

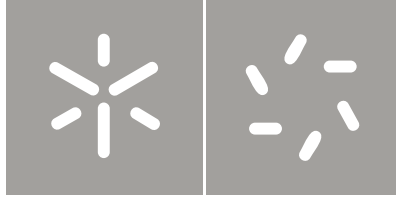


Universidade do Minho
Escola de Ciências

Miguel Basto Pereira

Estimação da Sensibilidade,
da Especificidade e da curva ROC

Miguel Basto Pereira Estimação da Sensibilidade,
da Especificidade e da curva ROC



Universidade do Minho
Escola de Ciências

Miguel Basto Pereira

Estimação da Sensibilidade,
da Especificidade e da curva ROC

Tese de Mestrado
Estatística / Mestrado em Estatística

Trabalho efetuado sob a orientação do
Professor Doutor Luís Filipe Meira Machado

DECLARAÇÃO

Nome: Miguel Basto Pereira

Correio electrónico: miguelbastopereira@hotmail.com

Tlm.: 919722470

Número do Bilhete de Identidade: 13036706 OZZO

Título da dissertação:

Estimação da Sensibilidade, da Especificidade e da curva ROC

Ano de conclusão: 2013

Orientador: Professor Doutor Luís Filipe Meira Machado

Designação do Mestrado: Mestrado em Estatística

Área de Especialização: Estatística

Escola: Escola de Ciências

Departamento: Departamento de Matemática e Aplicações

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.

Guimarães, ___/___/_____

Assinatura: _____

Agradecimentos

Ao meu orientador, Professor Doutor Luís Filipe Meira Machado, pelo conhecimento partilhado, pela aprendizagem e sobretudo pela inteira disponibilidade demonstrada ao longo de todo o processo de planeamento e desenvolvimento da dissertação.

Ao Professor Doutor Jaime Correia de Sousa, por ter possibilitado a conciliação do meu trabalho de investigação na Escola de Medicina da Universidade do Minho com o Mestrado em Estatística. Agradeço igualmente à minha atual casa profissional, a Faculdade de Psicologia e de Ciências da Educação da Universidade do Porto e à Doutora Mariana Martins, por terem também possibilitado a conciliação entre as minhas tarefas laborais e as minhas tarefas académicas.

O mais especial agradecimento aos meus pais e também à minha irmã, por todo o apoio prestado aos mais diversos níveis e sem os quais esta experiência enriquecedora nunca teria sido possível.

Aos colegas de Mestrado em Estatística, pelo companheirismo e pela ajuda prestada ao longo de dois anos de percurso. Em especial, um agradecimento aos colegas: Fernando Gomes, Cristiana Vieira, Celine Queirós, Daniela Catalão, Helena Piairo e Cristina Campelos.

A quem me incentivou a entrar nesta aventura, tão diferente mas ao mesmo tempo tão complementar da minha profissão e da minha formação de base.

Obrigado a todos.

Estimação da Sensibilidade, da Especificidade e da curva ROC

Resumo:

A tomada de decisões é um processo complexo que surge frequentemente ligado ao problema da classificação de indivíduos em grupos a partir de um marcador ordinal ou contínuo associado. Neste contexto os testes diagnósticos assumem especial relevância.

A escolha das provas de diagnóstico e do ponto de corte é geralmente realizada através da avaliação da sensibilidade e especificidade e pela representação da curva ROC (*Receiver Operating Characteristic*), uma poderosa ferramenta estatística.

Na Medicina estas ferramentas assumem um papel central, ao estimar, comparar e calibrar o impacto de diferentes soluções, permitindo que a solução mais eficaz seja disponibilizada.

Este trabalho pretendeu estudar e desenvolver estas ferramentas estatísticas e está estruturado, além da Introdução e Conclusão, em 5 secções que visam responder aos objetivos propostos.

O segundo Capítulo é composto pela revisão bibliográfica e histórica sobre o tema, à medida que são apresentados os conceitos e fórmulas base sobre as curvas ROC.

No Capítulo 3 é efetuada a caracterização e a análise descritiva de uma base de dados de cardiologia que será posteriormente utilizada para testar exemplificar as funções desenvolvidas.

No Capítulo 4 programamos e aplicamos um conjunto de funções que têm por objetivo estimar e representar a sensibilidade, especificidade e a curva ROC, bem como as respetivas bandas de confiança, através dos métodos: *Wald*, *Score* e *Bootstrap*.

O desempenho destes métodos é avaliado por meio de estudos de simulação. Realizou-se um estudo de simulação que será descrito no Capítulo 5. Pretendeu-se avaliar o nível de cobertura dos diferentes métodos e estudar qual é o melhor método sob diferentes condições.

Por fim, o Capítulo 6 é constituído pela programação e aplicação da função, *slife*. A função visa avaliar o custo/retorno unitário esperado num rastreio populacional, a função foi exemplificada recorrendo à base de dados de cardiologia, de modo a aprofundar as suas funcionalidades.

Nesta dissertação foram revistos e programados métodos de estimação da sensibilidade, especificidade e curvas ROC, e a relação destas com o custo e com o retorno esperado. Pretendeu-se disponibilizar informação e ferramentas estatísticas de apoio ao diagnóstico.

Palavras-Chave: Curvas ROC; Sensibilidade; Especificidade; Custo e Retorno Esperado

Estimation of Sensitivity, Specificity and ROC Curve

Abstract:

Decision making is a complex process that is frequently connected to the issue of subject classification into groups, starting from an ordinal or continuous associated marker. Within this context the diagnostic unit tests become particularly relevant.

The choice of diagnostic tests and the cutting location is usually performed by evaluating the sensitivity, specificity and by the ROC (Receiver Operating Characteristic) Curve representation, therefore becoming a powerful statistic tool.

In Medicine these tools acquire a central role by estimating, comparing and calibrating the varied solutions impact, thus providing the most effective solution.

This work sought to study and develop these statistic tools. It is structured, besides the Introduction and Conclusion, in five sections, aiming to meet the proposed objectives.

The second Chapter is composed by the historical and bibliographic revision of the theme, while concepts and base formulas are presented over the ROC curves.

On the third Chapter a characterization is made, and a descriptive analysis is made on a cardiology database that will be used to exemplify and test the developed functions.

On the Chapter four a set of functions are programmed and applied, with the objective of estimating and representing the sensitivity, specificity and ROC curve, as well as the respective confidence intervals through the *Wald*, *Score* and *Bootstrap* methods. The performance of these methods is assessed through simulation studies. One simulation study has been performed and described on Chapter 5. It was aimed to assess the level of coverage from different methods and study which is the best method under various circumstances.

Finally, the Chapter 6 is formed by the programming and implementation of the *slife* function. This function seeks to evaluate the expected cost/benefit per unit on a population screening. The function was exemplified using the cardiology database, in order to enhance its functionalities.

This dissertation revises and programs methods for the sensitivity estimation, specificity and ROC curves, as well as the relationship regarding cost and expected investment return. It was intended to provide information and statistical tools to support the diagnosis.

Keywords: ROC Curves; Sensitivity; Specificity; Cost and Expected Benefit.

Conteúdo

CAPÍTULO 1. INTRODUÇÃO.....	1
CAPÍTULO 2: REVISÃO BIBLIOGRÁFICA E CONCEITOS-CHAVE.....	3
2.1. RESENHA HISTÓRIA	3
2.2 CONCEITOS BASE	4
2.2.1. <i>Função de densidade e Discriminação dos Grupos</i>	6
2.2.2. <i>Sensibilidade e Especificidade</i>	9
2.2.3. <i>Curva ROC</i>	10
2.3. APLICAÇÕES INFORMÁTICAS NO CÁLCULO E REPRESENTAÇÃO GRÁFICA DA CURVA ROC	13
2.3.1. <i>Análise ROC no software estatístico R</i>	13
2.3.2. <i>SPSS</i>	14
2.3.3. <i>EXCEL</i>	14
2.3.4. <i>Outros softwares</i>	14
CAPÍTULO 3: BASE DE DADOS: <i>LEUKOCYTE ELASTASE DATA</i>.....	15
3.1. ESCOLHA DA BASE DE DADOS.....	15
3.2. INFORMAÇÕES GERAIS SOBRE A BASE DE DADOS: “LEUKOCYTE ELASTASE DATA”	16
3.3. ANÁLISE DESCRITIVA BREVE DA BASE DE DADOS	16
CAPÍTULO 4. MÉTODOS DE ESTIMAÇÃO DA SENSIBILIDADE, DA ESPECIFICIDADE E DA CURVA ROC.....	19
4.1. CURVA ROC.....	19
4.1.1 <i>Função Programada: myroc</i>	19
4.2. ÍNDICE DE YODEN E PRECISÃO	23
4.2.1 <i>Função Programada: ypindex</i>	23
4.3. ESTIMAÇÃO DOS INTERVALOS DE CONFIANÇA PARA A SENSIBILIDADE E ESPECIFICIDADE	26
4.3.1. <i>Método Wald para a estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade</i>	26
4.3.2. <i>Método Score para a estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade</i>	27

4.3.3. Método Bootstrap Simples para a estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade.....	28
4.3.4. Função Programada: <i>scut</i>	30
4.4. BANDAS DE CONFIANÇA.....	31
4.4.1. Função programada: <i>plot.secut</i> , <i>plot.spcut</i>	32
4.4.1.1 Função programada: <i>plot.secut</i>	33
4.4.1.1.1 Sensibilidade: Banda de Confiança para os pontos de corte da variável contínua.....	34
4.4.1.1.2 Representação Gráfica das Bandas de Confiança para a Sensibilidade.....	35
4.4.1.2. Função programada: <i>plot.spcut</i>	35
4.4.1.2.1. Especificidade: Bandas de Confiança para os pontos de corte da variável contínua.....	37
4.4.1.2.2. Representação Gráfica das Banda de Confiança para a Especificidade.....	38
4.4.2. Função Programada: <i>SpSe</i>	38
CAPÍTULO 5. ESTUDO DE SIMULAÇÃO AOS INTERVALOS DE CONFIANÇA DA SENSIBILIDADE E ESPECIFICIDADE.....	43
5.1. MÉDIA DOS NÍVEIS DE COBERTURA DOS INTERVALOS DE CONFIANÇA.....	44
5.2. AVALIAÇÃO GRÁFICA DOS NÍVEIS DE COBERTURA.....	49
5.2.1. Nível de Cobertura médio por método de cálculo dos intervalos de confiança e tamanho da amostra.....	50
5.2.2. Nível de Cobertura médio por ponto de corte.....	51
5.2.2.1. Em relação ao método de cálculo dos intervalos de confiança.....	51
5.2.2.2. Em relação ao tamanho da amostra.....	52
5.2.2.3. Níveis médios de Cobertura por proporção.....	53
5.3. AVALIAÇÃO GERAL DOS NÍVEIS DE COBERTURA.....	54
CAPÍTULO 6. RELAÇÃO ENTRE OS ÍNDICES DA CURVA ROC E O CUSTO/RETORNO ESPERADO.....	57
6.1 FUNÇÃO PROGRAMADA: SLIFE.....	59
6.1.1. Custo e retorno unitário esperado por ponto de corte.....	61
6.1.2. Custo unitário esperado em função do retorno unitário esperado.....	65
6.1.3. Relação entre sensibilidade, especificidade e o Custo/ Retorno Unitário Esperado.....	67

CAPÍTULO 7. CONCLUSÕES E PROJETOS FUTUROS	69
BIBLIOGRAFIA	71
ANEXO A. CÓDIGO DE SINTAXE E PROGRAMAÇÃO	75
ANEXO B. VERSÃO NÃO REDUZIDA DAS TABELAS COM OS VALORES DE SENSIBILIDADE, ESPECIFICIDADE, LIMITES DE CONFIANÇA E PONTOS DE CORTE.....	109
ANEXO C. VERSÃO NÃO REDUZIDA DA TABELA COM OS VALORES DA PRECISÃO E DO ÍNDICE DE YODEN	113
ANEXO D. VERSÃO NÃO REDUZIDA DA TABELA COM OS VALORES DO CUSTO UNITÁRIO ESPERADO E RETORNO UNITÁRIO ESPERADO	117

Lista de Notação

Amplit: Amplitude

AUC: área abaixo da curva ROC

BC: Bandas de confiança

C: Custo

Custunit: Custo unitário esperado

elas: Denominação reduzida da base de dados *Leukocyte Elastase Data* e simultaneamente denominação da variável contínua com o mesmo nome

FN: Falsos negativos

FP: Falsos positivos

IY: Índice de Youden

IRCE: Incremento da razão de custo-efetividade

LI: Limite inferior

LS: Limite Superior

nboot: número de subamostras a gerar pelo método *bootstrap*

nn: tamanho amostral dos casos negativos

np: tamanho amostral dos casos positivos

R: retorno

Retunit: Retorno unitário esperado

ROC: Receiver Operating Characteristic

Se: Sensibilidade

Sp: Especificidade

status: estado da doença (1: presente, 0: ausente)

