico. In *XXVIII Congresso Brasileiro de Urologia*, Fortaleza, CE, Brasil.

Esteves, S., Sharma, R.K., Thomas, A.J.J., and Agarwal, A. (2000). Improvement in motion characteristis and acrosome status in cryopreserved human spermatozoa by swimup processing before freezing. *HumProd*, 15:2173–2179.

Félix, L. C. M. (1998). Data mining no processo de extração de conhecimento de bases de dados. Dissertação de Mestrado, ICMC-USP.

Gomes, A. K. (2001). Medidas de avaliação de regras. Exame de Qualificação de Mestrado, ICMC-USP.

Gomes, A. K., Bernardini, F. C., and Monard, M. C. (2002). Uma sintaxe padrão Prolog para classificadores simbólicos. Technical Report 154, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_154.ps.zip.

Gomes, A. K. and Monard, M. C. (2002a). Um estudo de caso utilizando o módulo de análise de regras do $\mathcal{R}_{ule}\mathcal{S}_{ystem}$. Technical Report 158, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_158.ps.zip.

Gomes, A. K. and Monard, M. C. (2002b). Um módulo de análise de regras extraídas de classificadores simbólicos. Technical Report 155, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_155.ps.zip.

Horst, P. S. (1999). Avaliação do conhecimento adquirido por algoritmos de aprendizado de máquina utilizando exemplos. Dissertação de Mestrado, ICMC-USP.

Imamura, C. Y. (2001). Pré-processamento para extração de conhecimento de bases textuais. Dissertação de Mestrado, ICMC-USP.

Lee, H. D. (2000). Seleção e construção de features relevantes para o aprendizado de máquina. Dissertação de Mestrado, ICMC-USP.

Lee, H. D. and Monard, M. C. (2000a). Indução construtiva guiada pelo conhecimento: Um estudo de caso do processamento de sinen diagnóstico. In *Proceedings IBERAMIA-SBIA 2000, Open Discussion Track*, pages 167–176, Atibaia, SP, Brasil.

Lee, H. D. and Monard, M. C. (2000b). A practical approach for knowledge-driven constructive induction. In *Procedings Argentine Symposium on Artificial Intelligence, ASAI'2000, 29th International Conference SADIO*, pages 71–86, Argentina.

Martins, C. A. (2001). Clustering conceitual em aprendizado de máquina. Exame de Qualificação de Doutorado, ICMC-USP.

Milaré, C. R. (2000). Extração de conhecimento de redes neurais. Exame de Qualificação de Doutorado, ICMC-USP.

Nagai, W. A. (2000). Avaliação do conhecimentos extraído de problemas de regressão. Dissertação de Mestrado, ICMC-USP.

Pila, A. D. (2000). Seleção de atributos relevantes para aprrendizado de máquina utilizando a abordagem de rough sets. Dissertação de Mestrado, ICMC-USP. http://www.teses.usp.br/teses/disponiveis/55/55134/tde-13022002-153921/publico/dissertacao_ADP.pdf.

Prati, R. C., Baranauskas, J. A., and Monard, M. C. (1999). BibView: Um sistema para auxiliar a manutenção de registros para o BibTeX. Technical Report 95, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_95.ps.zip.

Prati, R. C., Baranauskas, J. A., and Monard, M. C. (2001a). Extração de informações padronizadas para a avaliação de regras induzidas por algoritmos de aprendizado de máquina simbólico. Technical Report 145, ICMC-USP. `ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/RT_145.ps.zip`.

Prati, R. C., Baranauskas, J. A., and Monard, M. C. (2001b). Uma proposta de unificação da linguagem de representação de conceitos de algoritmos de aprendizado de máquina simbólicos. Technical Report 137, ICMC-USP. `ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/RT_137.ps.zip`.

Pugliesi, J. B. (2001). O pós-processamento em extração de conhecimento de bases de dados. Exame de Qualificação de Doutorado, ICMC-USP.