

Curva ROC

George Darmiton da Cunha Cavalcanti
Tsang Ing Ren
CIn/UFPE



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO

CIn.ufpe.br

Introdução

- **ROC (*Receiver Operating Characteristics*)**
- **Curva ROC é uma técnica para a visualização e a seleção de classificadores baseado no seu desempenho**
- **A Curva ROC tem sido bastante usada pela comunidade de Aprendizagem de Máquina**
 - Pois, em geral, avaliar apenas a taxa de acerto de um classificador é uma métrica muito simples
- **A Curva ROC é bastante útil no trato com domínios cujas classes estejam desbalanceadas e que possuam custos de classificação diferentes por classe**

Classificadores

- Um **classificador** atribui um objeto a uma das categorias ou classes pré-definidas
- **Exemplos:**
 - Uma assinatura pode ser genuína ou falsa;
 - A leitura do valor do cheque pode ser efetuada de forma correta ou incorreta;
 - Uma transação de cartão de crédito pode ser aprovada ou negada;
 - Um teste médico pode dar um parecer positivo ou negativo.

Todos os problemas de classificação podem ser reduzidos para problemas com duas classes.

Performance de um Classificador

■ Saída

- Contínua
 - Uma estimativa da classe do padrão avaliado
 - Nesse caso, diferentes *thresholds* podem ser usados para prever a classe
- Discreta
 - Informa o rótulo da classe

Performance de um Classificador

- Verdadeiro Negativo
- Falso Negativo
- Falso Positivo
- Verdadeiro Positivo

Performance de um Classificador

■ Problema com duas classes

- p – classe positiva
- n – classe negativa

$$tp\ rate \approx \frac{\text{Positives correctly classified}}{\text{Total positives}}$$

$$fp\ rate \approx \frac{\text{Negatives incorrectly classified}}{\text{Total negatives}}$$

		<u>True class</u>	
		p	n
<u>Hypothesized class</u>	Y	True Positives	False Positives
	N	False Negatives	True Negatives

$$fp\ rate = \frac{FP}{N}$$

$$tp\ rate = \frac{TP}{P}$$

$$precision = \frac{TP}{TP+FP} \quad recall = \frac{TP}{P}$$

$$accuracy = \frac{TP+TN}{P+N}$$

Column totals: P N

$$F\text{-measure} = \frac{2}{1/precision+1/recall}$$

Exemplos

		Real	
		pos	neg
Predita	pos	40	30
	neg	60	70



TP = 0.4
FP = 0.3

		Real	
		pos	neg
Predita	pos	70	50
	neg	30	50



TP = 0.7
FP = 0.5

Como avaliar classificadores?

■ Medidas escalares

- Taxa de Acerto (*Accuracy*)
- Área sobre a curva ROC

■ Técnicas de Visualização

- Curvas ROC
- Curvas de Custo

Por que não escalares?

■ Um escalar não conta toda a história

- Existem dois números em jogo (FP e TP). Quando apenas um número é usado, informação preciosa pode ser descartada.
- Como está o comportamento do erro nas classes?
- Como o classificador se comportará em diferentes ambientes?

■ Um escalar impõe uma ordenação linear dos classificadores

- O que se quer é identificar sobre quais condições um é melhor do que o outro

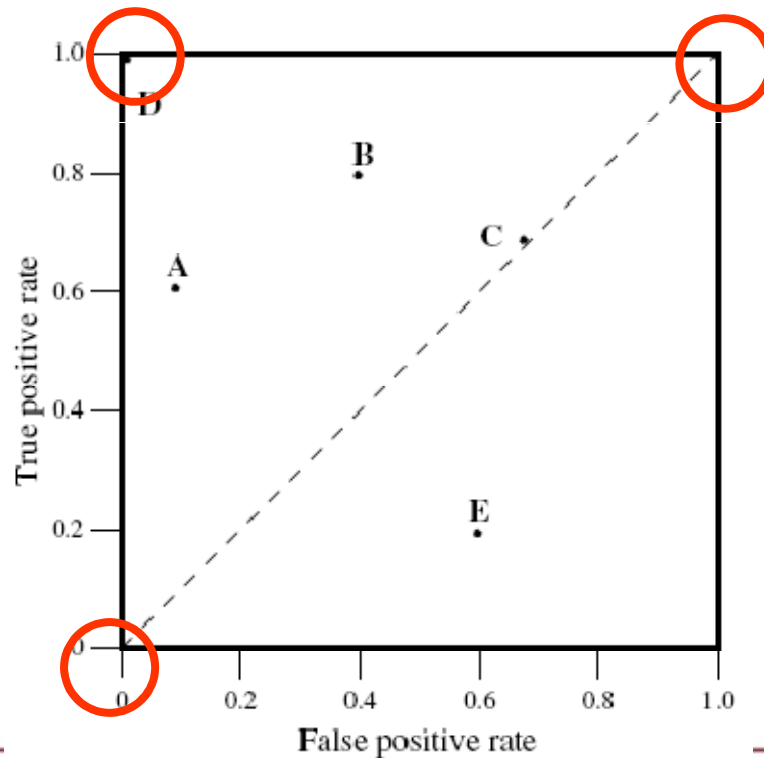
Por que visualizar?