Var (Se): Variância da Sensibilidade

Var (Sp): Variância da especificidade

VN: Verdadeiros negativos

VP: Verdadeiros positivos

Índice de Figuras

Figura 1. Árvore de decisão com o universo de eventos possíveis na análise ROC.....	4
Figura 2. Distribuição estatística da variável contínua para cada um dos níveis da variável categórica (Grupo Doentes versus Grupo se a Doença).....	7
Figura 3. Variação da sensibilidade e especificidade em função dos pontos de corte.....	10
Figura 4. Duas Curvas ROC, calculadas em R, cada uma associada a um teste de avaliação diferente.....	11
Figura 5. Curva ROC, calculada com recurso ao R. No eixo dos “XX” encontra-se a taxa dos falsos positivos e no eixo dos “YY” o nível de especificidade.	12
Figura 6. Representação gráfica da caixa de bigodes entre a variável Status e a variável Leukocyte Elastase Data para o sexo masculino e feminino.	17
Figura 7. Curva ROC, no eixo dos “XX” os falsos positivos (1-Especificidade) e no eixo dos “YY” o valor da sensibilidade.	22
Figura 8. Os gráficos 3D apresentam a relação entre 1-Especificidade, a Sensibilidade e a Precisão (Azul) / Índice de Youden (Amarelo).....	25
Figura 9. Curva ROC e bandas de confiança para a sensibilidade e especificidade/falsos positivos a 95%.....	32
Figura 10. Representação gráfica das bandas de confiança para a sensibilidade através dos 3 métodos (score, bootstrap (nboot=500) e Wald).	35
Figura 11. Representação gráfica das bandas de confiança para a especificidade através dos 3 métodos (Score, Bootstrap (nboot=500) e Wald).	38
Figura 12. Representação gráfica da curva ROC e das bandas de confiança ROC para o método Wald, Score e Bootstrap na versão em 2D e com a variável ponto de corte no gráfico 3D.	42
Figura 13. Níveis médios de cobertura por método e tamanho de amostra.....	50
Figura 14. Níveis médios de cobertura por método e ponto de corte.....	51
Figura 15. Níveis médios de cobertura por tamanho de amostra e ponto de corte.....	52
Figura 16. Níveis médios de cobertura por método e proporção.	53

Figura 17. Níveis médios de cobertura por ponto de corte, tamanho da amostra e método.	55
Figura 18. Gráfico 3D que analisa a relação entre o custo e o retorno unitário esperado em função de cada ponto de corte.....	63
Figura 19. Gráfico que analisa a relação entre o custo e o retorno esperado em 10000 habitantes.	64
Figura 20. Custo unitário esperado em função do retorno unitário esperado e Retorno unitário esperado em função dos pontos de corte.....	65
Figura 21. Representação gráfica com a relação entre os pontos de corte, o custo unitário esperado e o retorno unitário esperado (focado entre 0.020 e 0.030).....	66
Figura 22. Representação gráfica com a relação entre a sensibilidade, especificidade em relação ao retorno unitário esperado e em relação ao custo unitário esperado.	67

Índice de tabelas

Tabela 1. Tabela de Contingência 2x2.....	6
Tabela 2. Análise descritiva da base de dados Leukocyte Elastase Data.....	16
Tabela 3. Lista reduzida dos pares sensibilidade/especificidade da curva ROC.....	21
Tabela 4. Esta tabela apresenta uma lista reduzida dos resultados da função <i>ypindex</i>	24
Tabela 5. Ponto de corte selecionado, respectiva Especificidade, Sensibilidade e limites de confiança (superior e inferior) para cada um dos métodos.....	31
Tabela 6. Sensibilidade, limites de confiança para a sensibilidade em cada ponto de corte	34
Tabela 7. Lista reduzida com valores da especificidade e limites de confiança, superior e inferior para a especificidade nos três métodos e pontos de corte associados. Calculou-se a amplitude.	37
Tabela 8. Lista reduzida com valores da sensibilidade e especificidade e respectivos limites de confiança para os 3 métodos.....	40
Tabela 9. Resultado do estudo de simulação para uma proporção de 0,2 e um nível de confiança de 0.95.	45
Tabela 10. Resultado do estudo de simulação para uma proporção de 0,4 e um nível de confiança de 0.95.	46
Tabela 11. Resultado do estudo de simulação para uma proporção de 0,6 e um nível de confiança de 0.95.	47
Tabela 12. Resultado do estudo de simulação para uma proporção de 0,8 e um nível de confiança de 0.95.	48
Tabela 13. Tabela reduzida com o custo e retorno unitário esperado para cada ponto de corte.	62

CAPÍTULO 1. Introdução

Avaliar e compreender a forma como uma variável contínua pode classificar objetos ou pessoas em grupos distintos é uma necessidade científica comum a diversas áreas. Nas ciências da saúde são utilizados testes de rastreio, através da avaliação de biomarcadores, que permitem detetar patologias numa fase precoce do seu desenvolvimento [1-4].

Mas não é só na Medicina que a capacidade de discriminar e classificar corretamente dois grupos é importante, mas também na economia [4], nomeadamente na avaliação da desigualdade de salários e avaliação de consumo, na meteorologia e na capacidade de discriminação de eventos meteorológicos raros e perigosos [5], na informática, por exemplo no reconhecimento de padrões e processos de aprendizagem do computador [6-7] e na Psicologia e Psiquiatria, na capacidade de discriminação e classificação de determinadas doenças mentais ou défices cognitivos [8].

No entanto para uma utilização eficaz dos testes de classificação não chega saber que determinada medida tem um papel importante na discriminação de grupos, é necessário também conhecer qual é a capacidade de discriminação da medida e qual o melhor ponto de corte na execução dos objetivos. Por ponto de corte considerou-se o valor quantitativo a partir do qual um determinado teste deixa de se presumir negativo para ser considerado positivo. Por, exemplo, em alguns casos poderá ser mais dispendioso ou danoso classificar erradamente uma pessoa doente como saudável, do que classificar incorretamente uma pessoa saudável como doente. O ponto de corte que separa os dois grupos deverá ter em conta todos estes fatores.

A *Receiver Operating Characteristic* (ROC) é uma das principais soluções para o problema apresentado. É uma ferramenta poderosa para avaliar o desempenho de testes e procedimentos e definir pontos de corte que satisfaçam as necessidades de diagnóstico e

avaliação, demarcando-se como uma das principais soluções para os problemas de diagnóstico [9].

Sendo uma técnica marcadamente utilizada atualmente, no próximo tópico vamos fazer uma resenha histórica sobre o desenvolvimento da Curva ROC até aos dias de hoje.

CAPÍTULO 2: Revisão Bibliográfica e Conceitos-Chave

2.1. Resenha história

Os primeiros estudos que deram origem à análise ROC foram realizados na área da Psicologia, mais precisamente no campo da Psicologia Experimental. Fechner, matemático, mas também um dos primeiros cientistas da história da psicologia, tinha por objetivo estabelecer uma relação entre o corpo e a mente [6].

Fechner, nas suas primeiras experiências pretendia saber a partir de que grau de intensidade um estímulo é reconhecido pelo sujeito e discriminado de outros estímulos. Estabeleceu assim o critério de *escolha-forçada* que viria a ser muito relevante nos estudos consequentes da análise ROC. O método de escolha-forçada obrigava um sujeito a escolher entre uma opção ou o seu oposto (e.g. Reconheço determinado estímulo *vs* Não reconheço). Estes resultados eram subseqüentemente analisados com recurso à estatística, em que se relacionava a intensidade do estímulo com o seu reconhecimento. Thurstone foi precursor de Fechner, tendo desenvolvido uma série de estudos psicométricos que continuaram com o desenvolvimento dos estudos da análise discriminante [6].

No entanto, a primeira utilização prática da curva ROC aparece durante a segunda guerra mundial e foi aplicada na área das tecnologias, nomeadamente para responder à necessidade de deteção correta de aviões japoneses através dos sinais de radar. Entre a década de 40 e 70, Wald e Green e Sweets deram um forte impulso a análise ROC, nomeadamente na área da deteção do sinal, estabelecendo e sistematizando os mais modernos procedimentos da construção e análise da curva ROC, a maioria desses contributos são ainda úteis (e.g. maximização da combinação esperada, maximização do valor esperado) [6,10].

Desde a década de 70 e até aos dias de hoje a curva ROC tem sido utilizada nas mais diversas aplicações práticas, tendo atualmente um papel muito importante na Medicina e Engenharia Biomédica, onde permitiu calibrar e validar modelos de diagnóstico e deteção precoce que têm vindo a salvar muitas vidas [1-2], uma vez que a análise ROC permite avaliar qual a capacidade que tem um teste de diagnóstico/avaliação para discriminar entre duas populações (por exemplo, doentes *vs* não doentes) [3].

2.2 Conceitos Base

Na análise ROC pondera-se essencialmente duas variáveis. Uma variável dicotómica, representativa de um evento que se encontra presente ou ausente, por exemplo o estado de saúde: Doença (D) *versus* Ausência de Doença (\bar{D}) e uma variável contínua ou ordinal, com o determinado ponto de corte que categoriza os indivíduos identificados como portadores (I) ou identificados como não portadores da doença (\bar{I}) [3,10]. A árvore de decisão que se segue apresenta a sua aplicação na deteção de uma doença [6].

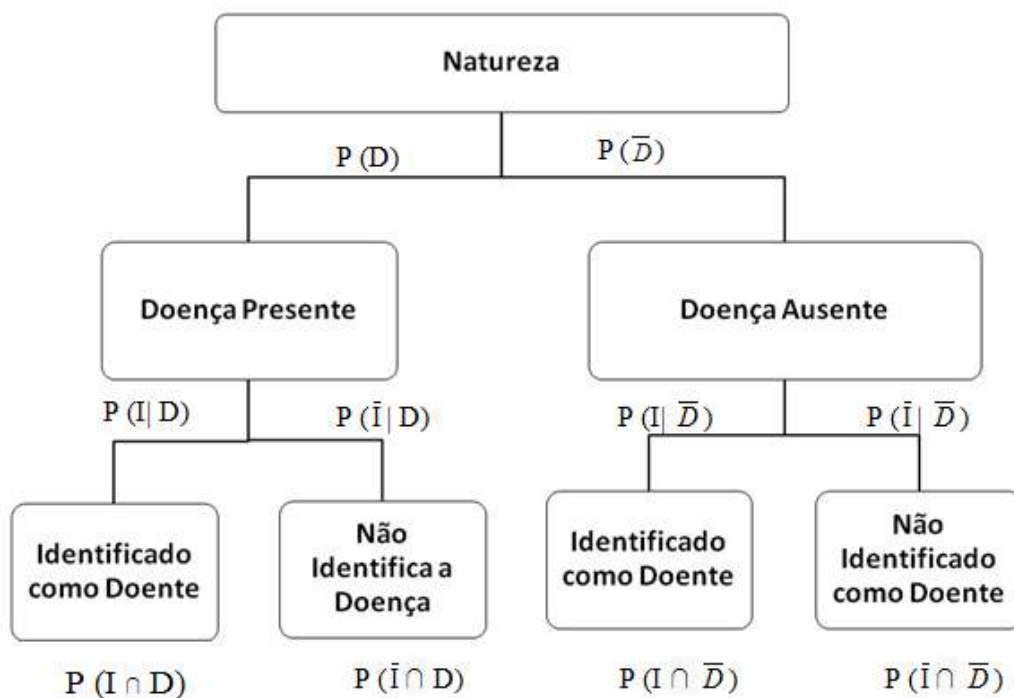


Figura 1. Árvore de decisão com o universo de eventos possíveis na análise ROC.

