

Aula 8 – Mineração de Dados

Classificação - Avaliação

Profa. Elaine Faria

UFU

- Os slides a seguir consistem em adaptações dos slides do prof. Andre Carlos Ponce de Leon Ferreira Carvalho
- Agradecimento ao prof. Andre Carvalho por gentilmente ceder os seus slides

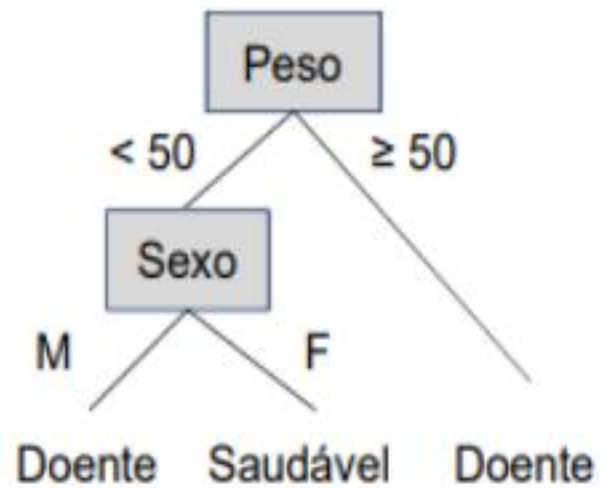
Algoritmos de AM

- Induzem modelos (hipóteses) a partir de um conjunto de dados
- Dados precisam
 - Ser estruturados
 - Ter boa qualidade
 - Ser representativos
- Algoritmos de AM indutivo possuem um viés
 - Tendência a privilegiar uma dada hipótese ou conjunto de hipóteses

Viés Indutivo

- “Quando um algoritmo de AM está aprendendo a partir de um conjunto de dados de treinamento, ele está procurando uma hipótese, no espaço de possíveis hipóteses, capaz de descrever as relações entre os objetos e que melhor se ajuste aos dados de treinamento.”
- “Cada algoritmo utiliza uma forma ou representação para descrever a hipótese induzida”

Viés Indutivo



Árvore de decisão

0.45	-0.40	0.54	0.12	0.98	0.37
-0.45	0.11	0.91	0.34	-0.20	0.83
-0.29	0.32	-0.25	-0.51	0.41	0.70

Redes neurais

Se $\text{Peso} \geq 50$ então Doente
Se $\text{Peso} < 50$ e $\text{Sexo} = \text{M}$ então Doente
Se $\text{Peso} < 50$ e $\text{Sexo} = \text{F}$ então Saudável

Conjunto de regras

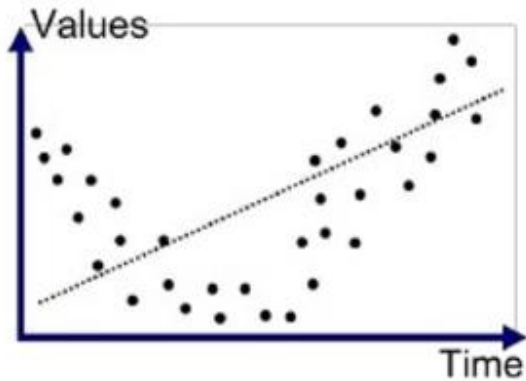
Algoritmos de AM

- Fontes de erro de algoritmos AM
 - Viés
 - Quando algoritmo aprende um modelo incorreto
 - Associado a *underfitting*
 - Variância
 - Quando algoritmo presta atenção a detalhes sem importância
 - Associado a *overfitting*
- Precisam ser reduzidos

