

INDUÇÃO DE REGRAS

Cristiane Neri Nobre

Indução de regras

- Uma regra assume a forma **if L then R** que é equivalente a $L \rightarrow R \equiv R \leftarrow L \equiv R:- L$
- Normalmente, as partes esquerda L e direita R são complexos sem atributos comuns entre eles, ou seja $\text{atributos}(L) \cap \text{atributos}(R) = \emptyset$
- A parte esquerda L é denominada **condição, premissa, antecedente, cauda** ou **corpo** da regra
- A parte direita R é denominada **conclusão** ou **cabeça** da regra

Indução de regras

- Assim, podemos ter por exemplo:
- if L then classe = C_i
- ou simplesmente
 - if L then C_i
- onde C_i pertence ao conjunto de k valores de classe $\{C_1, C_2, \dots, C_k\}$

Indução de regras

- Tal como nas árvores de decisão, o conjunto de regras é disjunto (FND – Forma Normal Disjuntiva)

Regra_1 ou regra_2 ou ... Regra_n

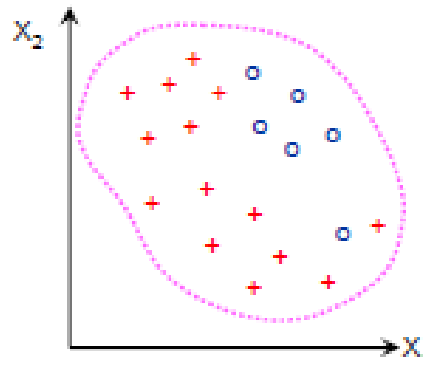
- A indução de **árvores de decisão** recursivamente divide os exemplos em subconjuntos menores, tentando separar cada classe das demais
- A **indução de regras**, por outro lado, o faz diretamente, usando um **algoritmo de cobertura**
- Nesse processo, cada regra cobre um subconjunto de exemplos que pertencem a uma classe específica

Indução de regras

- Na indução de regras, o algoritmo identifica novos testes (condições) a serem adicionados à regra atual (sendo induzida) de forma a melhorar a precisão

Exemplo

- Considere um conjunto com duas classes (+, o) e dois atributos X_1 e X_2 conforme figura abaixo:

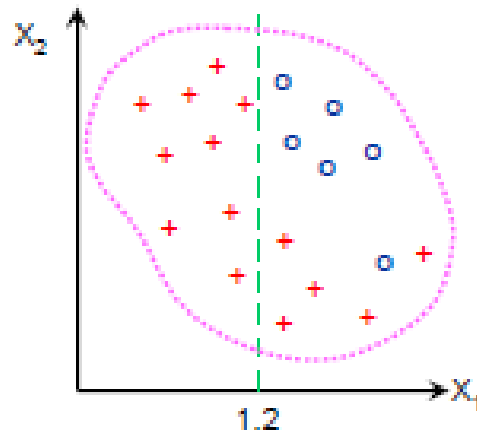


Indução de regras

Exemplo:

Passo 1:

- **if** $X_1 > 1.2$ **then** classe=**o**



Indução de regras

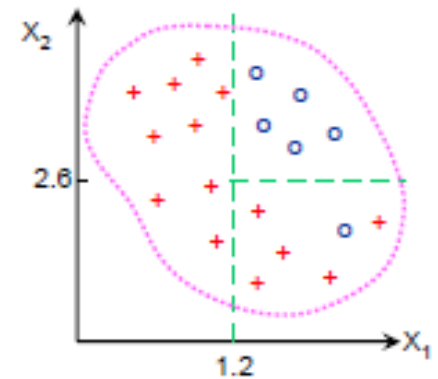
Exemplo:

Passo 1:

- **if** $X_1 > 1.2$ **then** classe = 0

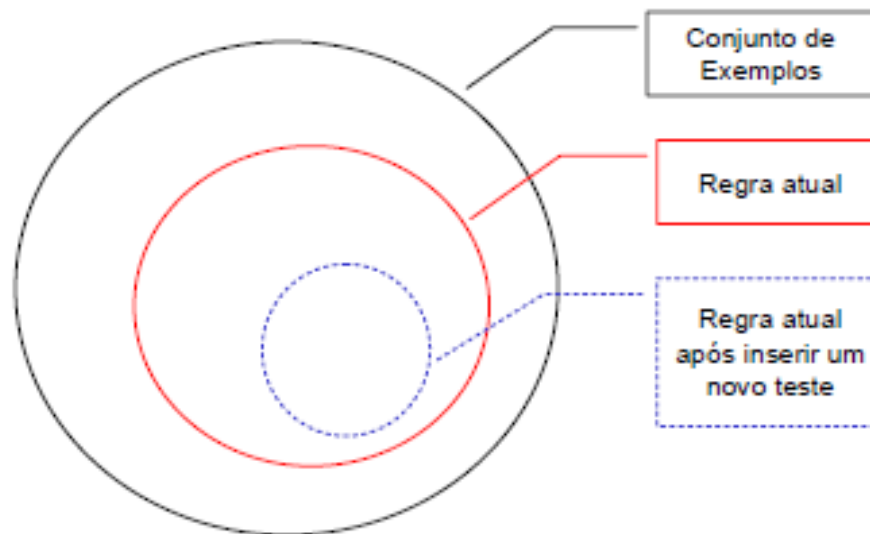
Passo 2: (especialização)