Esta árvore de probabilidade apresenta quatro cenários possíveis: 1) Indivíduo Doente, corretamente identificado, denominado *Verdadeiro Positivo*; 2) Indivíduo Doente avaliado incorretamente como não doente ou *Falso Negativo*; 3) Sem a doença mas avaliado como doente ou *Falso Positivo*; e 4) Sem a doença e está avaliado corretamente diagnosticado ou *Verdadeiro Negativo* [1,11]. Cada um dos possíveis eventos pode ser definido probabilisticamente do seguinte modo.

$$\text{Verdadeiro Positivo: } P(I \cap D) = P(D) P(I | D) \quad (1)$$

$$\text{Falso Negativo: } P(\bar{I} \cap D) = P(D) P(\bar{I} | D) \quad (2)$$

$$\text{Verdadeiros Negativos: } P(\bar{I} \cap \bar{D}) = P(\bar{D}) P(\bar{I} | \bar{D}) \quad (3)$$

$$\text{Falso Positivo: } P(I \cap \bar{D}) = P(\bar{D}) P(I | \bar{D}) \quad (4)$$

Com base na árvore de decisão podemos também definir o Universo de acontecimentos possíveis em cada grupo:

$$\text{Universo de Acontecimentos: } P(D) + P(\bar{D}) = 1 \quad (5)$$

$$\text{Universo de Acontecimentos na Doença: } P(I | D) + P(\bar{I} | D) = 1 \quad (6)$$

$$\text{Probabilidade de Doença: } P(I \cap D) + P(\bar{I} \cap D) = P(D) \quad (7)$$

$$\text{Probabilidade de Avaliação como doente: } P(I \cap D) + P(I \cap \bar{D}) = P(I) \quad (8)$$

$$\text{Universo de Acontecimentos na Ausência de Doença: } P(\bar{I} | \bar{D}) + P(I | \bar{D}) = 1 \quad (9)$$

$$\text{Probabilidade de Ausência de Doença: } P(\bar{I} \cap \bar{D}) + P(I \cap \bar{D}) = P(\bar{D}) \quad (10)$$

$$\text{Probabilidade de Avaliação como não doente: } P(\bar{I} \cap D) + P(\bar{I} \cap \bar{D}) = P(\bar{I}) \quad (11)$$

Tabela de Contingência 2x2:

A fração de acertos e erros, acima enunciados pode ser avaliada pela próxima tabela de contingência [10]:

		Avaliação Médica	
		Avaliado doente	Avaliado não doente
Estado de Saúde	Doente	Verdadeiro-Positivo $P(I D)$	Falso Negativo $(\bar{I} D)$
	Não Doente	Falso Positivo $P(I \bar{D})$	Verdadeiro Negativo $P(\bar{I} \bar{D})$

Tabela 1. Tabela de Contingência 2x2.

2.2.1. Função de densidade e Discriminação dos Grupos

A distinção correta entre grupos é uma das principais necessidades em diversos campos científicos, por exemplo no caso de diagnóstico médico, necessidade de distinção entre portadores e não portadores de uma determinada doença, com base numa variável de avaliação discriminante, o nosso problema pode assim ser definido em termos de função de densidade [6]. Assim se considerarmos X uma variável aleatória, que caracteriza a pontuação dos indivíduos num determinado teste de avaliação médica, podemos definir isso em termos de funções de densidade, assim $f(x | \bar{D})$ caracteriza a distribuição dos valores de não doentes no exame de avaliação e $f(x | D)$ que caracteriza a distribuição dos valores no exame de avaliação para os indivíduos doentes.

Por exemplo, se soubermos que a média de pontuação no teste de avaliação médica para não doentes é de 10 e o desvio-padrão de 3 e que a média no teste de avaliação para doentes é 20 e o desvio padrão 4 e que ambas as distribuições são gaussianas, podemos então definir um ponto de corte, construir um gráfico e estimar qual a probabilidade associada a cada um dos acontecimentos possíveis.

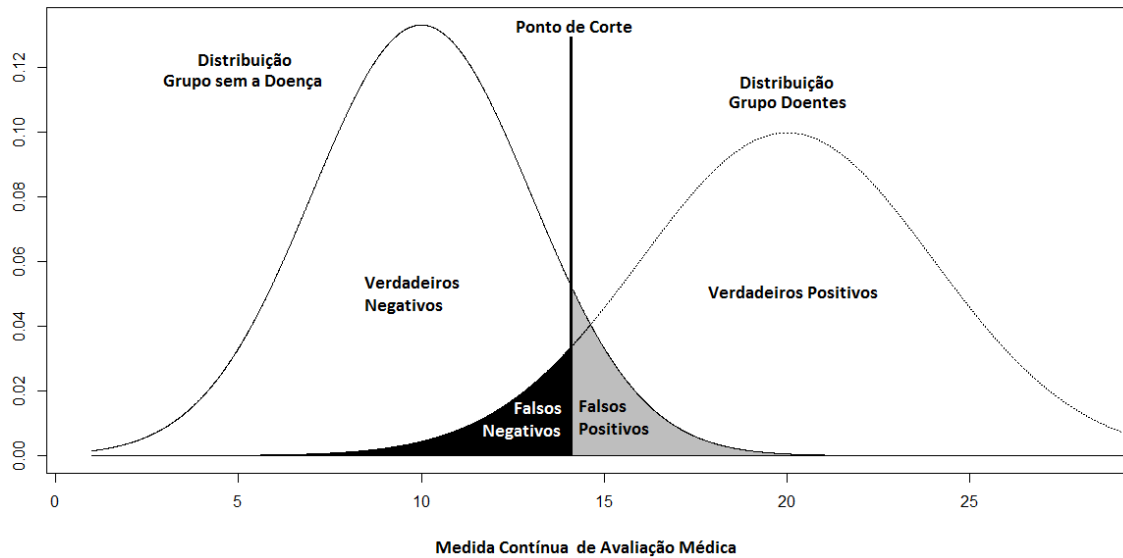


Figura 2. Distribuição estatística da variável contínua para cada um dos níveis da variável categórica (Grupo Doentes versus Grupo se a Doença).

Uma avaliação breve do gráfico dá-nos conta da fração de verdadeiros negativos e de falsos positivos na distribuição *gaussiana* do grupo de não doentes (indivíduos não doentes, corretamente ou incorretamente avaliados) e verdadeiros positivos ou falsos negativos (indivíduos doentes correta ou incorretamente avaliados) no grupo dos portadores da doença. Podemos então calcular, com base nas amostras de doentes e não doentes, as várias probabilidades de interesse, nomeadamente:

Fração de Verdadeiros Positivos

$$P(I | D) = \frac{P(I \cap D)}{P(D)} = \frac{\frac{\text{Número de Verdadeiros Positivos}}{\text{Número Total}}}{\frac{\text{Número de Casos Positivos}}{\text{Número total}}} = \frac{\text{Número de Verdadeiros Positivos}}{\text{Número de Casos Positivos}} \quad (12)$$

Fração de Falsos Negativos

$$P(\bar{I} | D) = \frac{P(\bar{I} \cap D)}{P(D)} = \frac{\frac{\text{Número de Falsos Negativos}}{\text{Número Total}}}{\frac{\text{Número de Casos Positivos}}{\text{Número total}}} = \frac{\text{Número de Falsos Negativos}}{\text{Número de Casos Positivos}} \quad (13)$$

Fração de Verdadeiros Negativos

$$P(\bar{I} \cap \bar{D}) = \frac{P(\bar{I} \cap \bar{D})}{P(\bar{D})} = \frac{\frac{\text{Número de Verdadeiros Negativos}}{\text{Número Total}}}{\frac{\text{Número de Casos Negativos}}{\text{Número total}}} = \frac{\text{Número de Verdadeiros Negativos}}{\text{Número Casos Negativos}} \quad (14)$$

Fração de Falsos Positivos

$$P(I \cap \bar{D}) = \frac{P(I \cap \bar{D})}{P(\bar{D})} = \frac{\frac{\text{Número de Falsos Positivos}}{\text{Número Total}}}{\frac{\text{Número de Casos Negativos}}{\text{Número total}}} = \frac{\text{Número de Falsos Positivos}}{\text{Número de Casos Negativos}} \quad (15)$$

Definição de Hipóteses:

Estas probabilidades podem ser avaliadas em termos de testes de hipóteses:

H_0 : Indivíduo não ser portador da doença

H_1 : O Indivíduo é portador da doença

Deste modo podemos elaborar as nossas hipóteses como probabilidades condicionadas [3].

Verdadeiros Positivos

$$P(\text{Aceitar } H_1 | H_1) \quad (16)$$

Verdadeiros Negativos

$$P(\text{Aceitar } H_0 | H_0) \quad (17)$$

Fração de falsos negativos

$$P(\text{Aceitar } H_0 | H_1) \text{ ou erro de tipo I } (\alpha) \quad (18)$$

Falsos positivos

$$P(\text{Rejeitar } H_0 | H_0) \text{ ou Erro de tipo II.} \quad (19)$$

Por norma o objetivo é que o ponto de corte permita que as probabilidades de $P(\text{Aceitar } H_1 | H_1)$ e $P(\text{Aceitar } H_0 | H_0)$ sejam as maiores possíveis.

2.2.2. Sensibilidade e Especificidade

Um dos principais métodos de avaliação da capacidade de discriminação dos grupos é a fração de verdadeiros positivos ou *Sensibilidade* (Se), ou seja, a probabilidade condicional do resultado ser positivo quando o evento está presente e a fração de verdadeiros negativos ou *Especificidade* (Sp), que designa a probabilidade condicional de um teste ser negativo quando o evento não está presente [12-13].

$$Se = P (I | D) = \frac{\text{Número de Verdadeiros Positivos}}{\text{Número de Casos Positivos}} \quad (20)$$

$$Sp = P (\bar{I} | \bar{D}) = \frac{\text{Número de Verdadeiros Negativos}}{\text{Número de Casos Negativos}} \quad (21)$$

Pode ainda ser determinada a variância (Var) associada a cada um destes índices [12]:

$$\text{Var (Se)} = \frac{Se(1-Se)}{\text{Número de Casos Positivos}} \quad (22)$$

$$\text{Var (Sp)} = \frac{Sp(1-Sp)}{\text{Número de Casos Negativos}} \quad (23)$$

A especificidade e sensibilidade representam todo o universo de casos possíveis, uma vez que:

$$\text{Fração de Verdadeiros Positivos} = \text{Sensibilidade} \quad (24)$$

$$\text{Fração de Falsos Negativos} = 1 - \text{Sensibilidade} \quad (25)$$

$$\text{Fração de Verdadeiros Negativos} = \text{Especificidade} \quad (26)$$

$$\text{Fração de Falsos Positivos} = 1 - \text{Especificidade} \quad (27)$$

Por esse motivo, a *Sensibilidade* e a *Especificidade* são dos índices mais utilizados na avaliação da discriminação entre grupos.

Como se pode verificar pela fórmula da variância, o ideal é que a Sensibilidade e a Especificidade assumam o valor de 1. No caso de ser 1 a capacidade de discriminação é perfeita e a variância de ambos os índices será nula.

No entanto raramente a capacidade de discriminação é perfeita, e como a sensibilidade e especificidade são dependentes, o aumento de um dos índices implica geralmente uma diminuição do outro índice [3]. Deste modo na discriminação entre grupos o ponto de corte torna-se essencial, porque vai delimitar o nível de sensibilidade e de especificidade esperado e deverá ser escolhido de acordo com as necessidades do utilizador [1,9].

2.2.3. Curva ROC

A discriminação entre grupos é o principal objetivo da análise ROC. Os índices de *sensibilidade* (fração de verdadeiros positivos) e de *especificidade* (fração de verdadeiros negativos) variam em função da variação do ponto de corte [3,6]. No entanto, o aumento do grau de sensibilidade (verdadeiros positivos) implica uma diminuição no grau de especificidade (verdadeiros negativos) e vice-versa. A próxima representação gráfica explicita a situação.

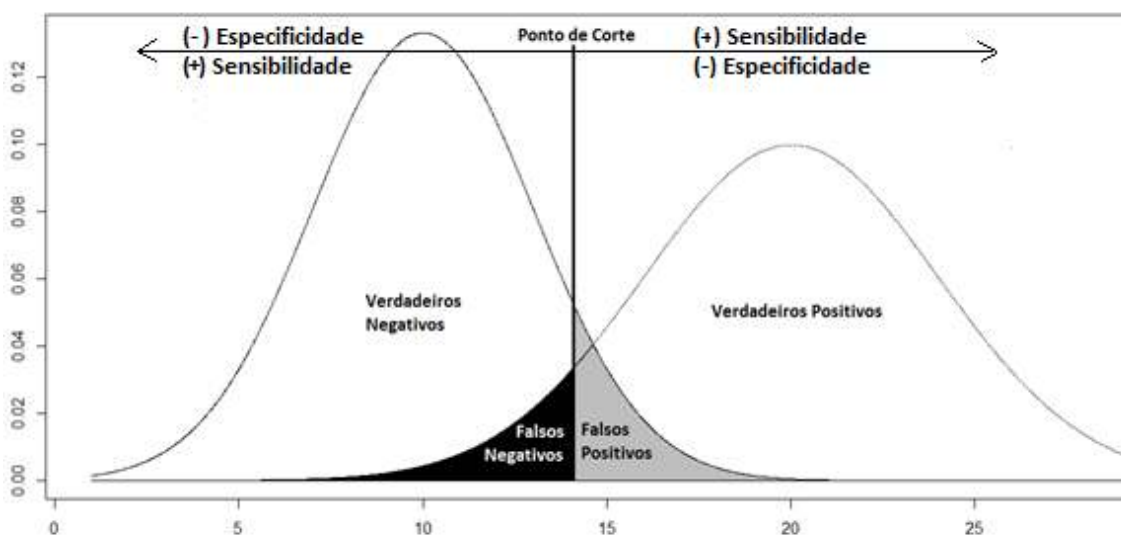


Figura 3. Variação da sensibilidade e especificidade em função dos pontos de corte.

Deste modo torna-se necessária uma medida que explicita a relação entre sensibilidade e especificidade, de modo a avaliar corretamente ambos os índices e escolher o ponto de corte que melhor responde as necessidades de discriminação.

