SCC0173 – Mineração de Dados Biológicos

Classificação III: Árvores de Decisão

Prof. Ricardo J. G. B. Campello

SCC / ICMC / USP

Créditos

- O material a seguir consiste de adaptações e extensões dos originais:
  - gentilmente cedidos pelo Prof. André C. P. L. F. de Carvalho
  - de (Tan et al., 2006)

Aula de Hoje

- Introdução
- Algoritmo Básico
- Medidas para Escolha de Atributos
- Divisão de Atributos de Diferentes Tipos
- Regras de Decisão
- Características de ADs

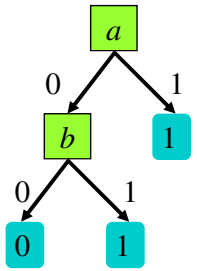
Introdução

- Árvores de Decisão (ADs) são algoritmos de classificação e tomada de decisão que utilizam a estratégia de **divisão e conquista**
  - Divide problemas difíceis em problemas mais simples
    - Problema complexo é decomposto em sub-problemas menores
    - Estratégia é aplicada recursivamente a cada subproblema
- Uma das técnicas de classificação mais utilizadas
  - Eficaz, eficiente e produz modelos interpretáveis

Exemplo Simples

a OR b

a	b	a v b
0	0	0
0	1	1
1	0	1
1	1	1



Exercício

- Encontrar árvore de decisão para:
  - A AND b
  - A XOR b
  - (a AND b) OR (b AND c)

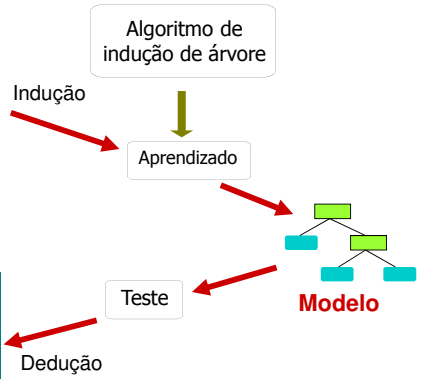
Treinamento e Teste

Conjunto de treinamento

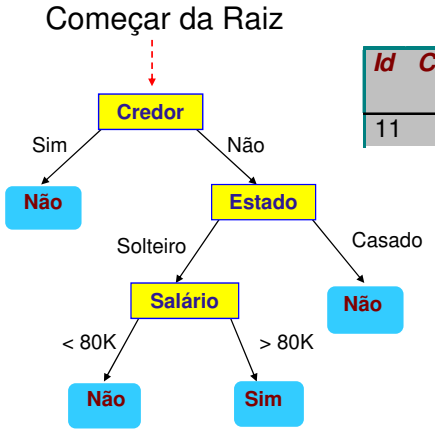
Id	É Credor	Estado Civil	Salário	Calote
1	Sim	Solteiro	125K	Não
2	Não	Casado	100K	Não
3	Não	Solteiro	70K	Não
4	Sim	Casado	120K	Não
5	Não	Solteiro	95K	Sim
6	Não	Casado	60K	Não
7	Sim	Solteiro	220K	Não
8	Não	Solteiro	85K	Sim
9	Não	Casado	75K	Não
10	Não	Solteiro	90K	Sim

Conjunto de teste

Id	É Credor	Estado Civil	Salário	Calote
11	Não	Casado	80K	?
12	Não	Solteiro	100K	?
13	Sim	Solteiro	100K	?
14	Não	Casado	120K	?
15	Sim	Solteiro	80K	?