- **if** $X_1 > 1.2$ and $X_2 > 2.6$ **then** classe = 0



Indução de regras

Algoritmo de cobertura



O algoritmo de cobertura

- O algoritmo de cobertura define o processo de aprendizado como um processo de procura: dado um conjunto de exemplos classificados e uma linguagem para representar generalizações dos exemplos, o algoritmo procede, para cada classe, uma procura heurística.
- O processo de procura é guiado por uma função de avaliação das hipóteses. Essa função estima a qualidade das regras que são geradas durante o processo.

O algoritmo de cobertura

- **Ideia Básica do algoritmo:**
- Dado um conjunto de exemplos de classes diferentes, o algoritmo de cobertura consiste em aprender uma regra para uma das classes, removendo o conjunto de exemplos cobertos pela regra, e repetir o processo.
- O processo termina quando só há exemplos de uma única classe.

O algoritmo de cobertura

Algoritmo de cobertura: construção de um conjunto de regras

Entrada: um conjunto de treinamento $\mathbf{D} = \{(x_i, y_i), i=1, \dots, n\}$

Saída: um conjunto de regras: **Regras**

1. **Regras** $\leftarrow \{\}$
2. Seja **Y** o conjunto das classes em **D**;
3. **Para cada** $y_i \in \mathbf{Y}$ faça
4. **repita**
5. $Regra = \text{Aprende_uma_regra}(\mathbf{D}, Y_i);$
6. **Regras** $\leftarrow \mathbf{Regras} \cup \{Regra\}$
7. $\mathbf{D} \leftarrow \text{Remove exemplos cobertos pela } Regra \text{ em } \mathbf{D};$
8. **até** não haver exemplos de y_i ;
9. **fim**
10. Retorna: **Regras**;

Como construir as regras?

- **Método direto**

Extrai regras diretamente dos dados

- ✓ RIPPER, RIPPER, OneR

- **Método indireto**

Extrai regras a partir de outros modelos de classificação

- ✓ C4.5Rules

Algoritmo ZeroR

Algoritmo ZeroR – Como funciona?

- Este atributo não cria nenhuma regra
- Ele atribui todo mundo a classe majoritária
- Serve como um **Baseline** para avaliar a qualidade dos modelos

Algoritmo OneR

Algoritmo OneR

A ideia do algoritmo é testar coisas simples primeiro.

Ele tenta selecionar um atributo para classificar a base

Vamos considerar a base abaixo:

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Desconhecida	Baixa	Nenhuma	>=15000 a <=35000	Moderado
Desconhecida	Baixa	Nenhuma	>35000	Alto
Desconhecida	Baixa	Nenhuma	>35000	Baixo
Desconhecida	Baixa	Adequada	>35000	Baixo
Ruim	Baixa	Nenhuma	<15000	Alto
Ruim	Baixa	Adequada	>35000	Moderado
Boa	Baixa	Nenhuma	>35000	Baixo
Boa	Alta	Adequada	>35000	Baixo
Boa	Alta	Nenhuma	<15000	Alto
Boa	Alta	Nenhuma	>=15000 a <=35000	Moderado
Boa	Alta	Nenhuma	>35000	Baixo
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

