



Ministério da  
**Ciência e Tecnologia**



sid.inpe.br/mtc-m19/2011/06.10.18.36-TDI

## ÁRVORES DE DECISÃO APLICADAS AO PROBLEMA DA SEPARAÇÃO ESTRELA/GALÁXIA

Eduardo Charles Vasconcellos

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada,  
orientada pelos Drs. Haroldo Fraga de Campos Velho, e Reinaldo Ramos de  
Carvalho, aprovada em 15 de abril de 2011.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/39RNQ52>>

INPE  
São José dos Campos  
2011

**PUBLICADO POR:**

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

**CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):****Presidente:**

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

**Membros:**

Dr<sup>a</sup> Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr<sup>a</sup> Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr<sup>a</sup> Regina Célia dos Santos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Dr. Ralf Gielow - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr. Wilson Yamaguti - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr. Horácio Hideki Yanasse - Centro de Tecnologias Especiais (CTE)

**BIBLIOTECA DIGITAL:**

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Deicy Farabello - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

**REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:**

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

**EDITORAÇÃO ELETRÔNICA:**

Vivéca Sant´Ana Lemos - Serviço de Informação e Documentação (SID)



Ministério da  
**Ciência e Tecnologia**



sid.inpe.br/mtc-m19/2011/06.10.18.36-TDI

## ÁRVORES DE DECISÃO APLICADAS AO PROBLEMA DA SEPARAÇÃO ESTRELA/GALÁXIA

Eduardo Charles Vasconcellos

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada,  
orientada pelos Drs. Haroldo Fraga de Campos Velho, e Reinaldo Ramos de  
Carvalho, aprovada em 15 de abril de 2011.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/39RNQ52>>

INPE  
São José dos Campos  
2011

Dados Internacionais de Catalogação na Publicação (CIP)

---

Vasconcellos, Eduardo Charles.

V441a Árvores de decisão aplicadas ao problema da separação estrela/galáxia / Eduardo Charles Vasconcellos. – São José dos Campos : INPE, 2011.

xxvi+72 p. ; (sid.inpe.br/mtc-m19/2011/06.10.18.36-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2011.

Orientadores : Drs. Haroldo Fraga de Campos Velho, e Reinaldo Ramos de Carvalho.

1. Mineração de dados. 2. Árvores de decisão. 3. SDSS data. 4. Catálogo estrela-galáxia. I.Título.

CDU 004.62

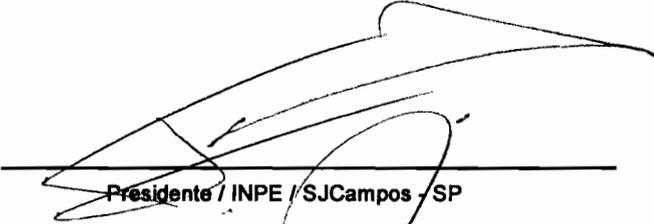
---

Copyright © 2011 do MCT/INPE. Nenhuma parte desta publicação pode ser reproduzida, armazenada em um sistema de recuperação, ou transmitida sob qualquer forma ou por qualquer meio, eletrônico, mecânico, fotográfico, reprográfico, de microfilmagem ou outros, sem a permissão escrita do INPE, com exceção de qualquer material fornecido especificamente com o propósito de ser entrado e executado num sistema computacional, para o uso exclusivo do leitor da obra.

Copyright © 2011 by MCT/INPE. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, microfilming, or otherwise, without written permission from INPE, with the exception of any material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use of the reader of the work.

Aprovado (a) pela Banca Examinadora  
em cumprimento ao requisito exigido para  
obtenção do Título de Mestre em  
Computação Aplicada

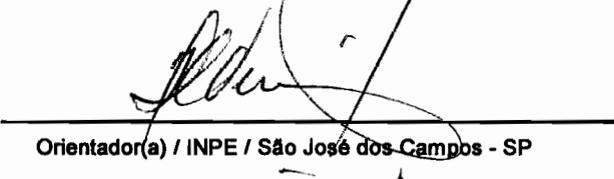
Dr. Horacio Hideki Yanasse



---

Presidente / INPE / SJCampos - SP

Dr. Haroldo Fraga de Campos Velho



---

Orientador(a) / INPE / São José dos Campos - SP

Dr. Reinaldo Ramos de Carvalho



---

Orientador(a) / INPE / SJCampos - SP

Dr. Reinaldo Roberto Rosa



---

Membro da Banca / INPE / SJCampos - SP

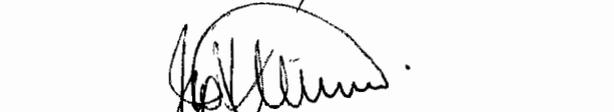
Dr. Hugo Vicente Capelato



---

Membro da Banca / INPE / SJCampos - SP

Dr. João Luiz Kohl Moreira



---

Convidado(a) / ON / Rio de Janeiro - RJ

Este trabalho foi aprovado por:

( ) maioria simples

(x) unanimidade

Aluno (a): Eduardo Charles Vasconcellos

São José dos Campos, 15 de abril de 2011



*“Try not. Do or do not, there is no try.”*

MASTER YODA  
em “*Star Wars - Episode V*”, 1980



*A meus pais Acir e Kátia, à minha esposa Fernanda e à  
memória de minha amada avó Elvira e de meu amigo e  
mentor Dr. Francisco C. de Araújo*



## **AGRADECIMENTOS**

Agradeço à minha esposa por todo apoio durante a produção deste trabalho; a meus pais por poder chegar até aqui; e aos meus amigos por estarem sempre por perto. Agradeço também, o auxílio do Dr. Roy Gal, Dr. Francesco L. LaBarbera e Dr. Hugo V. Capelato, que contribuíram com dados e sugestões ao longo do trabalho, e ao Dr. Nick Ball que forneceu seus próprios resultados para compararmos com os nossos. Por fim, agradeço aos meus orientadores Dr. Haroldo F. de Campos Velho e Dr. Reinaldo R. de Carvalho, que me guiaram durante todas as etapas da elaboração desta dissertação, e a CAPES por financiar meu trabalho através do curso de Computação Aplicada (CAP) do Instituto Nacional de Pesquisas Espaciais (INPE).



## RESUMO

Neste trabalho estudamos a eficiência de 13 algoritmos distintos para construção de árvores de decisão quando aplicados a tarefa de separar estrelas de galáxias. Esta separação é procedida com base nos dados fotométricos da sétima disponibilização de dados do “Sloan Digital Sky Survey” (SDSS-DR7). Cada um dos algoritmos testados está contido no software WEKA e é definido por um conjunto de parâmetros que, quando variados, podem modificar a árvore de decisão gerada. Nós exploramos extensivamente o espaço de parâmetros de cada algoritmo usando um conjunto de treinamento composto por 880.715 objetos do SDSS-DR7 que possuem espectroscopia disponível. Buscamos a configuração otimizada que permitiria a construção de uma árvore capaz de separar estrelas e galáxias com a maior precisão possível. A eficiência da separação estrela/galáxia é medida usando a função de completudeza (fração de galáxias classificadas corretamente). Nossos resultados mostraram que o algoritmo FT (*Functional Trees*) obteve os melhores resultados, com base na função de completudeza, para dois intervalos de magnitude:  $14 \leq r \leq 21$  (85.2%) e  $r \geq 19$  (82.1%). Comparamos o desempenho da árvore gerada pelo FT com sua melhor configuração com separações obtidas pelo método paramétrico do SDSS, pelo 2DPHOT e por Ball et al. (2006). Essa comparação mostrou que nossa árvore gerada com o FT tem, para magnitudes no intervalo  $14 \leq r < 19$ , um desempenho similar aos demais métodos em termos de completudeza, mas com uma contaminação (fração de estrelas classificadas como galáxias) muito inferior a todas menos a obtida por Ball et al. (2006). Para magnitudes fracas ( $r \geq 19$ ), nossa árvore é o único separador que obtém uma alta completudeza ( $>80\%$ ) e uma baixa contaminação ( $\sim 2.5\%$ ). Executamos também um experimento utilizando uma máquina de comitê composta por árvores de decisão treinadas com todos os 13 algoritmos do WEKA. Este experimento resultou em uma queda na função de completudeza de  $\sim 5\%$  para magnitudes maiores que  $19.5^m$  quando comparada à completudeza obtida pelo FT. Por outro lado, a amostra de galáxias gerada pelo comitê apresenta uma contaminação aproximadamente 6% menor que a gerada pelo FT. Por fim examinamos o separador paramétrico do SDSS (`psfMag - modelMag`) com intuito de verificar se a linha divisória que separa os objetos poderia ser mais precisa. Identificamos que muitos pares de estrelas próximas estão sendo classificados erroneamente como galáxias, e sugerimos um novo valor de corte para melhorar a precisão do separador do SDSS. Por fim, aplicamos nossa árvore de decisão gerada com o FT para separar estrelas de galáxias em todo o conjunto de 69.545.326 objetos da amostra fotométrica do SDSS-DR7 com magnitudes no intervalo  $14 \leq r \leq 21$ .



# DECISION TREES APPLIED AS STAR/GALAXY SEPARATOR

## ABSTRACT

We study the star/galaxy classification efficiency of 13 different decision tree algorithms applied to photometric objects in the Sloan Digital Sky Survey Data Release Seven (SDSS DR7). Each algorithm is defined by a set of parameters which, when varied, produce different final classification trees. We extensively explore the parameter space of each algorithm, using the set of 884,126 SDSS objects with spectroscopic data as the training set. The efficiency of star-galaxy separation is measured using the completeness function. We find that the Functional Tree algorithm (FT) yields the best results as measured by the mean completeness in two magnitude intervals:  $14 \leq r \leq 21$  (85.2%) and  $r \geq 19$  (82.1%). We compare the performance of the tree generated with the optimal FT configuration to the classifications provided by the SDSS parametric classifier, 2DPHOT and Ball et al. (2006). We find that our FT classifier is comparable or better in completeness over the full magnitude range  $15 \leq r \leq 21$ , with much lower contamination than all but the Ball et al. (2006) classifier. At the faintest magnitudes ( $r > 19$ ), our classifier is the only one that maintains high completeness ( $> 80\%$ ) while simultaneously achieving low contamination ( $\sim 2.5\%$ ). We carried out an experiment with a decision tree committee machine designed with trees trained with all thirteen WEKA algorithms. The result was: for magnitudes greater than  $20.5^m$ , in both a completeness  $\sim 5\%$  and a contamination  $\sim 6\%$  lower than our pure FT tree. Finally we examine the SDSS parametric classifier (`psfMag - modelMag`) to see if the dividing line between stars and galaxies can be adjusted to improve the classifier. We find that currently, stars in close pairs are often misclassified as galaxies, and suggest a new cut to improve the classifier. Finally, we apply our FT classifier to separate stars from galaxies in the full set of 69,545,326 SDSS photometric objects in the magnitude range  $14 \leq r \leq 21$ .



## LISTA DE FIGURAS

	<u>Pág.</u>	
1.1	Comparação visual entre estrelas e galáxias. A coluna da esquerda mostra 3 imagens de galáxias, e a coluna da direita 3 imagens de estrelas. Para efeito de comparação, os pares estrela/galáxia mostrados tem magnitudes similares ( $\Delta r \leq 0.5^m$ ). . . . .	2
1.2	A distribuição de $\log(\mathbf{Área})$ vs $\mathbf{MTot}$ nas seções das placas J380 e J442.	7
1.3	A figura mostra a distribuição de objetos em três espaços bidimensionais definidos por atributos fotométricos medidos pelo COSMOS: (a) $\mathbf{G}$ versus magnitude; (b) $\log \mathbf{A}$ versus magnitude; (c) $\mathbf{S}$ versus magnitude. . . . .	14
1.4	A distribuição do parâmetro de classificação $\psi$ para o campo 78 do UKSTU. As estrelas estão dispostas em uma faixa bem definida com $\psi \approx 0$ . As galáxias com magnitudes entre 9 e 13 ( $20.5 > B_j > 16.0$ ) jazem em uma faixa separada com $\psi > 1500$ . Próximo ao limite da placa, o seeing atmosférico borra qualquer estrutura na imagem, logo as duas faixas se confundem e nenhuma distinção pode ser feita para imagens mais fracas que 9 ( $B_j \gtrsim 20.5$ ). . . . .	15
2.1	Uma AD simples, construída com o algoritmo J48 (WITTEN; FRANK, 2000). Esta árvore foi treinada com 50.000 objetos da amostra espectroscópica, como descrito na Seção 2.5, e com um número mínimo de objetos por folha igual a 50. As possíveis classes aqui são 1 (estrela) ou 2 (galáxia).	18
2.2	A figura mostra exemplos de testes executados por uma AD sobre os dois tipos de atributos (numérico e nominal). A figura (a) mostra um exemplo de teste para um atributo nominal com três elementos. As figuras (b) e (c) mostram possíveis testes sobre um atributo numérico. As palavras sim e não representam a decisão de sair de casa. As classes são seguidas por números que representam quantos exemplos de cada uma existem naquela folha. Quando existir uma barra separando dois números, o primeiro representa o número de exemplos com a classe indicada, e número após a barra indica o número de exemplos da outra classe naquela folha. . . .	21
2.3	A figura mostra o espaço de atributos <b>temperatura X umidade</b> . A reta <b>umidade = 84</b> representa o teste proposto na 2.2(b) e divide o espaço de atributos em duas partes. . . . .	28

2.4	As figuras mostram: (a) uma AD treinada com o C4.5; (b) o espaço definido pelos atributos <b>A</b> e <b>B</b> com as respectivas divisões associadas a árvore apresentada em (a); (c) a expansão da árvore mostrada em (a) com a aplicação da técnica de <i>grafting</i> ; (d) o espaço definido pelos atributos <b>A</b> e <b>B</b> com as respectivas divisões associadas a árvore apresentada em (c). Os símbolos $\bullet$ , $*$ e $\diamond$ representam as três diferentes classes do conjunto de treinamento. . . . .	30
2.5	A figura mostra o espaço de atributos <b>Temperatura X Umidade</b> . A reta <b>Umidade</b> = 84 representa o teste proposto na 2.2(b) e representa um teste univariante. A reta <b>Umidade</b> = $-0.77 * \text{Temperatura} + 106.08$ representa um teste multivariante. Ambos dividem o espaço de atributos em duas regiões. . . . .	32
2.6	A figura mostra uma AD construída pelo WEKA com os exemplos apresentados na Tabela 2.1 usando ADTree. O texto “(-ve = sim, +ve = não)” contido no nó raiz significa que o sinal negativo no resultado da soma dos nós de previsão classifica uma instância como <i>sim</i> , e o sinal positivo como <i>não</i> . . . . .	36
2.7	Função resposta dos filtros do SDSS. A figura mostra a resposta de cada filtro usado pelo SDSS de acordo com o comprimento de onda. . . . .	39
2.8	Resultados obtidos para a exploração do espaço de parâmetros para cada um dos 13 algoritmos do WEKA. As áreas escuras são os lugares geométricos das funções de completeza obtidas com o procedimento de CV para cada conjunto de parâmetros testado. . . . .	47
2.9	Funções de completeza (curvas superiores) e contaminação (curvas inferiores) para todos os quatro conjuntos de dados perturbados usados para treinar um AD com o algoritmo FT. Cada conjunto de dado é representado por um tipo de linha diferente. . . . .	52
2.10	Funções de completeza (curvas superiores) e contaminação (curvas inferiores) para todos os cinco subconjuntos de dados retirados do conjunto total de treinamento e usados para treinar um AD com o algoritmo FT. O subconjunto de 240.712 objetos contém 100% dos objetos do conjunto total para $r > 19$ , enquanto que para $r \leq 19$ a distribuição de objetos por classe do conjunto total é utilizado. Os demais subconjuntos utilizam a distribuição de objetos por classe do conjunto total em todo o intervalo de magnitudes estudado [14, 21]. . . . .	53

2.11	A figura mostra a completeza (curvas superiores) e a contaminação (curvas inferiores) para uma amostra de 10.391 de objetos do SDSS reprocessados com o 2DPHOT e classificados unicamente com estrela ou galáxia. As linhas contínuas representam as funções de completeza e contaminação para nossa a classificação fornecida pela nossa AD, enquanto que a linha tracejada representa as mesmas funções para a classificação fornecida pelo 2DPHOT. . . . .	55
2.12	Completeza (curvas superiores) e contaminação (curvas inferiores) para 561.070 objetos classificados por Ball et al. (2006) e com espectroscopia fornecida pelo SDSS. As linhas contínuas mostram as funções de completeza e de contaminação para a classificação da nossa AD, enquanto que as linhas tracejadas mostram o mesmo para a classificação de BALL et al..	56
2.13	As funções de completeza (curvas superiores) e de contaminação (curvas inferiores) para o método paramétrico usado pelo <i>pipeline</i> do SDSS (linha tracejada) e para nosso método de AD (linha contínua) quando aplicados à separação de 880.715 objetos do conjunto espectroscópico (veja o texto).	57
2.14	Completeza e contaminação para o método paramétrico do SDSS quando assumimos que a classificação fornecida pela nossa AD é 100% correta. .	58
2.15	A figura mostra o espaço definido por dois atributos A e B. As semi-retas representam as divisões do espaço determinadas pelos testes propostos pelo algoritmo C4.5 (a) e pelo C4.5 mais o <i>graft</i> (b). Cada símbolo representa uma classe diferente. Para visualizar as árvores associadas à divisão desses espaços, o leitor pode consultar a Figura 2.4. . . . .	59
2.16	A figura mostra as curvas de completeza e contaminação para uma AD treinada com o FT e para um comitê de árvores treinadas com cada um dos 13 algoritmos testados do WEKA quando configurados com seus conjuntos de parâmetros otimizados. A linha cheia superior representa a função de completeza do FT e a inferior a de contaminação. As linhas tracejadas representam as funções de completeza (superior) e contaminação (inferior) do comitê. . . . .	60
2.17	A figura mostra nove campos celestes do SDSS DR7 para nove objetos que são classificados erroneamente pelo método paramétrico do SDSS. Quase todos estão sobrepostos a outros objetos ou possuem um ou mais vizinhos próximos mais brilhantes que afetam a fotometria. . . . .	62

2.18 O valor de  $psfMag - modelMag$  utilizado pelo classificador paramétrico do SDSS é representado como uma função da magnitude para os objetos do conjunto espectroscópico. As estrelas são apresentadas como pontos vermelhos, enquanto que as galáxias como pontos verdes. As linhas divisórias usadas pelo classificador SDSS ( $psfMag - modelMag = 0,145$ ) e a calculada pelo Decision Stump ( $psfMag - modelMag = 0,376$ ) também são mostradas. O classificador do SDSS incorretamente classifica muitas estrelas relativamente brilhantes como galáxias, a maioria das quais tem vizinhos próximos. . . . . 63

## LISTA DE TABELAS

	<u>Pág.</u>
2.1 Pequeno conjunto de exemplos de treinamento. Esse conjunto de dados relaciona condições meteorológicas com a decisão de sair ou não de casa.	20
2.2 Distribuição dos objetos do DR7 com espectroscopia (1.030.220). A distribuição é baseada na classificação espectral fornecida pelo CAS na forma do atributo <code>specClass</code> . O <code>specClass</code> possui seis diferentes classes: <code>unknown</code> , <code>star</code> , <code>galaxy</code> , <code>qso</code> , <code>hiz_qso</code> , <code>sky</code> , <code>star_late</code> e <code>gal_em</code> . A descrição foi adaptada da página <a href="http://cas.sdss.org/dr7/en/help/browser/browser.asp">http://cas.sdss.org/dr7/en/help/browser/browser.asp</a> .	42
2.3 Atributos SDSS-DR7 usados para separação estrela/galáxia.	44
2.4 Exploração do espaço de parâmetros para cada algoritmo do WEKA. As colunas são: o nome do algoritmo e o número de testes realizados, seus parâmetros, os intervalos estudados, o número de valores no intervalo, e o valor do parâmetro na melhor configuração testada. Devido aos limitados recursos computacionais disponíveis, somente analisamos as variações dos parâmetros contínuos com passos que causassem uma modificação no valor da função de completeza maior que 5%. Para os parâmetros nos quais somente um valor é exibido, qualquer variação somente modifica a função de completeza em menos de 5%, e o valor padrão do WEKA é usado.	64
2.5 Principais resultados do estudo comparativo realizado com os algoritmos do WEKA. As colunas são, respectivamente; os nomes dos algoritmos testados, o número de parâmetros do algoritmo em questão que causam uma modificação significativa da AD resultante; o tempo médio de processamento calculado sobre todos as combinações de parâmetros testadas; a completeza média calculada no intervalo de magnitudes [14, 19[; a completeza média calculada no intervalo de magnitudes [19, 21]; e a completeza no bin de magnitudes mais fracas ( $20.5 \leq r \leq 21.0$ ).	65

- 2.6 Classificação estrela/galáxia fornecida pelo SDSS e pela nossa AD gerada pelo FT. A coluna 1 lista a identificação única do SDSS e a coluna 2 a magnitude `modelMag` na banda  $r$ . As colunas 3 e 4 `typer` e `type` contém, respectivamente, a classificação paramétrica do SDSS utilizando somente a banda  $r$  e todas as cinco bandas. A coluna 5 mostra a classificação fornecida pela nossa AD, onde o número 1 representa estrelas e o 2 galáxias. 65
- 2.7 Classificação estrela/galáxia paramétrica fornecida pelo SDSS. A coluna 1 mostra as possíveis classificações, enquanto que a coluna 2 contém uma pequena descrição da respectiva classe. A separação estrela ou galáxia é baseada na diferença entre as magnitudes `psfMag` e `modelMag` (veja 2.7.3). 66

## LISTA DE ABREVIATURAS E SIGLAS

2MASS	–	Two Micron All Sky Survey
AAO	–	Anglo-Australian Observatory
APM	–	SERC Automatic Plate Measuring
CCD	–	Charge-coupled device
DES	–	Dark Energy Survey
DSS	–	Digitized Sky Survey
DR2	–	Data Release 2
DR3	–	Data Release 3
DR7	–	Data Release 7
ESO	–	European Southern Observatory
FOCAS	–	Faint Object Classification and Analysis System
LSST	–	Large Survey Telescope
MBAM	–	Método Baseado em Aprendizado de Máquina
PNSC	–	Palomar-Norris Sky Catalog
POSS	–	Palomar Observatory Sky Survey
SDSS	–	Sloan Digital Sky Survey
SKICAT	–	Sky Image Cataloging and Analysis Tool
SERC	–	Science and Engineering Research Council
TEH	–	Telescópio Espacial Hubble
UKSTU	–	UK Schmidt Telescope Unit
WEKA	–	Waikato Environment for Knowledge Analysis



## LISTA DE SÍMBOLOS

- Å – angstrom ( $1,0 \times 10^{-10}$  metros)
- Gb – gigabyte
- Tb – terabyte
- $1^m$  – uma unidade de magnitude



# SUMÁRIO

	<u>Pág.</u>
<b>1 INTRODUÇÃO</b> . . . . .	<b>1</b>
1.1 Levantamentos Astronômicos . . . . .	1
1.1.1 Levantamentos Astronômicos de Placas Fotográficas . . . . .	3
1.1.2 Levantamentos Astronômicos em Imagens Digitais . . . . .	5
1.2 Métodos de Classificação . . . . .	5
1.2.1 Métodos Paramétricos . . . . .	6
1.2.2 Métodos Não Paramétricos . . . . .	8
1.2.3 Métodos Baseados em Aprendizado de Máquina . . . . .	9
1.3 Objetivos . . . . .	11
<b>2 ÁRVORES DE DECISÃO APLICADAS AOS DADOS DO SDSS-DR7</b> . . . . .	<b>17</b>
2.1 O Que é Uma Árvores de Decisão? . . . . .	17
2.2 Construindo uma Árvore de Decisão . . . . .	19
2.3 O WEKA e Seus Algoritmos de Construção de Árvores de Decisão . . . . .	23
2.3.1 J48 . . . . .	26
2.3.2 J48graft . . . . .	28
2.3.3 BFTree . . . . .	31
2.3.4 FT . . . . .	31
2.3.5 LMT . . . . .	32
2.3.6 Simple Cart . . . . .	33
2.3.7 REPTree . . . . .	33
2.3.8 Random Tree . . . . .	34
2.3.9 Random Forest . . . . .	34
2.3.10 NBTree . . . . .	34
2.3.11 ADTree . . . . .	35
2.3.12 LADTree . . . . .	37
2.3.13 Decision Stump . . . . .	37
2.4 Precisão e Performance de um Algoritmo: o Método Cross-Validation . . . . .	37
2.5 Os Dados do SDSS-DR7 Utilizados . . . . .	39
2.6 Separação Estrela/Galáxia para o SDSS-DR7 . . . . .	43

2.6.1	Atributos . . . . .	44
2.6.2	Seleção do Melhor Algoritmo Aplicado a Separação Estrela/Galáxia . . . . .	48
2.6.3	Construindo a Árvore de Decisão Final . . . . .	50
2.7	Comparação com Outros Métodos de Separação Aplicados ao SDSS . . . . .	53
2.7.1	Algoritmo FT Versus o Método do 2DPHOT . . . . .	54
2.7.2	O Algoritmo FT Versus O Algoritmo Axis-Parallel . . . . .	54
2.7.3	O Algoritmo FT Versus o Método Paramétrico do SDSS . . . . .	55
2.8	Máquinas de Comitê . . . . .	57
2.9	Um Teste Simples do Método Paramétrico do SDSS . . . . .	61
<b>3</b>	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>67</b>
3.1	Trabalhos Futuros . . . . .	68
	<b>REFERÊNCIAS BIBLIOGRÁFICAS . . . . .</b>	<b>69</b>

# 1 INTRODUÇÃO

As últimas décadas têm assistido a uma verdadeira revolução na obtenção de dados astrofísicos, tanto em quantidade como em complexidade. Os modernos detectores digitais, que substituíram as placas fotográficas usadas até o final dos anos 80, foram os principais responsáveis por essa revolução. Além disso, os avanços na área de computação, principalmente na armazenagem e transmissão de dados, mudaram radicalmente o próprio desenvolvimento da astrofísica observacional, que atualmente, gera cerca de centenas de Terabytes de dados por ano. Vários projetos de construção de telescópios de grande porte estão sendo desenvolvidos e espera-se que nos próximos dez anos vários levantamentos fotométricos profundos sejam realizados. Neste contexto é que se coloca a necessidade crescente de se desenvolver novos métodos de tratamento e análise de imagens astronômicas.

Um dos elementos básicos da análise de imagens em astronomia é a separação entre objetos pontuais (estrelas) e extensos (galáxias). Esta separação torna-se cada vez mais difícil na medida em que examinamos objetos de menor fluxo (Figura 1.1). Por outro lado, objetos distantes e de baixo fluxo são os que trazem maior informação do ponto de vista da cosmologia observacional. Assim, temos um problema específico: quais métodos nos permitiriam separar estrelas e galáxias, mesmo em um conjunto de objetos com baixos níveis de fluxo, com uma confiabilidade tal que possamos, por exemplo, identificar aglomerados de galáxias no Universo distante com uma alta completude (fração de galáxias classificadas corretamente) e uma baixa contaminação (fração de estrelas classificadas como galáxias)<sup>1</sup>. Neste trabalho, examinamos o uso de um método computacional, mais especificamente árvores de decisão, para tratar o problema específico de separação estrela/galáxia, estabelecendo sua performance e limitações quando aplicado em grandes quantidades de dados.