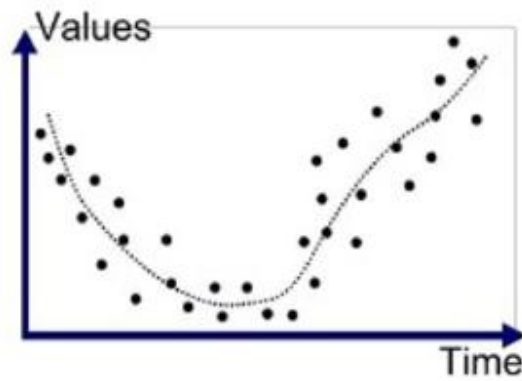
Overfitting e Underfitting

- Quando uma hipótese apresenta uma baixa capacidade de generalização
 - Pode ser que ela está superajustada aos dados de treinamento (*overfitting*)
 - A hipótese memorizou ou se especializou nos dados de treinamento
- Quando uma hipótese apresenta uma baixa taxa de acerto mesmo no subconjunto de treinamento
 - Pode ser que ela está subajustada (*underfitting*).
 - Ex: os exemplos de treinamento disponíveis são pouco representativos ou o modelo usado é muito simples e não captura os padrões existentes nos dados.

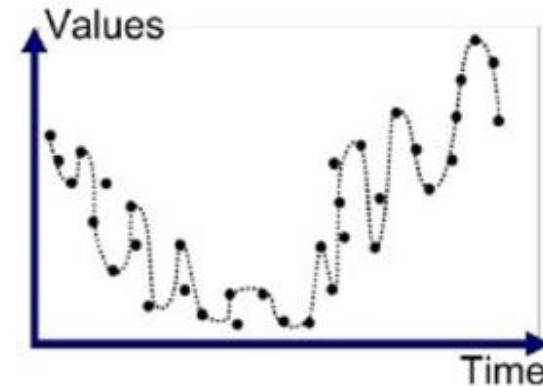
Overfitting e underfitting



Underfitted



Good Fit/Robust



Overfitted

Bom algoritmo de AM

- Está sempre percorrendo um caminho estreito entre:
 - *Overfitting*
 - *Underfitting*
- Buscando o melhor compromisso que busca reduzir ambos

Avaliação de desempenho

- Espera-se de um classificador que ele apresente desempenho adequado para dados não vistos
 - Acurácia, pouca sensibilidade ao uso de diferentes amostras de dados, ...
- Desempenho do classificador deve ser avaliado
 - Utiliza-se conjuntos distintos de exemplos de treinamento e exemplos de teste
 - Permitem estimar a capacidade de generalização do classificador
 - Permitem avaliar a variância (estabilidade) do classificador

Avaliação de classificadores

- Existem diferentes métodos para organização e utilização dos dados (exemplos) disponíveis em conjuntos de treinamento e teste
- Por exemplo:
 - *Holdout*
 - *Random Subsampling*
 - *Cross-Validation*

Holdout

- Também conhecido como *split-sample*
- Técnica mais simples
- Faz uma única partição da amostra em:
 - Conjunto de treinamento
 - geralmente $1/2$ ou $2/3$ dos dados
 - Conjunto de teste
 - dados restantes

Holdout

- Problema: dependência da composição dos conjuntos
- É mais crítico em “pequenas” quantidades de dados...
 - Quanto menor o conjunto de treinamento, maior a variância (sensibilidade / instabilidade) do classificador a ser obtido
 - Quanto menor o conjunto de teste, menos confiável a acurácia estimada do classificador para dados não vistos
 - Conjuntos de treinamento e teste podem não ser independentes
 - Classe sub-representada em um será super-representada no outro

Random Subsampling

- Múltiplas execuções de *Holdout*
 - Diferentes partições treinamento-teste são escolhidas de forma aleatória
 - Não pode haver interseção entre os dois conjuntos
 - Desempenho de classificação é avaliado para cada partição
 - Desempenho estimado para dados não vistos é o desempenho médio para as diferentes partições
 - Permite uma estimativa de erro mais precisa
 - Porém, não controla número de vezes que cada exemplo é utilizado nos treinamentos e testes

Random Subsampling

- Exemplo:
 - Supor que o conjunto de dados original seja formado pelos dados: $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8$
 - Possíveis partições:

	Treinamento	Teste
Part. 1	x_2, x_4, x_6, x_7	x_5, x_8, x_1, x_3
Part. 2	x_3, x_4, x_5, x_8	x_1, x_7, x_2, x_6
Part. 3	x_3, x_4, x_5, x_7	x_2, x_8, x_1, x_6

Cross-Validation

- Validação cruzada
- Classe de métodos para estimativa da taxa de erro
- k-fold cross-validation
 - Cada objeto participa o mesmo número de vezes do treinamento ($k - 1$ vezes)
 - Cada objeto participa o mesmo número de vezes do teste (1 vez)

Cross-Validation

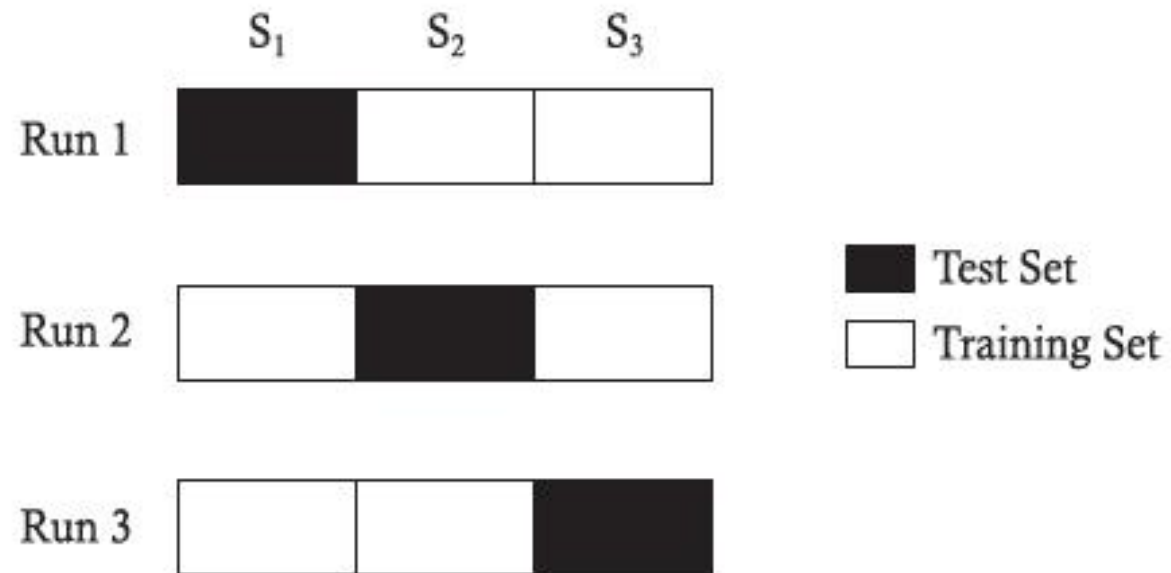
- Divide conjunto de dados em k partições mutuamente exclusivas
 - A cada iteração, uma das k partições é usada para testar o modelo
 - As outras $k - 1$ são usadas para treinar o modelo
 - Taxa de erro é tomada como a média dos erros de teste das k partições
- Exemplo Típico
 - *10-fold cross-validation*

Cross-Validation

- *k-fold cross-validation* estratificada
 - Mantém nas pastas as proporções de exemplos das classes presentes no conjunto total de dados

Cross-Validation

- 3-fold cross-validation



Leave-one-out

- N iterações são utilizadas para uma amostra de tamanho N
 - N-fold cross-validation
 - A cada iteração, um dos exemplos é utilizado para testar o modelo
 - Os outros $N-1$ exemplos são utilizados para treinamento
- Taxa de erro é obtida dividindo o número total de erros de validação observados por N