Algoritmo OneR

Cálculos para o atributo Histórico de crédito

Algoritmo OneR

História do crédito	História de crédito	Boa	5		
Ruim		Ruim	4		
Desconhecida		Desconhecida	5		
Desconhecida					
Desconhecida	História do crédito	Dívida	Garantias	Renda Anual	Risco
Desconhecida	Boa	Baixa	Nenhuma	>35000	Baixo
Desconhecida	Boa	Alta	Adequada	>35000	Baixo
Ruim	Boa	Alta	Nenhuma	<15000	Alto
Ruim	Boa	Alta	Nenhuma	>=15000 a <=35000	Moderado
Boa	Boa	Alta	Nenhuma	>35000	Baixo
Boa					
Boa	História do crédito	Dívida	Garantias	Renda Anual	Risco
Boa	Ruim	Alta	Nenhuma	<15000	Alto
Boa	Ruim	Baixa	Nenhuma	<15000	Alto
Boa	Ruim	Baixa	Adequada	>35000	Moderado
Ruim	Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto
	História do crédito	Dívida	Garantias	Renda Anual	Risco
	Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
	Desconhecida	Baixa	Nenhuma	>=15000 a <=35000	Moderado
	Desconhecida	Baixa	Nenhuma	>35000	Alto
	Desconhecida	Baixa	Nenhuma	>35000	Baixo
	Desconhecida	Baixa	Adequada	>35000	Baixo

Regras:

- Se História do crédito é BOA então Risco=alto (1)
- Se História do crédito é BOA então Risco=baixo (3)**
- Se História do crédito é BOA então Risco=moderado (1)
- Se História do crédito é RUIM então Risco=alto (3)**
- Se História do crédito é RUIM então Risco=baixo (0)
- Se História do crédito é RUIM então Risco=moderado (1)
- Se História do crédito é DESCONHECIDA então Risco=alto (2)**
- Se História do crédito é DESCONHECIDA então Risco=baixo (2)
- Se História do crédito é DESCONHECIDA então Risco=moderado (1)

Algoritmo

História do crédito
Ruim
Desconhecida
Desconhecida
Desconhecida
Desconhecida
Desconhecida
Ruim
Ruim
Boa
Boa
Boa
Boa
Boa
Ruim

História de crédito	Boa	5
	Ruim	4
	Desconhecida	5

Regras:

Se História do crédito é BOA então Risco=alto (1)
Se História do crédito é BOA então Risco=baixo (3)
Se História do crédito é BOA então Risco=moderado (1)

Se História do crédito é RUIM então Risco=alto (3)
Se História do crédito é RUIM então Risco=baixo (0)
Se História do crédito é RUIM então Risco=moderado (1)

Se História do crédito é DESCONHECIDA então Risco=alto (2)
Se História do crédito é DESCONHECIDA então Risco=baixo (2)
Se História do crédito é DESCONHECIDA então Risco=moderado (1)

Atributo	Quantidade Regras			Erro	Erro total
História de crédito	Boa	5	Se História do crédito é BOA então Risco=baixo (3)	2/5	6/14
	Ruim	4	Se História do crédito é RUIM então Risco=alto (3)	1/4	
	Desconhecida	5	Se História do crédito é DESCONHECIDA então Risco=alto (2)	3/5	

Algoritmo OneR

Cálculos para o atributo Dívida

Algoritmo OneR

Dívida	Dívida	Alta	7
Alta		Baixa	7

Alta
Baixa
Baixa
Baixa
Baixa
Baixa
Baixa
Baixa
Baixa
Alta
Alta
Alta
Alta

História do crédito	Dívida	Garantias	Renda Anual	Risco
Desconhecida	Baixa	Nenhuma	>=15000 a <=35000	Moderado
Desconhecida	Baixa	Nenhuma	>35000	Alto
Desconhecida	Baixa	Nenhuma	>35000	Baixo
Desconhecida	Baixa	Adequada	>35000	Baixo
Ruim	Baixa	Nenhuma	<15000	Alto
Ruim	Baixa	Adequada	>35000	Moderado
Boa	Baixa	Nenhuma	>35000	Baixo

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Boa	Alta	Adequada	>35000	Baixo
Boa	Alta	Nenhuma	<15000	Alto
Boa	Alta	Nenhuma	>=15000 a <=35000	Moderado
Boa	Alta	Nenhuma	>35000	Baixo
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

Regras:

- Se Dívida Baixa então Risco=baixo (3)**
- Se Dívida Baixa então Risco=moderado (2)
- Se Dívida Baixa então Risco=alto (2)

- Se Dívida alta então Risco=baixo (2)
- Se Dívida alta então Risco=moderado (1)
- Se Dívida alta então Risco=alto (4)**

Algoritmo OneR

Dívida	Dívida	Alta	7
Alta		Baixa	7
Alta			
Baixa			
Baixa			
Baixa			
Baixa			
Baixa			
Baixa			
Alta			
Alta			
Alta			
Alta			
Alta			

Regras:
Se Dívida Baixa então Risco=baixo (3)
Se Dívida Baixa então Risco=moderado (2)
Se Dívida Baixa então Risco=alto (2)

Se Dívida alta então Risco=baixo (2)
Se Dívida alta então Risco=moderado (1)
Se Dívida alta então Risco=alto (4)