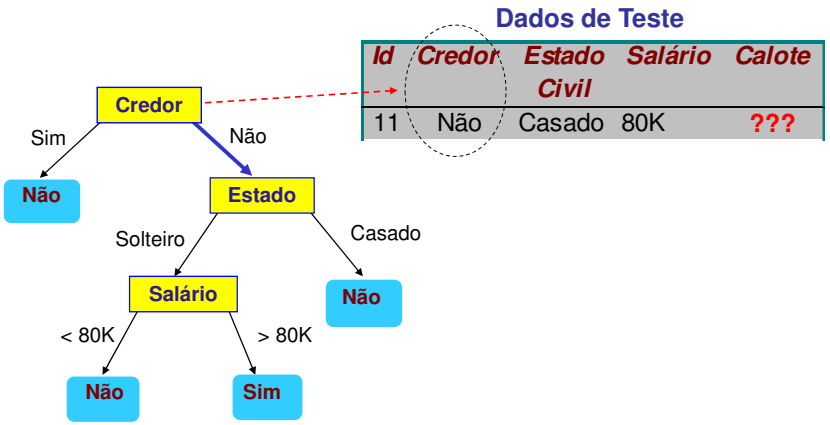
Classificação de Novos Dados



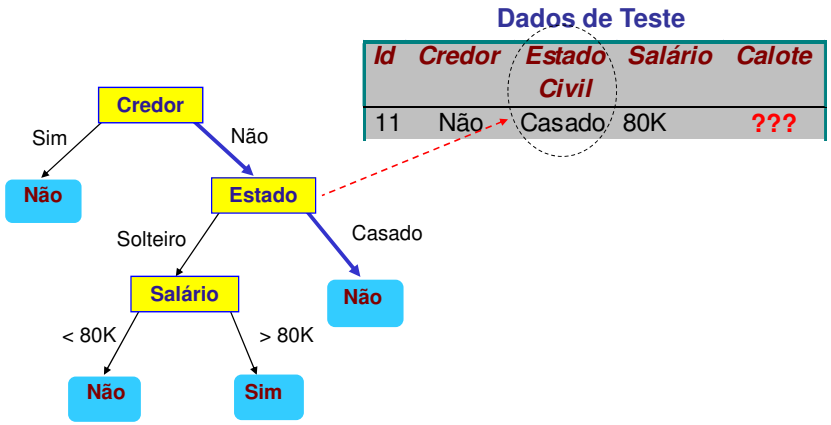
Dados de Teste

Id	Credor	Estado Civil	Salário	Calote
11	Não	Casado	80K	???

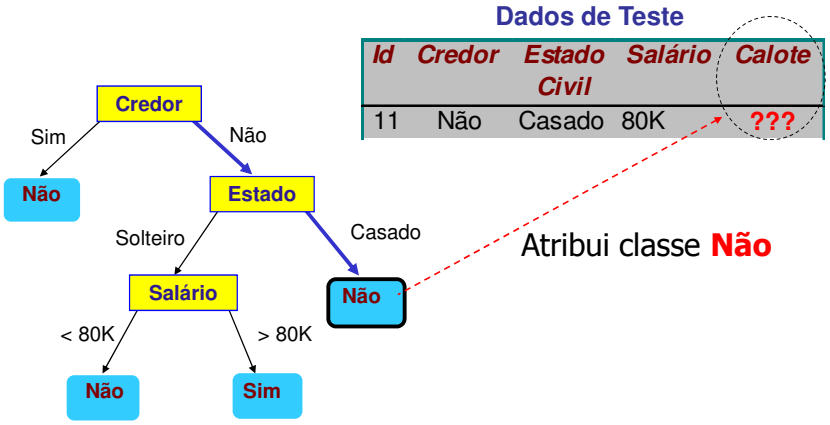
Classificação de Novos Dados



Classificação de Novos Dados



Classificação de Novos Dados



Indução de ADs

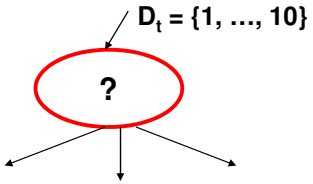
- Existem vários algoritmos
  - Hunt's Concept Learning System
    - Um dos primeiros
    - Base de vários algoritmos atuais
  - ID3, C4.5, J4.8, C5.0
  - CART, Random-Forest
  - ...