## 1.1 Levantamentos Astronômicos

Os levantamentos astronômicos são, na atualidade, os maiores produtores de dados no campo da astrofísica. Um levantamento consiste na aquisição de dados, geralmente fotométricos, medidos a partir de imagens de campos celestes. Até a metade do século passado, as observações eram compostas de imagens tomadas de pequenas áreas (campos) do céu que, geralmente, compreendiam minutos de arco. Essas áreas continham dezenas de objetos e suas respectivas imagens eram armazenadas

---

<sup>1</sup>Definições mais formais da completude e da contaminação são encontradas na Seção 2.4.

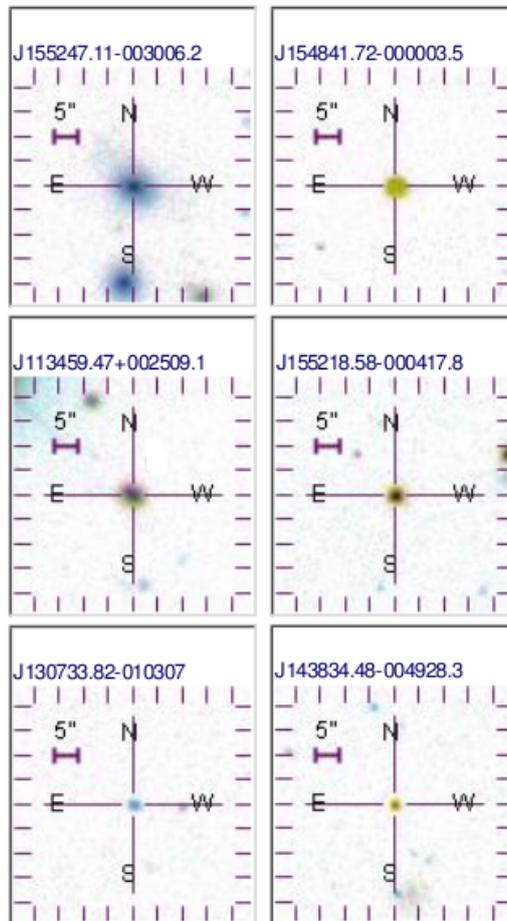


Figura 1.1 - Comparação visual entre estrelas e galáxias. A coluna da esquerda mostra 3 imagens de galáxias, e a coluna da direita 3 imagens de estrelas. Para efeito de comparação, os pares estrela/galáxia mostrados tem magnitudes similares ( $\Delta r \leq 0.5^m$ ).

Fonte: SDSS SkyServer (<http://cas.sdss.org/dr7/en/>).

em placas fotográficas. Com o desenvolvimento tecnológico, novos e mais potentes telescópios permitiram expandir as áreas observadas de minutos de arco para graus, que, juntamente com placas fotográficas de maior sensibilidade, possibilitaram um aumento do número de objetos contidos em uma imagem. Contudo, o grande salto nas observações astronômicas foi o desenvolvimento dos detectores digitais CCD (Charge-Couple Device). Os CCDs são capazes de obter imagens digitais em alta resolução, e possuem uma sensibilidade muitas ordens de grandeza maior que as antigas placas fotográficas. Os principais levantamentos astronômicos são apresentados nas próximas seções e estão divididos em duas partes: levantamentos feitos a partir

de imagens de placas fotográficas e feitos a partir imagens digitais.

### 1.1.1 Levantamentos Astronômicos de Placas Fotográficas

O primeiro levantamento astronômico foi inicializado em 1949 utilizando o telescópio de 1,2 metros do Observatório Palomar, sendo denominado “Palomar Observatory Sky Survey” (POSS). Em sua primeira fase (POSS-I), o projeto gerou 935 pares de placas fotográficas, cada uma contendo uma imagem de um campo celeste de  $6,5^\circ \times 6,5^\circ$ . Essas observações foram feitas em duas bandas fotométricas<sup>2</sup> (uma placa para cada banda) correspondentes ao azul e vermelho (MINKOWSKI; ABELL, 1963).

Em 1970, estimulado pelo sucesso do POSS-I, surgiu o “ESO/SERC Southern Sky Survey”. O projeto foi conduzido pelo telescópio British Schmidt de 1,2 metros do Observatório Anglo-Australiano (em inglês AAO, *Anglo-Australian Observatory*), na Austrália, e pelo telescópio Schmidt de 1,0 metro do “European Southern Observatory” (ESO), no Chile.

O último levantamento astronômico a utilizar placas fotográficas foi a segunda fase do projeto POSS (POSS-II). O projeto foi conduzido ao longo da década de 80 e contava com placas fotográficas mais sensíveis que as utilizadas no POSS-I. Foram observados 894 campos celestes de  $6,5^\circ \times 6,5^\circ$  nas bandas correspondentes ao azul, vermelho e infravermelho (REID et al., 1991).

Apesar de terem contribuído para o desenvolvimento da astronomia da época, os projetos POSS e ESO/SERC Southern Sky Survey não produziram uma quantidade de dados significativa. As medidas de características dos objetos presentes nas imagens geradas por estes projetos eram feitas a olho nu, por comparação com padrões previamente estabelecidos, tornando o processo de obtenção de dados lento e impreciso. Estes levantamentos começaram a gerar um volume de dados realmente expressivo a partir do final da década de 80, com a ajuda de aparelhos chamados microdensitômetros. Os microdensitômetros eram capazes de digitalizar as imagens contidas nas placas fotográficas. As imagens digitalizadas puderam então ser processadas por computadores que mediam as características dos objetos com maior velocidade e precisão.

---

<sup>2</sup>Uma banda fotométrica consiste em um determinado intervalo de comprimentos de onda no qual um fóton pode ser capturado pelo detector.

Dois importantes microdensitômetros foram a APM e o COSMOS. A APM (*Automatic Plate Measuring Machine*) da Universidade de Cambridge na Inglaterra foi utilizada para digitalizar 185 placas do ESO/SERC Southern Sky Survey. O processamento dessas digitalizações permitiu a determinação de parâmetros que caracterizavam a forma e o perfil de brilho para aproximadamente 20 milhões de objetos com magnitudes  $B \leq 22^m$  (MADDOX et al., 1990). O COSMOS<sup>3</sup> foi utilizado em um número maior de trabalhos que a APM, atendendo de forma mais significativa a comunidade astronômica. O trabalho de maior destaque utilizando o COSMOS foi a criação do “Edinburg/Durham Southern Galaxy Catalog” com base em digitalizações do ESO/SERC Southern Sky Survey (HEYDON-DUMBLETON et al., 1989). Contudo, tanto a APM quanto o COSMOS foram utilizados para cumprir propósitos específicos e não atenderam de forma abrangente a comunidade astronômica.

O primeiro projeto a fornecer digitalizações de alta-definição no ótico<sup>4</sup> abertamente para a comunidade científica foi o “Digitized Sky Survey” (DSS). A quantidade total de dados gerados pela versão original do DSS (DSS-I) chegou a 600Gb. Esses dados foram obtidos a partir de digitalizações das placas da banda azul do POSS-I e do ESO/SERC Southern Sky Survey. A segunda fase do projeto (DSS-II) utilizou placas do POSS-II nas três bandas, gerando por volta de 5Tb de dados (LASKER et al., 1996), possibilitando o estudo das regiões observadas em diferentes bandas fotométricas. Os dados do DSS-I e DSS-II estão disponíveis na url <http://archive.stsci.edu/dss/>.

O “Digitized Palomar Sky Survey” (DPOSS) foi outro projeto dedicado a digitalizar placas do POSS-II e disponibilizar os dados. A principal diferença entre o DPOSS e o DSS-II está na calibração fotométrica, que no caso do DPOSS é feita por meio de imagens CCD de observações conduzidas no Observatório Palomar. O DPOSS inclui um banco de dados com aproximadamente 3Tb de dados e diversos catálogos, sendo o mais importante deles o catalogo PNSC (*Palomar-Norris Sky Catalog*), contendo todos os objetos observados pelo POSS até a magnitude limite  $B \approx 22^m$ .

---

<sup>3</sup>O nome COSMOS vem das palavras da língua inglesa “Co-Ordinates”, “Sizes”, “Magnitudes”, “Orientations” e “Shapes”, ou respectivamente: coordenadas, tamanhos, magnitudes, orientações e formas. Essas palavras fazem referência as informações que a máquina COSMOS podia obter de um objeto contido em uma imagem do céu.

<sup>4</sup>O ótico refere-se ao intervalo de comprimentos de onda entre 40Å e 70Å.

### 1.1.2 Levantamentos Astronômicos em Imagens Digitais

A substituição das antigas placas fotográficas pelos detectores CCD, marcou uma nova fase na astronomia, inclusive nos levantamentos astronômicos. Esses detectores fornecem imagens digitais de alta resolução e não sofrem com os problemas intrínsecos das placas fotográficas, como baixa sensibilidade, faixa dinâmica restrita e não-linearidade. Os CCDs, juntamente com os novos e mais potentes telescópios, têm permitido aos astrônomos observar objetos cada vez mais distantes da Terra.

O 2MASS (Two Micron All Sky Survey) é um importante levantamento astronômico desta nova fase. Ele é o resultado de um esforço conjunto da Universidade de Massachusetts e do “Infrared Processing and Analysis Center at Caltech (Californian Institute of Technology)”. O 2MASS é um projeto de levantamento puramente fotométrico cobrindo todo o céu nos filtros J ( $1, 25\mu\text{m}$ ), H ( $1, 65\mu\text{m}$ ) e  $K_s$  ( $2, 17\mu\text{m}$ ) (SKRUTSKIE et al., 1997). As observações foram conduzidas com uso de dois telescópios robóticos de 1,3 metros; um no Arizona, EUA, e outro no Chile. O projeto foi de 1997 a 2001, produzindo  $\sim 10\text{Tb}$  de dados, entre imagens e catálogos.

No entanto, o projeto mais importante dessa nova fase é o “Sloan Sky Digital Survey” (SDSS; YORK et al., 2000). O SDSS foi inicializado em 2000 e já está na sua terceira etapa (SDSS-I, 2000-2005; SDSS-II, 2005-2008) que deve ser finalizada em 2014. Ele é o maior fornecedor de dados astronômicos da atualidade e constitui a fonte de dados deste trabalho, sendo abordado com mais detalhes na Seção 2.5.

## 1.2 Métodos de Classificação

Os dados fornecidos pelos levantamentos astronômicos têm possibilitado estudos cada vez mais profundos do universo. Esses dados são compostos por atributos medidos a partir da imagem de um objeto celeste, tais como fluxo e brilho superficial. A primeira etapa na análise desses dados consiste na determinação da natureza do objeto observado, ou seja, é preciso determinar se o objeto em questão é uma estrela ou uma galáxia. Weir et al. (1995) afirmam que a precisão na separação de objetos em estrelas e galáxias limita a qualidade e a profundidade<sup>5</sup>, em termos de utilidade científica, de um levantamento astronômico.

---

<sup>5</sup>A profundidade de um levantamento astronômico está relacionada com o limite superior das magnitudes dos objetos observados. Um levantamento é dito mais profundo quanto maior o seu limite superior de magnitudes observadas. Quanto mais profundo, maior é a capacidade do levantamento de observar objetos distantes.

Na época das placas fotográficas, a separação estrela/galáxia era completamente subjetiva, cabendo a um astrônomo a classificação de um objeto por inspeção visual (veja Figura 1.1). Contudo, as imagens digitais fornecidas pelos microdensitômetros e posteriormente pelas câmeras CCD lançaram um novo desafio na separação de objetos. Nos últimos 25 anos, muitos trabalhos, como os de MacGillivray et al. (1976), Heydon-Dumbleton et al. (1989) e Maddox et al. (1990), aplicam métodos paramétricos para separar estrelas e galáxias. Já outros, como os de Sebok (1979), Jarvis e Tyson (1981) e Weir et al. (1995), utilizam métodos não-paramétricos.

### 1.2.1 Métodos Paramétricos

A abordagem clássica para tratar o problema da separação estrela/galáxia em imagens digitais, ou digitalizadas, consiste na construção de um gráfico contendo o posicionamento de cada objeto em um espaço bi-dimensional definido por dois de seus atributos medidos. Um desses atributos é, em geral, a magnitude do objeto. Neste tipo de gráfico é possível definir um lugar geométrico onde se encontram as fontes pontuais (estrelas), como mostra a Figura 1.2. Dessa maneira, pode-se utilizar uma função discriminante para separar estrelas e galáxias. A maioria dos trabalhos publicados até o início da década de 90 utiliza esse método.

MacGillivray et al. (1976) utilizaram um gráfico definido por dois atributos medidos em imagens digitalizadas pela COMOS para separar 220 objetos. Os autores obtiveram 90% de completeza e  $\sim 10\%$  de contaminação, mas não definem a faixa de magnitudes dos objetos utilizados.

Heydon-Dumbleton et al. (1989) construíram gráficos de magnitude versus outros atributos medidos pelo COSMOS para 200 placas fotográficas do “ESO/SERC Southern Sky Survey”. Os autores dividiram o espaço de magnitudes em três intervalos de forma a otimizar a separação. Para objetos brilhantes, eles utilizaram o parâmetro geométrico  $\mathbf{G}$ , definido para medir a eficiência com a qual a imagem do objeto reproduz a elipse definida pelos seus semi-eixos maior e menor. Para magnitudes medianas foi usado o logaritmo na base decimal da área da imagem em pixels ( $\log \mathbf{A}$ ). Finalmente, para magnitudes mais fracas é empregado o parâmetro  $\mathbf{S}$ , definido como a largura do ajuste gaussiano à distribuição de luz do objeto. Cada um destes parâmetros é plotado contra a magnitude, como mostra a Figura 1.3, retirada

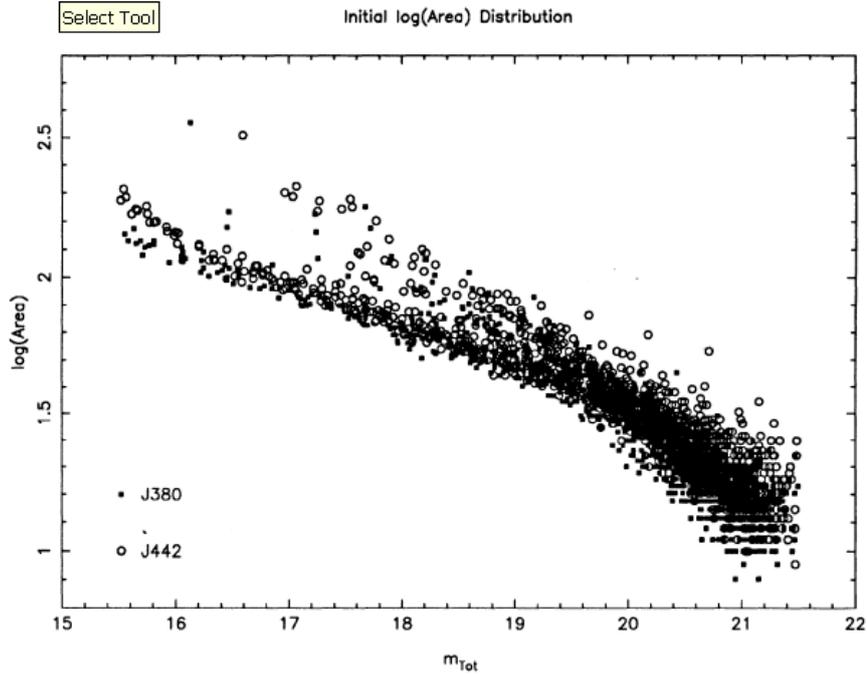


Figura 1.2 - A distribuição de  $\log(\text{Área})$  vs  $m_{\text{Tot}}$  nas seções das placas J380 e J442.  
 Fonte: Adaptada de (WEIR et al., 1995).

do referido trabalho. Para tentar resolver a questão da subjetividade na escolha da função discriminante, os autores aplicaram um método mais objetivo para determinar o lugar geométrico das estrelas. Um conjunto de pontos é selecionado ao longo do intervalo de magnitudes e em cada ponto uma distribuição estatística é feita para o parâmetro em questão. A partir desta distribuição o ponto de separação é escolhido. Por fim, a curva de separação é ajustada por uma spline cúbica que interpola os pontos de separação. Os autores conseguiram atingir em média uma fração de completude de  $98 \pm 2\%$  com uma contaminação de  $8 \pm 2\%$  para  $\log \mathbf{A}$  e  $9 \pm 2\%$  para  $\mathbf{S}$ . Esses resultados foram obtidos para um limite de magnitude aproximadamente igual a  $18.6^m$ .

Utilizando a APM, Maddox et al. (1990) digitalizaram aproximadamente 600 placas fotográficas contendo imagens capturadas pelo UK Schmidt Telescope Unit (UKSTU) na Austrália. Para separar estrelas e galáxias, os autores fizeram uso de dez atributos medidos pela APM: densidade de pico, raio de giro e a área da imagem calculada sobre oito diferentes superfícies de brilho. A cada um destes atributos foi associada uma variável  $p_i$  de modo que  $p_i = A_i$  se  $1 \leq i \leq 8$  (área),  $p_9 = P$  (den-

sidade de pico)  $p_{10} = K$  (raio de giro). Um gráfico  $p_i$  pela magnitude foi construído para cada  $i$  e o ponto de separação localizado em função da magnitude, resultando na função discriminante  $p_i^*(m)$ . A partir da diferença entre o valor de  $p_i^*(m)$  e o valor do respectivo parâmetro  $p_i$  na magnitude  $m$  é calculado o perfil residual  $\psi$  definido como

$$\psi = 1000 \log \left\{ \sum_{i=1}^{10} w_i(m) [p_i(m) - p_i^*(m)]^2 \right\}, \quad (1.1)$$

onde  $w_i = 1/\delta_i^2$  é chamado peso residual, e  $\delta_i$  é o erro estimado de  $p_i$ . Com o perfil residual  $\psi$  (Figura 1.4) os autores conseguiram atingir uma completude de 90% com uma contaminação por volta de 10% para magnitudes  $B_j$  menores que  $20.5^m$ .

### 1.2.2 Métodos Não Paramétricos

Outra abordagem para o problema da separação estrela/galáxia, são os métodos ditos não paramétricos. Esse tipo de abordagem caminhou junto com a abordagem paramétrica, mas só ganhou força a partir da década de 90 com a aplicação de técnicas de aprendizado de máquina ao problema em questão.

No final da década de 70, [Sebok \(1979\)](#) propôs um classificador baseado na teoria Bayesiana de reconhecimento de padrões. Um funcional numérico foi construído com base em um modelo estatístico contendo a probabilidade de o objeto ser uma estrela ou uma galáxia. O valor do funcional indicava a classificação do objeto.

[Jarvis e Tyson \(1981\)](#) utilizaram a idéia de agrupamento na implementação da ferramenta FOCAS (“Faint Object Classification and Analysis System”). Segundo o método proposto pelos autores, a classificação é feita em um espaço multidimensional formado por sete diferentes atributos medidos para cada objeto estudado. Desta forma, cada objeto é caracterizado por um vetor dentro deste espaço, podendo ser classificado como estrela, galáxia ou ambíguo. O método separa um grupo de estrelas usando uma abordagem paramétrica, como descrita na Seção 1.2.1. Este conjunto de estrelas é então separado em grupos com auxílio de um algoritmo de agrupamento<sup>6</sup>, sendo cada grupo classificado com base no número de estrelas que possui. Os demais objetos são alocados em cada um destes grupos com base na distância euclidiana entre a posição do objeto no espaço de atributos e a posição do centro estimado do grupo.

---

<sup>6</sup>No referido trabalho, o algoritmo utilizado foi, segundo os autores, uma variação do algoritmo ISODATA ([BALL; HALL, 1967](#)).

### 1.2.3 Métodos Baseados em Aprendizado de Máquina

Os métodos baseados em aprendizado de máquina (MBAMs) começaram a ser aplicados ao problema de separação estrela/galáxia no final do século passado. Estes métodos, como citado por Weir et al. (1995), podem trabalhar com um número relativamente grande de parâmetros de entrada, permitindo a criação de um classificador mais complexo e preciso do que os baseados em métodos paramétricos. Contudo, no atual estado em que se encontram os dados astronômicos, os MBAMs deixaram de ser métodos alternativos para ocupar um lugar de destaque na separação estrela/galáxia.

Os levantamentos astronômicos digitais vêm produzindo, de acordo com Donalek (2006), uma quantidade de dados que cresce exponencialmente em volume, complexidade e qualidade. Donalek ainda afirma que em 2006 os levantamentos estavam produzindo um volume de dados que dobrava a cada ano. Hoje, somente a sétima disponibilização de dados do SDSS, o SDSS-DR7 (<http://www.sdss.org/dr7/>), provê um volume de dados de aproximadamente 60.5Tb. A análise desses dados por meio dos métodos paramétricos não seria eficiente. A aplicação destes seria contrária ao grande esforço que tem havido nos últimos anos para aumentar a quantidade de medidas para um dado objeto (o SDSS-DR7 possui dezenas), aumentando assim a quantidade de informação sobre o mesmo, pois os métodos paramétricos estariam restritos a analisar apenas duas dessas medidas de cada vez, deixando de lado a informação contida nas demais. Por esse motivo, os MBAMs, que podem trabalhar com todas essas medidas e extrair regras de classificação complexas que não seriam detectadas por humanos (WITTEN; FRANK, 2000), vêm sendo largamente utilizados para criação de separadores estrela/galáxia.

Um dos primeiros trabalhos publicados utilizando MBAMs como separador estrela/galáxia foi Weir et al. (1995). Os autores aplicaram um método baseado em árvores de decisão (AD; QUINLAN, 1986; QUINLAN, 1993) como um separador automatizado que foi implementado no “Sky Image Cataloging and Analysis Tool” (SKICAT) e aplicado nos dados do DPOSS. Os autores usaram os algoritmos GID\*3 (FAYYAD, 1994) e o O-Btree (FAYYAD; IRANI, 1992) para construção da árvore, obtendo uma separação estrela/galáxia com 90% de completeza e 10% de contaminação. WEIR et al. também testaram redes neurais<sup>7</sup> do tipo Perceptron de duas camadas utilizando

---

<sup>7</sup>Tipo de algoritmo de aprendizado de máquina inspirado no mecanismo de aprendizado do cérebro humano.

os algoritmos de treinamento “backpropagation”, “conjugate gradient optimization” e “variable metric optimization”. Os resultados mostraram que as redes neurais testadas apresentavam valores de completeza e contaminação similares aos da AD, mas demandavam um tempo consideravelmente maior de treinamento.

Odewahn et al. (1999) aplicaram redes neurais para separar de objetos do DPOSS. Os autores utilizaram imagens CCD (imagens digitais obtidas com uma câmera CCD) de 550 campos observados com o telescópio Palomar de 1,5m para obter um conjunto de treinamento. Eles estabeleceram catálogos de estrelas e galáxias em 1000 graus quadrados do céu. Mais tarde, Odewahn et al. (2004), em uma terceira classificação dos dados DPOSS, utilizaram 3000 estrelas e 4000 galáxias classificadas com o FOCAS em 52 imagens CCD profundas<sup>8</sup> para treinar uma rede neural e uma árvore de decisão. O trabalho classificou objetos em 341 campos do DPOSS cobrindo um total de 7756 graus quadrados em ambos os hemisférios galácticos. Essa separação deu origem ao PNSC. Os resultados do trabalho mostraram um desempenho similar entre a rede neural e a árvore de decisão na separação estrela/galáxia, como já havia sido constatado por Weir et al. (1995).

Suchkov et al. (2005) foram os primeiros a usar um classificador baseado em AD para separar objetos do SDSS. Os autores aplicaram um classificador composto por 10 ADs oblíquas, construídas com o algoritmo OC1 (MURTHY et al., 1994), aos dados fotométricos da segunda disponibilização de dados do SDSS (SDSS-DR2; ABAZAJIAN et al., 2004). As ADs foram treinadas com 63.852 objetos da amostra espectroscópica do SDSS-DR2. Cada AD foi treinada para separar os objetos em 25 classes diferentes, sendo elas: estrelas, estrelas frias (“red stars”), 10 classes diferentes de galáxias, obtidas separando-se as galáxias da amostra espectroscópica do SDSS-DR2 em 10 intervalos de *redshift*<sup>9</sup>; e 13 classes de AGN (“Active Galactic Nuclei”, ou núcleo ativo de galáxia), separando-se os AGNs em 13 intervalos de “redshift”, como feito para as galáxias. A classificação de um objeto é feita de forma independente por cada AD e a classificação final é obtida por um esquema de votação baseado no trabalho de (WHITE et al., 2000). Os autores relatam a obtenção de classificações corretas para 98.1% das estrelas, 98.5% das galáxias, e 96.5% dos AGNs, em uma sub-amostra de 20.000 objetos retirada da amostra espectroscópica do SDSS-DR2.

---

<sup>8</sup>Imagens profundas, é um termo largamente utilizado em astronomia para imagens contendo objetos muito distantes da Terra, em geral objetos fora da nossa galáxia.

<sup>9</sup>valor numérico que expressa o desvio para maiores comprimentos de onda observado nas linhas espectrais. Esse desvio é causado pelo movimento relativo dos objetos observados em relação à Terra (Efeito Doppler).

Mais recentemente, Ball et al. (2006) aplicaram uma *axis-parallel decision tree*<sup>10</sup> na separação de objetos da terceira disponibilização de dados do SDSS (SDSS-DR3; ABAZAJIAN et al., 2005). Este foi o primeiro trabalho a aplicar este método a uma disponibilização de dados completa do SDSS. Os autores treinaram a árvore usando 477.068 objetos com classificação estrela/galáxia baseada em espectroscopia. Foram classificados 143 milhões de objetos fotométricos gerando um catálogo completo para a DR3. Os dados foram analisados em um espaço de parâmetros composto pelos índices de cor<sup>11</sup> baseados nas magnitudes disponíveis no banco de dados do SDSS<sup>12</sup>. Segundo os autores, o separador apresentou uma completude de 93.8% para galáxias e 95.4% para estrelas.

### 1.3 Objetivos

Neste trabalho, empregamos o método de AD para separar objetos da sétima disponibilização de dados do SDSS (SDSS-DR7) em estrelas e galáxias. Avaliamos 13 diferentes algoritmos para construção de ADs disponíveis com a ferramenta de mineração de dados WEKA (*Waikato Environment for Knowledge Analysis*) na versão 3.6.0. Um conjunto de dados de treinamento contendo somente objetos SDSS-DR7 com espectroscopia disponível foi utilizado na construção das árvores. O algoritmo com o melhor desempenho nos testes sobre o conjunto de treinamento foi então utilizado para separar todos os objetos do SDSS-DR7 Legacy<sup>13</sup> com magnitudes  $14 \leq r \leq 21$ , onde somente os dados fotométricos estão disponíveis. Este foi o primeiro trabalho publicado a testar tamanha variedade de algoritmos usando o volume total de dados do SDSS-DR7. Este trabalho é uma extensão do estudo de Ruiz et al. (2009), que fez uso somente do algoritmo J48 no desenvolvimento de classificadores para separar estrelas e galáxias com base em atributos fotométricos utilizando dados do SDSS-DR6.