Atributo	Quantidade		Regras	Erro	Erro total
História de crédito	Boa	5	Se História do crédito é BOA então Risco=baixo (3)	2/5	6/14
	Ruim	4	Se História do crédito é RUIM então Risco=alto (3)	1/4	
	Desconhecida	5	Se História do crédito é DESCONHECIDA então Risco=alto (2)	3/5	

Atributo	Quantidade		Regras	Erro	Erro total
Dívida	baixa	7	Se Dívida Baixa então Risco=baixo (3)	4/7	7/14
	Alta	7	Se Dívida alta então Risco=alto (4)	3/7	

Algoritmo OneR

Cálculos para o atributo Garantias

Algoritmo OneR

Garantias	Garantias	Nenhuma	11
Nenhuma		Adequada	3

Nenhuma	História do crédito	Dívida	Garantias	Renda Anual	Risco
Nenhuma	Ruim	Alta	Nenhuma	<15000	Alto
Nenhuma	Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Adequada	Desconhecida	Baixa	Nenhuma	>=15000 a <=35000	Moderado
Nenhuma	Desconhecida	Baixa	Nenhuma	>35000	Alto
Adequada	Desconhecida	Baixa	Nenhuma	>35000	Baixo
Nenhuma	Ruim	Baixa	Nenhuma	<15000	Alto
Adequada	Boa	Baixa	Nenhuma	>35000	Baixo
Nenhuma	Boa	Alta	Nenhuma	<15000	Alto
Nenhuma	Boa	Alta	Nenhuma	>=15000 a <=35000	Moderado
Nenhuma	Boa	Alta	Nenhuma	>35000	Baixo
Nenhuma	Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

História do crédito	Dívida	Garantias	Renda Anual	Risco
Desconhecida	Baixa	Adequada	>35000	Baixo
Ruim	Baixa	Adequada	>35000	Moderado
Boa	Alta	Adequada	>35000	Baixo

Regras:

- Se Garantias =Nenhuma ENTÃO Risco = baixo (3)
- Se Garantias =Nenhuma ENTÃO Risco = alto (6)**
- Se Garantias =Nenhuma ENTÃO Risco = moderado (2)

- Se Garantias = adequada ENTÃO Risco = baixo (2)**
- Se Garantias =adequada ENTÃO Risco = alto (0)
- Se Garantias =adequada ENTÃO Risco = moderado (1)

Algoritmo OneR

Garantias	Garantias	Nenhuma	11
Nenhuma		Adequada	3
Nenhuma			
Nenhuma			
Nenhuma			
Nenhuma			
Adequada			
Nenhuma			
Adequada			
Nenhuma			
Adequada			
Nenhuma			
Nenhuma			
Nenhuma			
Nenhuma			

Regras:

- Se Garantias =Nenhuma ENTÃO Risco = baixo (3)
- Se Garantias =Nenhuma ENTÃO Risco = alto (6)**
- Se Garantias =Nenhuma ENTÃO Risco = moderado (2)
- Se Garantias = adequada ENTÃO Risco = baixo (2)**
- Se Garantias =adequada ENTÃO Risco = alto (0)
- Se Garantias =adequada ENTÃO Risco = moderado (1)

Atributo	Quantidade		Regras	Erro	Erro total
História de crédito	Boa	5	Se História do crédito é BOA então Risco=baixo (3)	2/5	6/14
	Ruim	4	Se História do crédito é RUIM então Risco=alto (3)	1/4	
	Desconhecida	5	Se História do crédito é DESCONHECIDA então Risco=alto (2)	3/5	

Atributo	Quantidade		Regras	Erro	Erro total
Dívida	baixa	7	Se Dívida Baixa então Risco=baixo (3)	4/7	7/14
	Alta	7	Se Dívida alta então Risco=alto (4)	3/7	

Atributo	Quantidade		Regras	Erro	Erro total
Garantias	Nenhuma	11	Se Garantias =Nenhuma ENTÃO Risco = alto (6)	5/11	6/14
	Adequada	3	Se Garantias = adequada ENTÃO Risco = baixo (2)	1/3	

Algoritmo OneR

Cálculos para o atributo Renda anual

Algoritmo OneR

Renda Anual
<15000
>=15000 a <=35000
>=15000 a <=35000
>35000
>35000
>35000
<15000
>35000
>35000
>35000
<15000
>=15000 a <=35000
>35000
>=15000 a <=35000

Renda Anual	<15000	3
	>=15000 a <=35000	4
	>35000	7

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Ruim	Baixa	Nenhuma	<15000	Alto
Boa	Alta	Nenhuma	<15000	Alto