Algoritmo de Hunt

- Seja D_t o conjunto de objetos que atingem o nó t
  - porque ainda não foram classificados em um nó folha acima na árvore
- Algoritmo Rudimentar:
  - **Passo 1.** Se todos os objetos de D_t pertencem à mesma classe c_t, então t é um nó folha rotulado como c_t
  - **Passo 2.** Se D_t contém objetos que pertencem a mais de uma classe, então t deve ser um nó interno
    - **Passo 2.1.** O nó deve conter uma condição de teste sobre algum dos atributos que não houverem sido selecionados acima na árvore
    - **Passo 2.2.** Um nó filho é criado para cada possível saída da condição de teste (valor do atributo) e os objetos em D_t são distribuídos neles
    - **Passo 2.3.** O algoritmo é aplicado recursivamente para cada nó filho

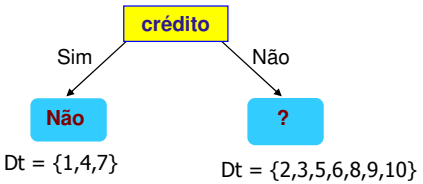
Algoritmo de Hunt

Id	Crédito	Estado Civil	Renda	Deve
1	Sim	Solteiro	Alta	Não
2	Não	Casado	Alta	Não
3	Não	Solteiro	Baixa	Não
4	Sim	Casado	Alta	Não
5	Não	Solteiro	Alta	Sim
6	Não	Casado	Baixa	Não
7	Sim	Solteiro	Alta	Não
8	Não	Solteiro	Alta	Sim
9	Não	Casado	Baixa	Não
10	Não	Solteiro	Alta	Sim



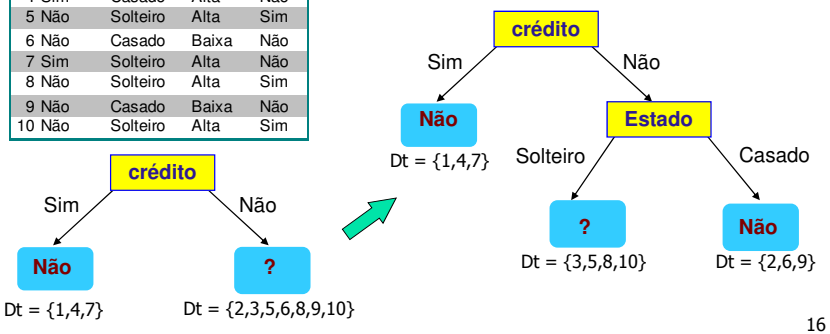
Algoritmo de Hunt

Id	Crédito	Estado Civil	Renda	Deve
1	Sim	Solteiro	Alta	Não
2	Não	Casado	Alta	Não
3	Não	Solteiro	Baixa	Não
4	Sim	Casado	Alta	Não
5	Não	Solteiro	Alta	Sim
6	Não	Casado	Baixa	Não
7	Sim	Solteiro	Alta	Não
8	Não	Solteiro	Alta	Sim
9	Não	Casado	Baixa	Não
10	Não	Solteiro	Alta	Sim



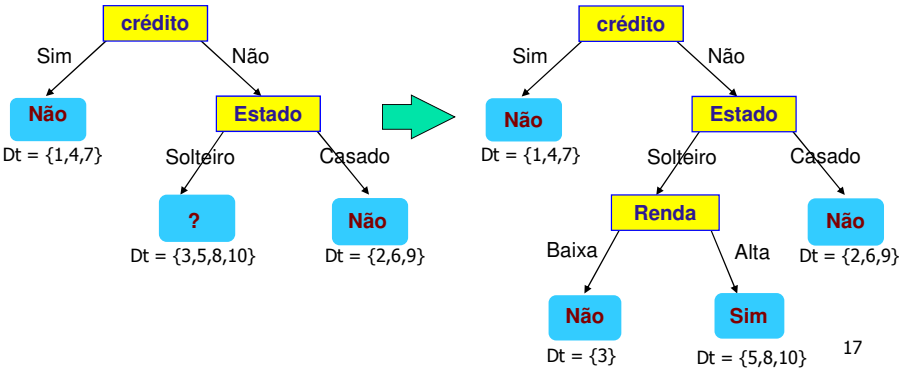
Algoritmo de Hunt

Id	Crédito	Estado Civil	Renda	Deve
1	Sim	Solteiro	Alta	Não
2	Não	Casado	Alta	Não
3	Não	Solteiro	Baixa	Não
4	Sim	Casado	Alta	Não
5	Não	Solteiro	Alta	Sim
6	Não	Casado	Baixa	Não
7	Sim	Solteiro	Alta	Não
8	Não	Solteiro	Alta	Sim
9	Não	Casado	Baixa	Não
10	Não	Solteiro	Alta	Sim