- **Curvas são mais informativas do que um simples número**
- **Informações contidas na curva**
 - Todos os possíveis custos de erro de classificação
 - Sobre quais condições $C1$ supera $C2$

Curva ROC

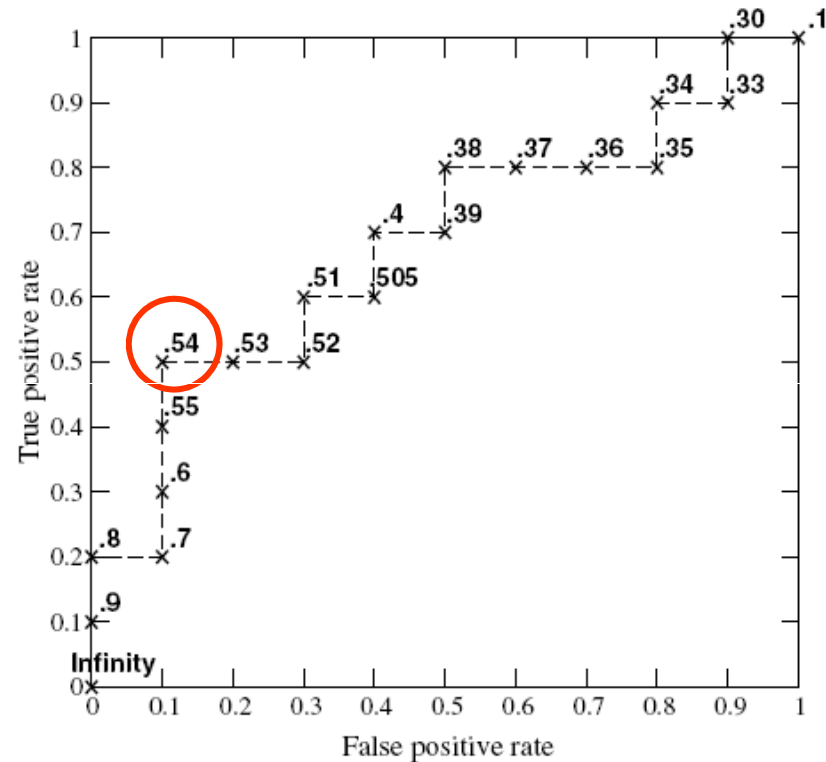
■ Curva ROC

- É um gráfico bidimensional no qual *tp rate* é colocada no **eixo Y** e *fp rate* é colocada no **eixo X**



Curva ROC

Inst#	Class	Score	Inst#	Class	Score
1	p	.9	11	p	.4
2	p	.8	12	n	.39
3	n	.7	13	p	.38
4	p	.6	14	n	.37
5	p	.55	15	n	.36
6	p	.54	16	n	.35
7	n	.53	17	p	.34
8	n	.52	18	n	.33
9	p	.51	19	p	.30
10	n	.505	20	n	.1



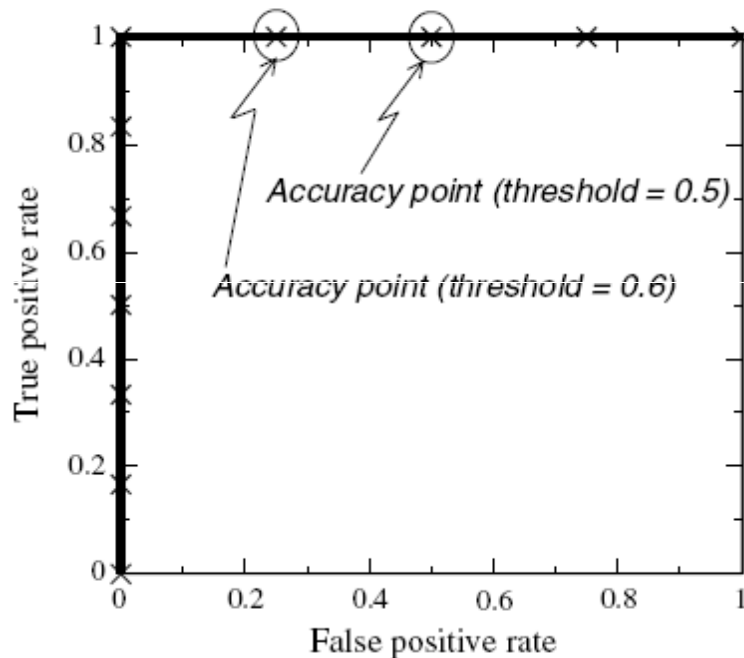
A curva ROC no ponto (0.1,0.5) produz sua maior taxa de acerto: 70%

classe p – 1 erro (1/10 = 0.1)

classe n – 5 erros (5/10 = 0.5)

Total de 6 erros de 20, ou seja, 6/20=0.3 (30% de erro)

Escore Relativo versus Absoluto



Inst no.	Class		Score
	True	Hyp	
1	p	Y	0.99999
2	p	Y	0.99999
3	p	Y	0.99993
4	p	Y	0.99986
5	p	Y	0.99964
6	p	Y	0.99955
7	n	Y	0.68139
8	n	Y	0.50961
9	n	N	0.48880
10	n	N	0.44951

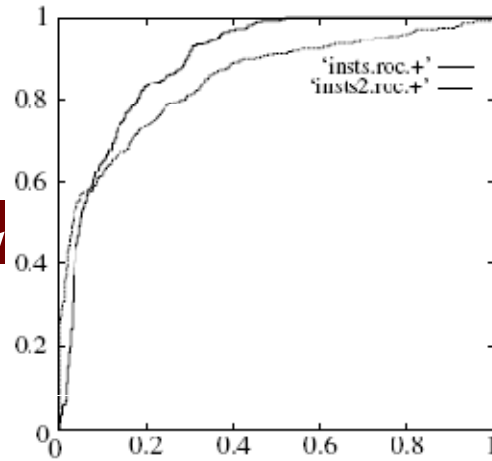
Uma maneira de evitar isso é calibrando os escores dos classificadores
Pois comparar a performance de modelos não calibrados tomando como base um limiar comum é incorreto