História do crédito	Dívida	Garantias	Renda Anual	Risco
Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Desconhecida	Baixa	Nenhuma	>=15000 a <=35000	Moderado
Boa	Alta	Nenhuma	>=15000 a <=35000	Moderado
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

História do crédito	Dívida	Garantias	Renda Anual	Risco
Desconhecida	Baixa	Nenhuma	>35000	Alto
Desconhecida	Baixa	Nenhuma	>35000	Baixo
Desconhecida	Baixa	Adequada	>35000	Baixo
Ruim	Baixa	Adequada	>35000	Moderado
Boa	Baixa	Nenhuma	>35000	Baixo
Boa	Alta	Adequada	>35000	Baixo
Boa	Alta	Nenhuma	>35000	Baixo

Regras:

- Se Renda < 15000 ENTÃO Risco = baixo (0)
- Se Renda < 15000 ENTÃO Risco = alto (3)**
- Se Renda < 15000 ENTÃO Risco = moderado (0)

- Se Renda >= 15000 a <=35000 ENTÃO Risco = baixo (0)
- Se Renda >= 15000 a <=35000 ENTÃO Risco = alto (2)**
- Se Renda >= 15000 a <=35000 ENTÃO Risco = moderado (2)

- Se Renda >35000 ENTÃO Risco = baixo (5)**
- Se Renda >35000 ENTÃO Risco = alto (1)
- Se Renda >35000 ENTÃO Risco = moderado (1)

Algoritmo OneR

Renda Anual
<15000
>=15000 a <=35000
>=15000 a <=35000
>35000
>35000
>35000
<15000
>35000
>35000
>35000
<15000
>=15000 a <=35000
>35000
>=15000 a <=35000

Renda Anual	<15000	3
	>=15000 a <=35000	4
	>35000	7

Regras:

Se Renda < 15000 ENTÃO Risco = baixo (0)

Se Renda < 15000 ENTÃO Risco = alto (3)

Se Renda < 15000 ENTÃO Risco = moderado (0)

Se Renda >= 15000 a <=35000 ENTÃO Risco = baixo (0)

Se Renda >= 15000 a <=35000 ENTÃO Risco = alto (2)

Se Renda >= 15000 a <=35000 ENTÃO Risco = moderado (2)

Se Renda >35000 ENTÃO Risco = baixo (5)

Se Renda >35000 ENTÃO Risco = alto (1)

Se Renda >35000 ENTÃO Risco = moderado (1)

Atributo	Quantidade		Regras	Erro	Erro total
História de crédito	Boa	5	Se História do crédito é BOA então Risco=baixo (3)	2/5	6/14
	Ruim	4	Se História do crédito é RUIM então Risco=alto (3)	1/4	
	Desconhecida	5	Se História do crédito é DESCONHECIDA então Risco=alto (2)	3/5	
Atributo	Quantidade		Regras	Erro	Erro total
Dívida	baixa	7	Se Dívida Baixa então Risco=baixo (3)	4/7	7/14
	Alta	7	Se Dívida alta então Risco=alto (4)	3/7	
Atributo	Quantidade		Regras	Erro	Erro total
Garantias	Nenhuma	11	Se Garantias =Nenhuma ENTÃO Risco = alto (6)	5/11	6/14
	Adequada	3	Se Garantias = adequada ENTÃO Risco = baixo (2)	1/3	
Atributo	Quantidade		Regras	Erro	Erro total
Renda anual	<15000	3	Se Renda < 15000 ENTÃO Risco = alto (3)	0/3	4/14
	>=15000 a <=35000	4	Se Renda >= 15000 a <=35000 ENTÃO Risco = alto (2)	2/4	
	>35000	7	Se Renda >35000 ENTÃO Risco = baixo (5)	2/7	

Algoritmo PRISM

Algoritmo PRISM

Primeiro, vamos encontrar a regra para a classe RISCO=alto

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Desconhecida	Baixa	Nenhuma	>35000	Alto
Ruim	Baixa	Nenhuma	<15000	Alto
Boa	Alta	Nenhuma	<15000	Alto
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

Vamos calcular a abrangência:

Atributo	Valores	Abrangência
História de crédito	Boa	1
	Ruim	3
	Desconhecida	2
Dívida	baixa	2
	Alta	4
→ Garantias	Nenhuma	6
	Adequada	0
Renda anual	<15000	3
	>=15000 a <=35000	2
	>35000	1

Regra: Se Garantia = Nenhuma ENTÃO o RISCO é alto

Algoritmo PRISM

Tentamos encontrar agora o segundo atributo:

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Desconhecida	Baixa	Nenhuma	>35000	Alto
Ruim	Baixa	Nenhuma	<15000	Alto
Boa	Alta	Nenhuma	<15000	Alto
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

Atributo	Valores	Abrangência
História de crédito	Boa	1
	Ruim	3
	Desconhecida	2
Dívida	baixa	2
	Alta	4
Garantias	Nenhuma	6
	Adequada	0
Renda anual	<15000	3
	>=15000 a <=35000	2
	>35000	1

Regra: Se Garantia = Nenhuma E Dívida = alta ENTÃO o RISCO é alto

Algoritmo PRISM

Agora, vamos selecionar apenas onde a garantia = nenhuma e dívida = alta

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Desconhecida	Alta	Nenhuma	>=15000 a <=35000	Alto
Boa	Alta	Nenhuma	<15000	Alto
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

Atributo	Valores	Abrangência
História de crédito	Boa	1
	Ruim	2
	Desconhecida	1
Renda anual	<15000	2
	>=15000 a <=35000	2
	>35000	0

Regra: Se Garantia = Nenhuma E Dívida = alta E História de crédito = ruim ENTÃO o RISCO é alto

Algoritmo PRISM

Agora, vamos selecionar apenas onde a garantia = nenhuma e dívida = alta e história de crédito = ruim

História do crédito	Dívida	Garantias	Renda Anual	Risco
Ruim	Alta	Nenhuma	<15000	Alto
Ruim	Alta	Nenhuma	>=15000 a <=35000	Alto

Atributo	Valores	Abrangência
→ Renda anual	<15000	1
	>=15000 a <=35000	1
	>35000	0

Regra: Se Garantia = Nenhuma E Dívida = alta E História de crédito = ruim E Renda anual < 15000 ENTÃO o RISCO é alto

Algoritmo PRISM

Encontre as regras para as classes RISCO **baixo e moderado**

Algoritmo PRISM – Algoritmo

1. Para cada classe C

Inicialize E como o conjunto de treinamento

Enquanto E contém instâncias na classe C

 Crie uma regra R com o lado esquerdo vazio que prediz a classe C

 Até que R seja perfeita (ou não haja mais atributos para usar), faça:

 Para cada atributo A não mencionado em R, e cada valor v,

 Considere adicionar a condição $A = v$ ao lado esquerdo de R

 Selecione A e v para **maximizar a precisão**

 Adicione $A = v$ a R

 Remova as instâncias cobertas por R de E

Algoritmo RIPPER¹ – método direto

- Para problemas de **2 classes**, escolha uma das classes como sendo positiva, e a outra como sendo negativa
 1. Aprenda regras da classe positiva
 2. A classe negativa será a classe default
- Para problemas **multi-classe**
 1. Ordene as classes de acordo com a quantidade de instâncias de cada classe
 2. Aprenda o conjunto de regras para a classe menor primeiro, trate o resto como classe negativa
 3. Repita com a próxima classe menor como classe positiva

¹Repeated Incremental Pruning to Produce Error Reduction

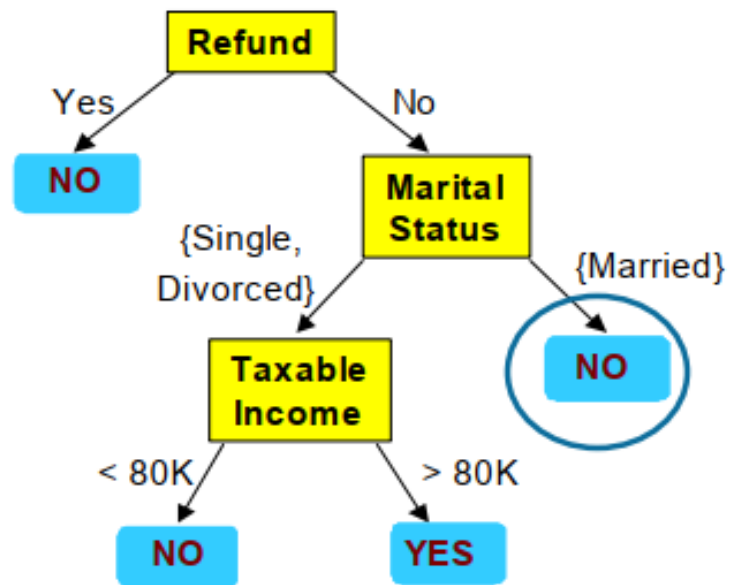
Algoritmo C4.5 Rules – método indireto

- Extrair regras de uma árvore de decisão não podada
- Para cada regra, considere podá-la
- Ordene as regras
 1. Cada subconjunto é uma coleção de regras com o mesmo conseqüente (classe)
 2. As classes descritas por conjuntos de regras mais simples tendem a aparecer primeiro

Quinlan R (1987). "Simplifying Decision Trees." *International Journal of Man-Machine Studies*, 27(3), 221-234.