Algoritmo de Hunt

Id	Crédito	Estado Civil	Renda	Deve
1	Sim	Solteiro	Alta	Não
2	Não	Casado	Alta	Não
3	Não	Solteiro	Baixa	Não
4	Sim	Casado	Alta	Não
5	Não	Solteiro	Alta	Sim
6	Não	Casado	Baixa	Não
7	Sim	Solteiro	Alta	Não
8	Não	Solteiro	Alta	Sim
9	Não	Casado	Baixa	Não
10	Não	Solteiro	Alta	Sim



Algoritmo de Hunt

- **Problema**: o algoritmo rudimentar apresentado anteriormente garantidamente funciona apenas se:
  1. Houver ao menos um objeto para cada combinação possível dos valores dos atributos preditores; e
  2. Havendo mais de um, devem pertencer todos à mesma classe
- **Solução** (dada que essas hipóteses são muito restritivas):
  - Se D_t for vazio para um determinado nó t, rotular o nó com a classe majoritária dos objetos do nó pai
  - Se D_t for composto de objetos pertencentes a classes distintas em um dado nó t e não há mais atributos disponíveis, rotular o nó com a classe majoritária desses objetos

Critério de Parada

- Chamada recursiva pode ser finalizada quando:
  - Quando os dados do nó atual possuem o mesmo rótulo
  - Quando os dados do nó atual ainda possuem rótulos de classes diferentes, porém possuem os mesmos valores (categóricos) para todos os atributos preditores
    - o que significa que todos os atributos já terão sido incluídos no caminho a partir da raiz, não havendo mais atributos disponíveis

Indução de ADs

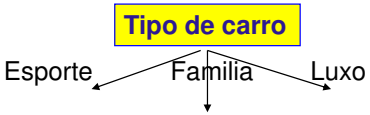
- Geralmente usa estratégia gulosa
  - Divide progressivamente objetos com base em uma condição de teste sobre os valores de um atributo
    - escolhido para maximizar ou minimizar algum critério
- Decisões importantes
  - Como dividir os objetos
    - Como escolher o atributo de divisão
    - Qual a melhor divisão para aquele atributo
  - Quando parar de dividir os objetos

Condição de Teste

- Depende do tipo do atributo
- Os testes mais comuns basicamente diferenciam entre os seguintes tipos:
  - Categóricos
  - Contínuos

Atributos Categóricos

- Pode assumir dois ou mais valores
- Forma usual em vários algoritmos:
  - Usar tantos ramos quantos forem os possíveis valores do atributo. P. ex.:



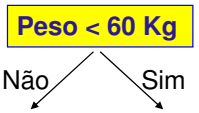
Atributos Contínuos

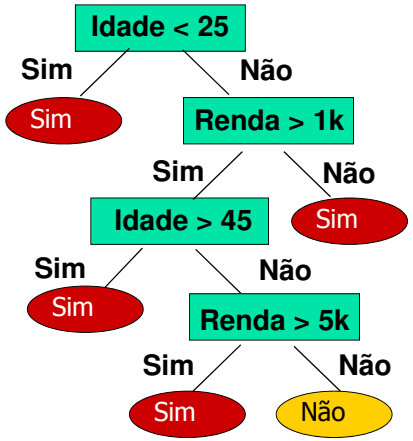
- Forma usual em vários algoritmos:
  - **Comparação**: $A_i < x$
    - Escolher valor x de A_i que gera melhor divisão
      - Ponto de referência
    - Exemplo:

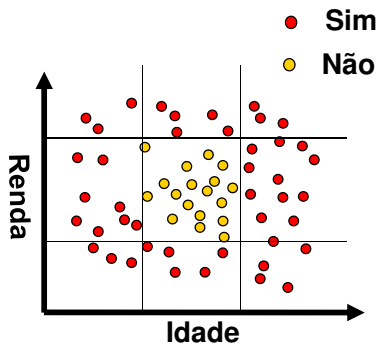
Atributos Contínuos

- **NOTA:**
  - Atributo não é removido do conjunto de candidatos
    - Pode gerar árvores mais profundas
      - regras mais complexas
    - Requer modificações no algoritmo básico
      - Interrupção da construção ou poda da árvore
        - detalhes estão além do escopo do nosso curso...