Curva ROC para Classes Desbalanceadas

- A curva ROC não é sensível a mudanças na proporção de exemplos positivos e negativos no conjunto de teste
- Ela é baseada nas taxas *tp rate* e *fp rate*, as quais são apenas uma razão que não dependem da distribuição das classes

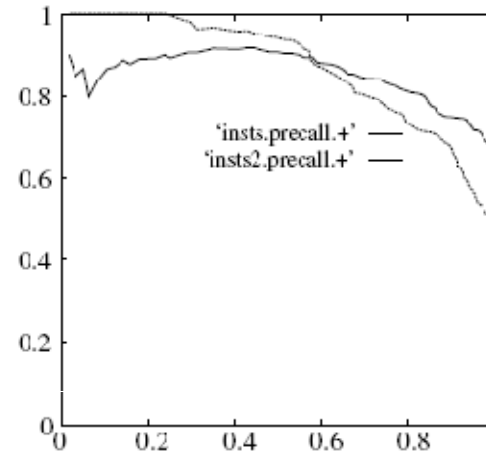
Curva ROC para Classes Desbalanceadas

Curva ROC 1:1



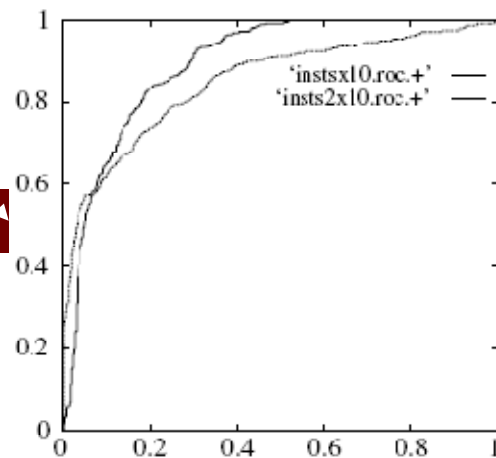
(a)

Constante

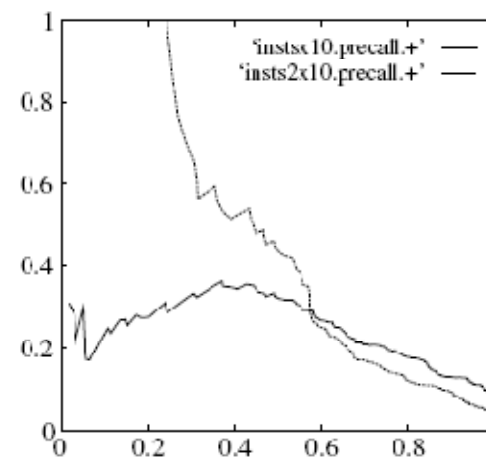


(b)

Curva ROC 1:10



(c)



(d)

Algoritmo Eficiente para Gerar Curvas ROC

Algorithm 1. Efficient method for generating ROC points

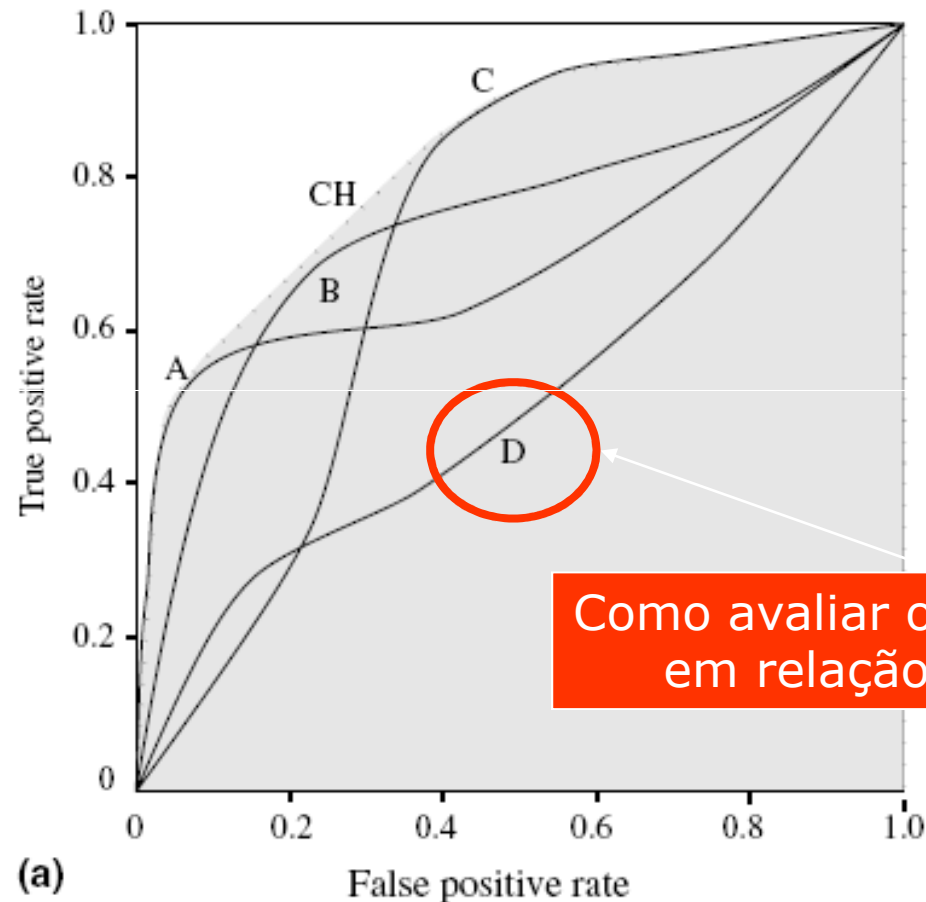
Inputs: L , the set of test examples; $f(i)$, the probabilistic classifier's estimate that example i is positive; P and N , the number of positive and negative examples.