Quinlan R (1993b). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

Algoritmo C4.5 Rules – método indireto



<i>Tid</i>	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Initial Rule: $(\text{Refund}=\text{No}) \wedge (\text{Status}=\text{Married}) \rightarrow \text{No}$

Simplified Rule: $(\text{Status}=\text{Married}) \rightarrow \text{No}$

Algoritmo PART– método indireto

```
1 repeat
2   | Create a pruned classification tree
3   | Determine the path through the tree with the largest coverage
4   | Add this path as a rule to the rule set
5   | Remove the training set samples covered by the rule
6 until all training set samples are covered by a rule
```

Frank E, Witten I (1998). "Generating Accurate Rule Sets Without Global Optimization." Proceedings of the Fifteenth International Conference on Machine Learning, pp. 144–151.

Exemplo Ilustrativo

Como gerar uma regra de decisão?

Considere o problema descrito na tabela:

Dia	Aparência	Temperatura	Umidade	Ventando	Jogar
d1	Ensolarado	Quente	Alta	Não	Não
d2	Ensolarado	Quente	Alta	Sim	Não
d3	Nublado	Quente	Alta	Não	Sim
d4	Chuvoso	Agradável	Alta	Não	Sim
d5	Chuvoso	Fria	Normal	Não	Sim
d6	Chuvoso	Fria	Normal	Sim	Não
d7	Nublado	Fria	Normal	Sim	Sim
d8	Ensolarado	Agradável	Alta	Não	Não
d9	Ensolarado	Fria	Normal	Não	Sim
d10	Chuvoso	Agradável	Normal	Não	Sim
d11	Ensolarado	Agradável	Normal	Sim	Sim
d12	Nublado	Agradável	Alta	Sim	Sim
d13	Nublado	Quente	Normal	Não	Sim
d14	Chuvoso	Agradável	Alta	Sim	Não

Exemplo Ilustrativo

Vamos utilizar como função de avaliação de hipóteses a **taxa de erro**

A procura inicia com a regra mais geral $\{\}$ \rightarrow Sim.

A regra não tem restrições, ou seja, tudo é da classe **Sim**

A taxa de erro é de 5/14

Exemplo Ilustrativo

Introduzindo uma restrição, o conjunto de hipóteses é:

Atributo *Tempo*

Tempo = *Ensolarado* → Sim; (Acertos = 2/5 Erro = 3/5)

Tempo = *Nublado* → Sim; (Acertos = 4/4 Erro = 0/4)

Tempo = *chuvoso* → Sim; (Acertos = 3/5 Erro = 2/5)

Dia	Tempo	Temperatura	Umidade	Vento	Jogar
d1	Chuvoso	Quente	Alta	Não	Não
d2	Ensolarado	Quente	Alta	Sim	Sim
d3	Ensolarado	Quente	Alta	Não	Não
d4	Nublado	Agradável	Alta	Não	Sim
d5	Chuvoso	Fria	Normal	Não	Sim
d6	Chuvoso	Fria	Normal	Sim	Não
d7	Nublado	Fria	Normal	Sim	Sim
d8	Nublado	Agradável	Alta	Não	Sim
d9	Ensolarado	Fria	Normal	Não	Sim
d10	Chuvoso	Agradável	Normal	Não	Sim
d11	Nublado	Agradável	Normal	Sim	Sim
d12	Ensolarado	Agradável	Alta	Sim	Não
d13	Ensolarado	Quente	Normal	Não	Não
d14	Chuvoso	Agradável	Alta	Sim	Sim

Exemplo Ilustrativo

Atributo *Temperatura*

Temperatura = Quente → Sim; (Acertos = 1/4 Erros = 3/4)

Temperatura = Agradável → Sim; (Acertos = 5/6 Erros = 1/6)

Temperatura = Fria → Sim; (Acertos = 3/4 Erros = 1/4)