Aumentar a precisão da separação estrela/galáxia no limite profundo de magnitudes mais fracas (objetos distantes) dos mapeamentos celestes, não é meramente um exercício acadêmico. Com uma elevação na completude nas amostras de galáxias fracas (valores elevados de magnitude), e uma redução na contaminação por estre-

---

<sup>10</sup>tipo padrão de algoritmo de AD onde cada nó de decisão contém testes sobre um único atributo. Para mais informações sobre ADs o leitor pode ler o Capítulo 2

<sup>11</sup>Um índice de cor nada mais é do que a diferença entre as magnitudes medidas em duas bandas distintas.

<sup>12</sup>O SDSS disponibiliza quatro diferentes magnitudes: PSF, petrosiana, modelo e de fibra.

<sup>13</sup>O Legacy é a parte do SDSS destinada à coleta de dados fotométricos de objetos celestes. Para mais informações veja <http://www.sdss.org/dr7/>.

las erroneamente classificadas, muitas questões astrofísicas importantes podem ser estudadas com uma maior confiabilidade. O mapeamento da assinatura de oscilações acústicas bariônicas, requer uma amostra com grande número de galáxias - e quanto mais completa em altos “redshifts” melhor. A medida das funções de correlação galáxia-galáxia pode ser melhorada, tanto pelo aumento do número de galáxias usadas como pela redução do ruído do sinal vinculada a suavização da distribuição de estrelas erroneamente classificadas. Mapeamentos de lentes **gravitacionais** fracas necessitam de uma amostra de galáxias de fundo (galáxias cuja radiação eletromagnética foi desviada pela lente) numerosa e o mais purificada possível (valores baixos de contaminação). De forma similar, a busca por aglomerados de galáxias por meio de superdensidade de galáxias também tem sua eficiência aumentada por uma amostra galáctica ampla e com baixa contaminação. Buscas por objetos raros, tanto estelares como extensos, também é melhorado com uma contaminação reduzida, assim como qualquer programa de seleção de objetos para espectroscopia baseado no tipo de fonte emissora.

Para os mapeamentos futuros, os classificadores otimizados irão exigir um novo tipo de conjunto de treinamento. Projetos como o “Dark Energy Survey” (DES), o “Large Survey Telescope” (LSST) e o **“Panoramic Survey Telescope & Rapid Response System”** (Pan-STARRS) cobrirão grandes áreas do céu, podendo utilizar toda espectroscopia disponível para criar conjuntos de treinamento, mesmo em magnitudes muito fracas. Sendo que a maioria espectroscopia tem como alvo as galáxias, a inclusão de estrelas deve ser feita de outra maneira. As imagens do Telescópio Espacial Hubble (TEH) possuem uma resolução excelente e podem ser usadas para determinar a classe morfológica (estrela ou galáxia) de quase todos os objetos observados pelo TEH. Embora as observações do TEH abranjam apenas uma pequena fração do céu, a profundidade de suas imagens juntamente com a área observada pelos grandes projetos de mapeamento proverão conjuntos de treinamento estrela/galáxia (e talvez até de morfologia galáctica) que serão mais do que suficientes para implementação de classificadores como o que desenvolvemos neste trabalho.

A estrutura desta dissertação é a seguinte: nas Seções 2.1 à 2.3 descrevemos as ADs e introduzimos os algoritmos do WEKA testados; na Seção 2.4 explicamos os critérios utilizados na escolha do melhor algoritmo para construção do classificador final e nos testes de comparação; na Seção 2.6, discutimos o processo de avaliação e os

resultados obtidos para cada um dos algoritmos testados; na Seção 2.7 é apresentada uma comparação entre a separação estrela/galáxia desempenhada pela AD criada com o algoritmo de melhor performance, o método de separação paramétrico do SDSS (YORK et al., 2000), o método paramétrico do 2DPHOT (BARBERA et al., 2008) e a AD *axis-parallel* de Ball et al. (2006); na Seção 2.8 descrevemos um experimento com máquinas de comitê realizado com os algoritmos; por fim, resumimos nossos resultados no Capítulo 3.

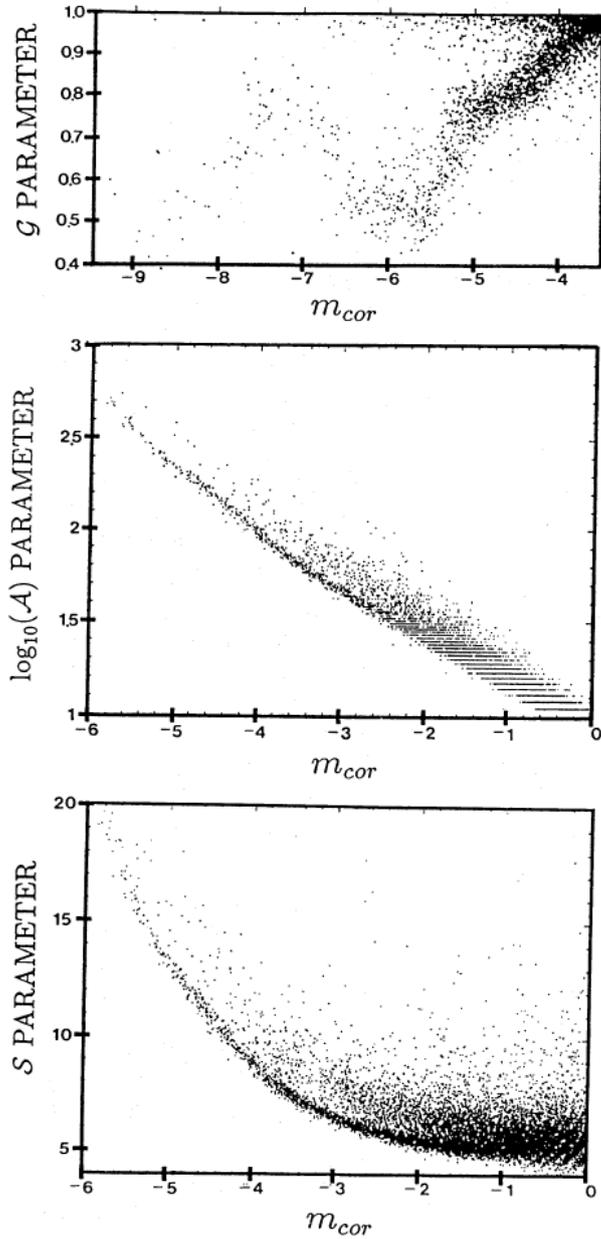


Figura 1.3 - A figura mostra a distribuição de objetos em três espaços bidimensionais definidos por atributos fotométricos medidos pelo COSMOS: (a) G versus magnitude; (b)  $\log A$  versus magnitude; (c) **S** versus magnitude.

Fonte: Adaptada de (HEYDON-DUMBLETON et al., 1989).

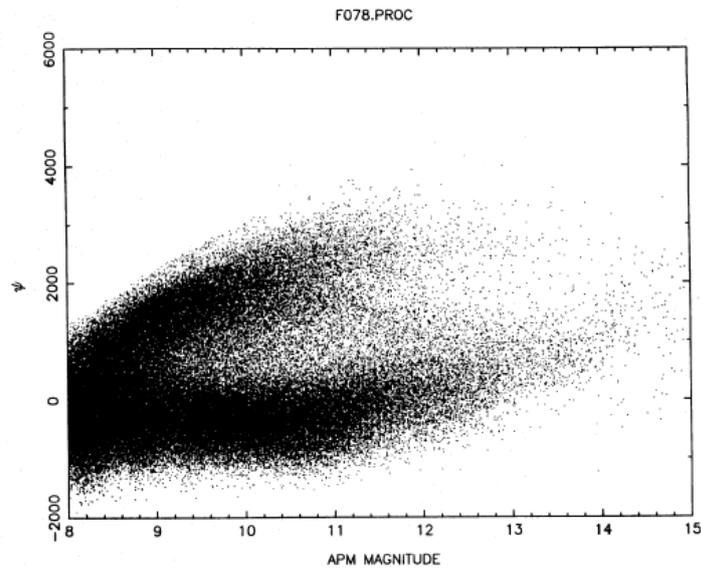


Figura 1.4 - A distribuição do parâmetro de classificação  $\psi$  para o campo 78 do UKSTU. As estrelas estão dispostas em uma faixa bem definida com  $\psi \approx 0$ . As galáxias com magnitudes entre 9 e 13 ( $20.5 > B_j > 16.0$ ) jazem em uma faixa separada com  $\psi > 1500$ . Próximo ao limite da placa, o seeing atmosférico borra qualquer estrutura na imagem, logo as duas faixas se confundem e nenhuma distinção pode ser feita para imagens mais fracas que 9 ( $B_j \gtrsim 20.5$ ).  
 Fonte: Adaptada de (MADDOX et al., 1990).



## 2 ÁRVORES DE DECISÃO APLICADAS AOS DADOS DO SDSS-DR7

Métodos de aprendizado de máquina são algoritmos que permitem ao computador “aprender” como identificar padrões dentro de um determinado conjunto de dados. Assim é possível agrupar, em classes distintas, dados com padrões semelhantes. O processo de aprendizado pode ser supervisionado ou não. Nos aprendizados supervisionados, como as AD e as redes neurais, o algoritmo exige um conjunto de exemplos de treinamento no qual a classe de cada exemplo seja conhecida. Nestes casos, o algoritmo tenta identificar padrões nos dados que estejam associados com cada uma das classes. No aprendizado não supervisionado, como o agrupamento, ou “clusterização”, as classes não são previamente conhecidas e o algoritmo tenta agrupar dados com padrões similares. Tanto no aprendizado supervisionado como no não supervisionado, novos dados podem ser remanejados para uma determinada classe, ou grupo, se apresentar os perfis definidos pelo algoritmo para aquela classe. Os métodos de aprendizado de máquina são ferramentas essenciais para buscar informações úteis em conjuntos de dados extensos e multidimensionais, uma vez que os computadores podem processar em minutos um volume de informações que um ser humano levaria décadas para analisar.

Escolhemos utilizar neste trabalho o método de árvore de decisão para separar estrelas de galáxias. Essa escolha não foi arbitrária, levamos em conta os bons resultados conseguidos com a aplicação deste método ao longo dos anos (veja Capítulo 1) e apresentados em trabalhos como os de [Weir et al. \(1995\)](#) e [Ball et al. \(2006\)](#). O principal concorrente da AD seria a rede neural, que segundo [Weir et al. \(1995\)](#) e [Odewahn et al. \(2004\)](#) apresenta um desempenho similar ao da AD, mas, segundo os mesmos autores, tem a necessidade de um maior tempo de treinamento. Contudo, o fator de maior peso na nossa escolha foi a necessidade da comunidade astronômica de conhecer os padrões encontrados pelos algoritmos de aprendizado de máquina nos dados utilizados na classificação. As AD, diferente das redes neurais<sup>1</sup>, exibem explicitamente os critérios de classificação, como mostra a Figura 2.1.

### 2.1 O Que é Uma Árvores de Decisão?

Uma AD consiste em um conjunto de regras geradas a partir de uma série de exemplos de treinamento para classificar dados que sejam definidos pelos mesmos atri-

---

<sup>1</sup>Para mais detalhes sobre redes neurais, o leitor pode consultar [Haykin \(1999\)](#)

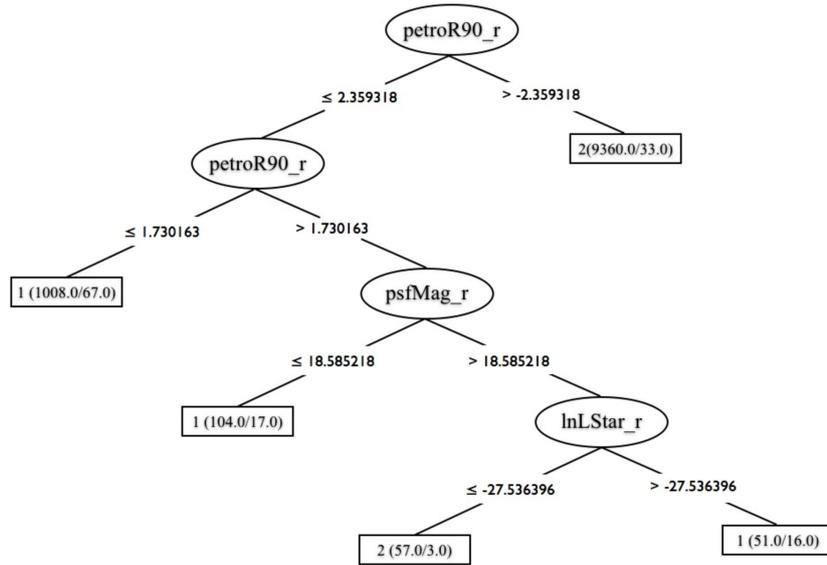


Figura 2.1 - Uma AD simples, construída com o algoritmo J48 (WITTEN; FRANK, 2000). Esta árvore foi treinada com 50.000 objetos da amostra espectroscópica, como descrito na Seção 2.5, e com um número mínimo de objetos por folha igual a 50. As possíveis classes aqui são 1 (estrela) ou 2 (galáxia).

butos dos exemplos. A Figura 2.1 mostra uma AD gerada com o algoritmo J48 do WEKA a partir de 50.000 exemplos (objetos) retirados do conjunto espectroscópico<sup>2</sup>. No caso da figura, a AD testa os atributos `petroR90_r`, `psfMag_r` e `lnLStar_r`, ou seja, esta árvore é capaz de classificar qualquer objeto do SDSS que possua estes três atributos. Como podemos observar, em uma AD as regras de classificação são estruturadas na forma de nós e ramos. Cada nó abriga um determinado teste sobre um ou mais atributos, e cada ramo os possíveis resultados para o teste. O nó contendo o primeiro teste (no caso do exemplo da Figura 2.1 o nó mais acima) é denominado de raiz, e os nós finais, dos quais não saem ramos, são chamados folhas. As folhas, em geral, não contêm um teste, mas sim uma classificação. Contudo, como veremos na Seção 2.3, há algoritmos que constroem árvores nas quais as folhas contêm novos testes, ou até mesmo novos classificadores. Cada caminho que começa na raiz e termina em uma folha pode ser definido como uma regra de classificação.

As árvores de decisão induzidas a partir de um conjunto de treinamento, ou simplesmente árvores de decisão induzidas, começaram a ganhar destaque em meados

<sup>2</sup>O conjunto espectroscópico é um conjunto de objetos do SDSS-DR7 cuja espectroscopia está disponível. Para mais detalhes o leitor pode consultar a Seção 2.5.

da década de 1980. Em 1984, quatro estatísticos (BREIMAN et al., 1984) publicaram um livro chamado “Classification and Regression Trees”. Dois anos depois, um proeminente pesquisador em aprendizado de máquina, J. Ross Quinlan, publicou um algoritmo capaz de inferir uma árvore classificadora a partir de um conjunto de exemplos (QUINLAN, 1986). Ambos os trabalhos, mesmo sendo independentes (WITTEN; FRANK, 2000, p. 27), deram origem a algoritmos similares. A idéia básica na qual se baseia a construção dessas árvores é “dividir para conquistar”, que visa resolver um problema complexo dividindo-o em uma série de subproblemas de complexidade menor. Os algoritmos de AD tentam dividir o conjunto de exemplos de treinamento até que seja fácil separar os exemplos nas classes predefinidas. O mesmo conceito é aplicado às árvores de regressão (*regression trees*). Contudo, as árvores de regressão são usadas para previsão numérica. Como um exemplo, podemos citar nosso trabalho de separação estrela/galáxia, no qual desejamos separar os exemplos (objetos celestes observados pelo SDSS com espectro disponível) em duas classes, estrela ou galáxia. Neste caso podemos dizer que as ADs usadas no nosso trabalho são árvores classificadoras. Por outro lado, se nosso objetivo fosse estimar o *redshift* fotométrico dos objetos do SDSS nossas ADs seriam árvores de regressão.

## 2.2 Construindo uma Árvore de Decisão

A construção de uma AD é um processo recursivo no qual o conjunto de exemplos é dividido até que seja fácil classificar os exemplos contidos nos subconjuntos gerados. A Tabela 2.1 mostra um conjunto simples de exemplos de treinamento. Os dados exibidos relacionam condições meteorológicas com duas possíveis decisões: sair ou não de casa. Neste contexto, as condições meteorológicas tempo, temperatura, umidade e vento são os atributos, ou seja, valores que devem ser analisados para se tomar a decisão. Já as decisões “sair” ou “não sair” são as possíveis classes.

O objetivo ao analisar os dados da Tabela 2.1 é aprender como decidir sair ou não de casa com base nos atributos meteorológicos contidos na tabela. Em outras palavras, dado uma combinação desses atributos que não esteja representada na tabela, como vamos decidir se devemos ou não sair? Para responder essa questão é que treinamos (construímos) uma AD. Os algoritmos de construção irão gerar um conjunto de testes sobre os atributos em um formato de nós e ramos (Figura 2.1), e os resultados destes testes irão nos dar a resposta desejada.

O conjunto de dados mostrado na Tabela 2.1 é muito interessante do ponto de vista

Tabela 2.1 - Pequeno conjunto de exemplos de treinamento. Esse conjunto de dados relaciona condições meteorológicas com a decisão de sair ou não de casa.

Tempo	Temp.( $^{\circ}C$ )	Umidade(%)	Vento?	Decisão
sol	23,9	70	sim	sair
sol	26,7	90	sim	não sair
sol	29,4	85	não	não sair
sol	22,2	95	não	não sair
sol	20,6	70	não	sair
nublado	22,2	87	sim	sair
nublado	28,3	78	não	sair
nublado	17,8	65	sim	sair
nublado	27,2	75	não	sair
chuva	21,7	80	sim	não sair
chuva	18,3	70	sim	não sair
chuva	23,9	80	não	sair
chuva	20,0	80	não	sair
chuva	21,1	96	não	sair

Fonte: Adaptado de [Quinlan \(1993\)](#).

acadêmico, pois ele contém atributos nominais e numéricos. Os atributos **Tempo**, **Vento** e a classe **Decisão** são chamados atributos nominais. Esses atributos possuem uma quantidade limitada e previamente conhecida de valores. No caso do atributo **Tempo**, as possibilidades são sol, nublado ou chuva. Neste caso, um teste sobre o atributo **Tempo** pode ter no máximo três resultados, como mostrado na Figura 2.2(a). Já os atributos ditos numéricos são assumidos como podendo apresentar qualquer valor real. Por exemplo, o atributo **Temperatura** pode ter qualquer valor  $x$  desde que  $x \in \mathbb{R} \mid x \geq -273$ . Os testes sobre atributos numéricos geralmente fornecem resultados binários, como os da Figura 2.1 e 2.2(b), mas podem também gerar resultados ternários como o mostrado na Figura 2.2(c). A classe do exemplo é sempre um atributo nominal, ou seja, o número possível de classes deve sempre ser limitado e bem definido.

A Tabela 2.1 exibe um conjunto de dados relativamente pequeno, e mesmo assim não é fácil para nós definir os critérios necessários para a tomada de decisão (classificação). Se já é difícil analisar um conjunto pequeno de dados como este, podemos imaginar com seria praticamente impossível para um ser humano inferir critérios para um conjunto com um grande volume de dados. Neste trabalho, por exemplo,

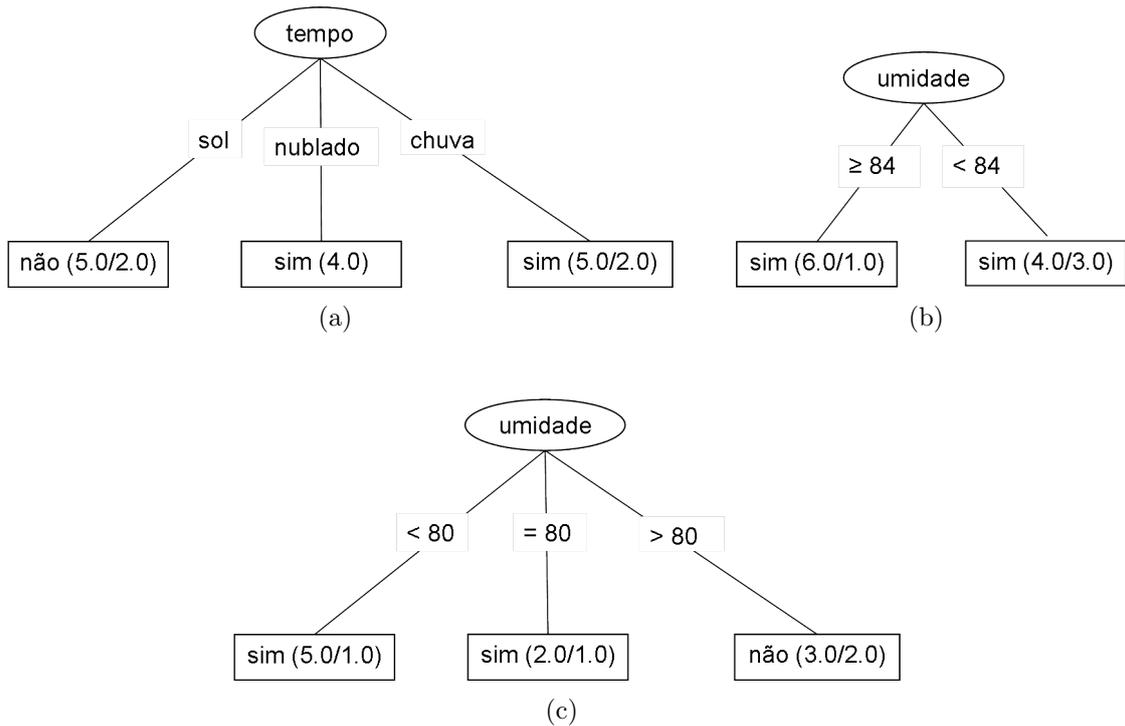


Figura 2.2 - A figura mostra exemplos de testes executados por uma AD sobre os dois tipos de atributos (numérico e nominal). A figura (a) mostra um exemplo de teste para um atributo nominal com três elementos. As figuras (b) e (c) mostram possíveis testes sobre um atributo numérico. As palavras sim e não representam a decisão de sair de casa. As classes são seguidas por números que representam quantos exemplos de cada uma existem naquela folha. Quando existir uma barra separando dois números, o primeiro representa o número de exemplos com a classe indicada, e número após a barra indica o número de exemplos da outra classe naquela folha.

desejamos analisar um conjunto com 880.715 exemplos, e 13 atributos distintos. Seria virtualmente impossível inferir critérios para classificação destes exemplos sem a ajuda de um computador. Os computadores são capazes de lidar com uma quantidade de informação gigantesca, e a cada dia esta capacidade tem aumentado. A criação de algoritmos para análise de dados é, portanto, uma consequência natural do desenvolvimento computacional. Os algoritmos de AD foram criados para inferir critérios de classificação em grandes volumes de dados de forma que estes critérios fiquem claros para o usuário. Esta é uma característica importante das ADs.

Um algoritmo de AD analisa um conjunto  $T$  de exemplos de treinamento para estabelecer o critério necessário para a classificação. Esse critério assume a forma de testes executados sobre os atributos  $A_i$  ( $i = 1, 2, \dots, l$ ) que definem o conjunto  $T$ . Um exemplo  $j$  contido no conjunto de treinamento  $T$ , qualquer que seja  $j$ , é definido por uma classe  $C_j$  e por um conjunto de valores  $a_{ij}$ , onde  $a_{ij}$  é o valor que o atributo  $A_i$  assume no exemplo  $j$  em questão. Os algoritmos constroem a AD nó por nó, sendo a raiz o primeiro a ser construído. O algoritmo de construção seleciona um dos atributos  $A_i$  para ser testado e define o teste a ser executado. Cada algoritmo da AD utiliza um determinado tipo de teste nos nós, como veremos na Seção 2.3. O C4.5 (QUINLAN, 1993) utiliza testes binários, como os da Figura 2.2(b), para atributos numéricos e testes de “valor”, como os da Figura 2.2(a), para atributos nominais. Já o FT (GAMA, 2004), utiliza combinações lineares dos atributos como teste nos nós. Outra diferença entre os diversos algoritmos de AD é a maneira de selecionar o atributo ou atributos a serem testados no nó. A forma mais difundida é a entropia da informação (QUINLAN, 1986; WITTEN; FRANK, 2005), que seleciona os atributos que quando testados tendem a minimizar a quantidade de informação necessária para a classificação dos exemplos.

A função do teste em um nó é dividir o conjunto de treinamento  $T$  em  $n$  subconjuntos, onde  $n$  é o número de possíveis resultados para o teste. Na Figura 2.2(a) vemos um exemplo da divisão do conjunto de treinamento em três subconjuntos de acordo com o valor do atributo **Tempo**. Os nós filho, ou seja, aqueles gerados a partir da raiz, poderão conter novos testes que iram dividir cada um dos  $n$  subconjuntos de  $T$  em novos subconjuntos gerando novos nós filhos. Este processo é executado recursivamente em cada nó filho até que o critério de parada usado pelo algoritmo seja satisfeito. Esse critério varia de algoritmo para algoritmo. Dois critérios de parada simples são:

- (i) se todos os membros do subconjunto pertencerem a mesma classe, uma folha é gerada;
- (ii) se o método de criação de testes do algoritmo não consegue determinar um teste capaz de separar o subconjunto, uma folha é criada.

Estes dois critérios são usados por muitos dos algoritmos avaliados neste trabalho. Em sua maioria, sempre que um algoritmo constrói uma folha ele associa uma classe a ela. No entanto, alguns algoritmos como o FT e o NBTree descritos na Seção 2.3, constroem folhas contendo novos testes. O FT e o NBTree utilizam, respectivamente, combinação linear dos atributos e classificadores Naive-Bayes nas folhas para a classificação final.

Alguns algoritmos de construção também utilizam uma técnica chamada poda. Quando construímos um AD a partir de um conjunto de exemplos de treinamento, pode ocorrer um fenômeno chamado de especialização ou, em inglês, “overfitting”. Nestes casos, a árvore é uma representação muito precisa do conjunto de exemplos, mas também muito específica. Contudo, como o principal objetivo de um AD é a classificação de dados fora do conjunto de exemplos, a técnica de poda é aplicada para garantir a generalização do modelo representado pela árvore. A poda, em geral, consiste em uma análise da AD após o treinamento visando a substituição de certos conjuntos de nós e ramos por folhas na tentativa de eliminar as especificidades geradas pelo treinamento. Este processo é um exemplo simples do que [Witten e Frank \(2000\)](#) chama de “postpruning”<sup>3</sup>. Alguns algoritmos, como o BFTree que abordaremos em breve, são capazes de proceder a poda durante o processo de treinamento da AD. Este tipo de poda é chamado “prepruning”.

### 2.3 O WEKA e Seus Algoritmos de Construção de Árvores de Decisão

O WEKA<sup>4</sup> é um aplicativo Java desenvolvido pela “University of Waikato”, na Nova Zelândia, para tarefas envolvendo mineração de dados. Ele consiste de uma coleção de algoritmos de aprendizado de máquina que podem ser aplicados diretamente, ou chamados dentro de um outro código JAVA. O WEKA contém ferramentas para pré-processamento, classificação, regressão, agrupamento (“clustering”) e visualização de dados.