Outputs: R , a list of ROC points increasing by fp rate.

Require: $P > 0$ and $N > 0$

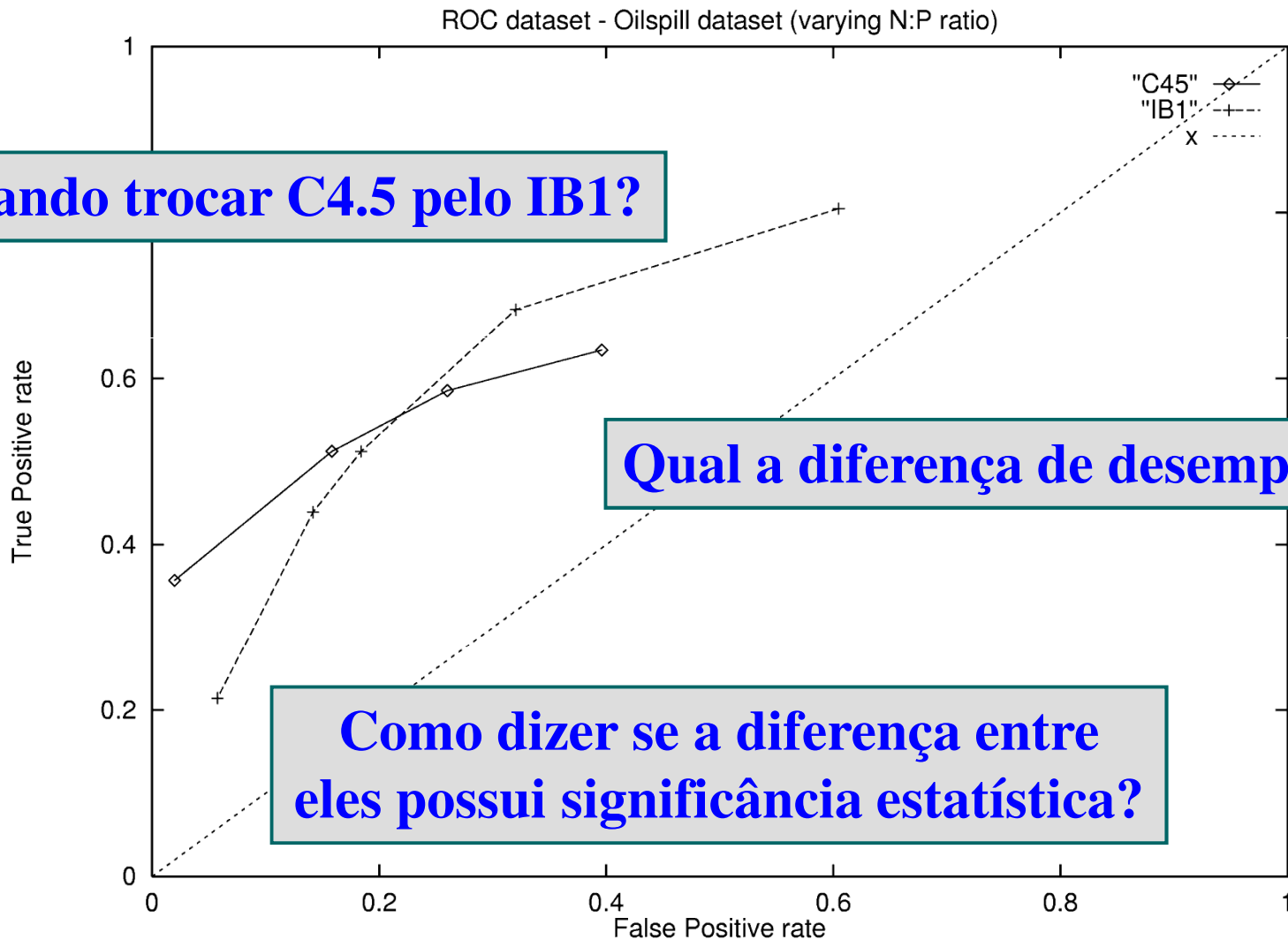
```
1:  $L_{\text{sorted}} \leftarrow L$  sorted decreasing by  $f$  scores
2:  $FP \leftarrow TP \leftarrow 0$ 
3:  $R \leftarrow \langle \rangle$ 
4:  $f_{\text{prev}} \leftarrow -\infty$ 
5:  $i \leftarrow 1$ 
6: while  $i \leq |L_{\text{sorted}}|$  do
7:   if  $f(i) \neq f_{\text{prev}}$  then
8:     push  $\left(\frac{FP}{N}, \frac{TP}{P}\right)$  onto  $R$ 
9:      $f_{\text{prev}} \leftarrow f(i)$ 
10:  end if
11:  if  $L_{\text{sorted}}[i]$  is a positive example then
12:     $TP \leftarrow TP + 1$ 
13:  else /*  $i$  is a negative example */
14:     $FP \leftarrow FP + 1$ 
15:  end if
16:   $i \leftarrow i + 1$ 
17: end while
18: push  $\left(\frac{FP}{N}, \frac{TP}{P}\right)$  onto  $R$  /* This is (1,1) */
19: end
```


The ROC Convex Hull



Um classificador é potencialmente ótimo se e somente se ele está na *convex hull* de um conjunto de pontos na curva ROC

Curvas ROC de dois classificadores



Quando trocar C4.5 pelo IB1?