Dia	Aparência	Temperatura	Umidade	Vento	Jogar
d1	Chuvoso	Quente	Alta	Não	Não
d2	Ensolarado	Quente	Alta	Sim	Sim
d3	Ensolarado	Quente	Alta	Não	Não
d4	Nublado	Agradável	Alta	Não	Sim
d5	Chuvoso	Fria	Normal	Não	Sim
d6	Chuvoso	Fria	Normal	Sim	Não
d7	Nublado	Fria	Normal	Sim	Sim
d8	Nublado	Agradável	Alta	Não	Sim
d9	Ensolarado	Fria	Normal	Não	Sim
d10	Chuvoso	Agradável	Normal	Não	Sim
d11	Nublado	Agradável	Normal	Sim	Sim
d12	Ensolarado	Agradável	Alta	Sim	Não
d13	Ensolarado	Quente	Normal	Não	Não
d14	Chuvoso	Agradável	Alta	Sim	Sim

Exemplo Ilustrativo

Atributo *Umidade*

Umidade = Alta → Sim; (Acertos = 4/7 Erros = 3/7)

Umidade = Normal → Sim; (Acertos = 5/7 Erros = 2/7)

Dia	Aparência	Temperatura	Umidade	Vento	Jogar
d1	Chuvoso	Quente	Alta	Não	Não
d2	Ensolarado	Quente	Alta	Sim	Sim
d3	Ensolarado	Quente	Alta	Não	Não
d4	Nublado	Agradável	Alta	Não	Sim
d5	Chuvoso	Fria	Normal	Não	Sim
d6	Chuvoso	Fria	Normal	Sim	Não
d7	Nublado	Fria	Normal	Sim	Sim
d8	Nublado	Agradável	Alta	Não	Sim
d9	Ensolarado	Fria	Normal	Não	Sim
d10	Chuvoso	Agradável	Normal	Não	Sim
d11	Nublado	Agradável	Normal	Sim	Sim
d12	Ensolarado	Agradável	Alta	Sim	Não
d13	Ensolarado	Quente	Normal	Não	Não
d14	Chuvoso	Agradável	Alta	Sim	Sim

Exemplo Ilustrativo

Atributo *Vento*

Vento = *Sim* → Sim; (Acertos = 4/6 Erros = 2/6)
Vento = *Não* → Sim; (Acertos = 5/8 Erros = 3/8)

Dia	Aparência	Temperatura	Umidade	Vento	Jogar
d1	Chuvoso	Quente	Alta	Não	Não
d2	Ensolarado	Quente	Alta	Sim	Sim
d3	Ensolarado	Quente	Alta	Não	Não
d4	Nublado	Agradável	Alta	Não	Sim
d5	Chuvoso	Fria	Normal	Não	Sim
d6	Chuvoso	Fria	Normal	Sim	Não
d7	Nublado	Fria	Normal	Sim	Sim
d8	Nublado	Agradável	Alta	Não	Sim
d9	Ensolarado	Fria	Normal	Não	Sim
d10	Chuvoso	Agradável	Normal	Não	Sim
d11	Nublado	Agradável	Normal	Sim	Sim
d12	Ensolarado	Agradável	Alta	Sim	Não
d13	Ensolarado	Quente	Normal	Não	Não
d14	Chuvoso	Agradável	Alta	Sim	Sim

Exemplo Ilustrativo

Como foi encontrada uma regra com taxa **de erro 0**,
Tempo = Nublado → **Sim**, o processo de encontrar a regra termina.

O algoritmo de cobertura remove os exemplos cobertos pela regra e retorna o processo de encontrar uma nova regra a partir do subconjunto de exemplos.

Biblioteca em Python

Que bibliotecas **em Python** podemos utilizar para indução de regras?

1. Algoritmo OneR

Você pode usar o algoritmo de árvore, normalmente, estabelecendo uma altura =1

```
oneR = DecisionTreeClassifier(random_state=0, max_depth=1)
```

2. Algoritmo RIPPER

Vocês pode utilizar a biblioteca “wittgenstein”

```
from wittgenstein import RIPPER
```

Biblioteca em Python

3. Algoritmo C4.5 Rules

Embora não haja uma implementação direta do algoritmo C4.5 na biblioteca scikit-learn, você pode utilizar a biblioteca "**sklearn-extensions**" para obter uma implementação do C4.5 e gerar regras a partir da árvore

```
pip install sklearn-extensions  
from skrules import SkopeRules
```


Referências:

Capítulo 6 do livro:

Data Mining: Practical Machine Learning Tools and Techniques

Disponível em: <https://www.cs.waikato.ac.nz/~ml/weka/book.html>

Links:

[JRIP](#): William W. Cohen: Fast Effective Rule Induction. In: Twelfth International Conference on Machine Learning, 115-123, 1995.

<https://christophm.github.io/interpretable-ml-book/rules.html>

<https://www.slideshare.net/gladysCJ/lesson-5-classification-rules>

<https://www.ime.unicamp.br/~wanderson/Aulas/>

<https://www.sciencedirect.com/science/book/9780123748560>