A curva ROC é composta por um gráfico, em que no eixo dos “XX” se encontra a proporção dos falsos positivos (*1-Especificidade*) e no eixo dos “YY” o grau de *Sensibilidade*. Neste sentido, cada ponto da curva representa um possível ponto de corte e a sua respetiva *Sensibilidade* e (*1-Especificidade*) [3,12-13,17].

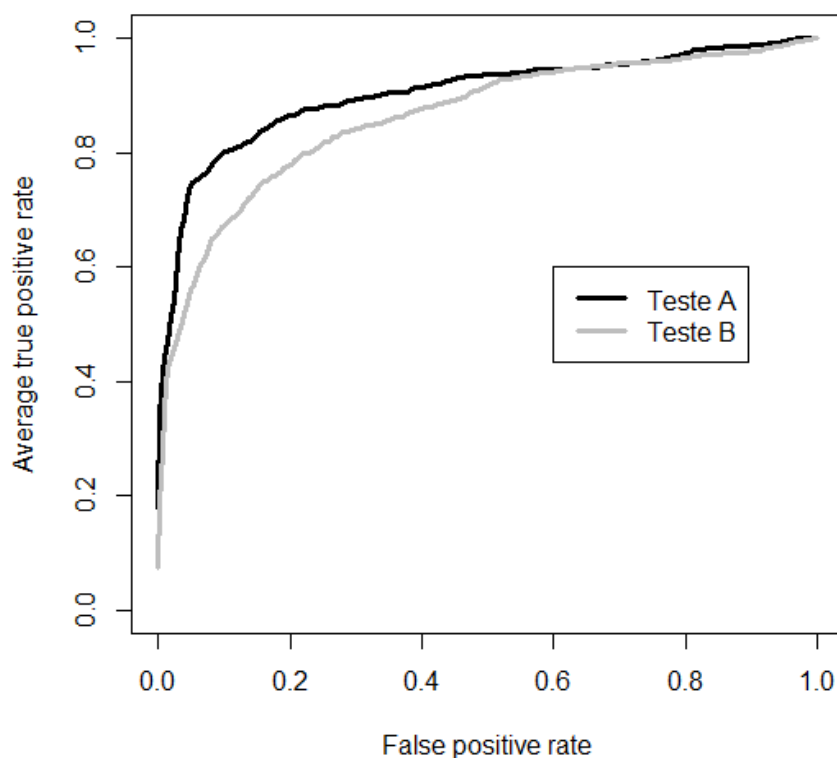


Figura 4. Duas Curvas ROC, calculadas em R, cada uma associada a um teste de avaliação diferente.

A curva ROC tem a grande vantagem de permitir estabelecer uma relação entre sensibilidade e especificidade e de representar todo o universo de acontecimentos possíveis [1,3].

Permite escolher o ponto de corte mais adequado às necessidades de discriminação, uma vez que cada ponto da curva representa um diferente par de sensibilidade e especificidade associado a um ponto de corte, como pode ser analisado na imagem acima [3,10].

Outra vantagem relevante é a possibilidade de comparar diferentes medidas de discriminação dos grupos, através da sobreposição dos diferentes gráficos da curva ROC [3,6-10]. Os pontos em que as linhas de um preditor são superiores ao de outro preditor, são os pares, sensibilidade e especificidade, em que um dos preditores é melhor que o outro [3]. Um índice geral de avaliação das curvas ROC é a área abaixo destas.

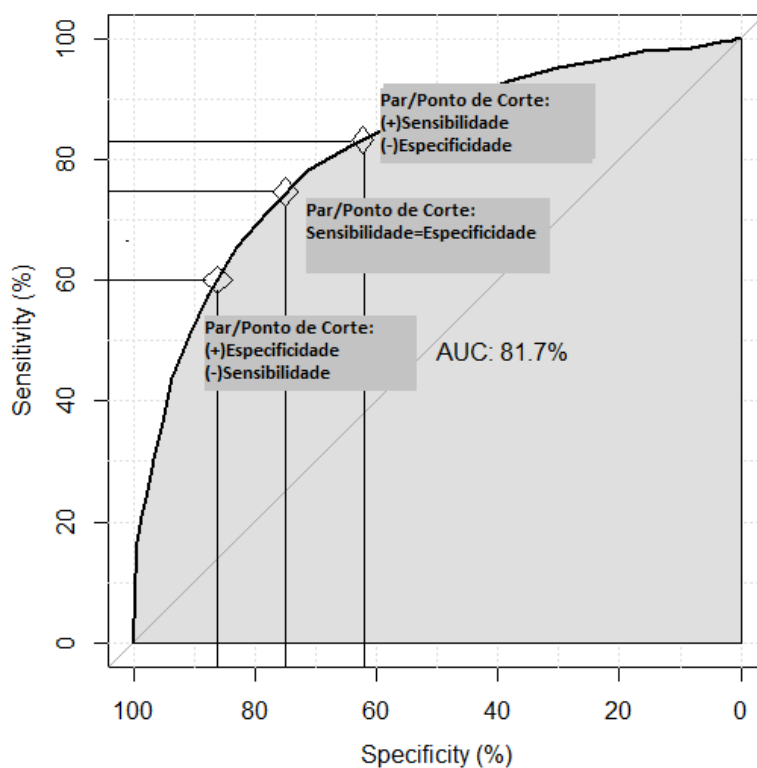


Figura 5. Curva ROC, calculada com recurso ao R. No eixo dos “XX” encontra-se a taxa dos falsos positivos e no eixo dos “YY” o nível de especificidade.

Uma das formas de avaliar a capacidade de discriminação da curva ROC é através da medição da área abaixo da curva. Quanto mais a curva se aproximar do canto superior esquerdo (área de 1) maior será a sua capacidade de discriminação. Se esta se aproximar da diagonal (área de 0.5) a sua capacidade de discriminação é nula, enquanto, que se assumir valores

inferiores, a curva ROC apresenta capacidade de discriminação, mas está a discriminar contrariamente (e.g. identifica melhor como não doente aquele que é doente e vice-versa).

Segundo Braga (2000) a área abaixo da curva pode ser medida através de quatro métodos principais: 1) Estimação de máxima verosimilhança, a 2) aproximação à estatística de Wilcoxon-Mann-Whitney e através 3) através do declive e da interceção do gráfico em plano binormal; e a 4) regra do trapézio [3,14,18].

2.3. Aplicações Informáticas no cálculo e representação gráfica da curva ROC

Atualmente a maioria dos *softwares* de cálculo estatístico permite realizar a análise ROC, bem como produzir uma representação gráfica da mesma. No próximo ponto abordaremos as soluções de análise ROC em três importantes *softwares*: R, SPSS e Excel.

2.3.1. Análise ROC no *software* estatístico R

O *software* estatístico R (www.r-project.org) é aquele que disponibiliza um maior número de opções para o cálculo da curva ROC. Na pesquisa realizada detetou-se um total de 13 pacotes com essa opção, podendo existir mais pacotes não detetados ou virem a surgir novas opções para este *software*.

Este conjunto de pacotes para o R pode ser subdividido em quatro grandes categorias: 1) Pacotes com várias opções gerais; 2) Análise ROC para amostras não paramétricas; 3) Análise ROC – Saúde (Sobrevivência e Genética); 4) Meta-análise da análise ROC.

Os pacotes com várias opções gerais fornecem uma seleção útil de cálculo e representação gráfica da maioria das aplicações de análise ROC. Nesta categoria contam-se seis pacotes: *ROCR*, *rocplus*, *caTools*, *pROC*, *cvAUC* e *HandTill2001*.

A categoria de Análise ROC para amostras não paramétricas, contém um conjunto de dois pacotes com opções especificamente para análise não paramétrica: *ROCwoGS*, *SROC*.

Os pacotes de análise ROC especificamente para a área da Saúde podem-se subdividir em dois tipos: Análise de Sobrevivência e Genética. Na análise de Sobrevivência foram encontrados três *packs*: *survivalROC*, *risksetROC* e *timeROC*. Na área da genética temos o pacote *ROCC*.

Por fim, para a realização de meta-análises da sensibilidade e especificidade existe o pacote *HSROC*.

2.3.2. SPSS

No SPSS também é possível calcular a curva ROC, a sensibilidade, especificidade e respectivos intervalos de confiança.

No entanto as opções são mais limitadas que no *R*, já que apenas se pode escolher a análise não paramétrica e a análise bi-exponencial negativa e não se consegue obter os intervalos de confiança para o gráfico, apenas para a tabela.

2.3.3. EXCEL

Também no programa EXCEL é possível fazer uma parte dos cálculos relativos à análise ROC, que pode ser programada através da introdução de fórmulas ou macros. No entanto comparativamente com o *R*, as opções são também muito limitadas.

2.3.4. Outros *softwares*

Existem outros *softwares* que calculam a curva ROC e os índices associados, com várias funcionalidades e opções, como o *SAS* e o *matlab*.

No *R*, existem ainda outros *packages* que avaliam a sensibilidade e especificidade, ainda que não tenham como função principal a análise ROC.

CAPÍTULO 3: Base de Dados:

Leukocyte Elastase Data

No decorrer dos próximos capítulos vamos apresentar um conjunto de funções programadas no âmbito da tese. Iremos recorrer à base de dados pré-existente no *software* R denominada: “*Leukocyte Elastase Data*”.

3.1. Escolha da Base de dados

A base de dados “*Leukocyte Elastase Data*” foi escolhida por ter características de interesse para este tema, nomeadamente: 1) por estar disponível no R (pacote *OptimalCutpoints*) e por isso pode ser acedida por qualquer pessoa; 2) por as variáveis terem as características necessárias para testar as funções programadas ao longo dos próximos capítulos; e 3) sobretudo porque as variáveis utilizadas resultam de um desenho de investigação respeitante a uma das formas mais importantes de utilização dos métodos matemáticos no âmbito das curvas ROC, a avaliação da presença/ausência de patologia.

Nesta base de dados será avaliada a capacidade do biomarcador/variável “*Leukocyte Elastase Data*”, usado para diagnosticar uma das doenças mais fatais do mundo, a Doença Arterial Cardíaca.

3.2. Informações gerais sobre a base de dados: “Leukocyte Elastase Data”

Os dados contidos na amostra, foram recolhidos no Departamento de Cardiologia do Hospital Geral da Galiza, em Santiago de Compostela, Espanha.

A base de dados contém informação sobre os níveis de *Leukocyte Elastase Data*, recolhido por angiograma, e tem por função avaliar a capacidade deste biomarcador em discriminar entre indivíduos com Doença Arterial Cardíaca (DAC) e indivíduos sem esta doença [19].

Depois de instalado o pacote *OptimalCutpoints*, a base de dados pode ser acedida com recurso aos seguintes comandos:

```
R> library (OptimalCutpoints)
```

```
R> data (elas)
```

3.3. Análise descritiva breve da base de dados

A base de dados é composta por 141 pacientes, avaliados por três variáveis, a variável *gender* que descreve o sexo dos pacientes, a variável *status*, que define se a doença arterial cardíaca está presente ou ausente, e por fim, a variável *elas*, que indica o nível de *Leukocyte Elastase Data* [19]. Segundo o artigo de Amaro (1995), a média de idades dos participantes é de 60,385 anos. Esta informação é resumida na Tabela 1.

Na secção "Média" da tabela é apresentada a média global para a variável "Elas" e a média da variável "Elas" em cada um dos níveis da variável "*status*" e "*gender*".

Tabela 2. Análise descritiva da base de dados Leukocyte Elastase Data

Variável	Tipo de Variável	Níveis	Tamanho Amostra	Percentagem	Desvio Padrão/	Média
<i>Elas</i>	Quantitativa	—	141	—	25.83	43.280
<i>status</i> (Doença)	Qualitativa	Sim	96	68.1 %	—	49.73
		Não	45	31.9 %	—	29.52
<i>Gender</i> (Sexo)	Qualitativa	Homem	105	73.8%	—	44.96
		Mulher	36	26,2%	—	38.38

Na Figura 6 apresentamos a caixa de bigodes, que apresenta a dispersão do biomarcador *elas* por sexo e estatuto da doença (presente/ausente).