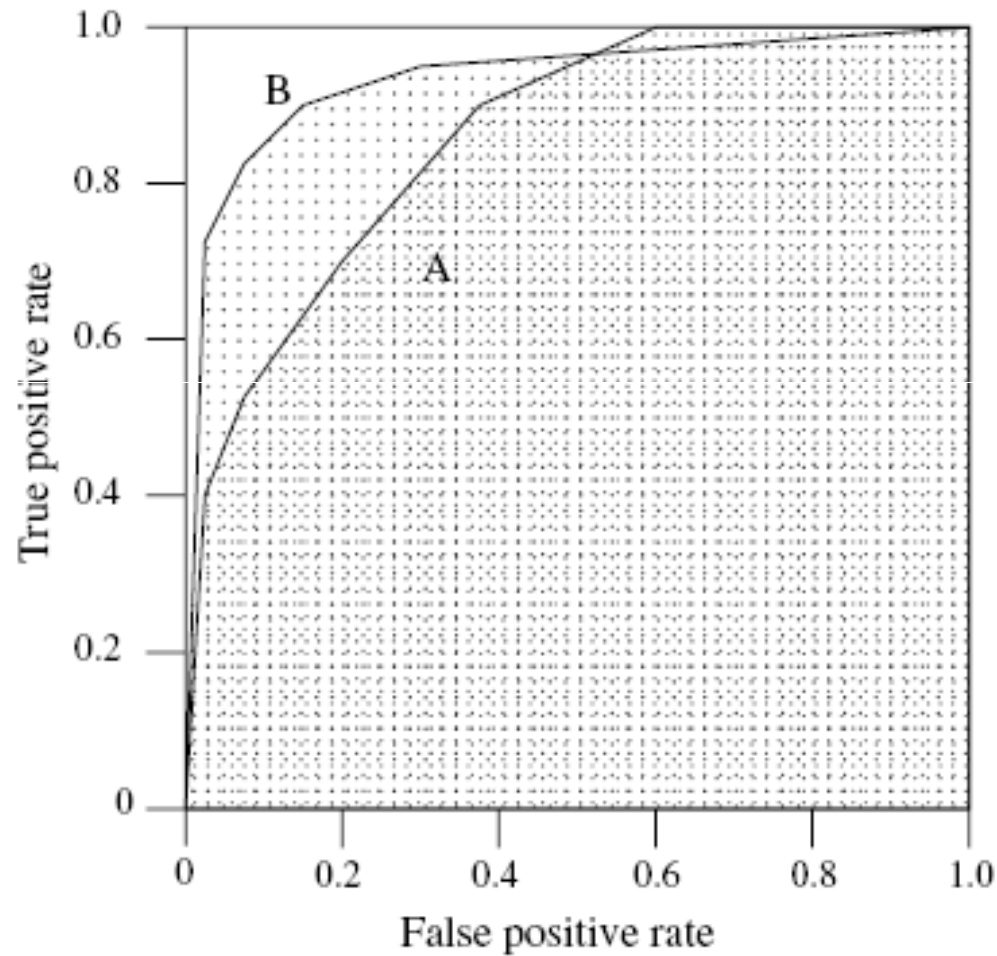
Qual a diferença de desempenho?

Como dizer se a diferença entre eles possui significância estatística?

Área sobre a curva ROC (AUC)

- Para comparar classificadores é desejável reduzir a curva ROC a um simples escalar
- Um método é calcular a área sobre a curva ROC

Área sobre a curva ROC (AUC)



Qual é o melhor classificador: A ou B?

Algoritmo Eficiente para Calcular AUC

Algorithm 2. Calculating the area under an ROC curve

Inputs: L , the set of test examples; $f(i)$, the probabilistic classifier's estimate that example i is positive; P and N , the number of positive and negative examples.

Outputs: A , the area under the ROC curve.

Require: $P > 0$ and $N > 0$

```
1: function TRAPEZOID_AREA( $X1, X2, Y1, Y2$ )
2:  $Base \leftarrow |X1 - X2|$ 
3:  $Height_{avg} \leftarrow (Y1 + Y2)/2$ 
4: return  $Base \times Height_{avg}$ 
5: end function
```