Leave-one-out

- Sua estimativa de erro é praticamente não tendenciosa
 - Média das estimativas tende a taxa de erro verdadeiro
- Computacionalmente caro
 - Geralmente utilizado para pequenos conjuntos de exemplos
 - 10-fold cross validation aproxima leave-one-out
- Variância tende a ser elevada

Bootstrap

- Funciona melhor que cross-validation para conjuntos muito pequenos
- Forma mais simples de bootstrap:
 - Ao invés de usar sub-conjuntos dos dados, usar sub-amostras
 - Cada sub-amostra é uma amostra aleatória com substituição do conjunto total de exemplos
 - Cada conjunto de treinamento têm o mesmo número de exemplos do conjunto total
 - Os exemplos que restarem são utilizados para teste

Bootstrap

- Se conjunto original tem N exemplos
 - Amostra de tamanho N tem $\approx 63,2\%$ dos exemplos originais
- Processo é repetido b vezes
- Resultado final = média dos b experimentos
- Existem diversas variações

Medidas de desempenho

- Principal objetivo de um modelo é prever com sucesso o valor de saída para novos exemplos
 - Errar o mínimo possível
- Existem várias medidas de “erro” e “acerto”
 - Diferentes medidas podem capturar diferentes aspectos do desempenho de classificadores

Taxa de Classificação Incorreta

- A medida mais básica para estimar a taxa de erro de um classificador é denominada de taxa de classificação incorreta (*misclassification rate*)
 - É simplesmente a proporção dos exemplos de teste que são classificados incorretamente pelo classificador
 - Usualmente é mensurada indiretamente através do seu complemento, a taxa de classificação correta
 - Denominada de Acurácia
 - $\text{Acurácia} = 1 - \text{taxa de classificação incorreta}$

Acurácia

- Também chamada de *accuracy* (do inglês)
 - Trata as classes igualmente...
 - Pode não ser adequada para classes desbalanceadas
 - Classe rara é normalmente mais interessante que a majoritária
 - No entanto, a medida tende a privilegiar a classe majoritária

Avaliação de desempenho

- Limitações da Acurácia
 - Considere um problema de duas classes
 - Número de exemplos da classe 0 = 9990
 - Número de exemplos da classe 1 = 10
 - Se o modelo predizer qualquer exemplo como da classe 0, a acurácia será $9990/10000 = 99.9 \%$
 - Acurácia pode ser enganadora

Tipos de Erros

- Em classificação binária, em geral se adota a convenção de rotular os exemplos da classe de maior interesse como positivos (+)
 - Normalmente a classe rara ou minoritária
 - Demais exemplos são rotulados como negativos (–)
- Em alguns casos, os erros têm igual importância
- Em muitos casos, no entanto, esse não é o caso
 - Ex. diagnóstico negativo para indivíduo doente...

Tipos de Erros

- Dois tipos de erro em classificação binária:
 - Classificação de um exemplo N como P
 - Falso Positivo (FP – alarme falso)
 - Ex.: Diagnosticado como doente, mas está saudável
 - Classificação de um exemplo P como N
 - Falso Negativo (FN)
 - Ex.: Diagnosticado como saudável, mas está doente


Matriz de Confusão

- Ou Tabela de Contingência
 - Pode ser usada para distinguir os tipos de erro
 - Base de várias medidas de erro
 - Pode ser usada com duas ou mais classes

Classe Prevista	Classe Verdadeira		
	1	2	3
1	25	10	0
2	0	40	0
3	5	0	20

Avaliação de desempenho

		Classe Verdadeira	
Classe Prevista		P	N
P		70	40
N		30	60



		Classe Verdadeira	
Classe Prevista	P	VP	FP
	N	FN	VN

$$\text{Acuracia} = \frac{VP + VN}{VP + VN + FP + FN}$$

Avaliação de desempenho

$$\text{Taxa de FP} = \frac{FP}{FP + VN}$$

(alarmes falsos)