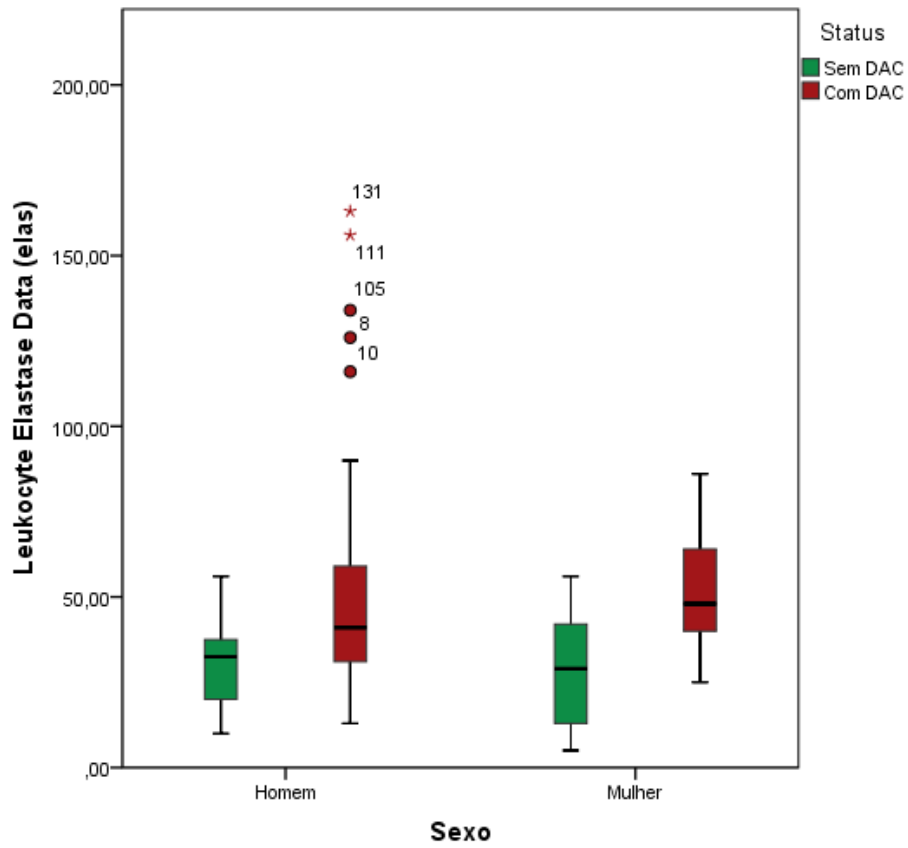


Figura 6. Representação gráfica da caixa de bigodes entre a variável Status e a variável Leukocyte Elastase Data para o sexo masculino e feminino.

CÁPITULO 4. Métodos de Estimação da Sensibilidade, da Especificidade e da Curva ROC

4.1. Curva ROC

Como já foi referido anteriormente, a curva ROC é um gráfico que representa no eixo dos “XX” a proporção de falsos positivos (*1-Especificidade*) e no eixo dos “YY” o grau de *Sensibilidade*. Cada ponto da curva representa a relação entre a sensibilidade e os falsos positivos de cada ponto de corte [3,12-13,17].

4.1.1 Função Programada: *myroc*

Neste Capítulo optamos por criar uma função, denominada, *myroc*, para calcular os pares de sensibilidade/especificidade. Os resultados serão apresentados na próxima tabela e de seguida será representada a curva ROC, num gráfico onde estão presentes os pares de sensibilidade/1-especificidade.

1.função:

myroc(y, x)

2.argumentos:

1.x="variável quantitativa que representa o marcador"

2.y="variável binária que nos indica o estatuto"

3.Esta função retorna uma lista com os seguintes itens:

1. Lista de valores da sensibilidade

2. Lista de valores da especificidade

3. Devolve o valor exato da área abaixo da curva ROC empírica (AUC)

Na Tabela 2 apresentamos a versão reduzida dos pares estimados para a sensibilidade/especificidade recorrendo à função *myroc*. Usamos o termo par, porque como se sabe, para cada ponto da sensibilidade existe um par, que é o valor associado de especificidade (e vice-versa), e a cada um destes pares corresponde um ponto de corte, onde se estima obter a especificidade e sensibilidade referida.

Foi também calculado o valor exato da área abaixo da curva ROC empírica a partir da soma da área do conjunto de retângulos que a constituem. Esta técnica tem a vantagem de retornar o valor exato da área do gráfico uma vez que não pressupõem um declive que poderá ser superior ao declive real entre os valores que distanciam um ponto de corte do ponto de corte seguinte.

À medida que o número de pontos de corte aumenta, a largura dos retângulos diminuiu, aproximando-se o gráfico de uma curva, tal como acontece no cálculo da área da curva ROC pela regra do trapézio. O valor exato é de 0.737963 e pode ser consultado na Figura 7.

Tabela 3. Lista reduzida dos pares sensibilidade/especificidade da curva ROC.

Pares	Sensibilidade	Especificidade
1	1.000	0.000
2	1.000	0.022
3	1.000	0.044
4	1.000	0.111
30	0.510	0.733
31	0.510	0.755
32	0.489	0.822
33	0.479	0.844
60	0.031	1.000
61	0.020	1.000
62	0.010	1.000
63	0.000	1.000

Na Figura 7 encontra-se a representação gráfica da curva ROC para os dados da base *Leukocyte Elastase Data*, apresentados na Tabela 2. No eixo dos “XX” são apresentados a taxa de falsos positivos (1-Especificidade) e no eixo dos YY o valor da sensibilidade.

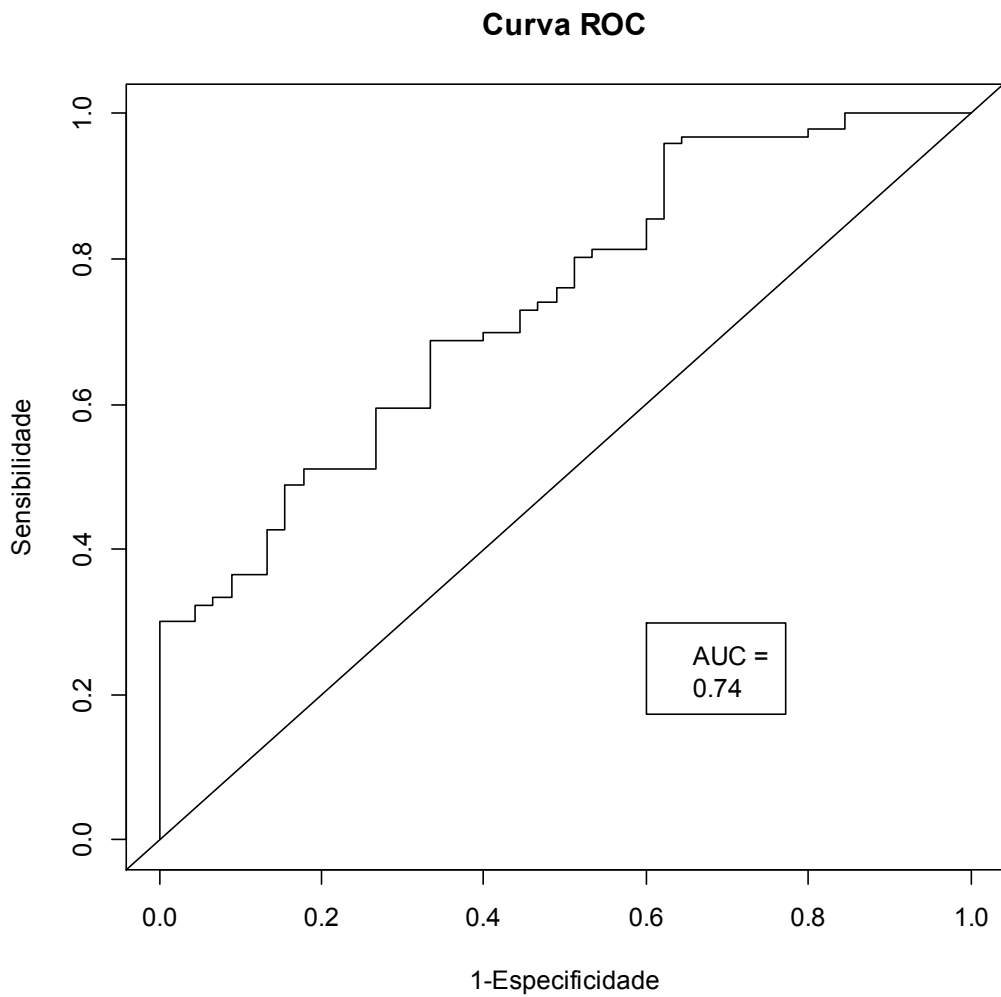


Figura 7. Curva ROC, no eixo dos “XX” os falsos positivos (1-Especificidade) e no eixo dos “YY” o valor da sensibilidade.

Na Figura 7, está representada a área acima da diagonal, que informa sobre a capacidade de discriminação da curva, quanto maior for esta área, maior será a capacidade de discriminação. Como seria de esperar à medida que aumenta a sensibilidade, aumenta o número de falsos positivos (1-especificidade) e vice-versa. Normalmente é escolhido o par sensibilidade/falsos positivos, intermédio, de modo a permitir uma boa capacidade de discriminação dos grupos em avaliação.

4.2. Índice de Youden e Precisão

Tanto a precisão como o Índice de Youden, são dois índices muito simples que têm por objetivos a avaliação da qualidade da curva ROC.

Cálculo da Precisão:

$$\text{Precisão} = \frac{\text{Número de Casos Corretamente Classificados}}{\text{Número total de casos}} \quad (28)$$

Cálculo do Índice de Youden:

$$\text{Índice de Youden} = (Se + Sp) - 1 \quad (29)$$

Em que Se é a sensibilidade e Sp a especificidade. Enquanto a precisão retorna valores entre 0 e 1 (quanto mais próximo de 1 melhor será a precisão), o Índice de Youden retorna valores entre -1 e 1, quanto mais próximo de 1 ou de -1 estiver o valor do Índice de Youden maior será a capacidade de discriminação das variáveis.

Foi criada uma função simples *ypindex* para calcular estes dois índices e permitir compará-los com cada conjunto de sensibilidade, especificidade e ponto de corte.

4.2.1 Função Programada: *ypindex*

1.função:

ypindex(y, x)

2.argumentos:

1.x="variável quantitativa que representa o marcador"

2.y="variável binária que nos indica o estatuto"

3. Esta função retorna uma lista com os seguintes itens:

1. Lista de valores da Sensibilidade
2. Lista de valores de Especificidade
3. Lista de valores do Índice de Youden
4. Lista de valores de Precisão
5. Lista de valores de Ponto de corte correspondentes

Tabela 4. Esta tabela apresenta uma lista reduzida dos resultados da função *ypindex*.

Pares	Ponto de Corte	Se	Sp	Precisão	IY
1	5	1.000	0.000	0.681	0.000
2	6	1.000	0.022	0.688	0.022
3	10	1.000	0.044	0.695	0.045
4	12	1.000	0.111	0.716	0.111
5	13	0.979	0.156	0.716	0.134
6	14	0.969	0.200	0.723	0.169
58	90	0.052	1.000	0.354	0.052
59	116	0.042	1.000	0.347	0.042
60	126	0.0313	1.000	0.340	0.031
61	134	0.0208	1.000	0.333	0.021
62	156	0.0104	1.000	0.326	0.010
63	163	0.0000	1.000	0.319	0.000

Legenda: Se (sensibilidade), Sp (especificidade), IY (Índice de Youden)

No entanto é importante salientar algumas limitações destes índices, nomeadamente o facto de não ilustrarem diferenças numéricas entre especificidade e sensibilidade, por exemplo, a pontuação destes índices será idêntica, tanto esteja presente um nível de especificidade e sensibilidade de 0.8 ou um nível de sensibilidade de 0.65 e um nível de especificidade de 0.95, o que tem implicações práticas na informação fornecida pelos índices e que poderá condicionar

a avaliação. A representação gráfica da curva ROC é uma alternativa na identificação destas limitações [3,7].

O próximo gráfico apresenta a relação a Sensibilidade, os falsos positivos (1-Especificidade) e dois Índices, a Precisão e o Índice de Youden.

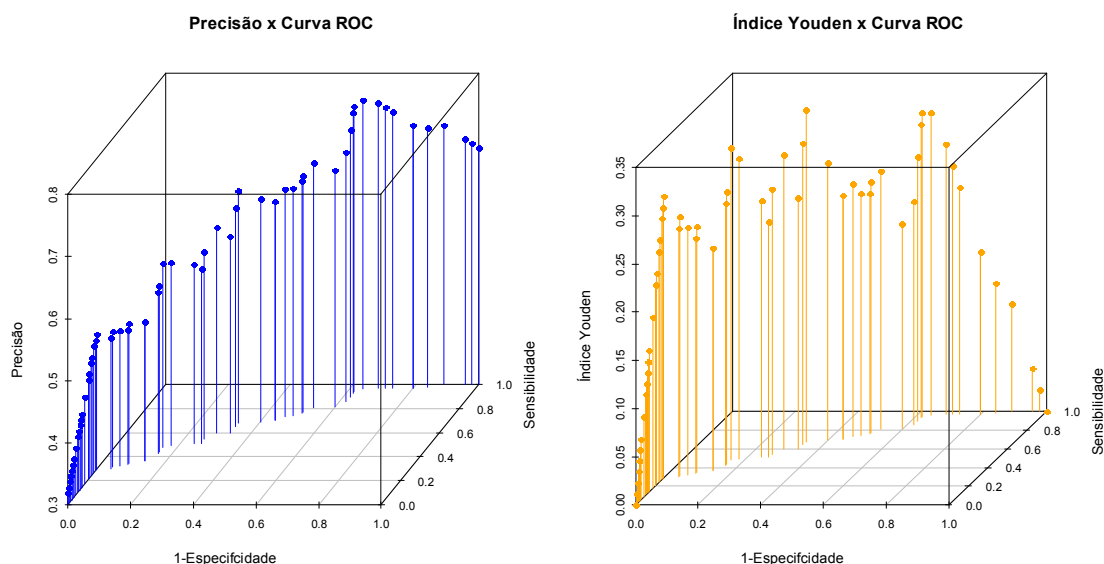


Figura 8. Os gráficos 3D apresentam a relação entre 1-Especificidade, a Sensibilidade e a Precisão (Azul) / Índice de Youden (Amarelo).