---

<sup>3</sup>Processo de poda realizado posteriormente a construção da árvore.

<sup>4</sup><http://www.cs.waikato.ac.nz/ml/weka/>

Utilizamos neste trabalho o conjunto de ferramentas do WEKA para construção de Árvores de Decisão. Estão disponíveis na versão 3.6.0, utilizada por nós, 15 algoritmos, diferentes e independentes, para treinamento de uma AD<sup>5</sup>. Contudo, uma vez que nosso conjunto de dados é composto por atributos numéricos<sup>6</sup>, aplicamos aos nossos dados somente os 13 algoritmos do WEKA capazes de lidar com este tipo de atributo.

O WEKA fornece ao usuário três diferentes modos de trabalho: o GUI (*Graphical User Interface*, ou, em português, Interface Gráfica de Usuário), as linhas de comando e a possibilidade de utilizá-lo como biblioteca JAVA.

O GUI pode ser considerado o modo de operação principal do WEKA, sendo uma interface gráfica de fácil utilização onde todas as ferramentas do WEKA podem ser acessadas. Uma grande vantagem do GUI é que ele fornece uma interface intuitiva, permitindo que um usuário iniciante consiga utilizar as ferramentas sem um entendimento profundo do funcionamento do WEKA.

O acesso as ferramentas por linha de comando requer um conhecimento bem mais profundo do WEKA. Para chamar as suas ferramentas via linha de comando, o usuário precisa conhecer um pouco de JAVA e também conhecer o código do WEKA. O comando:

```
$ java -cp /usr/local/weka-3-4-4/weka.jar
weka.classifiers.trees.J48 -U
-t /usr/local/weka-3-4-4/data/weather.arff ,
```

por exemplo, permite ao usuário utilizar o algoritmo de AD J48 para analisar os dados apresentados na Tabela 2.1. Dividimos a linha de comando para que fique mais fácil para descrevê-la. Na primeira parte, chama-se a máquina virtual Java definindo com o `-cp` que usaremos o pacote adicional `weka.jar`. A segunda parte indica que dentro do `weka.jar` a máquina virtual Java irá executar a classe J48 com a opção `-U`, que desliga o processo de poda da árvore. Observe que esse trecho contém todo caminho dentro do código para acessar a classe J48, incluindo pacotes e subpacotes. Por fim, na terceira parte, o `-t` indica que o arquivo a ser processado com o J48 está localizado em `/usr/local/weka-3-4-4/data/weather.arff`. O WEKA requer que os exemplos de treinamento estejam em um arquivo com extensão *arff*,

---

<sup>5</sup>É relevante para o leitor saber que na atual versão beta do WEKA, a 3.7.3, somente estão disponíveis 6 algoritmos: Decision Stump, J48, LMT, Random Forest, Random Tree e REPTree.

<sup>6</sup>O atributo `specClass` que armazena a classe do objeto é o único atributo nominal.

que deve ser um arquivo de texto puro contendo três partes:

- a relação, declarada na primeira linha do arquivo, deve ser composta pela expressão `@RELATION` seguida de uma palavra-chave que identifique a relação ou tarefa sendo analisada;
- os atributos, declarados em um conjunto de linhas sequenciais onde cada uma inicia com a expressão `@ATTRIBUTE` seguida do nome do atributo e seguida do seu tipo, que pode ser nominal (neste caso as alternativas devem aparecer como uma lista separada por vírgulas e cercada por chaves) ou numérico (neste caso o nome deve ser seguido da palavra-chave `real`). Geralmente, o último atributo contém a classificação para os exemplos;
- os dados, declarados depois de uma linha contendo a expressão `@DATA`. Cada linha subsequente deve corresponder a um exemplo de treinamento e deve ter valores separados por vírgula, de forma que a seqüência na qual os valores aparecem corresponda (na mesma ordem) a seqüência na qual os atributos foram declarados.

Abaixo segue o exemplo dos dados da Tabela 2.1 colocados no formato ARFF do WEKA.

```
@RELATION weather

@ATTRIBUTE Tempo sol,nublado,chuva
@ATTRIBUTE Temperatura REAL
@ATTRIBUTE Umidade REAL
@ATTRIBUTE Vento sim,não
@ATTRIBUTE Decisão sim,não

@DATA
sol,23.9,70,sim,sair
sol,26.7,90,sim,não sair
sol,29.4,85,não,não sair
sol,22.2,95,não,não sair
sol,20.6,70,não,sair
nublado,22.2,87,sim,sair
```

```
nublado,28.3,78,não,sair
nublado,17.8,65,sim,sair
nublado,27.2,75,não,sair
chuva,21.7,80,sim,não sair
chuva,18.3,70,sim,não sair
chuva,23.9,80,não,sair
chuva,20.0,80,não,sair
chuva,21.1,96,não,sair
```

O modo de trabalho escolhido para desenvolver nosso estudo foi utilizar a WEKA como uma biblioteca Java, ou seja, escrevemos nossos próprios códigos Java instanciando as classes do WEKA para processar os dados do SDSS. A GUI do WEKA é uma forma simples e dinâmica de trabalhar, mas implica em um excessivo consumo de memória por alocar espaço para as classes da interface gráfica. Tomando como exemplo um PC com 2Gb de memória RAM com sistema operacional Linux, e como conjunto exemplos de treinamento os objetos do SDSS citados na Seção 2.5 com os 13 atributos fotométricos descritos na Seção 2.6, nosso estudo nos mostra que alguns algoritmos, como o FT, não são capazes de processar mais de 60.000 exemplos, menos de 1/8 do conjunto de treinamento. Uma vez que desejamos testar os 13 algoritmos do WEKA, essa limitação nos fez abandonar a GUI. Escolhemos a implementação em Java instanciando as classes do WEKA porque este modo de trabalho nos dá uma maior flexibilidade do que as chamadas por linha de comando e consome menos memória RAM do que a GUI. A implementação de códigos próprios também nos permitiu otimizar o consumo de memória pelo processamento dos exemplos de treinamento. Como veremos na Seção 2.6, o consumo de memória RAM foi um obstáculo para o progresso do trabalho.

Os 13 algoritmos testados foram: J48, J48graft, BFTree, FT, LMT, Simple Cart, REPTree, Random tree, Random Forest, NBTree, ADTree, LADTree, Decision Stump. O estudo detalhado do funcionamento de cada uma destes algoritmos foge ao escopo desta dissertação. Nosso objetivo nos dois anos durante os quais trabalhamos foi avaliar o desempenho de cada um destes algoritmos quando aplicados na separação estrela/galáxia de objetos do SDSS-DR7, e posterior elaboração de um catálogo contendo essa classificação. Contudo, as subseções seguintes descrevem de forma resumida a metodologia de cada algoritmo.

### 2.3.1 J48

O J48 é a implementação WEKA do algoritmo C4.5 (QUINLAN, 1993). O C4.5 é a evolução do algoritmo ID3 (QUINLAN, 1986), que foi um dos primeiros algoritmos de AD induzida.

Ele seleciona o teste a ser executado em um nó com base no valor de um índice chamado “information gain ratio”, ou taxa de ganho de informação, que é calculado com base nas equações:

$$info(T) = - \sum_{j=1}^k \left[ \frac{freq(C_j, T)}{|T|} \times -\log_2 \left( \frac{freq(C_j, T)}{|T|} \right) \right] \text{ bits}, \quad (2.1)$$

$$gain(X) = info(T) - \sum_{i=1}^p \left( \frac{|T_i|}{|T|} \times info(T_i) \right) \text{ bits}, \quad (2.2)$$

$$split\ info(X) = - \sum_{i=1}^p \left[ \frac{|T_i|}{|T|} \times -\log_2 \left( \frac{|T_i|}{|T|} \right) \right] \text{ bits}, \quad (2.3)$$

$$gain\ ratio(X) = \frac{gain(X)}{split\ info(X)}, \quad (2.4)$$

onde  $T$  é o conjunto de exemplos de treinamento que “entra” no nó,  $C_j$  é a  $j$ -ésima classe,  $k$  é o número de classes,  $X$  é uma referência ao atributo testado, e  $p$  é o número de subconjuntos gerados pelo teste  $X$ . A idéia é selecionar o teste que minimize a quantidade de informação necessária para a classificação de um objeto<sup>7</sup>. Para a construção de um nó a taxa de ganho é calculada para cada atributo e aquele que apresentar maior valor será usado no nó para dividir o conjunto de exemplos de treinamento. Se o subconjunto que deverá ser testado em um determinado nó só possua exemplos da mesma classe ou se todos os exemplos nele contidos apresentam a mesma taxa de ganho, então nenhum teste é proposto e uma folha é criada.

O J48 é um algoritmo padrão que testa um único atributo em cada nó. As ADs geradas por este tipo de algoritmo são denominadas univariantes ou “axis-parallel” – eixo-paralela em uma tradução livre. Estas ADs recebem o nome univariante por testar um único atributo por nó. Este tipo de teste divide o espaço de atributos com um multiplano perpendicular ao eixo que representa o atributo em questão, ou seja, paralelo aos demais. Logo, a denominação *axis-parallel* refere-se a esta propriedade. Imagine que os exemplos de treinamento apresentados na Tabela 2.1 são definidos apenas pelos atributos **Umidade** e **Temperatura**. Neste caso, o teste proposto na Figura 2.2(b) divide o espaço bidimensional definido por estes dois atributos com uma reta paralela ao eixo **Temperatura** (Figura 2.3). Esse é um exemplo simples em um espaço bidimensional. No caso de espaço tridimensional, teríamos

---

<sup>7</sup>Neste texto, entendemos por objeto um conjunto de dados composto pelos 13 atributos fotométricos do SDSS-DR7 escolhidos por nós (veja Seção 2.6.1). Nossos exemplos de treinamento são casos especiais de objetos onde além dos atributos fotométricos temos também um atributo espectroscópico que nos diz a classificação verdadeira do objeto. Genericamente, um conjunto de atributos usado para classificação por meio de uma AD é chamado instância. Um exemplo de treinamento é simplesmente uma instância cuja classe é conhecida.

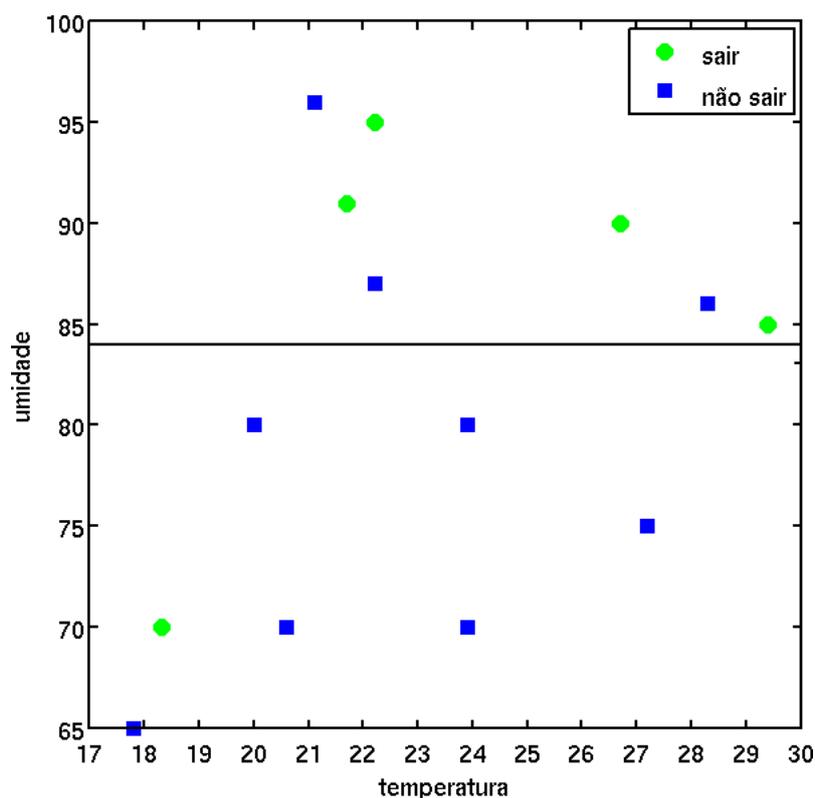


Figura 2.3 - A figura mostra o espaço de atributos temperatura X umidade. A reta umidade = 84 representa o teste proposto na 2.2(b) e divide o espaço de atributos em duas partes.

um plano no lugar de uma reta, e no de um espaço multidimensional, como o utilizado neste trabalho, teríamos um multiplano no lugar de um plano.

### 2.3.2 J48graft

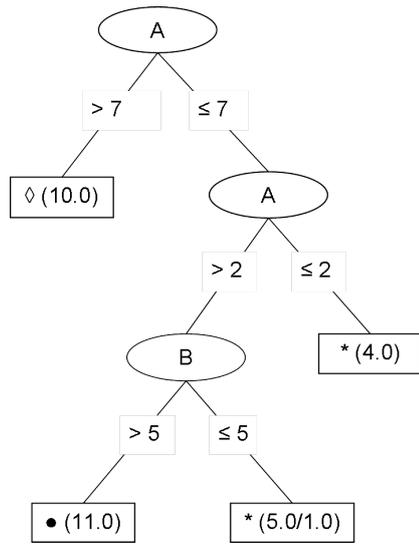
O J48graft aplica uma técnica chamada “grafting” (GEOFFREY, 1997; GEOFFREY, 1999) a uma AD treinada com o algoritmo J48 para reduzir o erro na classificação de objetos fora do conjunto de treinamento. Esta técnica consiste em adicionar novos testes (nós) a uma AD já treinada. Uma AD divide o espaço de atributos em um número de subespaços igual ao número de folhas da árvore. A Figura 2.4(b), adaptada de Geoffrey (1999), mostra um exemplo simples de um espaço definido por dois atributos e os subespaços criados por uma AD (Figura 2.4(a)) treinada com o C4.5 (QUINLAN, 1993). Esses subespaços podem não conter exemplos de treinamento ou conter um número elevado de exemplos classificados erroneamente, aumentando o erro de classificação. Para solucionar este problema, foram propostas técnicas de decisão por comitê (FREUND; SCHAPIRE, 1996; BREIMAN, 1996). Um comitê, como veremos em mais detalhes na seção 2.8, consiste em um conjunto de

árvores treinadas com os mesmos algoritmo e conjunto de treinamento, de forma que a classificação final de um objeto fora do conjunto de treinamento seja baseada em uma combinação das classificações de cada AD do comitê. Um exemplo da aplicação de um comitê de ADs é o trabalho de Suchkov et al. (2005), que foi discutido no capítulo 1, no qual os autores treinaram 10 ADs para estimar o *redshift* de objetos do SDSS-DR2. A criação de um comitê permite que os objetos erroneamente classificados em um subespaço definido por uma das árvores membro possam ser classificados corretamente por uma ou mais dentre as demais árvores. Desta forma, onde uma AD erra as outras podem acertar, o que pode resultar em uma redução do erro na classificação.

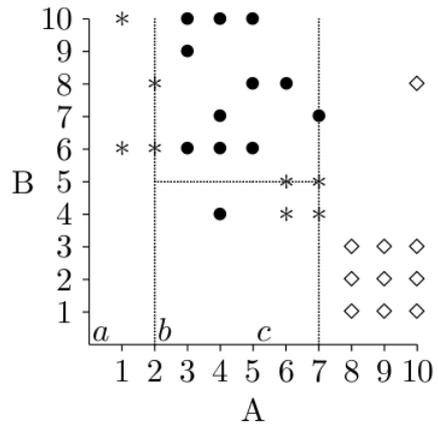
Contudo, enquanto uma única AD fornece um esquema de classificação de simples compreensão, a complexidade de um comitê, devido ao número de membros, pode ser muito elevada. Segundo (GEOFFREY, 1999), mesmo sendo possível gerar uma única árvore que represente todo o comitê, o tamanho (número de nós e ramos) dessa árvore aumenta exponencialmente com o número de membros do comitê.

A técnica “grafting” visa obter os mesmos benefícios de um comitê de ADs com uma única árvore. A idéia é examinar uma AD construída com o J48 buscando por regiões do espaço de atributos nas quais não existem exemplos de treinamento ou existe um número elevado de exemplos erroneamente classificados; propondo para estas regiões uma classificação alternativa. Para cada folha da AD original, o algoritmo examina cada nó pai no caminho inverso da folha a raiz propondo testes alternativos que irão criar novos subespaços para tentar reduzir o número de exemplos com classificação incorreta. Considere o exemplo mostrado na Figura 2.4. A AD apresentada na Figura 2.4(a) divide o espaço de atributos em quatro seções (Figura 2.4(b)), cada uma delas representada por uma folha da árvore. Pode-se observar que o subespaço  $X$  delimitado pelos intervalos  $2 < A \leq 7$  e  $B \leq 5$  contém um exemplo da classe ( $\bullet$ ) que é atribuído erroneamente pela AD à classe ( $*$ ). O algoritmo busca novas divisões do espaço de parâmetros que possam resolver o problema da imprecisão da árvore no subespaço  $X$ . A Figura 2.4(d) mostra a nova divisão do espaço de atributos após a aplicação da técnica *grafting*. A “grafted tree”, mostrada na Figura 2.4(c), foi gerada a partir da árvore original com a substituição de uma folha por um novo teste sobre o atributo A. O método foi bem sucedido e eliminou a inconsistência da AD original.

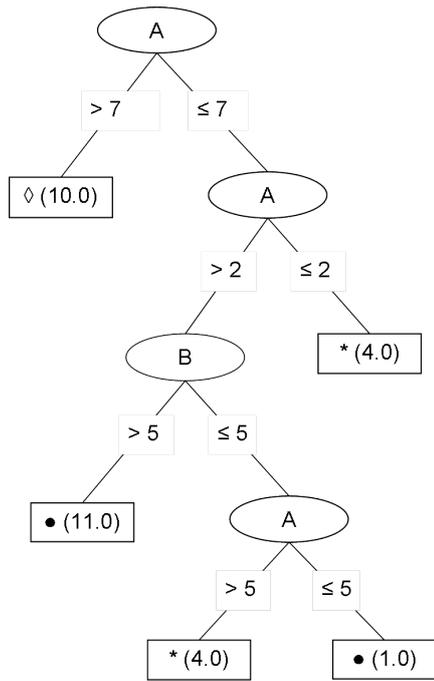
Segundo Geoffrey (1999), apesar da técnica *grafting* ser bem sucedida em seu propósito de reduzir o erro de classificação de uma AD previamente treinada, ela não apresenta resultados tão bons quanto os comitês. Contudo, é importante ter em mente que o algoritmo J48graft gera um AD de fácil interpretação, enquanto um comitê de árvores treinadas com o J48 pode ter uma interpretação altamente complexa.



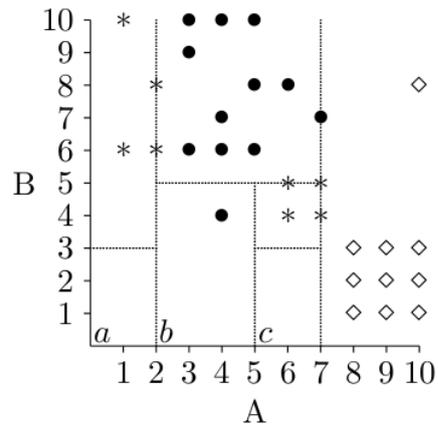
(a)



(b)



(c)



(d)

Figura 2.4 - As figuras mostram: (a) uma AD treinada com o C4.5; (b) o espaço definido pelos atributos A e B com as respectivas divisões associadas à árvore apresentada em (a); (c) a expansão da árvore mostrada em (a) com a aplicação da técnica de *grafting*; (d) o espaço definido pelos atributos A e B com as respectivas divisões associadas à árvore apresentada em (c). Os símbolos  $\bullet$ ,  $*$  e  $\diamond$  representam as três diferentes classes do conjunto de treinamento. Fonte: As Figuras (b) e (d) foram adaptadas de Geoffrey (1997).

### 2.3.3 BFTree

O BFTree (Best-First decision Tree; HAIJIAN, 2007) possui um processo de treinamento similar ao do J48. O algoritmo constrói uma AD de cima para baixo, começando com o nó raiz, e os testes a serem executados em cada nó são selecionados com base no valor do ganho de informação (veja Seção 2.3.1). A principal diferença entre o BFTree e o J48 está na ordem na qual os nós são construídos. O J48 constrói os nós em uma ordem fixa da esquerda para direita. Já o BFTree utiliza uma estratégia de construção chamada “best-first order” (HAIJIAN, 2007) na qual o algoritmo irá construir primeiro o nó com o maior ganho. Tomemos com exemplo a Figura 2.2(a), onde o nó raiz gera três nós filhos. Enquanto o J48 iria prosseguir a construção da árvore começando pelo nó da esquerda, o BFTree irá comparar o ganho de informação em cada nó e prosseguirá com a construção a partir do nó com o maior ganho. A AD final será idêntica em ambos os casos. O diferencial do BFTree é a possibilidade de poda durante a construção da árvore. Segundo Haijian (2007), usando a estratégia *best-first order* é possível prever e descartar os ramos que podem causar a especialização da árvore durante o processo de construção.

### 2.3.4 FT

O FT (Functional Trees; GAMA, 2004) é um algoritmo de construção de ADs oblíquas. Breiman et al. (1984), Murthy et al. (1994), e Brodley e Utgoff (1995) exploraram uma série de algoritmos capazes de construir ADs multivariantes. Uma AD padrão, como as geradas pelo C4.5 (QUINLAN, 1993), é chamada univariante por testar um único atributo por folha. Já uma AD dita multivariante utiliza testes baseados na combinação de diversos atributos. As AD multivariantes são também chamadas de oblíquas pela capacidade de dividir o espaço de atributos com multiplanos oblíquos aos eixos que o definem. Considerando novamente os exemplos da Tabela 2.1 definidos apenas pelos atributos *Umidade* e *Temperatura*, temos na Figura 2.5 a demonstração do aumento na precisão do classificador por utilizar uma divisão oblíqua em vez de uma paralela a eixos.

Witten e Frank (2000) estudaram o uso de um tipo diferente de árvores multivariantes, chamadas “model trees”, ou árvores modelo, para solução de problemas de previsão numérica. As árvores modelo, diferente das outras AD multivariantes, utilizam a combinação de atributos nas folhas e não nos nós de decisão<sup>8</sup>.

Gama (2004) propôs pela primeira vez a combinação da metodologia usada nas AD multivariantes padrão com as das árvores modelo, possibilitando a construção de uma AD que utilizasse combinações lineares dos atributos nos nós de decisão e também nas folhas.

---

<sup>8</sup>Os nós de decisão são todos aqueles que não são folhas.

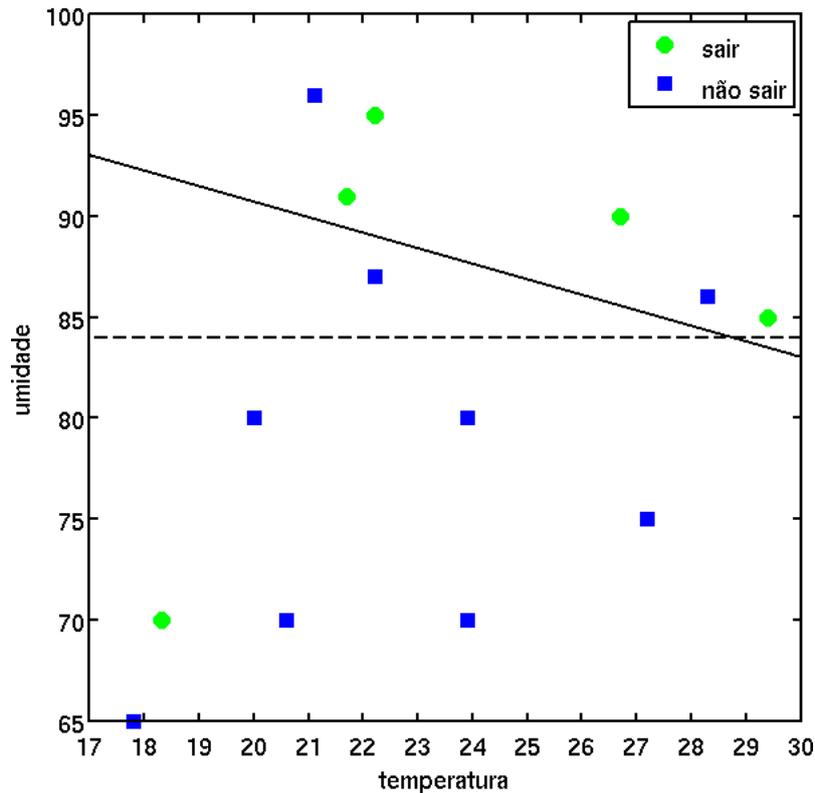


Figura 2.5 - A figura mostra o espaço de atributos Temperatura X Umidade. A reta  $Umidade = 84$  representa o teste proposto na 2.2(b) e representa um teste univariante. A reta  $Umidade = -0.77 * Temperatura + 106.08$  representa um teste multivariante. Ambos dividem o espaço de atributos em duas regiões.

O FT usa regressão linear para criar combinações lineares dos atributos, construindo nós multivariantes durante o crescimento da árvore, e folhas durante o processo de poda<sup>9</sup>. Os resultados apresentados por Gama (2004) mostram que árvores com nós e folhas multivariantes apresentam melhores resultados do que árvores univariantes, árvores multivariantes padrão e árvores modelo, principalmente para grandes conjuntos de dados.

### 2.3.5 LMT

O LMT (Logistic Model Trees; LANDWEHR et al., 2005) é um algoritmo de construção de árvores modelo. Como discutido na Seção 2.3.4, estas árvores possuem modelos de regressão nas folhas, ou seja, diferente das árvores univariantes padrão nas quais há uma classe associada a cada folha, a classificação de um objeto em uma AD construída com o LMT é fornecida pelo modelo linear associado a folha atingida pelo objeto.

<sup>9</sup>O FT, que é a implementação WEKA do método Functional Tree, pode utilizar regressão Logística (veja Seção 2.3.5) na construção das folhas.

A proposta do LMT é substituir os modelos de regressão linear por modelos de regressão logística (*logistic regression models*). Estes modelos fazem parte de uma categoria de modelos estatísticos chamados “generalized linear models”. Eles consistem em uma técnica estatística que tem como objetivo produzir, a partir de um conjunto de observações, um modelo que permita a predição de valores tomados por uma variável categórica, frequentemente binária, a partir de uma série de variáveis explicativas contínuas e/ou binárias. Uma ótima abordagem dos “generalized linear models” pode ser encontrada em [Agresti \(1996\)](#).