Erro do tipo I

Classe Verdadeira		P	N
Classe Prevista	P	VP	FP
	N	FN	VN

$$\text{Taxa de FN} = \frac{FN}{VP + FN}$$

Erro do tipo II

Classe Verdadeira		P	N
Classe Prevista	P	VP	FP
	N	FN	VN

Avaliação de desempenho

		Classe Verdadeira	
		P	N
Classe Prevista	P	20	15
	N	30	35

Classificador 1
TFN = 0.6
TFP = 0.3

		Classe Verdadeira	
		P	N
Classe Prevista	P	70	50
	N	30	50

Classificador 2
TFN = 0.3
TFP = 0.5

		Classe Verdadeira	
		P	N
Classe Prevista	P	60	20
	N	40	80

Classificador 3
TFN = 0.4
TFP = 0.2

Avaliação de desempenho

$$\text{Taxa de FP} = \frac{FP}{FP + VN}$$

(Erro tipo I)

$$\text{Precisão} = \frac{VP}{VP + FP}$$

$$\text{Acurácia} = \frac{VP + VN}{VP + VN + FP + FN}$$

$$\text{Especificidade} = \frac{VN}{VN + FP} = 1 - \text{TFP}$$

$$\text{Taxa de VP} = \frac{VP}{VP + FN}$$

(Sensibilidade)

$$\text{Revocação} = \frac{VP}{VP + FN}$$

(Recall)

$$\text{Medida-F} = \frac{2}{1/\text{prec} + 1/\text{rev}}$$

$$\text{Taxa de FN} = \frac{FN}{VP + FN} = 1 - \text{TVP}$$

(Erro tipo II)

Avaliação de desempenho

- Revocação (*recall*, sensibilidade, taxa de VP)
 - Taxa com que classifica como positivos todos os exemplos que são de fato positivos
 - Só considera os exemplos que são positivos
 - Normalmente classe de maior interesse
- Precisão (*precision*)
 - Taxa com que todos os exemplos classificados como positivos são realmente positivos
 - Só considera os exemplos classificados como positivos

Avaliação de desempenho

- Especificidade (*Specificity*)
 - Taxa com que classifica como negativos todos os exemplos que são de fato negativos
 - Só considera os exemplos negativos

Gráficos ROC

- Do inglês, *Receiver Operating Characteristics*
- Medida de desempenho originária da área de processamento de sinais
 - Muito utilizada na área médica
 - Mostra relação entre custo (taxa de FP) e benefício (taxa de VP)
 - Taxa de FP = Erro do Tipo I (alarmes falsos)
 - Taxa de VP (*Recall*, Sensibilidade) = $1 - \text{Erro do Tipo II}$