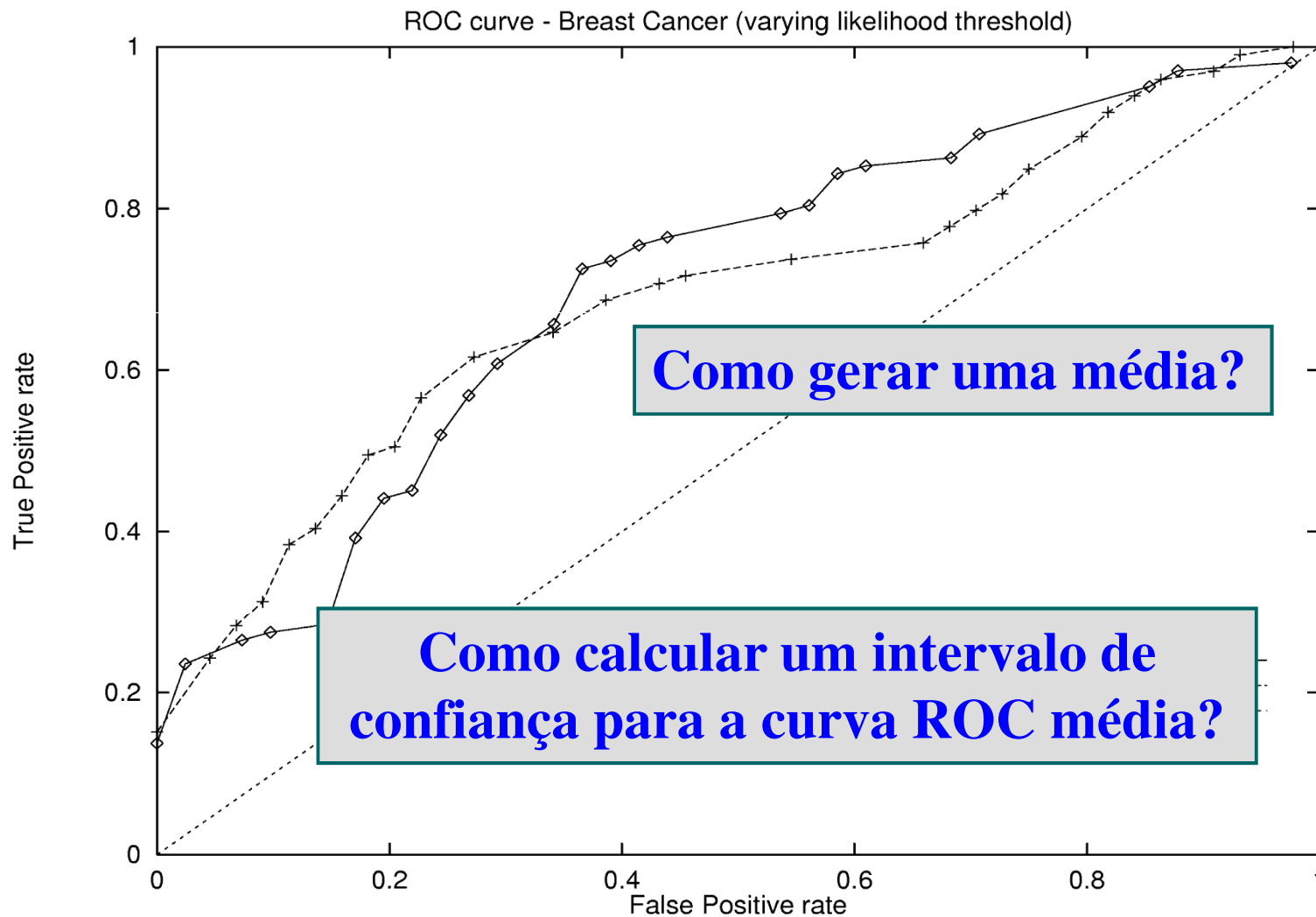
```
1:  $L_{sorted} \leftarrow L$  sorted decreasing by  $f$  scores
2:  $FP \leftarrow TP \leftarrow 0$ 
3:  $FP_{prev} \leftarrow TP_{prev} \leftarrow 0$ 
4:  $A \leftarrow 0$ 
5:  $f_{prev} \leftarrow -\infty$ 
6:  $i \leftarrow 1$ 
7: while  $i \leq |L_{sorted}|$  do
8:   if  $f(i) \neq f_{prev}$  then
9:      $A \leftarrow A + \text{TRAPEZOID\_AREA}(FP, FP_{prev}, TP, TP_{prev})$ 
10:     $f_{prev} \leftarrow f(i)$ 
11:     $FP_{prev} \leftarrow FP$ 
12:     $TP_{prev} \leftarrow TP$ 
13:   end if
14:   if  $i$  is a positive example then
15:      $TP \leftarrow TP + 1$ 
16:   else /*  $i$  is a negative example */
17:      $FP \leftarrow FP + 1$ 
18:   end if
19:    $i \leftarrow i + 1$ 
20: end while
21:  $A \leftarrow A + \text{TRAPEZOID\_AREA}(N, FP_{prev}, N, TP_{prev})$ 
22:  $A \leftarrow A / (P \times N)$  /* scale from  $P \times N$  onto the unit square */
23: end
```

Calculando a Média de Curvas ROC

- **Dados os conjuntos de treinamento**
 - T_1, T_2, \dots, T_k
 - *k-fold cross-validation*
- **Com vários conjuntos, k Curvas ROC podem ser calculadas**
- **Isso proporciona a vantagem de se ter uma medida de variância**

Curva ROC

2-fold-cross-validation



Média de Curvas ROC

■ Estratégias

- *Vertical Averaging*
- *Threshold Averaging*

Média de Curvas ROC (*Vertical Averaging*)

- *Vertical averaging* procura por sucessivos *FP rates* e calcula a média dos *TP rates* de Curvas ROC nos pontos *FP rates*
- Congelando a *FP rate*, é possível gerar um intervalo de confiança para a *TP rate* baseado na média e na variância
- Uma fraqueza desse método é que a *FP rate*, geralmente, não está sob o controle do pesquisador

Algoritmo Eficiente para Calcular Média de Curvas ROC (*vertical*)

Algorithm 3. Vertical averaging of ROC curves

Inputs: *samples*, the number of FP samples; *nrocs*, the number of ROC curves to be sampled, *ROCS*[*nrocs*], an array of *nrocs* ROC curves; *npts*[*m*], the number of points in ROC curve *m*. Each ROC point is a structure of two members, the rates *fpr* and *tpr*.