Atributos Contínuos



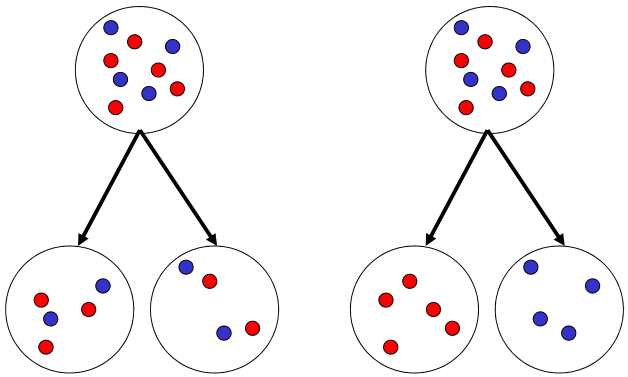




Medidas para Escolha de Atributo

- Existem várias medidas para determinar a melhor forma de dividir os objetos
  - Medidas de impureza
  - Definidas em termos da distribuição de classes dos dados antes e após a divisão
  - Baseadas na idéia que:
    - Quanto mais balanceadas as classes em uma partição, pior
    - A partição mais útil é aquela em que todos os exemplos pertencem a uma mesma classe

Medidas para Escolha de Atributo

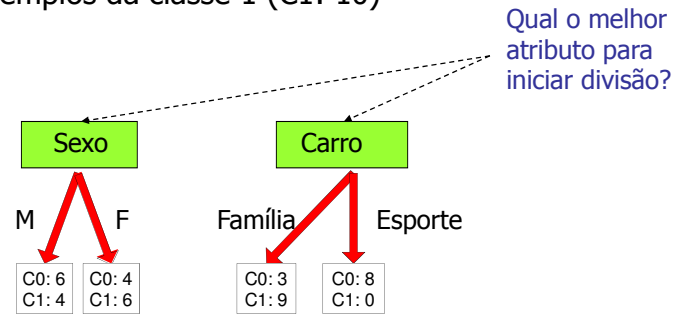


Medidas para Escolha de Atributo

- Medidas diferentes geram partições diferentes dos dados
- Exemplos de medidas de impureza
  - Entropia
  - Gini
  - Erro de classificação

Medidas para Escolha de Atributo

- Supor que D possui 20 exemplos antes da divisão:
  - 10 exemplos da classe 0 (C0: 10)
  - 10 exemplos da classe 1 (C1: 10)



Medidas para Escolha de Atributo

- Abordagem gulosa
  - Prefere nós com distribuição mais homogênea (pura) de classes
  - Necessário uma medida de (im)pureza

C0: 5 C1: 5	C0: 9 C1: 1
Muito heterogênea	**Muito homogênea**
Alto grau de impureza	**Baixo grau de impureza**

Medidas para Escolha de Atributo

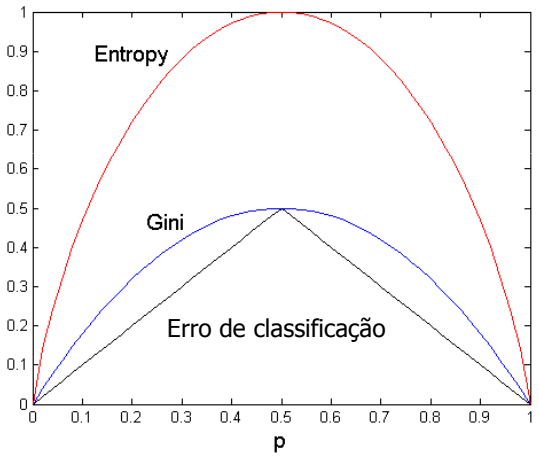
$$\text{Entropia}(t) = -\sum_{i=1}^{c} p(i \mid t) \log_2 p(i \mid t)$$

$$\text{Gini\_Index}(t) = 1 - \sum_{i=1}^{c} [p(i \mid t)]^2$$

$$\text{Erro\_Class}(t) = 1 - \max_{i \in \{1,\ldots,c\}} [p(i \mid t)]$$

onde:
p(i|t) = fração de dados pertencente à classe i em um nó t
c = número de classes
0 $\log_2$ 0 = 0