Exemplo

- Plotar no gráfico ROC os 3 classificadores do exemplo anterior

Classificador 1
TVP = 0.4
TFP = 0.3



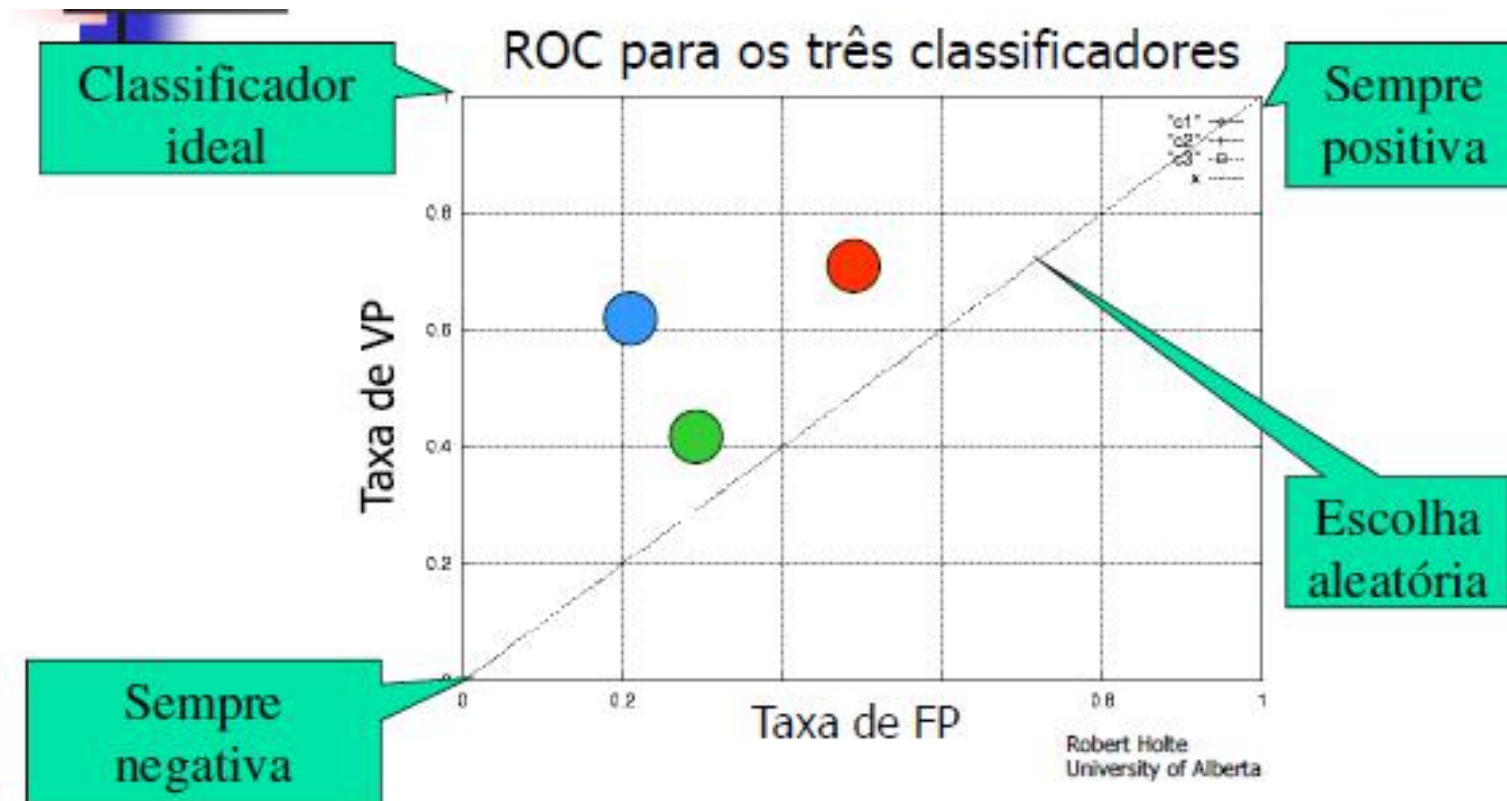
Classificador2
TVP = 0.7
TFP = 0.5



Classificador 3
TVP = 0.6
TFP = 0.2



Exemplo



Gráficos ROC

- Informalmente, melhor classificador é aquele cujo ponto está mais a noroeste
 - Classificadores próximos do canto inferior esquerdo são conservadores
 - Só fazem classificações positivas com forte evidência
 - Assim, cometem poucos erros de FP
 - Classificadores próximos ao canto superior direito são liberais (sob risco de alarme falso)