### 2.3.6 Simple Cart

O Simple Cart é a implementação WEKA do algoritmo CART ([BREIMAN et al., 1984](#)). O processo de construção é similar ao do J48. O algoritmo constrói a árvore nó a nó aplicando recursivamente seu critério de seleção de testes. As árvores geradas pelo Simple Cart são univariantes com divisões binárias, ou seja, cada nó testa um único atributo gerando apenas dois ramos. O algoritmo utiliza como critério de seleção de testes a variação do valor de um índice chamado **Gini**:

$$I(t) = \sum_{i,j} C(i|j) p(i|t) p(j|t) , \quad (2.5)$$

$$\Delta I(s, t) = I(t) - p_L I(t_L) - p_R I(t_R) , \quad (2.6)$$

onde  $C(i|j)$  é a probabilidade de se classificar erroneamente um objeto da classe  $j$  como sendo da classe  $i$ ,  $p(i|t)$  é a probabilidade de um objeto pertencer a classe  $i$  dado que ele “caiu” no nó  $t$ ,  $p(j|t)$  é a probabilidade de um objeto pertencer a classe  $j$  dado que ele “caiu” no nó  $t$ ,  $p_L$  e  $p_R$  são as probabilidades de se conduzir um exemplo de treinamento respectivamente para o nó a esquerda  $t_L$  ou a direita  $t_R$ . Esse critério usado pelo Simple Cart tem o mesmo objetivo da taxa de ganho de informação do J48, ambos selecionam testes que reduzem a quantidade de informação necessária à classificação dos exemplos.

### 2.3.7 REPTree

O REPTree (<http://www.cs.waikato.ac.nz/ml/weka/>) é um algoritmo para construção de AD univariantes que pode ser usado tanto para solução de problemas de classificação como de regressão. De forma similar ao J48, o REPTree baseia-se no ganho de informação para definir o teste a ser executado em cada nó de decisão.

### 2.3.8 Random Tree

Os algoritmos chamados Random Tree vêm sendo amplamente desenvolvidos nos últimos anos. A idéia é construir nós com atributos selecionados de maneira aleatória. O algoritmo Random Tree do WEKA constrói cada nó da árvore escolhendo um entre  $K$  atributos selecionados aleatoriamente. O algoritmo testa cada um dos  $K$  atributos usando como critério de avaliação a variação do índice GINI (veja Seção 2.3.6). O método é recursivo e uma folha só é criada se, após selecionar os  $K$  atributos, nenhum deles apresentar redução na quantidade de informação necessária à classificação dos exemplos.

### 2.3.9 Random Forest

O Random Forest (BREIMAN, 2001) utiliza a técnica de decisão por comitê<sup>10</sup> para tarefas de classificação. O algoritmo divide aleatoriamente o conjunto de treinamento em  $n$  subconjuntos distintos, e cada subconjunto gerado é então utilizado para construir uma AD.

As árvores são construídas com uma adaptação do algoritmo CART (BREIMAN et al., 1984). Na construção de cada nó de decisão o Random Forest seleciona randomicamente um número  $k$  de atributos ( $k < n^\circ$  total de atributos). O teste a ser realizado no nó é então selecionado com base na metodologia do algoritmo CART<sup>11</sup>. Em outras palavras, para um dado subconjunto  $i$  gerado pelo Random Forest, a AD correspondente é construída nó a nó com base na metodologia do algoritmo CART, de forma que o conjunto  $i$  de exemplos usados para construir o nó em questão é definido apenas pelos  $k$  atributos escolhidos aleatoriamente e não por todos os atributos do conjunto original. Como em outros algoritmos já vistos aqui, como por exemplo o J48, uma folha é criada quando o método CART não consegue determinar um novo teste capaz satisfazer seus critérios de seleção.

Com as AD treinadas e o comitê pronto, um objeto é classificado com base em votação majoritária. Cada árvore emite sua classificação do objeto e a de maior incidência é escolhida como a classificação final.

### 2.3.10 NBTree

O NBTree (“Naive Bayesian Tree learner algorithm”; KOHAVI, 1996) gera um classificador híbrido entre um classificador Naive-Bayes (LANGLEY et al., 1992) e uma AD. O algoritmo constrói uma AD cujos nós de decisão contêm testes univariantes, como no J48, mas as folhas contêm classificadores Naive-Bayes. O modelo construído por um algoritmo Naive-

---

<sup>10</sup>veja Seção 2.3.2.

<sup>11</sup>Veja Simple Cart na Seção 2.3.6.

Bayes consiste em um conjunto de probabilidades estimadas com base na frequência com a qual um valor de característico aparece quando analisamos os exemplos de treinamento. Dado um novo objeto, o classificador estima a probabilidade de este objeto pertencer a uma classe específica, com base no produto das probabilidades condicionais individuais para os valores característicos (atributos) do objeto. Se por exemplo queremos achar maçãs em um cesto de frutas, podemos dizer que uma fruta será considerada uma maçã se for vermelha, redonda e tiver aproximadamente 8 cm de diâmetro. Essas são os valores característicos “aprendidos” do conjunto de treinamento. O cálculo utiliza o teorema de Bayes. Durante o treinamento, o algoritmo considera que cada atributo contribui de forma independente para a probabilidade associada a uma classe. Segundo Kohavi (1996), os classificadores Naive-Bayes apresentam um bom desempenho em muitas tarefas de classificação em conjuntos pequenos de dados. Em uma árvore construída com o NBTree, um objeto é classificado usando um Naive-Bayes local na folha onde ele caiu. O NBTree frequentemente atinge uma precisão maior do que um classificador Naive-Bayes ou uma AD quando trabalhando sozinho (KOHAVI, 1996).

### 2.3.11 ADTree

O ADTree (“Alternating Decision Tree”; FREUND; MASON, 1999) constrói uma AD composta por nós de previsão e nós de divisão. Os nós de divisão são definidos pelo algoritmo como um teste univariante, de maneira similar ao J48, enquanto que os nós de previsão são definidos por um simples valor  $x \in \mathbb{R}$ . Em uma AD padrão, um objeto seguirá um determinado caminho da raiz a uma das folhas de acordo com o valor assumido por cada atributo, e neste caso a folha irá conter a classificação do referido objeto. Na classificação feita por uma ADTree o processo é similar, mas não há folhas, o caminho percorrido por um objeto começa em um nó de previsão e termina em outro nó de previsão (Figura 2.6). A classificação é obtida com base no sinal (positivo ou negativo) da soma de todos os nós de previsão existentes no caminho percorrido.

A Figura 2.6 mostra uma AD treinada com ADTree do WEKA, usando o conjunto de exemplos da Tabela 2.1. O WEKA organiza a visualização deste tipo de árvore colocando no nó raiz o critério de classificação. Neste exemplo, um objeto que após percorrer a árvore obtenha um resultado de sinal negativo para a soma dos nós de previsão será classificado como *sim*, e um objeto que obtenha um resultado de sinal positivo como *não*.

O método de construção do ADTree apresenta três características importantes, sendo duas delas facilmente percebidas na Figura 2.6.

- Uma árvore construída com o ADTree sempre começa por um nó de previsão e

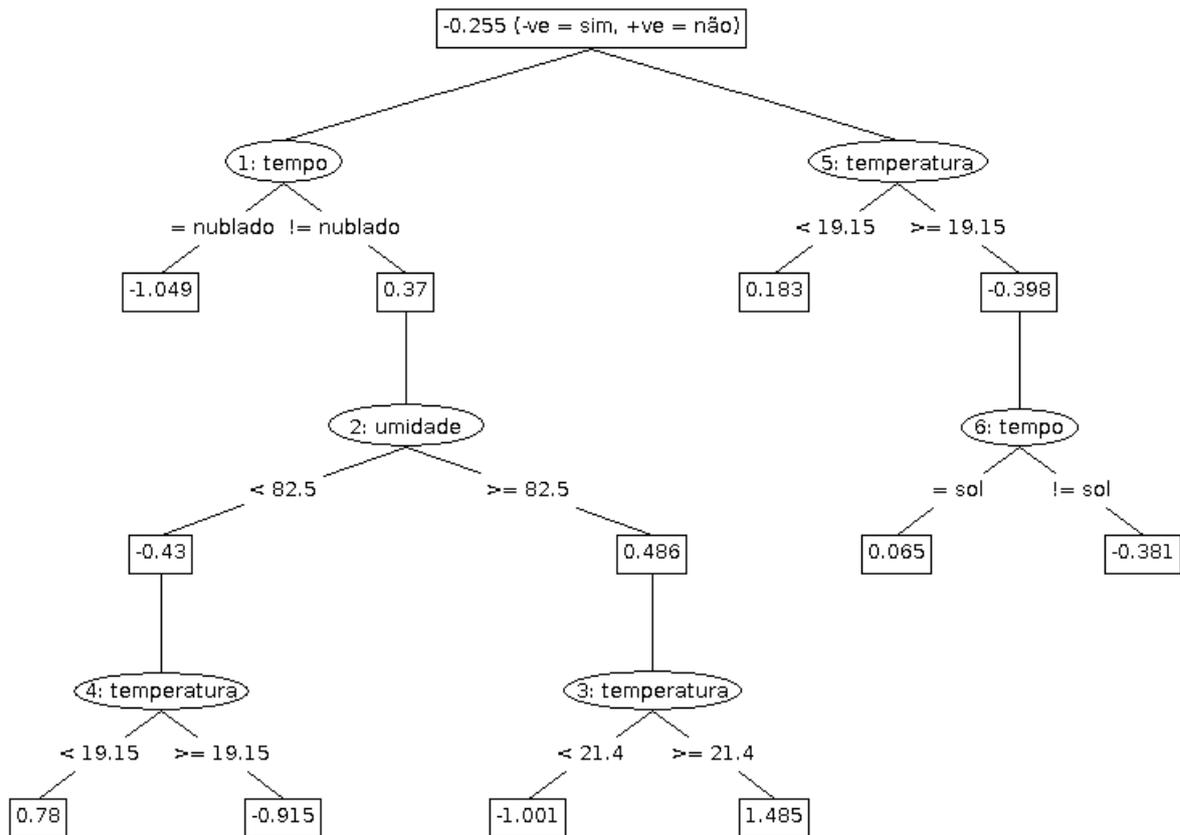


Figura 2.6 - A figura mostra uma AD construída pelo WEKA com os exemplos apresentados na Tabela 2.1 usando ADTree. O texto “(-ve = sim, +ve = não)” contido no nó raiz significa que o sinal negativo no resultado da soma dos nós de previsão classifica uma instância como *sim*, e o sinal positivo como *não*.

não por um de decisão, como é comum aos algoritmos de construção. Diferente de uma AD padrão, em uma árvore treinada com ADTree não existe somente um único caminho a partir da raiz que um objeto a ser classificado pode percorrer. No exemplo da Figura 2.6 um objeto irá percorrer o caminho que sai da raiz e passa pelo nó 1 e o que sai da raiz e passa pelo nó 5. Esse processo é possível porque a classificação é baseada no sinal da soma dos nós de previsão e não na classe associada a uma folha.

- Qualquer teste proposto pelo ADTree em um nó terá somente dois ramos, mesmo que o atributo testado seja nominal e tenha mais de dois valores possíveis.
- O ADTree somente trabalha com duas classes. Não poderíamos, por exemplo, usar o ADTree para construir uma AD com base no conjunto de exemplos apresentado na Figura 2.4(b), pois, este conjunto possui três classes distintas.

### 2.3.12 LADTree

O LADTree (HOLMES et al., 2001) é uma melhoria do ADTree capaz de treinar uma AD a partir de um conjunto de treinamento com mais de duas classes. Como dito na Seção 2.3.11, a formulação original da ADTree está restrita a problemas de classificação binária (duas classes). O algoritmo LADTree estende o algoritmo ADTree aos problemas multiclasse dividindo-os em vários problemas de classificação binária.

### 2.3.13 Decision Stump

O Decision Stump (<http://www.cs.waikato.ac.nz/ml/weka/>) gera AD binárias simples, compostas por um nó e duas folhas. O critério de seleção do teste contido no nó foi, no nosso caso, a taxa de ganho de informação, como no J48.

## 2.4 Precisão e Performance de um Algoritmo: o Método Cross-Validation

A precisão de qualquer método de separação estrela/galáxia depende da magnitude aparente dos objetos e é geralmente medida pelas funções de completudeza  $CP(m)$  (fração de galáxias classificadas corretamente) e de contaminação  $CT(m)$  (fração de estrelas classificadas como galáxias) em um intervalo de magnitudes  $\delta m$ <sup>12</sup>. Estas funções são definidas

---

<sup>12</sup>Os valores de completudeza e contaminação também são conhecidos respectivamente como “true positive rate” e “false positive rate”.

como:

$$CP(m) = 100 * \frac{N_{gal-gal}(m)\delta m}{N_{galaxy}^{tot}(m)\delta m} \quad (2.7)$$

e

$$CT(m) = 100 * \frac{N_{star-gal}(m)\delta m}{N_{star}^{tot}(m)\delta m} \quad (2.8)$$

onde  $N_{gal-gal}(m)\delta m$  é o número de imagens de galáxias corretamente identificadas no intervalo  $(m-\delta m/2, m+\delta m/2)$ ;  $N_{star-gal}(m)\delta m$  é o número imagens de estrelas falsamente identificadas como galáxias;  $N_{galaxy}^{tot}(m)\delta m$  é o número total de galáxias na amostra; e  $N_{star}^{tot}(m)\delta m$  é o número total de estrelas. Usamos, para calcular as funções de completeza e contaminação ao longo de todo o processo de pesquisa aqui descrito, um intervalo constante de magnitude<sup>13</sup> com lagura  $\delta m = 0,5^m$ .

É necessário, para discussões futuras neste texto, definir os valores médios destas funções dentro de um dado intervalo de magnitudes. Logo, a completeza média é definida como  $\langle \text{Compl} \rangle_{\Delta m} = (1/\Delta m) \sum CP(m_i)\delta m_i$ , com  $\Delta m = \sum \delta m_i$ . A mesma equação pode ser aplicada para definir a contaminação média. Observe que  $\langle \text{Compl} \rangle_{\Delta m} \cdot \Delta m$  fornece a área sobre a curva de completeza no intervalo  $\Delta m$ .

Nosso objetivo principal neste trabalho foi selecionar o algoritmo de melhor performance, em termos de precisão, especialmente no limite superior de magnitudes ( $r \geq 19$ ), dentre os 13 algoritmos descritos na Seção 2.3. Contudo, para aplicação em grandes volumes de dados, o tempo de treinamento pode ser uma informação importante na avaliação dos diferentes algoritmos.

Por fim, definimos nosso método de avaliação quanto a performance de cada algoritmo. Existem várias abordagens para se determinar esta performance. A mais usual consiste na divisão do conjunto de exemplos de treinamento em dois subconjuntos, normalmente numa fração 4:1, sendo o maior deles usado no treinamento da AD e o menor na avaliação da performance. No entanto, escolhemos um método mais sofisticado chamado Cross-Validation (CV; WITTEN; FRANK, 2005). O CV consiste em dividir o conjunto de treinamento em 20 subconjuntos, cada um com a mesma distribuição de classes que o conjunto original. Apesar de o número de subconjuntos, 20, ser arbitrário, cada subconjunto conterà um grande número de exemplos de treinamento. Para cada subconjunto, é criada uma AD, que por sua vez é aplicada aos outros 19. As funções de completeza e contaminação resultantes são coletadas e a média e a dispersão sobre a aplicação de todas as árvores sobre todos os

---

<sup>13</sup>A grandeza astronômica magnitude não possui uma unidade. Quando um valor de magnitude é seguido pelo expoente  $m$  entende-se “unidades de magnitude”. Por exemplo  $0,5^m$  lê-se 0,5 unidades de magnitude.

subconjuntos é calculada. Estes valores nos fornecem a estimativa do CV sobre o robustez de um algoritmo segundo a função de completiza.

## 2.5 Os Dados do SDSS-DR7 Utilizados

O SDSS é um levantamento astronômico conduzido pelo telescópio de 2,5 metros localizado no “Apache Point Observatory”, nos EUA, com a finalidade de gerar dados fotométricos e espectroscópicos de objetos celestes. As observações foram feitas por um detector composto por 30 câmeras CCD de  $2048 \times 2048$  pixels cada, capaz de cobrir uma região celeste de  $1,5^\circ \times 1,5^\circ$ . As câmeras CCD operam com um conjunto de cinco filtros (u, g, r, i e z), onde cada filtro cobre uma determinada banda (faixa de comprimentos de onda) do espectro eletromagnético (Figura 2.7). O SDSS conta ainda com um par de espectrógrafos alimentados por fibras óticas de 3” de diâmetro para obter espectros dos objetos observados. Todos os objetos observados pelo SDSS-DR7 possuem dados fotométricos, mas somente uma pequena fração deles possui dados espectroscópicos ( $\sim 1$  milhão).

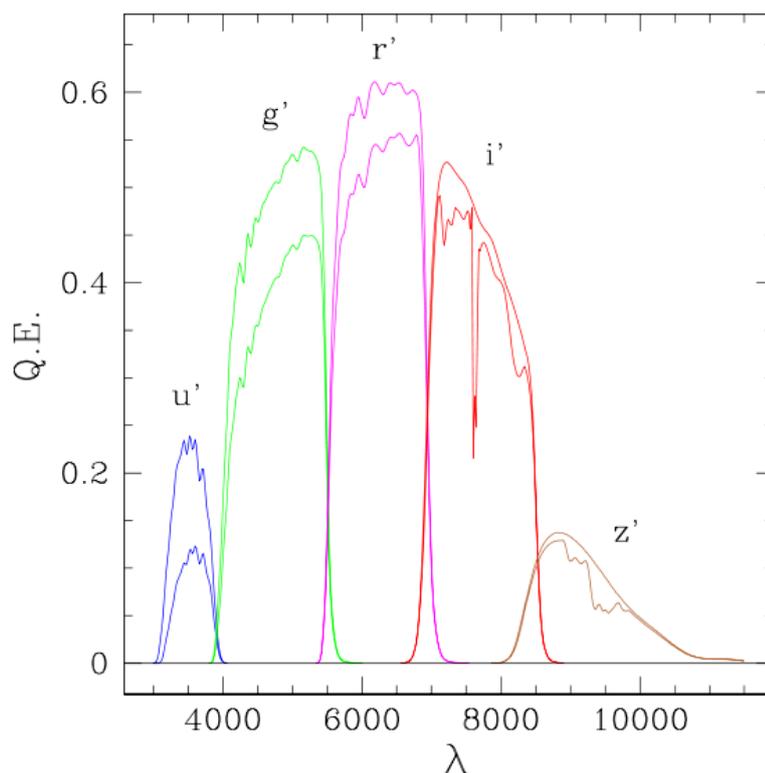


Figura 2.7 - Função resposta dos filtros do SDSS. A figura mostra a resposta de cada filtro usado pelo SDSS de acordo com o comprimento de onda.  
Fonte: Adaptada de (GUNN et al., 1998).

Utilizamos neste trabalho dados do SDSS-DR7 Legacy Survey obtidos a partir de requisições SQL simples (daqui em diante estas requisições serão chamadas, como no inglês, de “query” ou “queries”, no plural), feitas à base de dados SDSS “Sky Server” do DR7 “Catalog Archive Server” (CAS)<sup>14</sup>. Os objetos foram selecionados no intervalo de magnitudes  $r$  de  $14^m$  a  $21^m$ . Os dados foram então separados em dois conjuntos diferentes: o *espectroscópico*, ou de *treinamento*, e o de *aplicação*. O conjunto de treinamento foi construído com os dados de objetos que possuem tanto medidas fotométricas quanto espectroscópicas. Por outro lado, a construção do conjunto de aplicação foi desenvolvida sem vínculo algum, ou seja, nele estão todos os objetos do intervalo, tanto eles medidas espectroscópicas ou não.

A query usada na obtenção dos dados do conjunto de treinamento foi escrita com se segue:

```
SELECT
  p.objID, p.ra, p.dec, s.specObjID, p.r,
  p.psfMag_r, p.modelMag_r, p.petroMag_r,
  p.fiberMag_r, p.petroRad_r, p.petroR50_r,
  p.petroR90_r, p.lnLStar_r, p.lnLExp_r,
  p.lnLDeV_r, p.mE1_r, p.mE2_r, p.mRrCc_r,
  p.type_r, p.type, s.specClass
FROM PhotoObj AS p
  JOIN SpecObj AS s ON s.bestobjid = p.objid
WHERE
  p.r >= 14.0 AND p.r <= 21.0
```

Esta query retornou dados fotométricos e espectroscópicos para 1.030.220 objetos. Cada um destes objetos foi classificado pelo pipeline do SDSS-DR7 com base em seus espectros como estrelas, galáxias, quasares e desconhecidos<sup>15</sup>. Os quasares (abreviação de *quasi-stellar radio source*, ou fonte de rádio quase-estelar) são objetos astronômicos distantes, de alta energia e valores elevados de *redshift*. A imagem de um quasar é maior que a de uma estrela, porém menor do que o mínimo necessário para ser considerado uma galáxia, daí o termo “quase-estelar”. Contudo, neste trabalho, estamos interessados apenas em distinguir

<sup>14</sup>O CAS está hospedado no endereço <http://cas.sdss.org/astrodr7/en/>. Requistamos dados fotométricos obtidos através do `photoObj view` do CAS, e espectroscópicos através do `specObj view`.

<sup>15</sup>Uma descrição detalhada do processo de classificação espectroscópica do SDSS-DR7 foge ao escopo desta dissertação. Na Seção 2.6.1, fornecemos uma descrição sucinta da metodologia utilizada pelo pipeline do SDSS-DR7. Para uma descrição mais detalhada, o leitor pode buscar a página [http://www.sdss.org/dr7/algorithms/redshift\\_type.html](http://www.sdss.org/dr7/algorithms/redshift_type.html).

entre estrelas e galáxias, ou, em outras palavras, entre objetos pontuais e extensos. Os quasares são objetos distintos e representam uma fração de 9,1% do resultado da query, tendo a maioria deles um ou mais pixels saturados<sup>16</sup>. Os quasares, assim como os objetos classificados com desconhecidos foram descartados. Apesar dos quasares, descartamos também todos os demais objetos que apresentavam um ou mais pixels saturados. O pipeline do SDSS-DR7 fornece, além das referidas acima, a classe `STAR_LATE`, que engloba estrelas com bandas moleculares, do tipo M ou mais frias. Estes objetos também foram descartados, pois são um subtipo específico de estrelas e representam uma fração  $\sim 4\%$  da amostra retornada pela query. Nosso objetivo foi construir um conjunto espectroscópico o mais abrangente possível do ponto de vista de uma classificação unívoca estrela ou galáxia. No final ficamos com um conjunto de 880.715 objetos com classificação espectroscópica unicamente estrela ou galáxia, sem pixels saturados, e que apresentam somente valores com significado físico (foram retiradas 276 galáxias e 160 estrelas com valores  $-9999$ , indicativos de erro de medição, em um ou mais atributos). A Tabela 2.2 mostra a fração de objetos por classe do conjunto retornado pela query.

O conjunto de aplicação foi construído de forma similar ao de treinamento a partir seguinte query:

---

<sup>16</sup>Cada pixel de uma câmera CCD possui uma quantidade limite de energia que pode ser medida. Quando dizemos que um pixel está saturado, significa que ele atingiu este limite. Tratando-se de objetos pontuais que não são resolvidos pelo detector, os pixels saturados causam grande incerteza sobre as medidas fotométricas

```

SELECT
  objID, ra, dec, r, psfMag_r, modelMag_r,
  petroMag_r, fiberMag_r, petroRad_r,
  petroR50_r, petroR90_r, lnLStar_r,
  lnLExp_r, lnLDeV_r, mE1_r, mE2_r,
  mRrCc_r, type_r, type
FROM PhotoObj
WHERE
  r >= 14.0 AND r <= 21.0

```

Tabela 2.2 - Distribuição dos objetos do DR7 com espectroscopia (1.030.220). A distribuição é baseada na classificação espectral fornecida pelo CAS na forma do atributo `specClass`. O `specClass` possui seis diferentes classes: `unknown`, `star`, `galaxy`, `qso`, `hiz_qso`, `sky`, `star_late` e `gal_em`. A descrição foi adaptada da página <http://cas.sdss.org/dr7/en/help/browser/browser.asp>.

Classe	nº de objetos	Porcentagem(%)	Descrição
UNKNOWN	9.989	0,97%	Espectro não classificado ( $zConf < 0.25$ ).
STAR	84.135	8,16%	Espectro de uma estrela.
GALAXY	800.243	77,68%	Espectro de uma galáxia.
QSO	94.499	9,17%	Espectro de um objeto quase-estelar (quasar).
HIZ_QSO	7.549	0,73%	Espectro de um quasar com um desvio para o vermelho elevado ( <i>high-redshift</i> ; $z > 2.3$ ), cujo desvio foi confirmado pelo estimador Lyalpha (veja <a href="http://www.sdss.org/dr7/algorithms/redshift_type.html">http://www.sdss.org/dr7/algorithms/redshift_type.html</a> ).
STAR_LATE	33.805	3,28%	Espectro de uma estrela do tipo M ou mais fria dominado por linhas moleculares.

Foram retornados dados fotométricos para 69.545.326 objetos. Diferente do conjunto de treinamento, não impusemos qualquer restrição no conjunto de aplicação. Nosso desejo foi gerar um catálogo de estrelas e galáxias para estes objetos da forma mais geral possível. Desta maneira, pretendemos que a comunidade astronômica tenha liberdade para selecionar objetos correlacionando nossa classificação com a do SDSS, tendo como base os resultados apresentados nas Seções 2.6 e 2.7.

Ambos os conjuntos, de treinamento e de aplicação, foram compostos por objetos do SDSS Legacy. O SDSS-DR7 é composto por dados de três projetos de mapeamento celeste. O SEGUE (*the Sloan Extension for Galactic Understanding and Exploration*) tem por objetivo prover dados para o estudo da estrutura e da história da nossa galáxia (Via-Láctea). O “Sloan Supernova Survey” realizou repetidas observações de uma faixa equatorial sul com 300 graus quadrados com a finalidade de identificar e medir supernovas e outros processos variáveis. O “Sloan Legacy Survey”, chamado aqui de Legacy, é a parte principal do SDSS, completando o objetivo original do Sloan. O conjunto final de dados do Legacy inclui 230 milhões de objetos observados em 8.400 graus quadrados do céu. Escolhemos os dados do Legacy, pois nosso maior interesse reside nas observações de objetos distantes (extragalácticos) nos limite superior de magnitude. Uma descrição mais detalhada dos mapeamentos do SDSS pode ser encontrada na página <http://www.sdss.org/>.

## 2.6 Separação Estrela/Galáxia para o SDSS-DR7

Os dados espectroscópicos fornecidos pelo SDSS provêm uma classificação confiável de um objeto em estrela ou galáxia. Apesar do grande volume da amostra espectroscópica ( $\sim 1$  milhão de objetos), ela contém somente uma pequena fração dos objetos presentes na amostra fotométrica disponibilizada pelo SDSS-DR7 (230 milhões). Como podemos classificar com uma confiabilidade significativa os objetos para os quais não há dados espectroscópicos disponíveis? O *pipeline* do SDSS fornece uma classificação utilizando um método paramétrico baseado na diferença entre as magnitudes `psfMag` e `modelMag` (veja a Seção 2.6.1). Contudo, esta classificação não é muito precisa para objetos com magnitudes superiores a  $19.0^m$  (veja Figuras 2.13 e 2.18).