Comparação: Duas Classes



Comparação

- Valor máximo:
  - Entropia: ($\log_2$ c)
  - Gini e Erro de classificação: (1 – 1/c)
  - Quando os dados estão igualmente distribuídos entre todas as classes
    - Informação menos interessante (menos informação)
- Valor mínimo: 0 (para todos)
  - Quando todos os dados pertencem a uma classe
    - Informação mais interessante

$$\text{Gini\_Index}(t) = 1 - \sum_{i=1}^{c} [p(i \mid t)]^2$$

Exemplo

- Calcular a medida de impureza Gini para os dados abaixo

C1	0
C2	6
Gini=?	

C1	1
C2	5
Gini=?	

C1	2
C2	4
Gini=?	

C1	3
C2	3
Gini=?	

$$\text{Gini\_Index}(t) = 1 - \sum_{i=1}^{c} [p(i \mid t)]^2$$

Exemplo

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1
Gini = 1 – P(C1)² – P(C2)² = 1 – 0 – 1 = 0
P(C1) = 1/6 P(C2) = 5/6
Gini = 1 – (1/6)² – (5/6)² = 0.278
P(C1) = 2/6 P(C2) = 4/6
Gini = 1 – (2/6)² – (4/6)² = 0.444
P(C1) = 3/6 P(C2) = 3/6
Gini = 1 – (3/6)² – (3/6)² = 0.500

C1	0
C2	6
Gini=0.000	

C1	1
C2	5
Gini=0.278	

C1	2
C2	4
Gini=0.444	

C1	3
C2	3
Gini=0.500	

Exercício

- Fazer os mesmos cálculos para as medidas de entropia e de erro de classificação

$$\text{Entropia}(t) = -\sum_{i=1}^{c} p(i \mid t) \log_2 p(i \mid t)$$

$$\text{Erro\_Class}(t) = 1 - \max_{i \in \{1,\ldots,c\}} [p(i \mid t)]$$

C1	0
C2	6
E=?	

C1	1
C2	5
E=?	

C1	2
C2	4
E=?	

C1	3
C2	3
E=?	

C1	0
C2	6
Class=?	

C1	1
C2	5
Class=?	

C1	2
C2	4
Class=?	

C1	3
C2	3
Class=?	

Medidas para Escolha de Atributo

- As medidas de impureza são usadas para avaliar a qualidade de cada condição de teste candidata
  - Compara-se o grau de impureza antes e após a divisão
    - **Quanto maior a diferença, melhor o atributo**
    - Exemplos:
      - **Ganho de Informação:** usada, por exemplo, pelo algoritmo ID3
      - **Média Ponderada de Gini:** usada, por exemplo, pelo algoritmo CART

Medida de Ganho

$$\Delta = I(v_{pai}) - \sum_{t=1}^{k} \frac{N(v_t)}{N} I(v_t)$$

Soma ponderada pela proporção de objetos em cada um dos k nós filhos

onde:

$I(v_t)$: mede o grau de impureza do nó filho v_t

$N(v_t)$: no. de objetos do nó filho v_t

N: no. de objetos do nó original (v_{pai})

Quando a medida de impureza é Entropia, Δ mede o **Ganho de Informação** (Δ_{info})

Medida de Ganho

$$\Delta = I(v_{pai}) - \sum_{t=1}^{k} \frac{N(v_t)}{N} I(v_t)$$

- Note que o primeiro termo será constante para todos os atributos e, portanto, poderia ser omitido para comparar os Δs associados a cada atributo
  - Isso é feito no critério da média ponderada de Gini

Média Ponderada de Gini

- Quando um nó é dividido em k filhos, a qualidade da divisão é definida por:

$$\text{Gini}_{divisão} = \sum_{t=1}^{k} \frac{N(v_t)}{N} \text{Gini}(v_t)$$

Quanto menor melhor

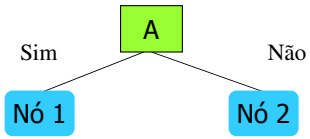
onde

$N(v_t)$: no. de objetos do nó filho v_t

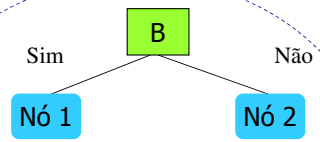
N: no. de objetos do nó original (pai)