Nesta representação gráfica é fácil perceber que os dois índices divergem em relação ao seu valor mais elevado, enquanto a precisão atinge o seu valor mais alto quando a sensibilidade (0.969) é mais elevada que a especificidade (0.311), o Índice de Youden atinge o valor mais alto quando a Especificidade (0.844) é mais elevada que a sensibilidade (0.479).

Este facto deve-se ao modo do cálculo dos Índices, porque enquanto a precisão considera o número de verdadeiros positivos e de verdadeiros negativos em conjunto, o Índice de Youden atribui a mesma ponderação à Sensibilidade e à Especificidade, independentemente do número de casos negativos (e.g. sem doença) em relação ao número de casos positivos.

A escolha do Índice de *Youden* é por isso mais útil quando pretendemos um ponto de corte que tenha em consideração a proporção de verdadeiros positivos em relação com a proporção de verdadeiros negativos ou vice-versa, enquanto a precisão é útil quando apenas interessa um ponto de discriminação que aumente a probabilidade de classificação correta.

4.3. Estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade

A distribuição associada à especificidade e a sensibilidade é bivariada, uma vez que se analisa a probabilidade de determinado evento fazer parte de um de dois grupos, neste sentido, ao longo desta secção vamos apresentar os diferentes métodos de cálculo dos intervalos de confiança para sensibilidade e especificidade, que foram programados em funções e que serão executados ao longo deste capítulo.

4.3.1. Método *Wald* para a estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade

Se estivermos perante uma amostra com tamanho igual ou superior a 50, é possível fazer uma aproximação à distribuição normal e pode-se usar o método *Wald* no cálculo dos Intervalos de Confiança para a proporção [12-14].

Os Intervalos de Confiança para a proporção p recorrendo à aproximação à normal para a distribuição binomial é [15]:

$$\left[\hat{p} - z_{1-\alpha/2} \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}; \hat{p} + z_{1-\alpha/2} \sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \right] \quad (30)$$

Onde \hat{p} denota a estimativa de uma proporção, n o tamanho da amostral e $z_{1-\alpha/2}$ o quantil $1-\alpha/2$ de uma distribuição normal reduzida.

A aplicação aos índices de sensibilidade e especificidade da curva ROC é dada por [12]:

Estimação dos Intervalos de Confiança para a Sensibilidade:

$$\left[\hat{S}_e - z_{1-\alpha/2} \sqrt{\widehat{\text{Var}}(\hat{S}_e)}; \hat{S}_e + z_{1-\alpha/2} \sqrt{\widehat{\text{Var}}(\hat{S}_e)} \right] \quad (31)$$

Onde \widehat{Se} representa a estimativa da Sensibilidade e a $\widehat{Var}(\widehat{Se})$, a estimativa da variância da sensibilidade estimada, sendo $z_{1-\alpha/2}$ o quantil $1-\alpha/2$ de uma distribuição normal reduzida.

Estimação dos Intervalos de Confiança para a Especificidade:

$$] \widehat{Sp} - z_{1-\alpha/2} \sqrt{\widehat{Var}(\widehat{Sp})}; \widehat{Sp} + z_{1-\alpha/2} \sqrt{\widehat{Var}(\widehat{Sp})} [\quad (32)$$

Onde \widehat{Sp} representa a estimativa da Sensibilidade e $\widehat{Var}(\widehat{Sp})$, a estimativa da variância da especificidade estimada, sendo $z_{1-\alpha/2}$ o quantil $1-\alpha/2$ de uma distribuição normal reduzida.

4.3.2. Método *Score* para a estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade

O intervalo de confiança de *Score* para proporção demonstrou ter boas características de estimação em amostras mais pequenas [15].

$$] \frac{\widehat{p} + \frac{z_{1-\alpha/2}^2}{2n} - z_{1-\alpha/2} \sqrt{\frac{\widehat{p}(1-\widehat{p}) + \frac{z_{1-\alpha/2}^2}{4n}}{n}}}{1 + \frac{z_{1-\alpha/2}^2}{n}} [\quad (33)$$

Em que \widehat{p} é a estimativa da proporção e n o tamanho amostral, $z_{\alpha/2}$ e $z_{1-\alpha/2}$ os quantis de uma distribuição normal reduzida. No próximo subtópico apresenta-se a aplicação desta fórmula à estimação da sensibilidade e especificidade [12]:

Estimação dos Intervalos de Confiança para a Sensibilidade:

$$\left[\frac{\widehat{Se} + \frac{z_{1-\alpha/2}^2}{2np} \pm z_{1-\alpha/2} \sqrt{\frac{\widehat{Se}(1-\widehat{Se}) + \frac{z_{1-\alpha/2}^2}{4np}}{np}}}{1 + \frac{z_{1-\alpha/2}^2}{np}} \right] \quad (34)$$

Onde np é o tamanho da amostra dos casos positivos, $z_{1-\alpha/2}$ o quantil $1-\alpha/2$ da distribuição normal reduzida.

Estimação dos Intervalos de Confiança para a Especificidade:

$$\left[\frac{\widehat{Sp} + \frac{z_{1-\alpha/2}^2}{2nn} \pm z_{1-\alpha/2} \sqrt{\frac{\widehat{Sp}(1-\widehat{Sp}) + \frac{z_{1-\alpha/2}^2}{4nn}}{nn}}}{1 + \frac{z_{1-\alpha/2}^2}{nn}} \right] \quad (35)$$

Em que \widehat{Sp} denota a estimativa da especificidade, nn o tamanho da amostra dos casos negativos, e $z_{1-\alpha/2}$ o quantil $1-\alpha/2$ da distribuição normal reduzida.

4.3.3. Método *Bootstrap Simples* para a estimação dos Intervalos de Confiança para a Sensibilidade e Especificidade

O método *bootstrap* é uma técnica estatística que permite a avaliação da variabilidade das estatísticas a partir dos dados de uma amostra existente [19-20].

Este método utiliza a amostra original e a partir dessa amostra gera n amostras do mesmo tamanho, que usualmente diferem entre si uma vez que resultam da seleção aleatória com reposição dos valores originais. De seguida é calculada a estimativa para cada amostra gerada. Este método é generalizável a um vasto leque de estimadores [19].

O método *bootstrap* tem aplicação paramétrica ou não paramétrica. É um método simples e muito útil, sobretudo em amostras pequenas e também quando as demais soluções matemáticas são complexas.

Neste ponto é abordado o método percentílico *bootstrap* para cálculo de intervalos de confiança. Este método não paramétrico é simples e muito útil para gerar intervalos e bandas de confiança para as curvas ROC [19-20].

$$] \hat{P}_{(\alpha/2)*100}^*; \hat{P}_{(1-\alpha/2)*100}^* [\quad (36)$$

Onde $\hat{P}_{(\alpha/2)*100}^*$ e $\hat{P}_{(1-\alpha/2)*100}^*$ são os percentis que se obtêm a partir da totalidade das amostras geradas.

O método percentílico *bootstrap* é um método que calcula os intervalos de confiança a partir dos valores dos percentis dos dados *bootstrap*. No caso de termos um intervalo de confiança simétrico de 95%, calcular-se-ia o percentil 2.5% e o percentil 97.5% dos dados obtidos pelas n amostras.

Estimação dos Intervalos de Confiança para a Sensibilidade:

$$] \widehat{Se}_{(\alpha/2)*100}^*; \widehat{Se}_{(1-\alpha/2)*100}^* [\quad (37)$$

Em que $\widehat{Se}_{(\alpha/2)*100}^*$ e $\widehat{Se}_{(1-\alpha/2)*100}^*$ são os percentis que se obtêm a partir da totalidade das amostras geradas.

Estimação dos Intervalos de Confiança para a Especificidade:

$$] \widehat{Sp}_{(\alpha/2)*100}^*; \widehat{Sp}_{(1-\alpha/2)*100}^* [\quad (38)$$

Onde $\widehat{Sp}_{(\alpha/2)*100}^*$ e $\widehat{Sp}_{(1-\alpha/2)*100}^*$ são os percentis que se obtêm para a especificidade a partir da totalidade das amostras geradas.

4.3.4. Função Programada: *scut*

Neste tópico foi desenvolvida a função *scut*. Esta função devolve os valores da especificidade, sensibilidade e dos limites de confiança, disponíveis nos três métodos acima descritos: *Wald*, *Score* e *Bootstrap*, para um determinado ponto de corte que se indique na função.

1. função:

scut (*y,x,cut,conf,conf.level, method,nboot*)

2.argumentos:

1.*x*="variável quantitativa que representa o marcador"

2.*y*="variável binária que nos indica o estatuto"

3.*cut*="ponto de corte em relação ao qual se pretende estimar a sensibilidade e especificidade"

4.*conf*="Variável lógica para indicar se pretendemos intervalos de confiança"

Opções: TRUE/FALSE (Verdadeiro/Falso), pré-definido em TRUE (Verdadeiro)

5.*conf.level*="nível de confiança pretendido", pré-definido em 95%

6.*method*="método para estimar os intervalos de confiança"

Opções=1.*Wald*/2.*Score*/3.*bootstrap* (caso *bootstrap*, usar *nboot* para seleccionar o número de amostras a gerar)

3.Esta função retorna os seguintes itens:

1. Sensibilidade para o ponto de corte escolhido

2. Especificidade para o ponto de corte escolhido

3. Limite de confiança inferior da sensibilidade e especificidade para o ponto de corte escolhido

4. Limite de confiança superior da sensibilidade e especificidade para o ponto de corte escolhido

5. Retorna o valor do ponto de corte seleccionado

Vamos experimentar a função *scut* para estimar os limites de confiança para 3 pontos de corte diferentes, 30, 60, 90, usando 3 diferentes métodos: *Wald*, *Score* e *Bootstrap* (500 amostras geradas), recorrendo à base que estamos a utilizar: "*Leukocyte Elastase Data*".

Tabela 5. Ponto de corte selecionado, respetiva Especificidade, Sensibilidade e limites de confiança (superior e inferior) para cada um dos métodos.

Método	PC	Se	LI Se	LS Se	Sp	LI Sp	LS Sp
Wald	30	0.771	0.687	0.855	0.488	0.343	0.635
Score	30	0.771	0.677	0.844	0.488	0.350	0.630
Bootstrap	30	0.771	0.678	0.856	0.488	0.340	0.631
Wald	60	0.250	0.163	0.337	1.000	1.000	1.000
Score	60	0.250	0.174	0.345	1.000	0.921	1.000
Bootstrap	60	0.250	0.166	0.333	1.000	1.000	1.000
Wald	90	0.052	0.008	0.097	1.000	1.000	1.000
Score	90	0.052	0.022	0.116	1.000	0.921	1.000
Bootstrap	90	0.052	0.011	0.099	1.000	1.000	1.000

Legenda: PC (Ponto de corte), Se (Sensibilidade), Sp (especificidade), LI (limite inferior) e LS (limite superior).

Esta função pode ser muito útil no caso de pretendermos comparar os valores de sensibilidade e especificidade, bem como os respetivos limites de confiança, de um determinado ponto de corte, o que permite determinar e comparar rapidamente diferentes pontos de corte, como foi apresentado na tabela acima apresentada.

4.4. Bandas de Confiança

Tratando-se cada ponto da curva ROC de um par Se-Sp associado a um ponto de corte da variável contínua, o conjunto de pontos da curva ROC dá origem a dois conjuntos de pontos de intervalos de confiança, originando duas linhas denominadas bandas de confiança [21].

O gráfico seguinte apresenta as bandas de confiança conjuntas, tanto para a especificidade como para a sensibilidade com recurso das médias limiaries otimizadas, foi calculada com recurso ao pacote estatístico *pROC* existente no *R*.