Nós tiramos proveito do grande número de objetos na amostra espectroscópica do SDSS-DR7, para os quais sabemos a verdadeira classificação de cada um, e treinamos uma AD para classificar todos os objetos do Legacy com base somente em seus atributos fotométricos. Esperamos que, utilizando esta amostra espectroscópica com um grande número de exemplos de treinamento, a AD resultante seja capaz de manter uma boa precisão, mesmo no limite superior de magnitude.

Tabela 2.3 - Atributos SDSS-DR7 usados para separação estrela/galáxia.

Atributo	Variável CAS
Magnitude PSF	<code>psfMag</code>
Magnitude de Fibra	<code>fiberMag</code>
Magnitude de Petrosian	<code>petroMag</code>
Magnitude Modelo	<code>modelMag</code>
Raio de Petrosian	<code>petroRad</code>
Raio contendo 50% do fluxo de Petrosian	<code>petroR50</code>
Raio contendo 90% do fluxo de Petrosian	<code>petroR90</code>
Verossimilhança PSF	<code>lnLStar</code>
Verossimilhança Exponencial	<code>lnLExp</code>
Verossimilhança deVaucouleurs	<code>lnLDeV</code>
Momentos Adaptativos	<code>mRrCc</code> , <code>mE1</code> e <code>mE2</code>
Classificação Espectroscopica	<code>specClass</code>

### 2.6.1 Atributos

Selecionamos como critérios de classificação 13 atributos fotométricos do SDSS e um único atributo espectrocópico (`specClass`), como mostrado na Tabela 2.3.

Este conjunto de atributos fotométricos é o mesmo, tanto para o conjunto espectrocópico (de treinamento) quanto para o de aplicação. Na escolha dos atributos é levantada a questão de quais atributos produziriam uma separação estrela/galáxia mais precisa, mas a grande variedade de atributos medidos pelo SDSS para cada objeto fotométrico coloca a análise desta questão fora do escopo deste trabalho. Escolhemos os atributos aqui utilizados com base em uma forte correlação, já conhecida ou esperada, entre o atributo e a classe a qual pertence o objeto. Esses atributos são:

- A *magnitude PSF* (`psfMag`), descrita em detalhes em Stoughton et al. (2002), é obtida através do ajuste de uma “Point Spread Function” (PSF) de perfil gaussiano à distribuição de brilho do objeto. Esperamos que a magnitude PSF seja uma boa medida de fluxo para estrelas (objetos pontuais), pois esta medida tende a superestimar o fluxo no caso das galáxias (objetos extensos), devido a suas formas irregulares.
- A *magnitude de fibra* (`fiberMag`) mede o fluxo através de uma fibra óptica de diâmetro 3”. A função principal das fibras ópticas do SDSS é a realização de espectroscopia de vários objetos simultaneamente e a separação de objetos

sobrepostos nas imagens.

- A *magnitude de petrosian* (**petroMag**) é uma medida de fluxo baseada em Petrosian (1976), e descrita em detalhes por Yasuda et al. (2001). Segundo Yasuda et al., define-se a função

$$\eta(r) = \frac{2\pi \int_{0,8r}^{1,25r} I(r')r' \{ \pi[(1,25r)^2 - (0,8r)^2] \}^{-1} dr'}{2\pi \int_0^r I(r')r'(\pi r^2)^{-1} dr'} , \quad (2.9)$$

onde  $I(r')$  é o perfil azimutal médio da superfície de brilho do objeto em questão. O raio de petrosian  $r_p$  é definido de forma que  $\eta(r_p) = 0,2$ , sendo o fluxo medido dentro de um raio  $2r_p$  a magnitude de petrosian<sup>17</sup>.

- O “pipeline” do SDSS ajusta dois diferentes modelos de brilho superficial as imagens bi-dimensionais de um objeto em cada uma das 5 bandas fotométricas: um perfil de deVaucouleurs e um perfil exponencial. Estes são modelos teóricos criados para reproduzir o perfil de galáxias, portanto, como no caso da PSF para as estrelas, é esperado que esta magnitude seja uma boa medida do fluxo galáctico, mas subestime o fluxo estelar devido as bordas irregulares reproduzidas pelos modelos galácticos. A *magnitude modelo* (**modelMag**) é obtida a partir do melhor dos dois ajustes, deVaucouleurs ou exponencial<sup>18</sup>.
- Os atributos **petroR50** e **petroR90** são os raios contendo respectivamente 50% e 90% do fluxo de Petrosian para cada banda. Estes dois atributos não são corrigidos para o “seeing” e podem fazer com que o brilho superficial de um objeto pontual comparável a uma PSF seja subestimado. No entanto, a amplitude deste efeito não é bem caracterizado, contudo, os algoritmos de aprendizado de máquina, ainda assim, podem localizar padrões para distinguir estrelas de galáxias.
- A *Verossimilhança* é a probabilidade de se alcançar o valor medido do chi-quadrado para cada um dos modelos de superfície de brilho: deVaucouleurs (**deV\_L**), exponencial (**exp\_L**) e PSF (**star\_L**)<sup>19</sup>. Por exemplo, **star\_L** é a probabilidade de que um objeto teria no mínimo o valor de chi-quadrado medido se sua distribuição de brilho é realmente bem representada pelo modelo PSF do SDSS. A *Verossimilhança Fracional* é uma grandeza calculada para cada modelo pelas equações 2.10, 2.11 e 2.12, e é um bom discriminante entre estrelas

<sup>17</sup>O uso dos valores  $\eta = 0,2$  e  $2r_p$  são explicados em mais detalhes em Yasuda et al. (2001) e na página [http://www.sdss.org/dr7/algorithms/photometry.html#mag\\_model](http://www.sdss.org/dr7/algorithms/photometry.html#mag_model)

<sup>18</sup>Para mais detalhes veja <http://www.sdss.org/dr7/algorithmsphotometry.html>.

<sup>19</sup>Os atributos SDSS: **lnLStar**, **lnLExp** e **lnLDeV**, são os logaritmos niperianos dos valores da *Verossimilhança*

e galáxias. Neste trabalho somente temos interesse em localizar padrões para separação estrela/galáxia com base nos atributos obtidos diretamente no banco de dados do SDSS-DR7, e não calcular novos atributos. Entretanto, se a *Verossimilhança Fracional* é um bom discriminante, esperamos que os algoritmos de AD sejam capazes de localizar padrões na *Verossimilhança* capazes de separar estrelas de galáxias.

$$f(\text{deV\_L}) = \frac{\text{deV\_L}}{\text{deV\_L} + \text{exp\_L} + \text{star\_L}} \quad (2.10)$$

$$f(\text{exp\_L}) = \frac{\text{exp\_L}}{\text{deV\_L} + \text{exp\_L} + \text{star\_L}} \quad (2.11)$$

$$f(\text{star\_L}) = \frac{\text{star\_L}}{\text{deV\_L} + \text{exp\_L} + \text{star\_L}} \quad (2.12)$$

- Os *Momentos Adaptivos* `mRrCc`, `mE1` e `mE2` são momentos de segunda ordem da intensidade da imagem de um objeto celeste, e são medidos a partir de uma função peso radial (“radial weight function”) adaptada a forma e tamanho da imagem. Uma descrição mais detalhada pode ser encontrada em [Bernstein e Jarvis \(2002\)](#). Os momentos adaptivos podem ser boas medidas de elipticidade, e são dados pelas equações:

$$\langle \text{col}^2 \rangle = \frac{\sum [I(\text{col}, \text{row}) * w(\text{col}, \text{row}) * \text{col}^2]}{\sum [I * w]}, \quad (2.13)$$

$$\text{mRrCc} = \langle \text{col}^2 \rangle + \langle \text{row}^2 \rangle, \quad (2.14)$$

$$\text{mE1} = \frac{\langle \text{col}^2 \rangle - \langle \text{row}^2 \rangle}{\text{mRrCc}}, \quad (2.15)$$

$$\text{mE2} = \frac{\langle \text{col}^2 \rangle * \langle \text{row}^2 \rangle}{\text{mRrCc}}, \quad (2.16)$$

onde `col` e `row` são as coordenadas de um pixel no detector,  $I(\text{col}, \text{row})$  é a intensidade no pixel e  $w(\text{col}, \text{row})$  é uma função peso. Logo, baseados no fato de imagens estelares apresentarem uma baixa elipticidade (objetos pontuais) e imagens galácticas apresentarem valores elevados de elipticidade; esperamos que estes atributos sejam bons discriminantes entre estrelas e galáxias.

- O *atributo espectroscópico* `specClass` armazena a classificação espectroscópica de cada objeto. As possíveis classificações são `unknown`, `star`, `galaxy`, `qso`, `hiz_qso`, `sky`, `star_late` ou `gal_em`, mas como mostrado na Tabela 2.2, as classes `sky` (espectro do céu puro) e `gal_em` (espectro de uma galáxia com linhas de emissão) não são atribuídas a nenhum dos objetos do conjunto espectroscópico. O pipeline do SDSS-DR7

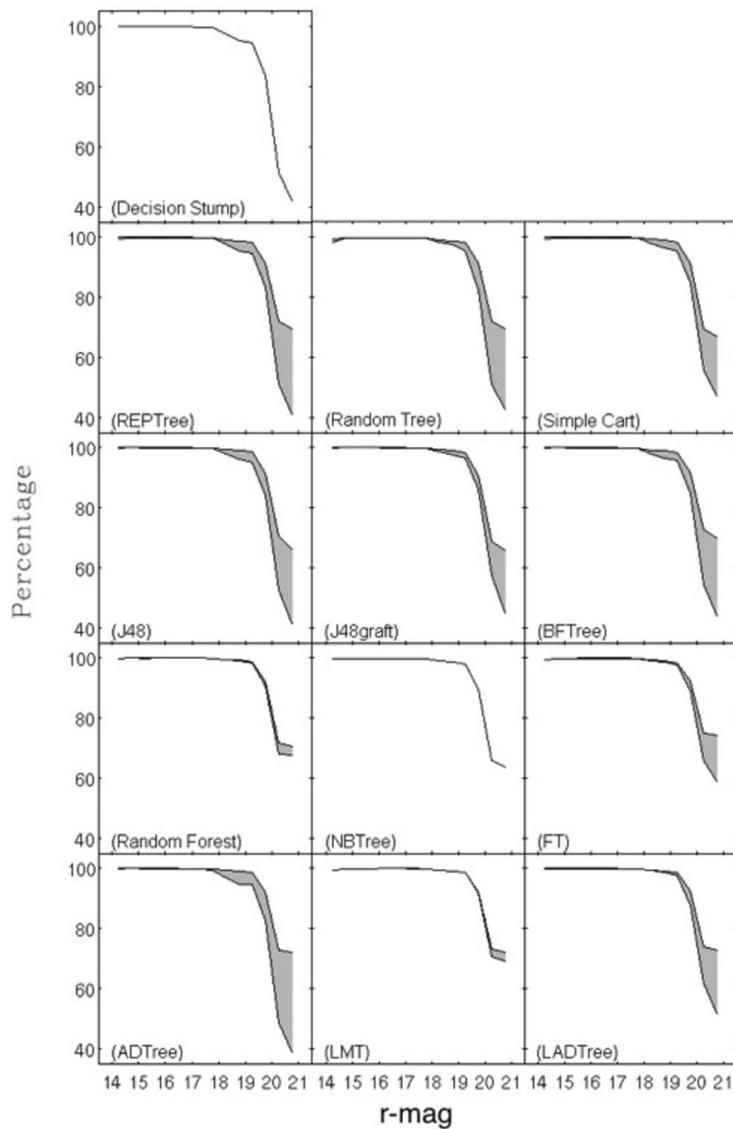


Figura 2.8 - Resultados obtidos para a exploração do espaço de parâmetros para cada um dos 13 algoritmos do WEKA. As áreas escuras são os lugares geométricos das funções de completude obtidas com o procedimento de CV para cada conjunto de parâmetros testado.

possui um algoritmo que analisa os espectros medindo linhas de emissão e absorção e estimando o *redshift* com base nessas medidas. Com base na correlação entre o *redshift* estimado e a presença de certas linhas, a classificação é atribuída. Para mais detalhes o leitor pode consultar a página do SDSS, [http://www.sdss.org/dr7/algorithms/redshift\\_type.html](http://www.sdss.org/dr7/algorithms/redshift_type.html).

## 2.6.2 Seleção do Melhor Algoritmo Aplicado a Separação Estrela/Galáxia

Como discutido anteriormente na Seção 2.3, o aplicativo de mineração de dados WEKA possui 13 diferentes algoritmos para construção (treinamento) uma AD a partir de um conjunto de exemplos de treinamento. Cada um destes algoritmos emprega procedimentos diferentes no treinamento da AD, e usando diferentes combinações de parâmetros internos um algoritmo pode construir AD distintas a partir do mesmo conjunto de exemplos. Para cada algoritmo, testamos várias combinações de parâmetros internos (sempre usando o mesmo conjunto de treinamento). Listamos na Tabela 2.4 todos os parâmetros internos testados, assim como os intervalos de variação e passos usados nos testes. Os intervalos e os passos utilizados no estudo da variação de cada parâmetro foram escolhidos com base na variação da função de completeza. Analisando a variação dos parâmetros contínuos e definimos passos que resultavam em uma variação da função de completeza de pelo menos 5%. Variações das funções de completeza e contaminação entorno de 5% são insignificantes do ponto de vista astrofísico. Tais variações são muito menores, por exemplo, do que a influência dos erros das magnitudes na contagem de galáxias. Caso a função de completeza apresentasse uma variação inferior à 5%, o valor padrão do WEKA para o parâmetro em questão era usado. Mostramos na primeira coluna, entre parênteses, o número de testes executados para cada algoritmo. Utilizamos o procedimento de CV descrito na Seção 2.4 para gerar, para cada combinação de parâmetros de cada algoritmo, uma função de completeza média. Estas funções médias nos permitiram comparar as diversas combinações de parâmetros para um mesmo algoritmo e também os algoritmos. Nesta seção, discutimos os testes realizados e analisamos seus resultados no intuito de selecionar o algoritmo de melhor desempenho – com seu conjunto de parâmetros otimizado – que será utilizado para separar estrelas e galáxias em todo SDSS-DR7 Legacy.

Primeiramente exploramos exaustivamente o espaço de parâmetros de cada algoritmo para determinar quais deles causam mudanças significativas na função de completeza, e descartar os irrelevantes<sup>20</sup>. Testamos a sensibilidade da função de completeza a variação de cada parâmetro individualmente (“testes simples”) e em combinação com variações de outros parâmetros (“testes compostos”). Para cada conjunto de funções de completeza geradas por um teste de parâmetros  $\nu$ , seja simples ou composto, calculamos a dispersão  $\sigma_{m_i}$  nos pontos médios  $m_i = 14.25^m + i * 0.5^m$  e a média sobre todos os pontos do intervalo:

$$\sigma_\nu = \frac{1}{14} \sum_{i=1}^{14} \sigma_{m_i} \quad . \quad (2.17)$$

---

<sup>20</sup>Note que os algoritmos Decision Stump e NBTree não possuem parâmetros internos.

Assim, um parâmetro é considerado irrelevante se  $\sigma_\nu \leq 5\%$ . Este procedimento nos permitiu descartar, em média, um parâmetro por algoritmo.

Em um segundo estágio, buscamos o valor otimizado para cada um dos parâmetros restantes. Para isso, definimos primeiramente uma faixa de variação para cada parâmetro restante utilizando o mesmo critério  $\sigma_\nu \leq 5\%$  usado anteriormente, ou seja, qualquer intervalo de valores em cuja variação da função de completude resulte em  $\sigma_\nu \leq 5\%$  foi descartado. Uma vez definidos os limites de variação, os parâmetros foram combinados de forma que pudéssemos comparar os resultados e selecionar a combinação otimizada (lembrando que cada procedimento descrito até aqui foi executado para cada um dos algoritmos de forma independente). Os resultados destes testes são apresentados na Figura 2.8, que mostra a mudança na função de completude obtida com o CV quando variamos os parâmetros internos de cada algoritmo. Pode-se observar que em magnitudes mais brilhantes,  $r \lesssim 19$ , os algoritmos se comportam de maneira muito similar, e a eficiência de cada um se mantém estável diante da variação de seus parâmetros internos.

Verificamos então o desempenho de cada um dos 13 algoritmos comparando suas funções de completude e contaminação, e também seus tempos de processamento, quando usamos suas configurações paramétricas ótimas. Definimos as quantidades  $\langle \text{Compl} \rangle_{\text{bright}}$ ,  $\langle \text{Compl} \rangle_{\text{faint}}$  e  $\text{Compl}_{20.75}$ , que são as completudes médias (veja Seção 2.4) nos intervalos  $14 \leq r < 19$ ,  $19 \leq r \leq 21$  e  $20.5 \leq r \leq 21$ , respectivamente. Os resultados deste estudo comparativo são mostrados na Tabela 2.5. A primeira coluna contém o nome do algoritmo, como apresentado na Seção 2.3. A segunda coluna contém o número de parâmetros internos significativos para cada algoritmo. A terceira coluna fornece o tempo de processamento que cada algoritmo gasta para construir uma AD com o conjunto de treinamento descrito na Seção 2.5 (os testes foram realizados usando um PC com processador AMD Phenom X3 8650 triple-core de 64 bits, com 2.3GHz cada núcleo e 4Gb de memória RAM). As colunas 4, 5 e 6 contém as médias  $\langle \text{Compl} \rangle_{\text{bright}}$ ,  $\langle \text{Compl} \rangle_{\text{faint}}$  e  $\text{Compl}_{20.75}$ , e seus respectivos desvios padrão. As linhas da Tabela 2.5 estão ordenadas de acordo com valor de  $\langle \text{Compl} \rangle_{\text{faint}}$ .

A análise dos resultados exibidos na Tabela 2.5 (e da Figura 2.8) mostra que todos os algoritmos apresentam um desempenho similar na faixa de magnitudes brilhantes ( $r < 19$ ). No entanto, para magnitudes  $r \geq 19$  a performance de cada algoritmo varia de forma significativa (variações maiores que 5%). O algoritmo Decision Stump, que não possui parâmetros internos, como esperado, é o que apresenta o pior desempenho, apesar de ser o de processamento mais veloz. O NBTree, que também não possui parâmetros internos, é consideravelmente melhor, mas tem um custo de processamento maior. Dentre os demais algoritmos, os mais rápidos, incluindo o Simple Cart, J48 e J48graft, REPTree e Random

Tree, apresentam uma completeza média no limite superior de magnitudes ( $\langle \text{Compl} \rangle_{\text{faint}}$ ) entre 81% e 83%, mas suas completezas no intervalo  $20.5 \leq r \leq 21$  ( $\text{Compl}_{20.75}$ ) ficam abaixo de 70%. Os demais algoritmos apresentam elevados valores de completeza, mesmo em magnitudes mais fracas, mas ao custo de um tempo de processamento mais elevado. Note que todos os algoritmos são igualmente robustos, como mostram as dispersões das funções de completeza na Tabela 2.5. Concluímos que o algoritmo FT foi o de melhor performance, pois apresenta uma boa precisão, mesmo em magnitudes mais fracas, com um modesto custo em tempo de processamento. De fato, como mostrado na Tabela 2.5, o algoritmo FT não é somente o mais preciso dentre os 13 testados, como também é muito robusto (classificado em segundo lugar como o de menor dispersão da completeza média).

Não é surpreendente que o FT produza o melhor resultado. Como dito anteriormente na Seção 2.3.4, Gama (2004) afirma que as *Functional trees* “apresentam melhores resultados do que árvores univariantes, árvores multivariantes padrão e árvores modelo, principalmente para grandes conjuntos de dados”. Quando comparamos, por exemplo, os desempenhos das AD geradas pelo FT e pelo NBTree, constatamos que apesar de ambos construírem árvores nas quais todos os 13 atributos estão presentes<sup>21</sup> seus desempenhos são muito diferentes. Apesar de não termos estudado a fundo a metodologia de cada algoritmo, acreditamos que uma das razões desta diferença é o emprego de combinações lineares de atributos pelo FT (veja Seção 2.3.4). A utilização de combinações lineares nos nós favorece uma divisão mais refinada do espaço de atributos (GAMA, 2004). Especificamente para a comparação entre o NBTree e o FT, Kohavi (1996) afirma que os classificadores Naive-Bayes usados nas folhas apresentam melhores resultados com conjuntos de dados pequenos, enquanto Gama (2004) apresenta resultados que mostram o bom desempenho das *Functional trees* na análise de grandes volumes de dados. Essa comparação e os demais resultados mostrados aqui reforçam a necessidade de confrontar diferentes algoritmos de AD.

### 2.6.3 Construindo a Árvore de Decisão Final

Sendo constatado que o algoritmo FT apresentou a melhor performance na separação estrela/galáxia durante o teste de “cross-validation”, devemos agora definir um conjunto de treinamento para a construção da AD definitiva para desempenhar a separação dos objetos do catálogo fotométrico do SDSS-DR7. Como descrito na Seção 2.5, o banco de dados CAS fornece 880.715 objetos com classificação espectroscópica estrela ou galáxia, e estes objetos compõem nosso conjunto total de treinamento. Contudo, a utilização do conjunto total de de treinamento com a implementação WEKA do algoritmo FT requer uma quantidade demasiadamente grande de memória RAM. Os 4Gb de memória do PC utilizado neste

---

<sup>21</sup>Todos os algoritmos constroem ADs nas quais todos os 13 atributos são testados. As exceções são o Decision Stump (01) e o ADTree (12).

trabalho não foram suficientes para gerar uma AD com todo conjunto de treinamento. Uma vez que uma análise do consumo de memória dos códigos do WEKA e sua otimização estão fora do escopo desta dissertação, procuramos determinar se a precisão da AD gerada depende fortemente do tamanho do conjunto de treinamento e dos valores exatos dos atributos procedendo um teste utilizando subconjuntos dos dados de treinamento com valores perturbados. Para cada atributo fotométrico discutido na Seção 2.6.1, geramos um valor perturbado,

$$X = X_{obs} + \sigma u \quad , \quad (2.18)$$

onde  $X_{obs}$  é o valor observado do atributo e  $u$  é um desvio Gaussiano randômico, com a dispersão  $\sigma$  calculada a partir do primeiro e quarto quartis da distribuição de incertezas do atributo em questão (obtida do CAS). Subdividimos então o conjunto total de treinamento com atributos perturbados em 4 subconjuntos com 221.029 objetos cada. Os objetos de cada subconjunto foram escolhidos randomicamente utilizando a classe *Math* do JAVA de forma que os subconjuntos apresentassem a mesma distribuição de objetos por classe do conjunto total de treinamento. Foram geradas 4 ADs, uma para cada subconjunto, que foram aplicadas no conjunto total de treinamento para separar estrelas e galáxias. Os resultados destes testes são mostrados na Figura 2.9, onde fica evidente que tanto a completeza quanto a contaminação são invariantes no intervalo de magnitudes  $14 \leq r < 20$ . No limite de magnitudes fracas,  $20 \leq r \leq 21$ , a função de completeza apresenta uma pequena variação, aproximadamente menor que 5%. Estes resultados confirmam que a AD gerada pelo FT é praticamente insensível a pequenas variações nos atributos de objetos individuais, o que sugere que podemos reduzir o conjunto de treinamento sem perda significativa de precisão. O conjunto de treinamento reduzido gasta menos memória e consome um tempo de processamento consideravelmente menor<sup>22</sup>.

Testamos também a dependência do sucesso da AD gerada pelo FT na separação de objetos com o tamanho do conjunto de treinamento. Esperamos que, nos mapeamentos astronômicos futuros, feitos no ótico, os dados espectroscópicos estejam disponíveis para uma fração bem pequena dos objetos observados fotometricamente, principalmente no limite de magnitudes mais fracas. Para  $r > 19$ , que constitui o limite de magnitudes fracas do nosso conjunto de treinamento, foram construídas ADs com o FT, utilizando amostras de treinamento com um número de objetos com  $r > 19$  variando entre 10% e 100% do conjunto total de treinamento. Os resultados exibidos na Figura 2.10 mostram que a completeza permanece essencialmente invariante quando 20% ou mais dos objetos do conjunto total de treinamento é utilizado (aproximadamente 7.100 objetos). Estes resultados sugere

---

<sup>22</sup>O FT rodando com o conjunto total no PC descrito na Seção 2.6.2, gasta em média 1.6 horas até registrar falta de memória. Já rodando com o conjunto reduzido, gasta em torno de 40 minutos para gerar a árvore

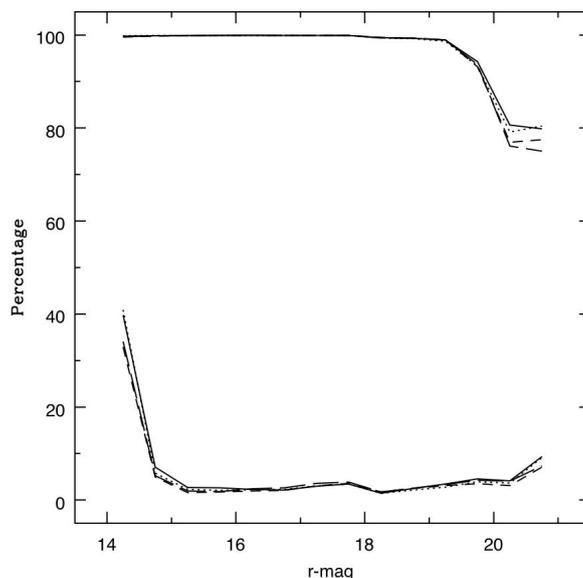


Figura 2.9 - Funções de completudeza (curvas superiores) e contaminação (curvas inferiores) para todos os quatro conjuntos de dados perturbados usados para treinar um AD com o algoritmo FT. Cada conjunto de dado é representado por um tipo de linha diferente.

rem que poderemos desempenhar a separação estrela/galáxia nos mapeamentos profundos vindouros, mesmo com um conjunto de treinamento modesto, desde que haja imagens com resolução suficiente.

Selecionamos os dados a serem utilizados no treinamento da AD final considerando separadamente os objetos fracos e os brilhantes. Para magnitudes  $14 \leq r < 19$  selecionamos 1/4 dos objetos do conjunto total de treinamento, mantendo a mesma distribuição de objetos por classe apresentada por este último. Obtivemos com este procedimento 205.348 objetos. Na faixa fraca de magnitudes, com  $19 \leq r \leq 21$ , mantivemos todos os objetos disponíveis no conjunto de treinamento. Objetos fracos possuem uma amostra espectroscópica pobre, portanto, optamos por fornecer ao algoritmo FT toda informação disponível na tentativa de manter a completudeza elevada para as magnitudes mais fracas.