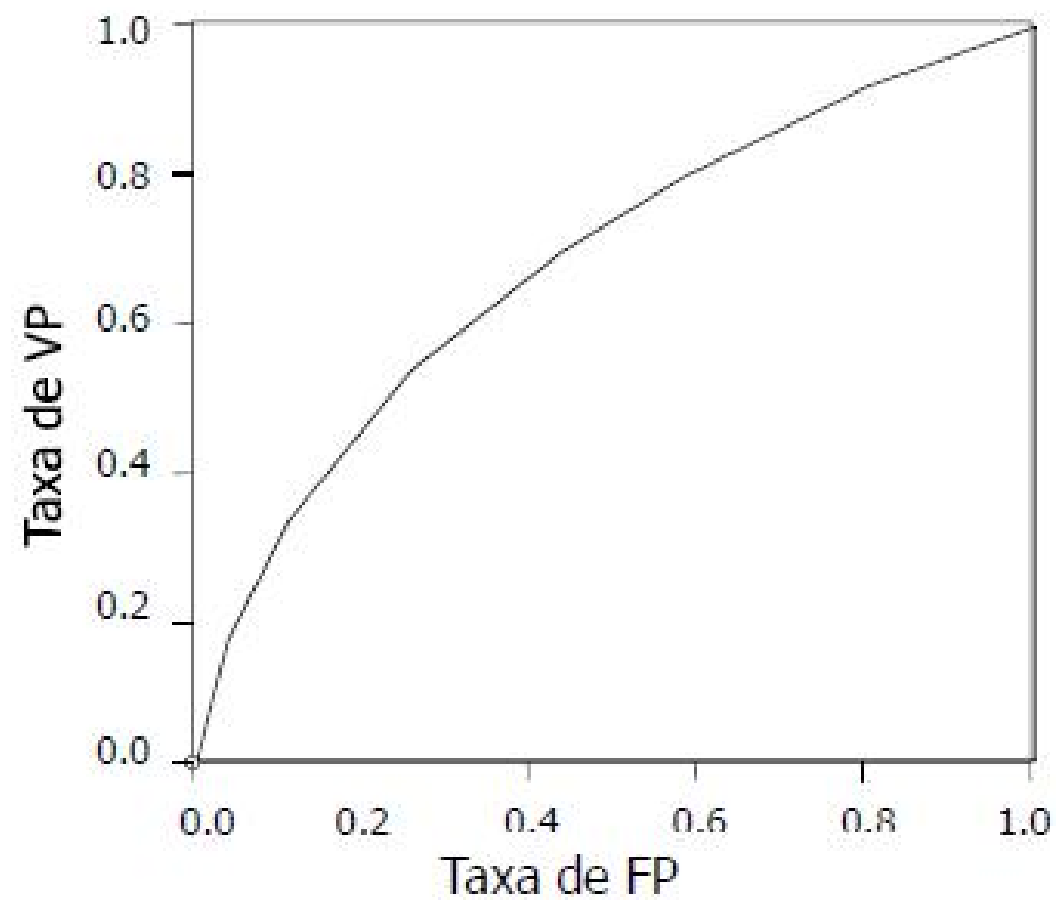
Curvas ROC

- Classificadores que geram escores:
 - Diferentes valores de limiar para os scores associados à classe Positiva podem ser utilizados para gerar um classificador
 - Cada valor produz um classificador diferente
 - Corresponde a um ponto diferente no gráfico ROC
 - Ligação dos pontos gera uma Curva ROC

Curvas ROC



Exemplo:



Classes Difíceis

- Alguns problemas de classificação são caracterizados por possuírem classes difíceis de serem aprendidas por um classificador
 - Duas das principais razões são:
 - Distribuição espacial complexa no espaço dos atributos
 - Classes desbalanceadas
 - Classes raras

Classes desbalanceadas

- No. de exemplos varia para as diferentes classes
 - Natural ao domínio; ou
 - Problema com geração / coleta de dados
- Várias técnicas de DM não conseguem ou têm dificuldade para lidar com esse problema
 - Tendência a classificar na(s) classe(s) majoritária(s)

Classes desbalanceadas

- Alternativa mais simples: Balanceamento Artificial

- **Sobre-amostragem**

Consiste em aumentar artificialmente os exemplos da classe minoritária (classe positiva) até que os dados de treinamento estejam balanceados

- **Sub-amostragem**

Diminui artificialmente os exemplos da classe majoritária (negativa) até que dados de treinamento estejam balanceados

- **Híbrido**

Mescla oversampling e undersampling para amenizar os possíveis problemas de cada abordagem