Divisão de Atributos Categóricos

	Pai
C1	6
C2	6
Gini = 0.500	



	Nó 1	Nó 2
C1	4	2
C2	3	3
Gini_d = 0.486		



	Nó 1	Nó 2
C1	1	5
C2	4	2
Gini_d = 0.375		

$Gini_{divisão}$ = (7/12)x0.49 + (5/12)x0.48
= 0.486

$Gini_{divisão}$ = (exercício)
= 0.375

Exercícios

- Calcular a média ponderada de Gini para os seguintes atributos candidatos:

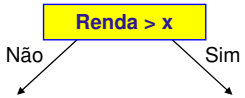
	Tipo de Carro		
	Família	Esporte	Luxo
C1	1	2	1
C2	4	1	1
$Gini_d$	???		

	Renda	
	Baixa	Alta
C1	3	1
C2	2	4
$Gini_d$	???	

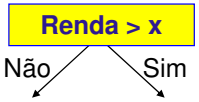
- Repetir para o Ganho de Informação

Divisão de Atributos Contínuos

Id	Crédito	Estado Civil	Renda	Deve
1	Sim	Solteiro	125K	Não
2	Não	Casado	100K	Não
3	Não	Solteiro	70K	Não
4	Sim	Casado	120K	Não
5	Não	Divorced	95K	Sim
6	Não	Casado	60K	Não
7	Sim	Divorced	220K	Não
8	Não	Solteiro	85K	Sim
9	Não	Casado	75K	Não
10	Não	Solteiro	90K	Sim



Divisão de Atributos Contínuos



- Vários candidatos a ponto de referência x
  - no. de valores distintos do atributo na BD...
- Cada valor candidato **x** possui uma matriz de contagens associada a ele
  - Contagens das classes em cada uma das duas partições ($A_i \leq$ **x** e $A_i >$ **x**)

Divisão de Atributos Contínuos

- Determinação de **x** por **Força Bruta**
  - Método mais simples
  - Testar todos os valores **x** candidatos
    - Para cada **x**, calcular sua medida de ganho (Δ_{info} ou $Gini_d$)
      - usando as matrizes de contagens das partições resultantes
  - Computacionalmente ineficiente:
    - $O(N^2)$
    - Trabalho repetitivo

Divisão de Atributos Contínuos

- **Um possível aprimoramento**:
  - Ordenar os valores do atributo em questão
  - Só é preciso considerar valores entre dois exemplos adjacentes com classes diferentes
    - Reduz de 11 para 2 o número de pontos de referência candidatos no exemplo a seguir

Exemplo e Exercício

Deve	Não	Não	Não	Sim	Sim	Sim	Não	Não	Não	Não
	Renda									
Valores Ordenados	60	70	75	85	90	95	100	120	125	220

$Gini_d$ = 0.343 **$Gini_d$ = 0.300**

Exercício: Calcular a média ponderada de Gini ($Gini_d$) para todos os valores e comprovar que **x** = 95 é de fato a melhor solução. Resolver tbém para o Ganho de Info (Δ_{info})

Parte do Exercício

Segundo Candidato: **x** = 60
Contagens do 2º candidato:
≤ 60
  Classe sim: 0
  Classe não: 1
  Gini N1 = ? (Calcular)
> 60
  Classe sim: 3
  Classe não: 6
  Gini N2 = ? (Calcular)
$Gini_d$ = 0.400

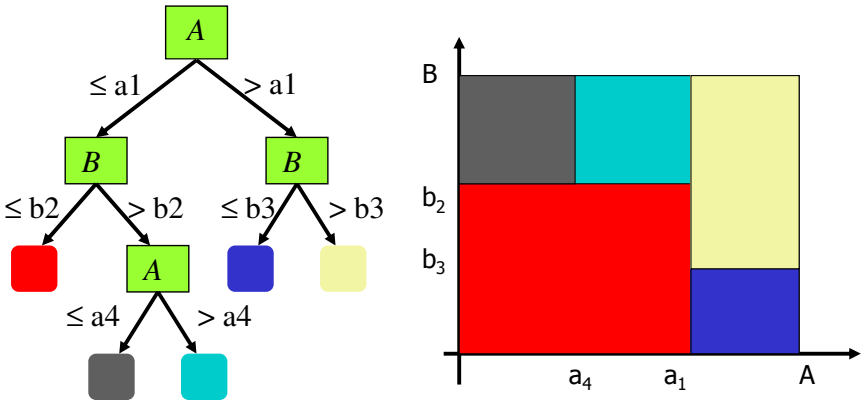
Taxa de Ganho (Gain Ratio)