Output: Array *tpravg*[*samples* + 1], containing the vertical averages.

```
1: s ← 1
2: for fprsample = 0 to 1 by 1/samples do
3:   tprsum ← 0
4:   for i = 1 to nrocs do
5:     tprsum ← tprsum + TPR_FOR_FPR(fprsample,
                                   ROCS[i], npts[i])
6:   end for
7:   tpravg[s] ← tprsum/nrocs
8:   s ← s + 1
9: end for
10: end
1: function TPR_FOR_FPR(fprsample, ROC, npts)
2: i ← 1
3: while i < npts and ROC [i + 1].fpr ≤ fprsample do
4:   i ← i + 1
5: end while
6: if ROC[i].fpr = fprsample then
7:   return ROC[i].tpr
8: else
9:   return INTERPOLATE(ROC[i], ROC [i + 1].fprsample)
10: end if
11: end function
1: function INTERPOLATE(ROCP1, ROCP2, X)
2: slope = (ROCP2.tpr - ROCP1.tpr) / (ROCP2.fpr -
                                       ROCP1.fpr)
3: return ROCP1.tpr + slope · (X - ROCP1.fpr)
4: end function
```

Média de Curvas ROC (*Threshold Averaging*)

- ***Threshold averaging*** procura superar uma potencial fraqueza do ***vertical averaging*** congelando o ***threshold*** ao invés do ***FP rate***
- Escolhe-se um conjunto de ***thresholds*** uniformemente distribuído entre todos os ***thresholds*** da Curva ROC
- Para cada uma dos ***thresholds***, é identificado o conjunto de pontos ROC que seriam gerados usando esse ***threshold*** sobre cada Curva ROC
- Desses pontos ROC, a média e o desvio-padrão são calculados para ***FP*** e ***TP rates***,
 - Gerando a média dos pontos ROC, bem como os intervalos de confiança vertical e horizontal
- Uma fraqueza desse método é que os escores dos classificadores devem estar numa mesma escala
 - Problema: escore relativo *versus* escore absoluto

Algoritmo Eficiente para Calcular Média de Curvas ROC (*threshold*)

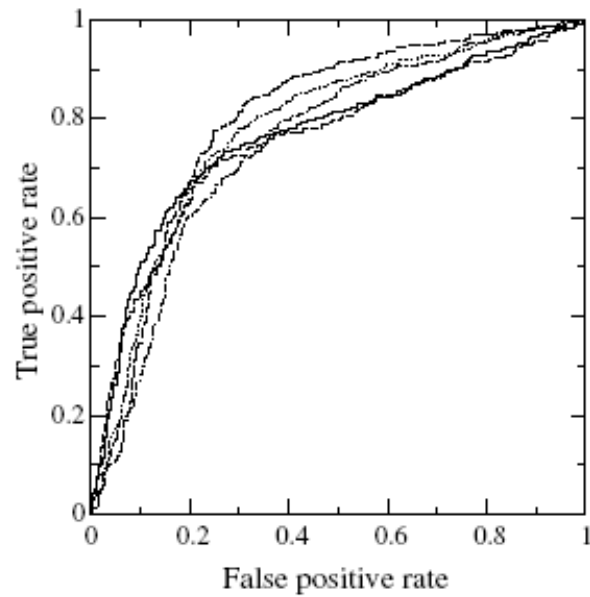
Algorithm 4. Threshold averaging of ROC curves

Inputs: *samples*, the number of threshold samples; *nrocs*, the number of ROC curves to be sampled; *ROCS*[*nrocs*], an array of *nrocs* ROC curves sorted by score; *npts*[*m*], the number of points in ROC curve *m*. Each ROC point is a structure of three members, *fpr*, *tpr* and score.

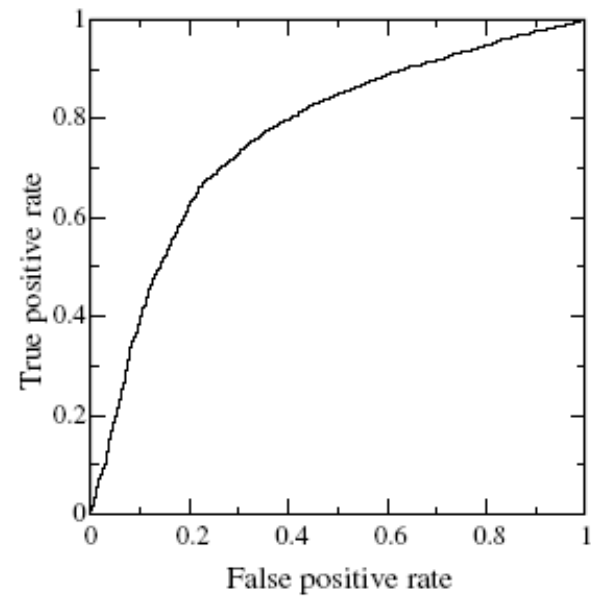
Output: *Avg*[*samples* + 1], an array of (*X*, *Y*) points constituting the average ROC curve.