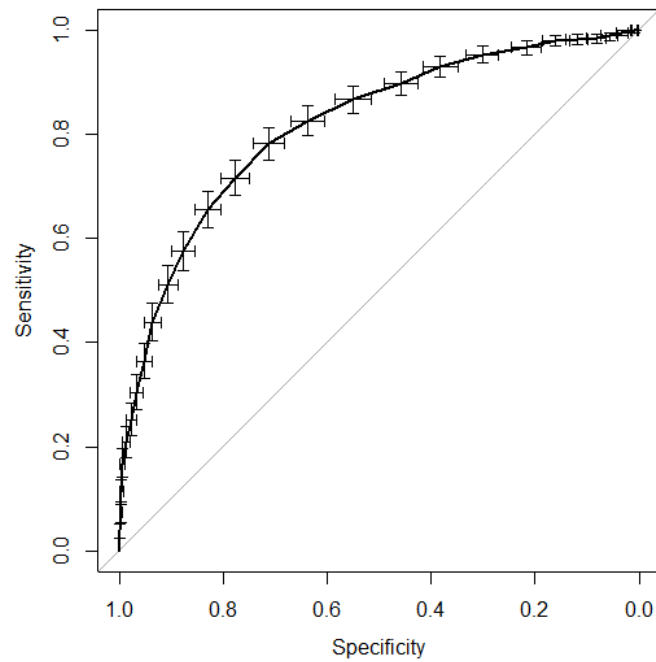


Figura 9. Curva ROC e bandas de confiança para a sensibilidade e especificidade/falsos positivos a 95%.

Na área interna das bandas de confiança, encontra-se a região de incerteza, com a qual o investigador terá que lidar, no entanto existe elevada confiança do valor não se situar fora da área desse intervalo.

4.4.1. Função programada: *plot.secuc*, *plot.spcuc*

Foram programadas as funções *plot.secuc* e *plot.spcuc*. As funções devolvem tabelas com informação sobre o nível de sensibilidade ou especificidade, e os respetivos limites de confiança, correspondentes a cada ponto de corte, devolve também um gráfico que estabelece a relação entre cada ponto de corte e as bandas de confiança da sensibilidade ou especificidade.

4.4.1.1 Função programada: *plot.secut*

1.função

plot.secut(x,y, conf, conf.level, method, nboot)

2.argumentos:

1.x="variável quantitativa que representa o marcador"

2.y="variável binária que nos indica o estatuto"

3.conf="Variável lógica para indicar se pretendemos intervalos de confiança"

Opções: TRUE/FALSE (Verdadeiro/Falso), pré-definido em TRUE (Verdadeiro)

4.conf.level="nível de confiança pretendido", pré-definido em 95%

5.method="método para estimar os intervalos de confiança"

Opções=1.Wald/2.Score/3.Bootstrap (caso bootstrap, usar nboot para seleccionar o número de amostras a gerar)

3.Esta função retorna uma lista com os seguintes itens:

1. Lista de valores da Sensibilidade
2. Lista de valores dos limites superiores da sensibilidade
3. Lista dos valores dos Limites inferiores da sensibilidade
4. Lista de valores de Ponto de corte correspondentes

Tabela 6. Sensibilidade, limites de confiança para a sensibilidade em cada ponto de corte

Pares	Ponto de Corte	Método <i>Wald</i>			Método <i>Score</i>			Método <i>Bootstrap</i> (<i>nboot</i> =500)			Amplit.
		Se	LI	LS	Amplit.	LI	LS	Amplit.	LI	LS	
1	5.0	1.000	1.000	1.000	0.000	0.962	1.000	0.038	1.000	1.000	0.000
24	36.0	0.688	0.595	0.780	0.185	0.589	0.771	0.182	0.586	0.782	0.176
25	37.0	0.677	0.583	0.771	0.188	0.578	0.762	0.184	0.592	0.770	0.178
26	38.0	0.646	0.550	0.742	0.192	0.546	0.734	0.188	0.541	0.733	0.192
63	163.0	0.000	0.000	0.000	0.000	0.000	0.038	0.038	0.000	0.000	0.000

Legenda: Se (sensibilidade), LI (limite inferior), LS (limite superior) e Amplit. (amplitude).

Os resultados obtidos, apresentados numa versão reduzida, para os limites de confiança a 95%, apresentam valores muito próximos entre os diferentes métodos de cálculo de bandas de confiança para a sensibilidade. Nenhum dos métodos apresenta consistentemente uma amplitude menor/maior ao longo dos intervalos de confiança, estes são variáveis ao longo dos diferentes pontos de corte. Foram selecionados um conjunto disperso de pontos de corte do espectro da variável contínua, a tabela completa encontra-se em anexo.

4.4.1.1.1 Sensibilidade: Banda de Confiança para os pontos de corte da variável contínua

Os próximos gráficos apresentam, para os 3 métodos, no eixo dos “YY”, as bandas com 95% de confiança, para a sensibilidade e no eixo dos “XX” o ponto de corte para cada um dos 3 métodos: *wald*, *score* e *bootstrap*. Os gráficos dos pontos de corte para a sensibilidade com as bandas de confiança, resultam na função *plot.secut*.

4.4.1.1.2 Representação Gráfica das Bandas de Confiança para a Sensibilidade

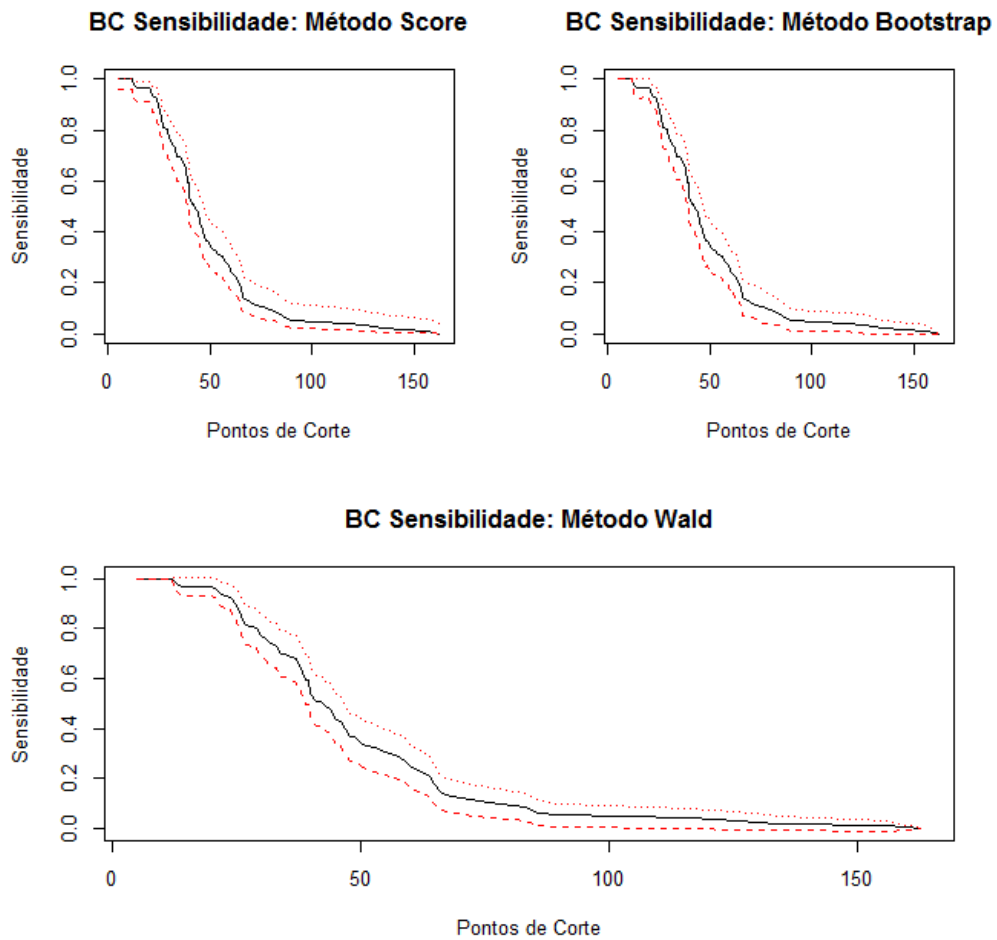


Figura 10. Representação gráfica das bandas de confiança para a sensibilidade através dos 3 métodos (score, bootstrap (nboot=500) e Wald).

4.4.1.2. Função programada: *plot.spcut*

1.função

plot.spcut(x,y, conf, conf.level, method, nboot)

2.argumentos:

1.x="variável quantitativa que representa o marcador"

2.y="variável binária que nos indica o estatuto"

3.conf="Variável lógica para indicar se pretendemos intervalos de confiança"

Opções: TRUE/FALSE (Verdadeiro/Falso), pré-definido em TRUE (Verdadeiro)

4.conf.level="nível de confiança pretendido", pré-definido em 95%

5.method="método para estimar os intervalos de confiança"

Opções=1.Wald/2.Score/3.bootstrap (caso bootstrap, usar nboot para seleccionar o número de amostras a gerar)

3.Esta função retorna uma lista com os seguintes itens:

1. Lista de valores da especificidade
2. Lista de valores dos limites superiores da especificidade
3. Lista dos valores dos limites inferiores da especificidade
4. Lista de valores de ponto de corte correspondentes

A próxima tabela apresenta uma versão reduzida para os valores dos limites de confiança da especificidade para cada ponto de corte, calculados através de três métodos: *Wald*, *Score* e *Bootstrap*. Os resultados desta tabela foram calculados através da função programada, *plot.spcut*. Acrescentou-se também a amplitude dos limites de confiança.

Tabela 7. Lista reduzida com valores da especificidade e limites de confiança, superior e inferior para a especificidade nos três métodos e pontos de corte associados. Calculou-se a amplitude.

Pares	Ponte de Corte	Método <i>Wald</i>				Método <i>Score</i>				Método <i>Bootstrap</i> (<i>nboot</i> =500)		Ampli t.
		Sp	LI	LS	Amplit.	LI	LS	Amplit.	LI	LS		
1	5	0.000	0.000	0.000	0.000	0.000	0.079	0.079	0.000	0.000	0.000	
32	43	0.822	0.711	0.934	0.223	0.687	0.907	0.220	0.705	0.927	0.226	
33	44	0.844	0.739	0.950	0.212	0.712	0.923	0.210	0.737	0.938	0.218	
34	45	0.844	0.739	0.950	0.212	0.712	0.923	0.210	0.725	0.930	0.218	
63	163	1.000	1.000	1.000	0.000	0.921	1.000	0.079	1.000	1.000	0.000	

Legenda: Sp (especificidade), LI (limite inferior), LS (limite superior) e Amplit. (amplitude).

Tal como acontece com a especificidade, os resultados obtidos para os limites de confiança a 95%, apresentam valores muito próximos entre os diferentes métodos de cálculo de bandas de confiança para a especificidade e como na sensibilidade, nenhum dos métodos apresenta uma amplitude consistentemente maior ou menor ao longo dos intervalos de confiança.

4.4.1.2.1. Especificidade: Bandas de Confiança para os pontos de corte da variável contínua

No próximo gráfico, sobre as bandas de confiança para a especificidade, visualiza-se no eixo dos “XX” o ponto de corte e no eixo dos “YY” as bandas com 95% de confiança, para a especificidade para cada um dos 3 métodos: *wald*, *score* e *bootstrap*. Os gráficos das bandas de confiança para a especificidade resultam da função *plot.secut*.

4.4.1.2.2. Representação Gráfica das Banda de Confiança para a Especificidade

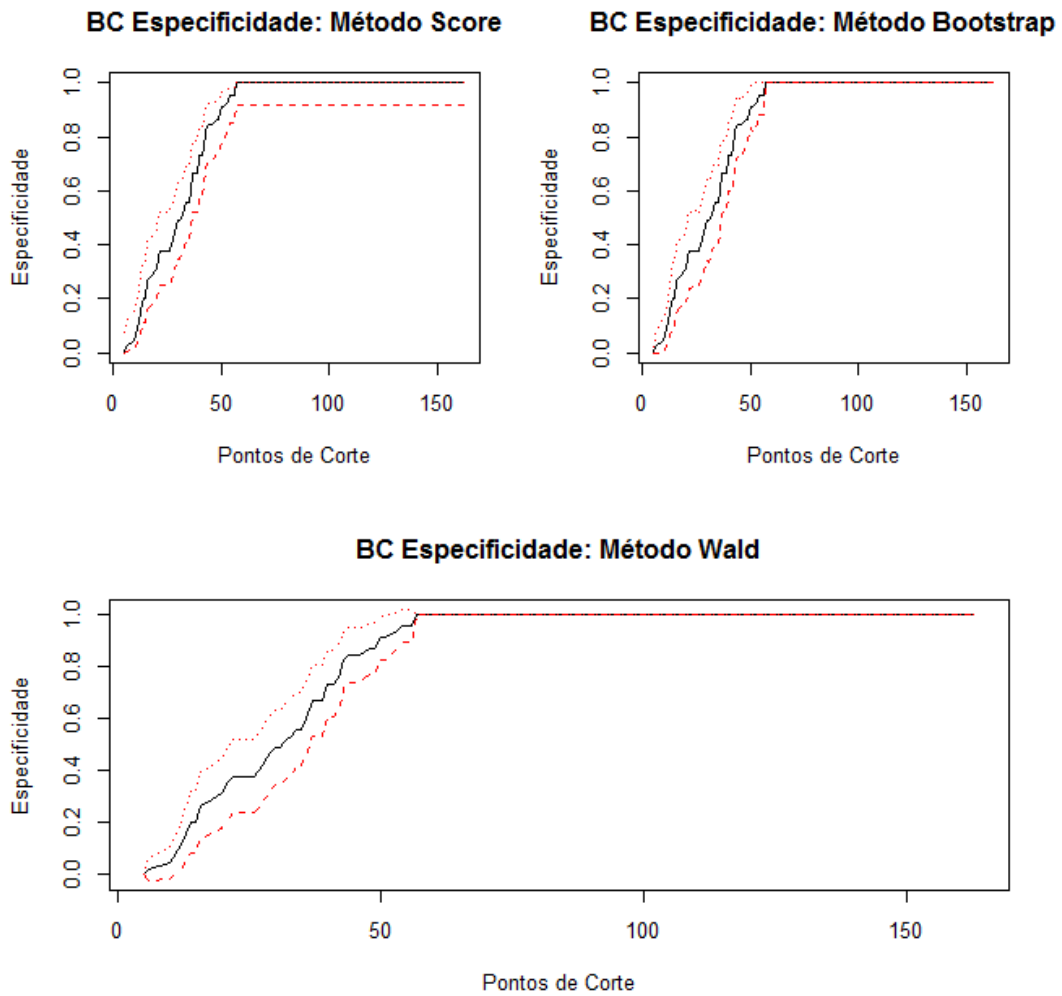


Figura 11. Representação gráfica das bandas de confiança para a especificidade através dos 3 métodos (Score, Bootstrap (nboot=500) e Wald).