O conjunto de exemplos de treinamento final contém 240.712 objetos com 13 atributos fotométricos cada, extraídos do conjunto espectroscópico. A AD resultante foi aplicada à separação de objetos do catálogo fotométrico do SDSS-DR7 com magnitudes  $r$  entre 14 e 21, criado a partir do mapeamento Legacy. A Tabela 2.6 mostra uma pequena fração do catálogo estrela/galáxia gerado pela classificação da AD; o catálogo completo está disponível como uma tabela eletrônica no endereço [http://www.lac.inpe.br/bravo/services\\_data.jsp](http://www.lac.inpe.br/bravo/services_data.jsp). A coluna 1 contém o ObjID único do SDSS e a coluna 2 a magnitude modelMag

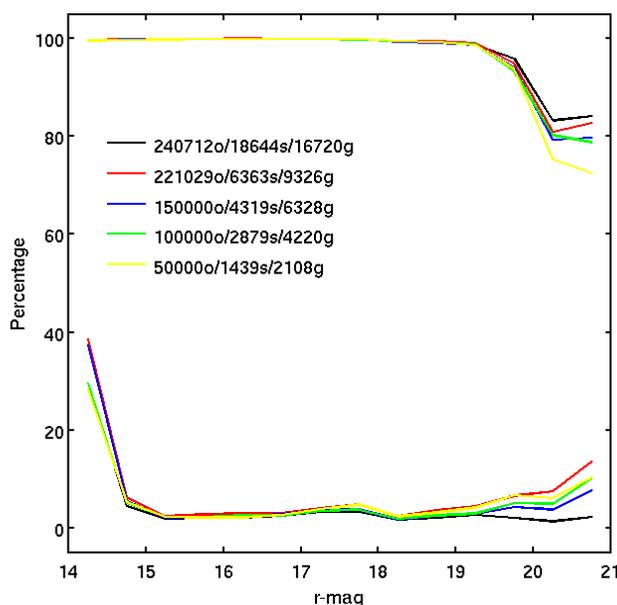


Figura 2.10 - Funções de completudeza (curvas superiores) e contaminação (curvas inferiores) para todos os cinco subconjuntos de dados retirados do conjunto total de treinamento e usados para treinar um AD com o algoritmo FT. O subconjunto de 240.712 objetos contém 100% dos objetos do conjunto total para  $r > 19$ , enquanto que para  $r \leq 19$  a distribuição de objetos por classe do conjunto total é utilizado. Os demais subconjuntos utilizam a distribuição de objetos por classe do conjunto total em todo o intervalo de magnitudes estudado [14, 21].

na banda  $r$ . As colunas 3 e 4 contém, respectivamente, a classificação paramétrica do SDSS utilizando somente a banda  $r$  e todas as cinco bandas. As classificações do SDSS encontradas no catálogo fotométrico utilizado são numéricas e possuem os valores 0 (Unknown), 3 (Galaxy), 6 (Star), 8 (Sky) – todas as classificações do SDSS são mostradas na Tabela 2.7. A coluna 5 fornece nossa classificação obtida pela AD gerada com o FT, onde o número 1 representa as estrelas e o 2 as galáxias. Todos os objetos do catálogo fotométrico foram classificados, independente da classificação fornecida pelo SDSS. Os objetos classificados pelo SDSS como 0 ou 8 devem ser vistos com cuidado, pois existe uma grande possibilidade de que esses objetos não sejam verdadeiramente objetos astronômicos.

## 2.7 Comparação com Outros Métodos de Separação Aplicados ao SDSS

Para verificar o real desempenho da nossa AD, comparamos a nossa separação de objetos para o catálogo fotométrico do SDSS-DR7 com outros gerados pela classificação paramé-

trica do próprio SDSS, pelo software para processamento de imagens 2DPHOT (BARBERA et al., 2008), e por Ball et al. (2006). Estas comparações utilizam somente objetos do conjunto espectroscópico onde a classificação verdadeira é conhecida. Todos os três métodos fornecem outras classificações diferentes de estrela ou galáxia. Contudo, como estamos interessados somente nestas duas classes, todas as amostras descritas neste capítulo são compostas exclusivamente por objetos classificados por todos os três métodos somente com estrelas ou galáxias.

### 2.7.1 Algoritmo FT Versus o Método do 2DPHOT

O 2DPHOT é um pacote de softwares para detecção e análise automática de fontes luminosas em imagens de campos profundos (“deep wide-field images”), ou seja, imagens de objetos distantes. Ele fornece fotometria integrada e de superfície para galáxias em uma imagem, e desempenha a separação estrela/galáxia definindo uma região no espaço de atributos<sup>23</sup> onde os objetos presentes são estrelas (BARBERA et al., 2008) – essa região é comumente chamada “stellar locus”. A comparação foi feita para um conjunto contendo 10.391 objetos da amostra espectroscópica que foram reprocessados com o 2DPHOT, e os resultados são mostrados na Figura 2.11. Observamos que ambos os classificadores apresentam funções de completude com um perfil similar em relação à magnitude, mas nossa AD gera quase nenhuma contaminação, enquanto que a contaminação gerada pelo 2DPHOT chega a  $\sim 40\%$ .

### 2.7.2 O Algoritmo FT Versus O Algoritmo Axis-Parallel

Ball et al. (2006) foram os primeiros a aplicar a metodologia de AD a um catálogo fotométrico inteiro do SDSS (DR3 - “Data Release 3”). Os autores utilizaram um tipo de AD conhecida como “axis-parallel Decision Tree” para fornecer uma probabilidade de que um objeto pertença a uma dentre três possíveis classes: estrelas, galáxias ou nsg (“neither star nor galaxy”, ou seja, nem estrela nem galáxia). Eles fizeram uso de seis índices de cor calculados a partir das magnitudes medidas pelo SDSS em cada banda para 477.068 objetos cuja classificação estrela/galáxia baseada em espectroscopia estava disponível. As funções de completude e contaminação para a separação de objetos realizada pelos autores e pela nossa AD, calculadas a partir de uma amostra de 561.070 objetos do catálogo dos autores, são mostradas na Figura 2.12. Estes resultados mostram que nossa AD tem uma precisão similar a AD de BALL et al. na classificação de galáxias, mas, no entanto, nossa separação apresenta uma contaminação menor do que a separação desempenhada pelos

---

<sup>23</sup>Esses atributos são parâmetros medidos pelo 2DPHOT. É importante observar que as imagens destes objetos foram reprocessadas com o 2DPHOT e, portanto, a classificação do 2DPHOT não utiliza nenhum dado fotométrico fornecido pelo SDSS.

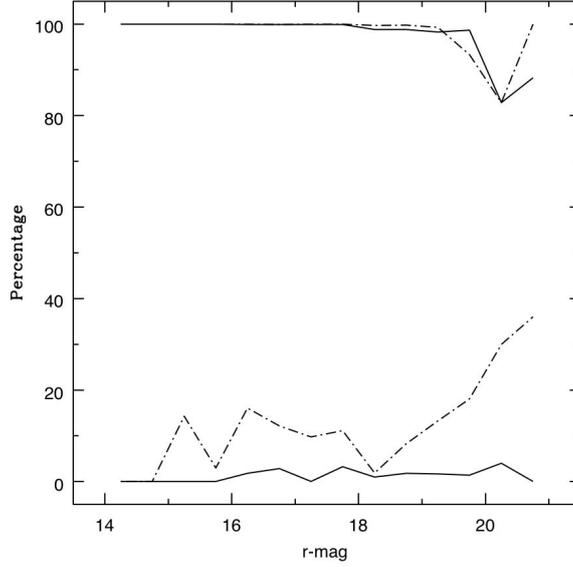


Figura 2.11 - A figura mostra a completeza (curvas superiores) e a contaminação (curvas inferiores) para uma amostra de 10.391 de objetos do SDSS reprocessados com o 2DPHOT e classificados unicamente com estrela ou galáxia. As linhas contínuas representam as funções de completeza e contaminação para nossa a classificação fornecida pela nossa AD, enquanto que a linha tracejada representa as mesmas funções para a classificação fornecida pelo 2DPHOT.

autores. No limite de magnitudes fracas,  $r \geq 19$ , nossa contaminação permanece constante e entorno de 3%, enquanto que a de BALL et al. chega a  $\sim 9\%$ .

### 2.7.3 O Algoritmo FT Versus o Método Paramétrico do SDSS

O “pipeline” do SDSS classifica um objeto em estrelas ou galáxias com base na diferença entre as magnitudes *psf* e *model* (veja Seção 2.6.1). Se a condição  $psfMag - modelMag > 0.145$  for satisfeita o objeto é classificado com galáxia; caso contrário é classificado como estrela.

Primeiramente analisamos o comportamento do FT quando comparado com o *pipeline* do SDSS na separação dos objetos do conjunto espectroscópico. Seleccionamos aleatoriamente 50 conjuntos de exemplos com 220.173 objetos do conjunto espectroscópico ( $\sim 20\%$ ) para construir 50 árvores. Cada árvore foi então aplicada nos 80% restantes, e funções médias de completeza e contaminação foram calculadas. Os resultados apresentados na Figura 2.13 mostram que a completeza atingida por nossa AD fica entorno de 78% para magnitudes  $r \geq 19$  com uma contaminação de no máximo 10%. Por outro lado, a completeza atingida pelo *pipeline* do SDSS cai para 60% para  $r \geq 19$ , e sua classificação apresenta uma alta

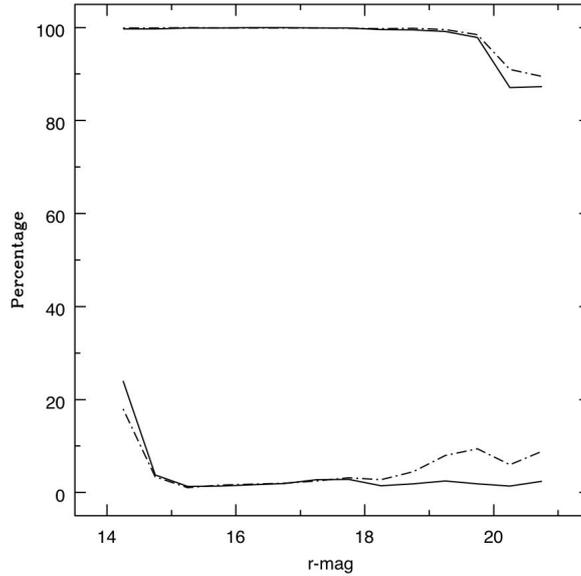


Figura 2.12 - Completeza (curvas superiores) e contaminação (curvas inferiores) para 561.070 objetos classificados por Ball et al. (2006) e com espectroscopia fornecida pelo SDSS. As linhas contínuas mostram as funções de completeza e de contaminação para a classificação da nossa AD, enquanto que as linhas tracejadas mostram o mesmo para a classificação de BALL et al..

contaminação em magnitudes mais brilhantes. Isso nos mostra que a aplicação da nossa AD possibilita um ganho de  $\sim 18\%$  em completeza para magnitudes mais fracas, quando comparada com o *pipeline* do SDSS.

Por fim, comparamos a classificação fornecida pela nossa AD com a do SDSS paramétrico para todos os objetos no conjunto de aplicação (veja Seção 2.5). Como, *a priori*, não há uma classificação verdadeira neste caso (diferente do conjunto espectroscópico), nós comparamos estes dois métodos assumindo que a nossa AD treinada com o FT fornece a classificação verdadeira. Essa decisão foi tomada com base nos resultados mostrados anteriormente neste capítulo. Na Figura 2.14 mostramos as curvas de completeza e contaminação para o método paramétrico do SDSS assumindo que nossa classificação é 100% correta. Somente consideramos objetos que fossem classificados pelo SDSS paramétrico unicamente como estrela ou galáxia, uma vez que estas são as únicas classificações fornecidas pela nossa AD. As demais classificações do SDSS paramétrico são irrelevantes, totalizando apenas 0.05% do conjunto de aplicação. Concluímos, que há um grande desacordo entre a classificação da nossa AD e o método paramétrico do SDSS quanto à classificação de estrelas, implicando em uma grande contaminação estelar na separação de objetos do SDSS. A curva de completeza mostra, para magnitudes entre 20.5 e 21, que os

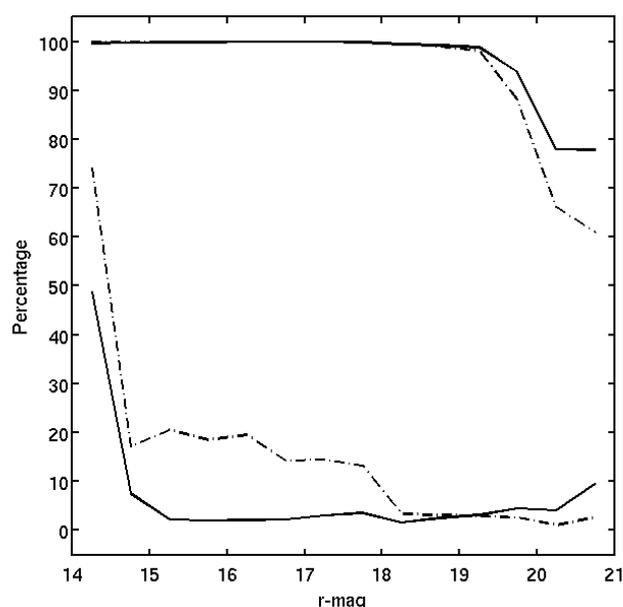


Figura 2.13 - As funções de completudeza (curvas superiores) e de contaminação (curvas inferiores) para o método paramétrico usado pelo *pipeline* do SDSS (linha tracejada) e para nosso método de AD (linha contínua) quando aplicados à separação de 880.715 objetos do conjunto espectroscópico (veja o texto).

dois separadores discordam para  $\sim 6\%$  de todo o conjunto de aplicação.

## 2.8 Máquinas de Comitê

Como um exame final dos algoritmos do WEKA, rodamos um experimento com uma máquina de comitê. Haykin (1999, cap. 7) introduz as máquinas de comitê como um conjunto de especialistas trabalhando juntos para solucionar um problema. Neste caso, os especialistas são redes neurais artificiais treinadas separadamente para resolver o mesmo problema. Segundo o autor, as máquinas de comitê são técnicas baseadas na estratégia dividir para conquistar onde uma tarefa complexa é dividida em um conjunto de tarefas simples. O resultado final é obtido combinando os resultados destas tarefas. Uma definição dada pelo autor em seu texto é:

Na aprendizagem supervisionada, a simplicidade computacional é alcançada distribuindo-se a tarefa de aprendizagem entre um número de *especialistas*, que por sua vez, divide o espaço de entrada em um conjunto de subespaços. Diz-se que a combinação de especialistas constitui uma *máquina de comitê*. Basicamente ela funde o conhecimento adquirido por especialistas para chegar a uma decisão global que é superior àquela alcançável por qualquer um deles atuando isoladamente. (HAYKIN, 1999, pág. 385)

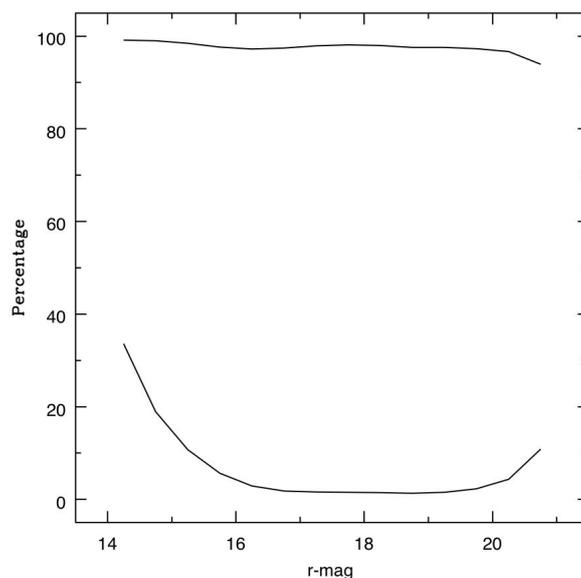


Figura 2.14 - Completude e contaminação para o método paramétrico do SDSS quando assumimos que a classificação fornecida pela nossa AD é 100% correta.

A abordagem dada por Haykin (1999) para as máquinas de comitê é a mesma das ADs. O objetivo de uma AD é, também, dividir uma tarefa complexa em uma série de tarefas simples (veja Seção 2.1). As redes neurais envolvem processos de treinamento complexos nos quais seus elementos (neurônios) são estimulados de forma que eles possam armazenar padrões, e posteriormente identifica-los, tentando replicar o funcionamento do cérebro humano. Em nenhum momento, uma rede neural tenta simplificar um problema para então resolve-lo. Por esta razão a idéia de criar máquinas de comitê que possam dividir um problema complexo em um número de problemas mais simples, de forma que seja mais fácil para as redes resolve-los, gera, segundo Haykin (1999), excelentes resultados. Contudo esta abordagem parece redundante do ponto de vista das ADs. Porque treinar um comitê de ADs se ele vai fazer o que uma AD sozinha já faz? A resposta para essa questão é simples e já foi introduzida na Seção 2.3.2. Uma AD é capaz de dividir o espaço atributos – que é nosso espaço de entrada –, mas eventualmente esta divisão pode gerar erros de classificação devido às estatísticas utilizadas pelos algoritmos de construção. A Figura 2.15(a) mostra um espaço de dois atributos dividido pelo C4.5 (QUINLAN, 1993). Na figura, ao espaço inferior central está associada à classe (\*). Contudo vemos que existe um objeto da classe (●) neste espaço. Isso significa que existe uma probabilidade de um objeto qualquer desta região ser classificado incorretamente. Entretanto, é possível que uma outra AD seja capaz de gerar uma divisão diferente para esta região do espaço de parâmetros e possa classificar o objeto (●) em questão de forma correta. É neste pensamento que se baseia a formulação das máquinas de comitê aplicadas as ADs. Esperamos que usando

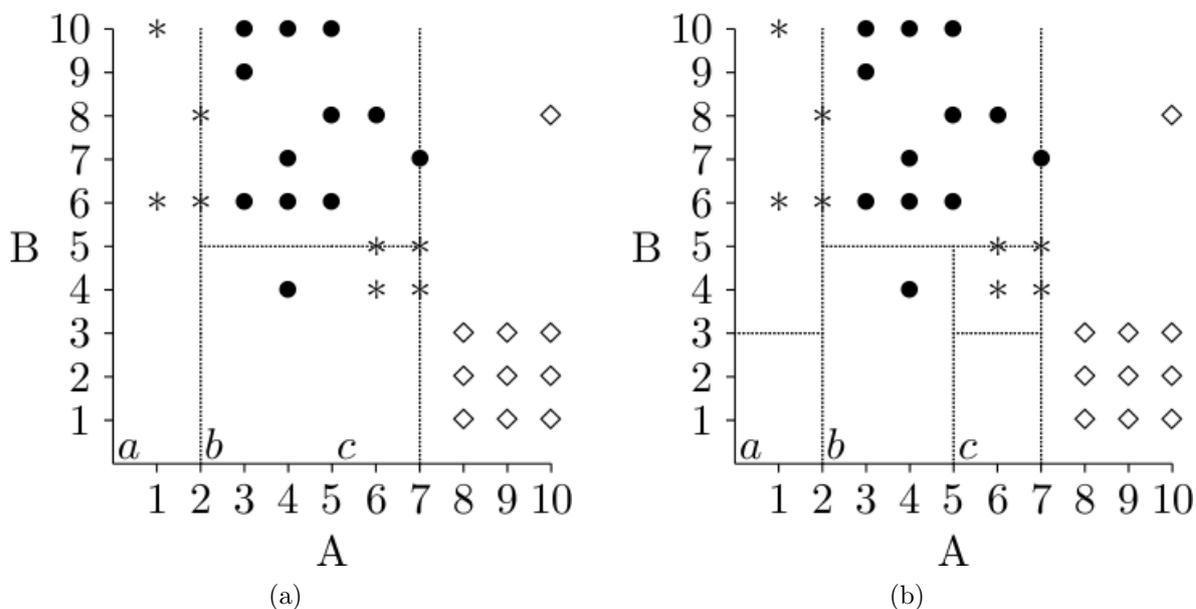


Figura 2.15 - A figura mostra o espaço definido por dois atributos A e B. As semi-retas representam as divisões do espaço determinadas pelos testes propostos pelo algoritmo C4.5 (a) e pelo C4.5 mais o *graft* (b). Cada símbolo representa uma classe diferente. Para visualizar as árvores associadas à divisão desses espaços, o leitor pode consultar a Figura 2.4.

Fonte: As figuras foram adaptadas de Geoffrey (1997).

um comitê, uma ou mais árvores possam acertar onde outras erraram. De fato, Geoffrey (1999) e Haykin (1999) afirmam que as máquinas de comitê obtêm excelentes resultados em uma grande diversidade de tarefas de aprendizado.

Em nosso experimento, utilizamos uma máquina de comitê construída com 13 ADs, cada uma treinada com um dos 13 algoritmos do WEKA configurado com o seu conjunto de parâmetros internos otimizado (Seção 2.6.2). Foi selecionada uma amostra aleatória com 25% dos objetos do conjunto espectroscópico, e esta mesma amostra foi empregada na construção de cada uma das árvores do comitê. Obtivemos a classificação final com base em uma votação majoritária, ou seja, a classe mais votada era escolhida. As funções de completeza e contaminação resultantes dos experimentos, e mostradas na Figura 2.16, apresentam um desempenho similar ao do FT. Os resultados dos testes de CV descritos na Seção 2.6.2 (veja Tabela 2.5) mostram que a maioria dos algoritmos testados apresenta um desempenho inferior ao do FT no limite superior de magnitudes. Logo, uma vez que utilizamos votação majoritária, era esperado que a função de completeza apresentasse a diferença que vemos na Figura 2.16. Contudo, o comitê obteve uma contaminação apro-

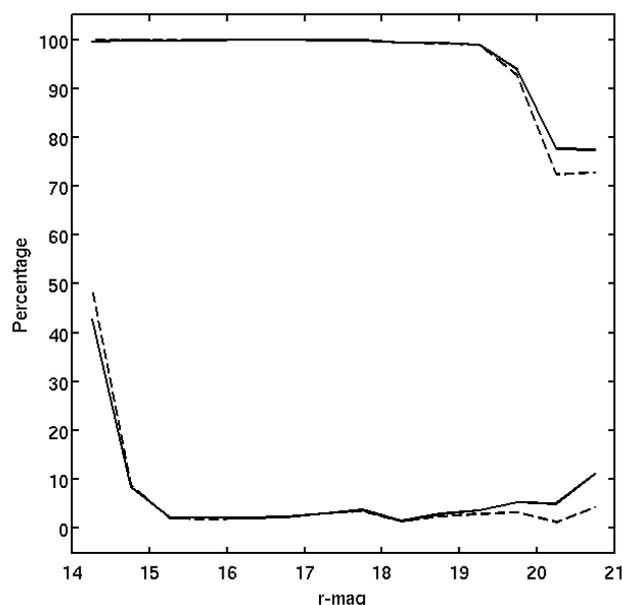


Figura 2.16 - A figura mostra as curvas de completudeza e contaminação para uma AD treinada com o FT e para um comitê de árvores treinadas com cada um dos 13 algoritmos testados do WEKA quando configurados com seus conjuntos de parâmetros otimizados. A linha cheia superior representa a função de completudeza do FT e a inferior a de contaminação. As linhas tracejadas representam as funções de completudeza (superior) e contaminação (inferior) do comitê.

ximadamente 6% menor que a do FT. Acreditamos que a maior variedade de estrelas no conjunto treinamento com magnitude  $r > 19.5$  (4.066 estrelas e 2.452 galáxias) torne a “multidivisão” do espaço de atributos obtida pela máquina de comitê mais eficiente.

Os resultados mostrados na Seção 2.6.2 e os obtidos pelo experimento descrito acima desacreditam que um comitê de árvores treinadas com os algoritmos do WEKA possa superar a completudeza atingida pelo FT. Podíamos esperar que onde o FT errasse um outro algoritmo iria acertar, mas nosso experimento mostra que, em geral, quando um algoritmo classifica uma galáxia erroneamente, os outros também o fazem. Talvez esse problema possa ser contornado utilizando um sistema de votação com peso, mas a atribuição de pesos pode não ser simples, sendo necessário analisar o comportamento de cada árvore no espaço de atributos. Estas considerações são importantes, mas não descartam a possibilidade de que bons resultados possam ser obtidos com as máquinas de comitê. Consideramos estudar futuramente outros aspectos do emprego das máquinas de comitê para classificação estrela/galáxia, inclusive o uso de comitês de redes neurais, uma vez que Weir et al. (1995) e Odewahn et al. (2004) constataam que estas podem, por sua vez, conseguir resultados similares as ADs. As redes neurais são classificadores complexos e organizados em um

comitê podem gerar bons resultados. O maior problema do emprego dessas redes é a impossibilidade de interpretar os critérios de classificação.

## 2.9 Um Teste Simples do Método Paramétrico do SDSS

O método paramétrico do SDSS consiste em um simples teste de atributo: se a condição  $psfMag - modelMag > 0.145$  é satisfeita, o objeto é classificado com galáxia; caso contrário é classificado como estrela. O valor de divisa (0.145) entre as duas classes foi escolhido com base em imagens simuladas de estrelas e galáxias. Contudo, acreditamos que a escolha desta linha de divisa pode ser melhorada usando o grande volume de informação sobre a natureza de cada objeto contido no conjunto espectroscópico. Para testar essa teoria, nós recuperamos o valor de  $psfMag - modelMag$  para todo o conjunto de treinamento, e utilizamos o algoritmo Decision Stump para gerar uma AD contendo um único nó<sup>24</sup>. O Decision Stump escolhe o valor de divisa de um atributo baseado na entropia da informação, como o J48, visando a maximização da completeza e a minimização da contaminação da AD gerada. Surpreendentemente, descobrimos que, segundo o Decision Stump, o requerimento ótimo para que um objeto seja classificado como galáxia é  $psfMag - modelMag > 0.376$ , significativamente maior do que o valor utilizado pelo SDSS.

Para verificar a causa desta grande diferença, nós examinamos imagens de objetos brilhantes classificados erroneamente pelo método paramétrico do SDSS, e que são responsáveis pela alta taxa de contaminação mostrada na Figura 2.13. Verificamos que muitos destes objetos possuem um vizinho muito próximo. A Figura 2.17 mostra nove exemplos destes objetos, onde cada painel jaz centralizado em um objeto erroneamente classificado pelo SDSS paramétrico. Vemos claramente que muitos objetos estão fortemente misturados (do inglês “blended”) com um companheiro ou pelo menos perto o suficiente de outro objeto para que o valor de  $psfMag - modelMag$  usado pelo método paramétrico do SDSS seja influenciado pela presença do objeto vizinho. Este fato fica evidenciado na Figura 2.18, que mostra  $psfMag - modelMag$  como função da magnitude para o conjunto espectroscópico. Os objetos classificados espectroscopicamente como estrelas são mostrados em vermelho, enquanto os classificados como galáxias são mostrados em verde. Um segundo aglomerado de estrelas é claramente visto na região onde  $psfMag - modelMag \sim 0.3$ ; todos erroneamente classificados pelo SDSS paramétrico. Esse problema já havia sido previamente identificado na disponibilização prévia de dados do SDSS (*SDSS Early Data Release*; STOUGHTON et al., 2002), onde os autores notaram que pares de estrelas não misturadas (do inglês *un-deblended star pairs*) e galáxias com núcleos brilhantes são impropriamente classificadas. Esta constatação aumenta a validade da nossa AD, que, através da investigação de um número grande de atributos é capaz de criar um conjunto de regras

---

<sup>24</sup>O algoritmo Decision Stump somente gera ADs de um único nó (veja Seção 2.3).