Require: *samples* > 1

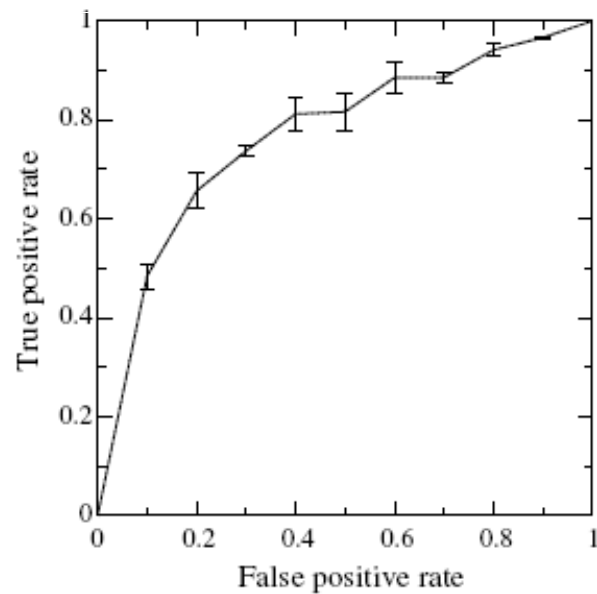
```
1: initialize array T to contain all scores of all ROC
   points
2: sort T in descending order
3: s ← 1
4: for tidx = 1 to length(T) by int(length(T)/samples) do
5:   fprsum ← 0
6:   tprsum ← 0
7:   for i = 1 to nrocs do
8:     p ← ROC_POINT_AT_THRESHOLD(ROCS[i], npts[i],
                                   T[tidx])
9:     fprsum ← fprsum + p.fpr
10:    tprsum ← tprsum + p.tpr
11:   end for
12:   Avg[s] ← (fprsum/nrocs, tprsum/nrocs)
13:   s ← s + 1
14: end for
15: end
1: function ROC_POINT_AT_THRESHOLD(ROC, npts, thresh)
2: i ← 1
3: while i ≤ npts and ROC[i]. score > thresh do
4:   i ← i + 1
5: end while
6: return ROC[i]
7: end function
```



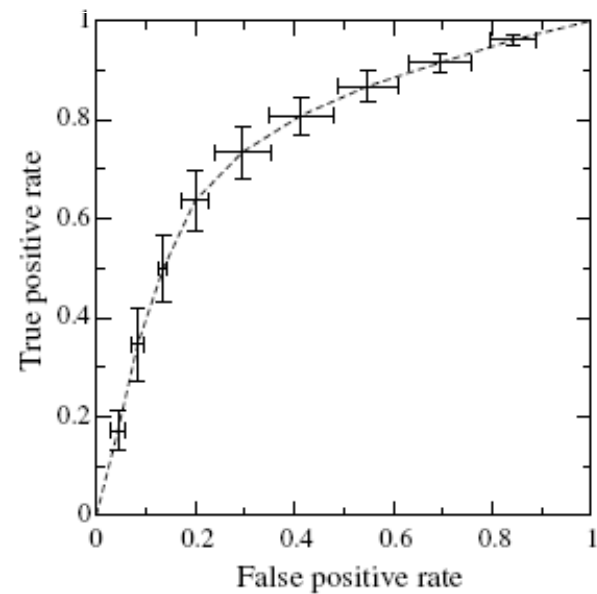
(a)



(b)



(c)



(d)

Fig. 9. ROC curve averaging. (a) ROC curves of five instance samples, (b) ROC curve formed by merging the five samples, (c) the curves of a averaged vertically and (d) the curves of a averaged by threshold.

Problemas com mais de Duas Classes

- Usando mais de duas classes, o problema fica bem mais complexo
- Uma alternativa é gerar n diferentes curvas ROC
 - Uma para cada classe (n classes)
 - Se C é o conjunto de todas as classes, a curva ROC i exibe a classificação usando a classe i como positiva e todas as outras como negativa
- Entretanto, essa abordagem é sensível ao desbalanceamento entre classes

Problemas com Mais de Duas Classes (*Area Under Curve – AUC*)

- Uma alternativa para calcular a AUC para problemas multi-classe é
 - Calcular a AUC para cada classe e depois somar essas áreas de forma ponderada

$$AUC_{\text{total}} = \sum_{c_i \in C} AUC(c_i) \cdot p(c_i)$$

Probabilidade de um elemento ser da classe i ou custo da classe i

- Uma desvantagem é que a classe de referência da ROC é sensível a distribuição das classes

Problemas com Mais de Duas Classes (*Area Under Curve – AUC*)

- Uma alternativa à limitação anterior é uma medida que não seja sensível a distribuição das classes

$$AUC_{\text{total}} = \frac{2}{|C|(|C| - 1)} \sum_{\{c_i, c_j\} \in C} AUC(c_i, c_j)$$

Esse cálculo é realizado para todos os pares de classes, ou seja, $|C| * (|C| - 1) / 2$ vezes

□ Desvantagens

- Não é possível visualizar as áreas calculadas
- Custo computacional

Considerações Finais

- **A curva ROC é uma ferramenta bastante útil para visualização e avaliação de classificadores**
- **Medida de desempenho baseada em escalares não devem ser usadas se o custo e a distribuição das classes não são completamente conhecidas, ou podem variar com o tempo e o ambiente**

Referências

- Tom Fawcett. *An introduction to ROC analysis*. Pattern Recognition Letters, Vol. 27, No. 8., pp. 861-874, 2006.
- Macskassy and Provost. *Confidence Bands for ROC Curves: Methods and an Empirical Study*. Proceedings of the First Workshop on ROC Analysis in AI at ECAI. 2004.
- Provost, Fawcett and Kohavi. *The case against accuracy estimation for comparing induction algorithms*. Proc. of the International Conference on Machine Learning, pp. 445–453. 1998.
 - <http://www.purl.org/NET/tfawcett/papers/ICML98-final.ps.gz>
- Hanley and McNeil. *The meaning and use of the area under a receiver operating characteristic (ROC) curve*. Radiology 143, 29–36. 1982.