- Medidas como Entropia e Gini favorecem atributos com muitos valores
  - podem gerar muitos subconjuntos dos dados de treinamento
  - subconjuntos menores tendem a ser mais puros
  - porém, são mais susceptíveis a se especializar nos dados de treinamento
    - preditores ruins da função de classificação para dados não vistos
    - **exemplo extremo**: no. do RG ou CPF para classificação de risco de crédito
- Alternativas para minimizar este problema
  - Estão além do escopo do nosso curso...

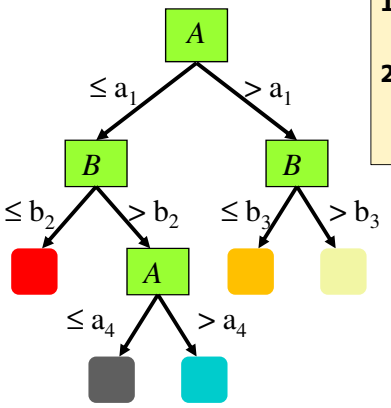
Árvores e Regras

- Cada percurso da raiz a um nó folha representa uma **regra de classificação**
- Cada nó folha
  - Está associado a uma classe
  - Corresponde a uma região do domínio dos atributos
    - Hiper-retângulo
      - Interseção de hiper-retângulos é um conjunto vazio
      - União é o espaço total

Árvores e Regras



Árvores e Regras



Regras: disjunções de conjunções lógicas

1. **Se** $A \le a_1$ **E** $B \le b_2$ **Então** Classe = Vermelha

OU

2. **Se** $A > a_1$ **E** $B \le b_3$ **Então** Classe = Laranja

OU

...

Exercício: complete as regras !

Vantagens das ADs

- Rápida classificação de novos dados
- Interpretação do modelo induzido
  - Fácil para árvores relativamente pequenas
    - ou seja, com poucas regras...
- Determina quais atributos são importantes
  - **Seleção de atributos embarcada** !!!
    - Pode ser estendida para também levar em conta o custo financeiro da utilização de cada atributo...

Vantagens das ADs

- Principais algoritmos tratam tanto atributos categóricos como atributos numéricos
- Desempenho muitas vezes comparável ou até superior a outros bons classificadores
  - depende da natureza dos dados
- Algoritmos podem ser adaptados para tratar instâncias com valores ausentes (e.g. C4.5)
  - tanto no treinamento como na classificação

Desvantagens das ADs

- Baixo desempenho em problemas com muitas classes e poucos dados
  - Representatividade dos nós folha...
- Custo computacional de indução e simplificação do modelo pode ser elevado
  - especialmente para os algoritmos mais sofisticados

Alguns Algoritmos

- ID3
- C4.5
  - http://www.cse.unsw.edu.au/~quinlan/c4.5r8.tar.gz
  - Versão implementada no software Weka – **J4.8**
- C5.0
- CART
- ...

Exercício

- Seja o seguinte cadastro de pacientes:

Nome	Febre	Enjôo	Manchas	Dores	Diagnóstico
João	sim	sim	pequenas	sim	doente
Pedro	não	não	grandes	não	saudável
Maria	sim	sim	pequenas	não	saudável
José	sim	não	grandes	sim	doente
Ana	sim	não	pequenas	sim	saudável
Leila	não	não	grandes	sim	doente

Exercício

- Usando medida de entropia:
  - Induzir uma árvore de decisão capaz de distinguir:
    - Pacientes potencialmente saudáveis
    - Pacientes potencialmente doentes
  - Testar a árvore para novos casos
    - (Luis, não, não, pequenas, sim)
    - (Laís, sim, sim, grandes, sim)
- Repita para medida de Gini