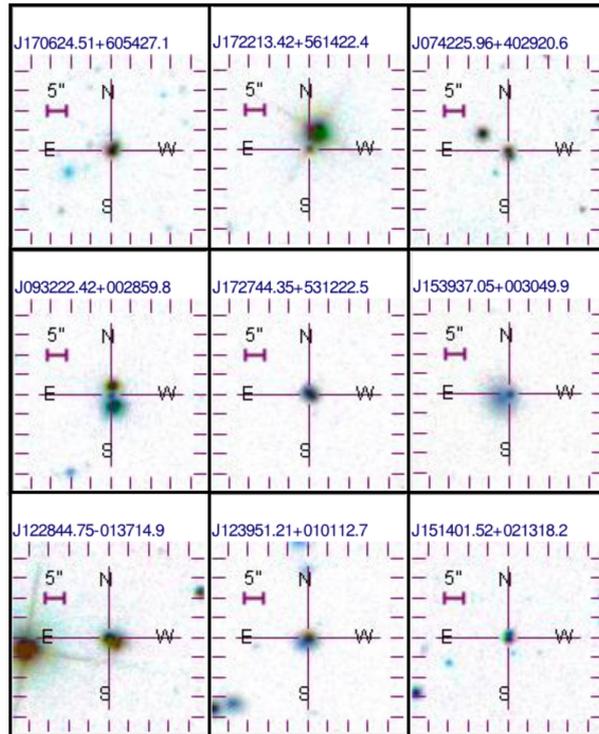


Figura 2.17 - A figura mostra nove campos celestes do SDSS DR7 para nove objetos que são classificados erroneamente pelo método paramétrico do SDSS. Quase todos estão sobrepostos a outros objetos ou possuem um ou mais vizinhos próximos mais brilhantes que afetam a fotometria.

(testes) que podem classificar corretamente mesmo os objetos com vizinhos próximos.

Acreditamos na possibilidade de que existam atributos no banco de dados do SDSS que não utilizamos e que poderiam ajudar a construir uma AD mais precisa. Pode também ser o caso, que atributos “PSF-deconvolved”, como aqueles medidos pelo 2DPHOT, possam melhorar a performance. Outros atributos “indiretos”, como os índices de cor utilizados por Ball et al. (2006) e a combinação de atributos usada pelo método paramétrico do SDSS podem ser consideradas. Contudo, testar todas estas possibilidades, está fora do escopo deste trabalho. Toda via, nossa AD obteve resultados significativamente melhores que todos os demais classificadores publicados aplicados aos dados fotométricos do SDSS.

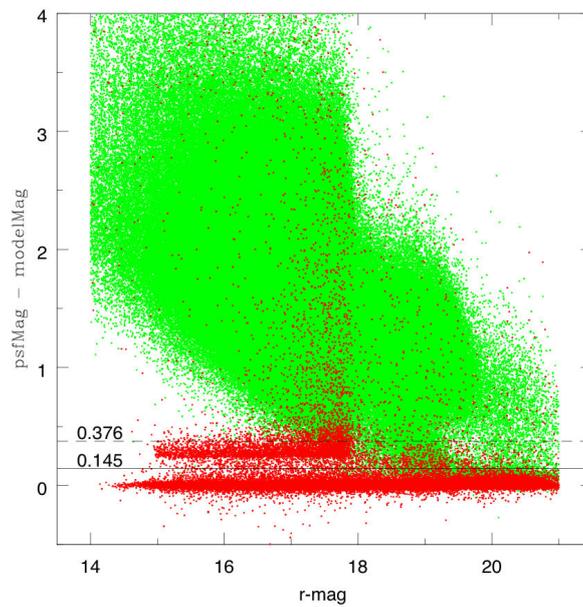


Figura 2.18 - O valor de  $psfMag - modelMag$  utilizado pelo classificador paramétrico do SDSS é representado como uma função da magnitude para os objetos do conjunto espectroscópico. As estrelas são apresentadas como pontos vermelhos, enquanto que as galáxias como pontos verdes. As linhas divisórias usadas pelo classificador SDSS ( $psfMag - modelMag = 0,145$ ) e a calculada pelo Decision Stump ( $psfMag - modelMag = 0,376$ ) também são mostradas. O classificador do SDSS incorretamente classifica muitas estrelas relativamente brilhantes como galáxias, a maioria das quais tem vizinhos próximos.

Tabela 2.4 - Exploração do espaço de parâmetros para cada algoritmo do WEKA. As colunas são: o nome do algoritmo e o número de testes realizados, seus parâmetros, os intervalos estudados, o número de valores no intervalo, e o valor do parâmetro na melhor configuração testada. Devido aos limitados recursos computacionais disponíveis, somente analisamos as variações dos parâmetros contínuos com passos que causassem uma modificação no valor da função de completude maior que 5%. Para os parâmetros nos quais somente um valor é exibido, qualquer variação somente modifica a função de completude em menos de 5%, e o valor padrão do WEKA é usado.

Algoritmo	Parâmetro	Values	N <sub>values</sub>	Melhor Valor
J48graft (1444)	confidenceFactor	0.1,0.15,0.2,...,0.5	9	0.45
	minNumObj	2, 3,...,10,20,...,100,120,...,200	19	8
	relabel	true, false	2	false
	subtreeRaising	true, false	2	true
	unpruned	true, false	2	false
	useLaplace	true, false	1	false
SimpleCart (190)	minNumObj	2,10,40,100,150,...,400	10	2
	numFoldsPruning	2,3,4,...,10	9	5
	sizePer	1	1	1
	useOneSE	true, false	2	false
	usePrune	true, false	2	false
J48 (2898)	confidenceFactor	0.1,0.15,0.2,...,0.5	9	0.45
	minNumObj	2,3,...,10,20,...,100,120,...,200	23	7
	numFolds	3,4,...,10,20,...,100,120,...,200	22	3
	reducedErrorPruning	true, false	2	false
	subtreeRaising	true, false	2	false
	unpruned	true, false	2	false
REPTree (288)	useLaplace	true, false	2	true
	maxDepth	-1,0,1,2,...,10	12	-1
	minNum	0,1,2,3	4	0
	minVarProp	0.001	1	0.001
	noPruning	true, false	2	true
RandomTree (35)	numFolds	10,20,...,50	5	50
	KValue	1,5,10,15,20	5	15
	maxDepth	0,5,10,...,30	7	0
RandomForest (60)	minNum	1	1	1
	maxDepth	0,10,20,30	4	20
	numTrees	5,10,20,30,40	5	40
BFTree (1008)	numFeatures	0,10,20	3	10
	minNumObj	2,10,20,...,40,60,80,100,125,...,200	12	2
	numFoldsPruning	5,10,20,30,40	5	5
	pruningStrategy	postPruned,prePruned,unpruned	3	postPruned
	sizePer	1	1	1
	useErrorRate	true, false	2	true
	useGini	true, false	2	false
ADTree (80)	useOneSE	true, false	2	false
	numOfBoostingIterations	5,10,50,100,150	5	150
	randomSeed	0,50,100,150	4	150
LMT (576)	searchPath	all, the heaviest, the best z-pure, a random	4	all Paths
	errorOnProbabilits	true, false	2	false
	fastRegression	true, false	2	true
	minNumInstances	0,15,30	3	30
	numBoostingIterations	-1,5,10	3	-1
	splitOnResiduals	true, false	2	false
	useAIC	true, false	2	false
LADTree (4)	weightTrimBeta	0,0.1,0.2,0.3	4	0.2
	numOfBoostingIterations	10,50,100,200	4	200
FT (864)	errorOnProbabilits	true, false	2	false
	minNumInstances	0,50,100	3	0
	modelType	FT, FTLeaves, FTInner	3	FT
	numBoostingIterations	10,20,...,60	6	60
	useAIC	true, false	2	false
	weightTrimBeta	0,0.1,0.2,0.3	4	0

Tabela 2.5 - Principais resultados do estudo comparativo realizado com os algoritmos do WEKA. As colunas são, respectivamente; os nomes dos algoritmos testados, o número de parâmetros do algoritmo em questão que causam uma modificação significativa da AD resultante; o tempo médio de processamento calculado sobre todos as combinações de parâmetros testadas; a completeza média calculada no intervalo de magnitudes [14, 19]; a completeza média calculada no intervalo de magnitudes [19, 21]; e a completeza no bin de magnitudes mais fracas ( $20.5 \leq r \leq 21.0$ ).

Algorithm	Number of Parameters	Processing Time (hours)	$\langle \text{Compl} \rangle_{\text{bright}}$ %	$\langle \text{Compl} \rangle_{\text{faint}}$ %	$\text{Compl}_{20.75}$ %
Decision Stump	0	0.03	99.20( $\pm 0.17$ )	68.06( $\pm 1.20$ )	42.29( $\pm 9.64$ )
NBTree	0	1.12	99.64( $\pm 0.16$ )	79.19( $\pm 1.39$ )	63.55( $\pm 14.90$ )
J48graft	6	0.09	99.74( $\pm 0.12$ )	80.93( $\pm 1.16$ )	65.84( $\pm 10.39$ )
Simple Cart	5	0.05	99.63( $\pm 0.16$ )	81.56( $\pm 1.13$ )	67.06( $\pm 9.51$ )
J48	7	0.08	99.73( $\pm 0.12$ )	81.70( $\pm 0.96$ )	66.30( $\pm 7.69$ )
REPTree	5	0.09	99.50( $\pm 0.18$ )	82.76( $\pm 1.09$ )	69.32( $\pm 8.80$ )
Random Tree	3	0.06	99.50( $\pm 0.18$ )	82.76( $\pm 1.09$ )	69.32( $\pm 8.80$ )
Random Forest	3	1.13	99.77( $\pm 0.12$ )	83.15( $\pm 1.14$ )	70.48( $\pm 9.91$ )
BFTree	7	0.24	99.69( $\pm 0.15$ )	83.18( $\pm 1.10$ )	69.85( $\pm 9.55$ )
ADTree	3	1.42	99.73( $\pm 0.12$ )	83.80( $\pm 1.12$ )	71.88( $\pm 9.81$ )
LMT	7	5.50	99.66( $\pm 0.15$ )	83.91( $\pm 1.14$ )	72.18( $\pm 9.39$ )
LADTree	1	7.90	99.70( $\pm 0.14$ )	84.39( $\pm 1.10$ )	72.74( $\pm 9.41$ )
FT	6	2.50	99.64( $\pm 0.15$ )	84.98( $\pm 1.08$ )	74.04( $\pm 8.45$ )

Tabela 2.6 - Classificação estrela/galáxia fornecida pelo SDSS e pela nossa AD gerada pelo FT. A coluna 1 lista a identificação única do SDSS e a coluna 2 a magnitude modelMag na banda  $r$ . As colunas 3 e 4  $type_r$  e  $type$  contém, respectivamente, a classificação paramétrica do SDSS utilizando somente a banda  $r$  e todas as cinco bandas. A coluna 5 mostra a classificação fornecida pela nossa AD, onde o número 1 representa estrelas e o 2 galáxias.

SDSS ObjID	ModelMag $_r$	Type $_r$	Type	FT Class
588848900971299281	20.230947	3	3	2
588848900971299284	20.988880	3	3	2
588848900971299293	20.560146	3	3	2
588848900971299297	19.934738	3	3	2
588848900971299302	20.039648	3	3	2
588848900971299310	20.714321	3	3	2
588848900971299313	20.742567	3	3	2
588848900971299314	20.342773	3	3	2
588848900971299315	20.425304	3	3	2
588848900971299331	20.582634	3	3	2

Tabela 2.7 - Classificação estrela/galáxia paramétrica fornecida pelo SDSS. A coluna 1 mostra as possíveis classificações, enquanto que a coluna 2 contém uma pequena descrição da respectiva classe. A separação estrela ou galáxia é baseada na diferença entre as magnitudes psfMag e modelMag (veja 2.7.3).

Classe SDSS	Descrição
UNKNOWN	Desconhecido: O tipo de objeto é desconhecido.
COSMIC_RAY	Raio Cósmico (não é utilizado).
DEFECT	Defeito: A imagem é gerada por um defeito no telescópio ou no “pipeline” de processamento (não é utilizado).
GALAXY	Galáxia: Um objeto extenso composto de várias estrelas e outros materiais.
GHOST	Fantasma: A imagem é gerada por reflexão ou refração de luz. (não é utilizado)
KNOWNOBJ	Objeto Conhecido: A imagem vem de outro catálogo (não do catálogo do SDSS). (ainda não utilizado)
STAR	Estrela: Um objeto celeste gasoso com luminosidade própria.
TRAIL	Rastro: Um rastro de satélite ou de um meteoro. (ainda não utilizado)
SKY	Céu: espectro de uma área vazia do céu (não há objetos na área demarcada por 1 segundo de arco).
NOTATYPE	Não é um tipo predefinido.

### 3 CONSIDERAÇÕES FINAIS

Analisamos o desempenho de 13 diferentes algoritmos de construção de árvores de decisão disponíveis publicamente na ferramenta de mineração de dados WEKA, quando aplicados ao problema de separação estrela/galáxia de objetos do SDSS-DR7 com base em dados fotométricos. Este é o primeiro trabalho a examinar a aplicação de uma ferramenta pública para mineração de dados a um catálogo astronômico tão vasto. Mostramos as funções de completeza e contaminação para todos os algoritmos em um extenso estudo do espaço de parâmetros de cada algoritmo. Essas funções foram obtidas utilizando testes de validação cruzada (*cross-validation*) e demonstram a capacidade de cada algoritmo para classificar os objetos do conjunto espectroscópico. Assim, nosso estudo pode ser usado como um guia para os astrônomos que o desejem aplicar tais algoritmos a tarefas de separação estrela/galáxia e outros problemas de mineração de dados similares. Os principais resultados do nosso trabalho são:

1. 13 algoritmos WEKA diferentes foram testados e Figura 2.8 mostra o lugar geométrico das funções de completeza resultantes;
2. Todos os algoritmos atingem a mesma precisão no intervalo de magnitudes  $14 \leq r < 19$ , mas com uma grande diferença nos tempos de processamento (Tabela 2.5 e Figura 2.8);
3. As funções de completeza no limite de magnitudes fracas ( $r \geq 19$ ) mostram que o LADTree, o LTM, e o FT são os mais robustos e possuem uma precisão similar (veja Tabela 2.5). Contudo, o FT necessita de quase metade do tempo de que os outros precisão para construir uma AD;
4. O algoritmo FT do WEKA foi então escolhido para treinar uma AD para classificar os objetos do SDSS-DR7 com base em seus atributos fotométricos;
5. Nós mostramos, usando o FT, que reduzir o tamanho do conjunto de treinamento por um fator  $\sim 5$  não altera de forma significativa as funções de completeza e contaminação (veja Figuras 2.9 e 2.10);
6. Usamos o algoritmo FT do WEKA para construir uma AD treinada com atributos fotométricos e a classificação espectroscópica de 240.712 objetos do SDSS-DR7, e separamos com ela todos os objetos do mapeamento Legacy do SDSS-DR7 no intervalo de magnitudes  $14 \leq r \leq 21$  (veja o endereço eletrônico na Seção 2.6.3);
7. Comparamos nosso resultado com aqueles obtidos pelo método paramétrico do SDSS, 2DPHOT e Ball et al. (2006). Nosso catálogo apresenta uma contaminação inferior as do 2DPHOT e de Ball et al. (2006) (Figuras 2.11 e 2.12) e pouco maior

que a do SDSS paramétrico (Figura 2.13), além de uma completeza maior que a do SDSS paramétrico para objetos fracos ( $r \geq 19$ );

8. Por fim, procedemos um experimento com uma máquina de comitê composta por árvores de decisão treinadas com cada um dos algoritmos do WEKA aqui testados, usando suas melhores configurações paramétricas e o mesmo conjunto de exemplos de treinamento. Os resultados obtidos somente reforçaram a eficiência do algoritmo FT do WEKA.

### 3.1 Trabalhos Futuros

Os bons resultados obtidos neste trabalho de pesquisa nos motivaram a estabelecer uma continuidade. O estudo da precisão e performance dos algoritmos do WEKA ainda pode ser mais aprofundado com um maior tempo de pesquisa. A maior abrangência de objetos com espectroscopia disponível em magnitudes mais brilhantes nos leva a considerar testar os 13 algoritmos do WEKA aqui estudados quanto treinados em diferentes intervalos de magnitude. O objetivo principal seria averiguar qual a precisão de uma AD treinada com objetos brilhantes quando aplicada na classificação de objetos de menor brilho.

O estudo adicional que desempenhamos para o FT pode ser estendido para os demais métodos. A verificação da variação das funções de completeza e contaminação de AD construídas com conjuntos de treinamento de diferentes tamanhos pode ser expandida para os outros 12 algoritmos.

Contudo é o estudo das máquinas de comitê que tem nossa maior atenção. Dois tópicos importantes nessa área são: (i) a criação de comitês com AD treinadas em diferentes intervalos de magnitude; (ii) o estudo de comitês de redes neurais artificiais na separação estrela/galáxia. A criação de comitês de classificadores é uma disciplina extensa e abrangente. Infelizmente tivemos pouco tempo para explorá-la, mas os resultados obtidos são promissores.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ABAZAJIAN, K. et al. The second data release of the sloan digital sky survey. **The Astronomical Journal**, v. 128, p. 502–512, 2004. 10
- \_\_\_\_\_. The third data release of the sloan digital sky survey. **The Astronomical Journal**, v. 129., p. 1755–1759., 2005. 11
- AGRESTI, A. **An Introduction to Categorical Data Analysis**. second. [S.l.]: John Wiley & Sons, 1996. 33
- BALL, G. H.; HALL, D. J. A clustering technique for summarizing multivariate data. **Behavioral Science**, v. 12, p. 153–155, 1967. 8
- BALL, N. M. et al. Robust machine learning applied to astronomical data sets. i. star-galaxy classification of the sloan digital sky survey dr3 using decision trees. **The Astrophysical Journal**, v. 650, p. 497–509, 2006. xiii, xvii, 11, 13, 17, 54, 55, 56, 62, 67
- BARBERA, F. L. et al. 2dphot: A multi-purpose environment for the two-dimensional analysis of wide-field images. **Publications of the Astronomical Society of the Pacific**, v. 120, p. 681–702, 2008. 13, 54
- BERNSTEIN, G. M.; JARVIS, M. Shapes and shears, stars and smears: Optimal measurements for weak lensing. **The Astronomical Journal**, v. 123, p. 583–618, 2002. 46
- BREIMAN, L. Baggin predictors. **Machine Learning**, v. 24, p. 123–140, 1996. 28
- \_\_\_\_\_. Random forests. **Machine Learning**, v. 45, p. 5–32, 2001. 34
- BREIMAN, L. et al. **Classifications and regression Trees**. Belmont, California: Wadsworth, 1984. 19, 31, 33, 34
- BRODLEY, C. E.; UTGOFF, P. E. Multivariate decision trees. **Machine Learning**, v. 19, p. 45–77, 1995. 31
- DONALEK, C. **Mining Astronomical Massive Data Sets**. Tese (Doutorado) — Università degli Studi di Napoli 'Federico II', 2006. 9
- FAYYAD, U. M. Branching on attribute values in decision tree generation. In: **Proceedings of the Eleventh National Conference on Artificial Intelligence**. Menlo Park, California: The AAAI Press, 1994. v. 1, p. 601–606. Disponível em: <<http://www.aaai.org/Papers/AAAI/1994/AAAI94-091.pdf>>. 9

- FAYYAD, U. M.; IRANI, K. B. The attribute selection problem in decision tree generation. In: **The 10th National Conference on Artificial Intelligence**. Menlo Park, California: The AAAI Press, 1992. v. 1, p. 104–110. Disponível em: <http://www.aaai.org/Papers/AAAI/1992/AAAI92-016.pdf>. 9
- FREUND, Y.; MASON, L. The alternating decision tree learning algorithm. In: BRATKO, I.; DZEROSKI, S. (Ed.). **The 16th International Conference on Machine Learning**. San Francisco, California: Morgan Kaufmann, 1999. p. 124–133. 35
- FREUND, Y.; SCHAPIRE, R. E. Experiments with a new boosting algorithm. In: **The 13th International Conference on Machine Learning**. [S.l.]: Morgan Kaufmann, 1996. p. 148–156. 28
- GAMA, J. a. Functional trees. **Machine Learning**, v. 55, p. 219–250, 2004. 22, 31, 32, 50
- GEOFFREY, I. W. Decision tree grafting. In: **The 15th International Joint Conference on Artificial Intelligence**. [S.l.: s.n.], 1997. p. 846–851. 28, 30, 59
- \_\_\_\_\_. Decision tree grafting from the all-tests-but-one partition. In: DEAN, T. (Ed.). **The 16th International Joint Conference on Artificial Intelligence**. San Francisco, California: Morgan Kaufmann, 1999. v. 2, p. 702–707. 28, 29, 59
- GUNN, J. E. et al. The sloan digital sky survey photometric camera. **The Astronomical Journal**, v. 116, p. 3040–3081, 1998. 39
- HAIJIAN, S. **Best-first Decision Tree Learning**. Dissertação (Mestrado) — The University of Waikato, Hamilton, New Zealand, 2007. 31
- HAYKIN, S. **Neural Networks: A Comprehensive Foundation**. second. [S.l.]: Pearson Education, 1999. 17, 57, 58, 59
- HEYDON-DUMBLETON, N. H.; COLLINS, C. A.; MACGILLIVRAY, H. T. The edinburgh/durham southern galaxy catalogue. ii - image classification and galaxy number counts. **Royal Astronomical Society, Monthly Notices**, v. 238, p. 379–406, 1989. 4, 6, 14
- HOLMES, G. et al. Multiclass alternating decision trees. In: **In:European Conference on Machine Learning**. [S.l.: s.n.], 2001. p. 161–172. 37
- JARVIS, J. F.; TYSON, J. A. Focas - faint object classification and analysis system. **Astronomical Journal**, v. 86, p. 476–495, 1981. 6, 8

- KOHAVI, R. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In: **In: Second International Conference on Knowledge Discovery and Data Mining**. [S.l.: s.n.], 1996. p. 202–207. 34, 35, 50
- LANDWEHR, N.; HALL, M.; FRANK, E. Logistic model trees. **Machine Learning**, v. 95, p. 161–205, 2005. 32
- LANGLEY, P.; W., I.; THOMPSON, K. An analysis of bayesian classifiers. In: **The 10th National Conference on Artificial Intelligence**. [S.l.: s.n.], 1992. p. 223–228. 34
- LASKER, B. M. et al. The palomar–st sci digitized sky survey (poss–ii): Preliminary data availability. In: JACOBY, G. H.; BARNES, J. (Ed.). **Astronomical Data Analysis Software and Systems V**. [S.l.: s.n.], 1996. (Astronomical Society of the Pacific Conference Series, v. 101), p. 88. 4
- MACGILLIVRAY, H. T. et al. A method for the automatic separation of the images of galaxies and stars from measurements made with the cosmos machine. **Monthly Notices of the Royal Astronomical Society**, v. 176, p. 265–274, 1976. 6
- MADDOX, S. J. et al. The apm galaxy survey. i - apm measurements and star-galaxy separation. **Royal Astronomical Society, Monthly Notices**, v. 243, p. 692–712, 1990. 4, 6, 7, 15
- MINKOWSKI, R. L.; ABELL, G. O. **The National Geographic Society-Palomar Observatory Sky Survey**. [S.l.]: the University of Chicago Press, 1963. 3
- MURTHY, S.; KASIF, S.; SALZBERG, S. A system for induction of oblique decision trees. **Journal of Artificial Intelligence Research**, v. 2, p. 1–32, 1994. 10, 31
- ODEWAHN, S. C. et al. The digitized second palomar observatory sky survey (dposs). iii. star-galaxy separation. **he Astronomical Journal**, v. 128, p. 3092–3107, 2004. 10, 17, 60
- \_\_\_\_\_. Galaxy number counts in 1000 sq.deg. of the digital palomar observatory sky survey (dposs). In: **Bulletin of the American Astronomical Society**. [S.l.: s.n.], 1999. (Bulletin of the American Astronomical Society, v. 31), p. 828–+. 10
- PETROSIAN, V. Surface brightness and evolution of galaxies. **Astrophysical Journal**, v. 209, p. L1–L5, 1976. 45
- QUINLAN, J. R. Induction of decision trees. **Machine Learning**, v. 1, p. 81–106, 1986. 9, 19, 22, 26
- \_\_\_\_\_. **C4.5: Programs for machine learning**. [S.l.]: Morgan Kaufmann, 1993. 9, 20, 22, 26, 28, 31, 58

- REID, I. N. et al. The second palomar sky survey. **Astronomical Society of the Pacific**, v. 103, p. 661–674, 1991. 3
- RUIZ, R. S. R. et al. Árvores de decisão na classificação de dados astronômicos. **Tendências em Matemática Aplicada e Computacional**, v. 10, n. 1, p. 75–86, 2009. 11
- SEBOK, W. L. Optimal classification of images into stars or galaxies - a bayesian approach. **Astronomical Journal**, v. 84, p. 1526–1536, 1979. 6, 8
- SKRUTSKIE, M. F. et al. The two micron all sky survey (2mass): Overview and status. In: GARZON, F. et al. (Ed.). **The Impact of Large Scale Near-IR Sky Surveys**. [S.l.: s.n.], 1997. (Astrophysics and Space Science Library, v. 210), p. 25. 5
- STOUGHTON, C. et al. Sloan digital sky survey: Early data release. **The Astronomical Journal**, v. 123, p. 485–548, 2002. 44, 61
- SUCHKOV, A. A.; HANISCH, R. J.; MARGON, B. A census of object types and redshift estimates in the sdss photometric catalog from a trained decision tree classifier. **The Astronomical Journal**, v. 130, p. 2439–2452, 2005. 10, 29
- WEIR, N.; FAYYAD, U. M.; DJORGOVSKI, S. Automated star/galaxy classification for digitized poss-ii. **Astronomical Journal**, v. 109, p. 2401, jun. 1995. 5, 6, 7, 9, 10, 17, 60
- WHITE, R. L. et al. The first bright quasar survey. ii. 60 nights and 1200 spectra later. **The Astrophysical Journal Supplement Series**, v. 126, p. 133–207, 2000. 10
- WITTEN, I. H.; FRANK, E. **Data Mining: practical machine learning tools and techniques with java implementations**. [S.l.]: Morgan Kaufmann, 2000. xv, 9, 18, 19, 23, 31
- \_\_\_\_\_. **Data Mining: practical machine learning tools and techniques**. second. [S.l.]: Morgan Kaufmann, 2005. 22, 38
- YASUDA, N. et al. Galaxy number counts from the sloan digital sky survey commissioning data. **The Astronomical Journal**, v. 122, p. 1104–1124, 2001. 45
- YORK, D. G. et al. The sloan digital sky survey: Technical summary. **The Astronomical Journal**, v. 120, p. 1579–1587, 2000. 5, 13

## **PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE**

### **Teses e Dissertações (TDI)**

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

### **Manuais Técnicos (MAN)**

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

### **Notas Técnico-Científicas (NTC)**

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

### **Relatórios de Pesquisa (RPQ)**

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

### **Propostas e Relatórios de Projetos (PRP)**

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

### **Publicações Didáticas (PUD)**

Incluem apostilas, notas de aula e manuais didáticos.

### **Publicações Seriadas**

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Constam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

### **Programas de Computador (PDC)**

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.

### **Pré-publicações (PRE)**

Todos os artigos publicados em periódicos, anais e como capítulos de livros.