4.4.2. Função Programada: $SpSe$

A função $SpSe$ devolve os valores da especificidade e sensibilidade bem como os limites de confiança superior e inferior para cada ponto de corte da nossa variável contínua. Calcula ainda a representação gráfica da curva ROC, com limites de confiança para a sensibilidade e para os falsos positivos (1-especificidade), através de três métodos: *Wald*, *Score* e *Bootstrap*.

1. função:

SpSe(y="variável qualitativa", x="variável continua", conf=TRUE, level.conf="escolher nível de confiança", method="wald/score/bootstrap", nboot=(numero de amostras/ apenas quando method=bootstrap))

2.argumentos:

1.x="variável quantitativa que representa o marcador"

2.y="variável binária que nos indica o estatuto"

3.conf="Variável lógica para indicar se pretendemos intervalos de confiança"

Opções: TRUE/FALSE (Verdadeiro/Falso), pré-definido em TRUE (Verdadeiro)

4.conf.level="nível de confiança pretendido", pré-definido em 95%

5.method="método para estimar os intervalos de confiança"

Opções=1.Wald/2.Score/3.bootstrap (caso bootstrap, usar nboot para seleccionar o número de amostras a gerar)

3.Esta função retorna uma lista com os seguintes itens:

- 1. Lista de valores da especificidade*
- 2. Lista de valores dos limites superiores da especificidade*
- 3. Lista dos valores dos limites inferiores da especificidade*
- 4. Lista de valores da sensibilidade*
- 5. Lista de valores dos limites superiores da sensibilidade*
- 6. Lista dos valores dos limites inferiores da sensibilidade*
- 7. Lista de valores de ponto de corte correspondentes*

Com o comando *matplot* retorna também a curva ROC com as bandas de confiança para a sensibilidade e 1-especificidade.

Tabela 8. Lista reduzida com valores da sensibilidade e especificidade e respectivos limites de confiança para os 3 métodos.

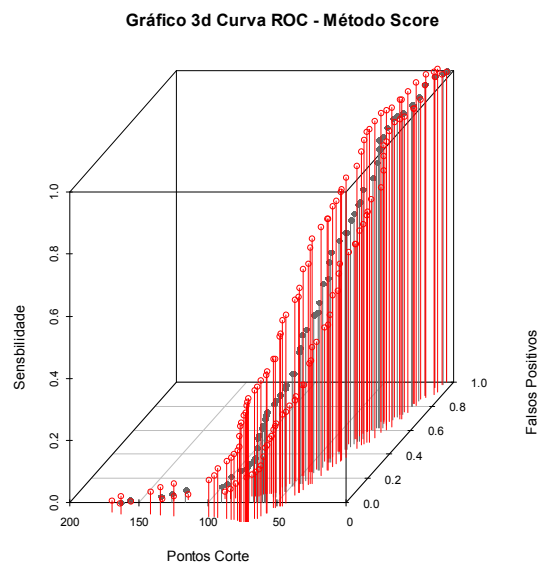
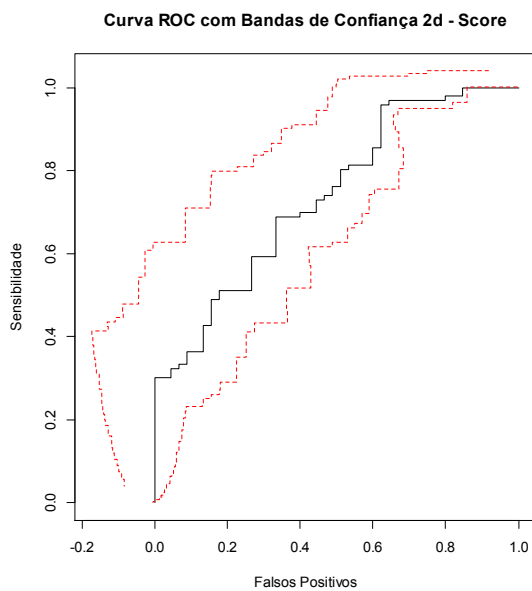
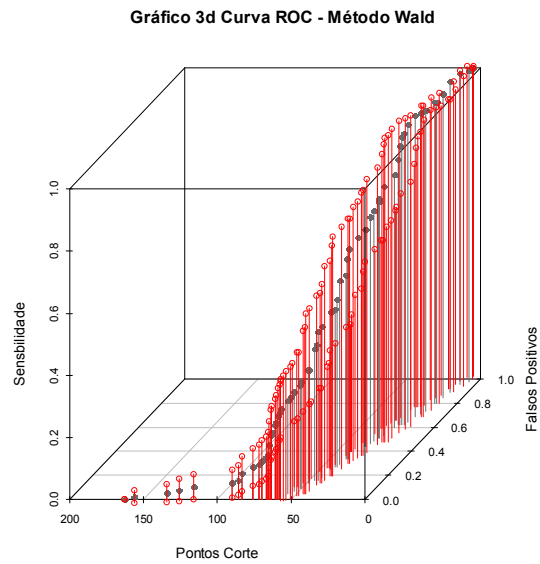
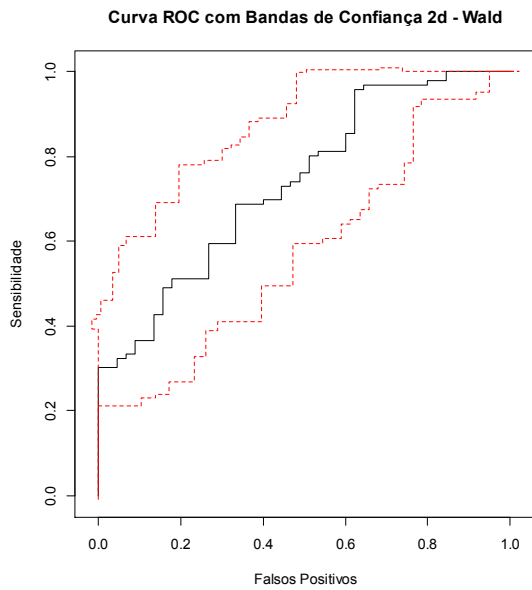
Ponto de						
Corte	Sensibilidade	Especificidade	LI Se	LS Se	LI Sp	LS Sp
<i>Método Wald</i>						
41	0.510	0.733	0.410	0.610	0.604	0.863
42	0.510	0.756	0.410	0.610	0.630	0.881
43	0.490	0.822	0.390	0.590	0.711	0.934
44	0.479	0.844	0.379	0.579	0.739	0.950
45	0.438	0.844	0.338	0.537	0.739	0.950
<i>Método Score</i>						
41	0.510	0.733	0.432	0.628	0.636	0.916
42	0.510	0.756	0.432	0.628	0.658	0.938
43	0.490	0.822	0.412	0.608	0.725	1.000
44	0.479	0.844	0.401	0.597	0.747	1.000
45	0.438	0.844	0.360	0.555	0.748	1.000
<i>Método Bootstrap (nboot=500)</i>						
41	0.510	0.733	0.409	0.611	0.598	0.857
42	0.510	0.756	0.407	0.602	0.618	0.865
43	0.490	0.822	0.383	0.596	0.705	0.927
44	0.479	0.844	0.393	0.563	0.737	0.938
45	0.438	0.844	0.347	0.538	0.725	0.930

Legenda: Se (Sensibilidade), Sp (especificidade), LI (limite inferior), LS (limite superior) e Amplit. (amplitude).

Esta função permite-nos estimar todos intervalos de confiança para a sensibilidade e especificidade e os respetivos pontos de corte para três métodos diferentes. Permite-nos ainda obter a as bandas de confiança para a curva ROC como será apresentado de seguida.

4.4.2.1. Representação gráfica da Curva ROC com Bandas de Confiança

Neste ponto é representado o gráfico com as Bandas de Confiança da curva ROC, numa versão *2D* e *3D* (acrescentando os pontos de corte), calculadas a partir de um *matplotlib* que utilizou a informação obtida através do comando *SpSe*. As bandas de confiança da curva ROC são de seguida apresentadas nos três métodos descritos:



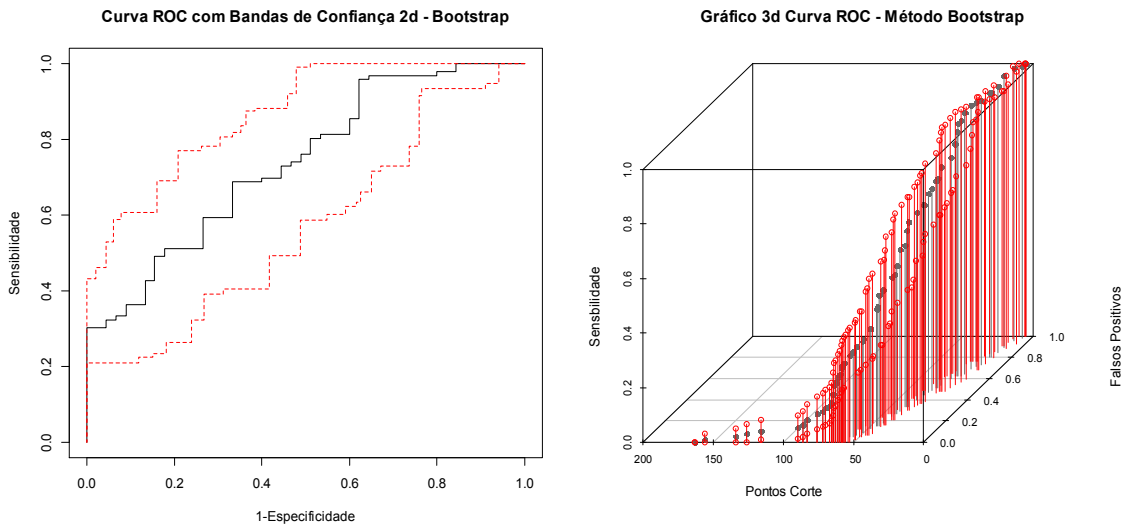


Figura 12. Representação gráfica da curva ROC e das bandas de confiança ROC para o método Wald, Score e Bootstrap na versão em $2D$ e com a variável ponto de corte no gráfico $3D$.

Como se verifica, os pontos de corte mais baixos traduzem-se numa maior sensibilidade, mas também se traduzem num maior número de falsos positivos, ou numa menor especificidade. Normalmente opta-se por valores de corte intermédios, onde ambos os índices apresentam os valores mais altos, para permitir uma identificação razoável, tanto nos casos negativos como nos casos positivos, mas depende se a valorização do investigador recaí mais pela correta identificação dos casos negativos ou dos casos positivos, ou igualmente por ambos.

Capítulo 5. Estudo de simulação aos intervalos de confiança da sensibilidade e especificidade

A simulação é um método muito utilizado em estatística para avaliar a capacidade e a qualidade de diferentes métodos. Neste capítulo procuramos realizar um estudo de simulação para avaliar a cobertura dos intervalos de confiança dos três métodos em estudo: *Wald*, *Score* e *Bootstrap*.

Do ponto de vista de programação, o estudo de simulação dos intervalos de confiança da curva ROC é um procedimento simples, ainda que demorado.

No problema colocado pela curva ROC, estamos perante uma variável qualitativa de dois níveis, uma na qual o evento ocorreu, *presença*, e outra na qual o evento não ocorre, *ausência*. A variável contínua assume uma distribuição diferente para cada um dos níveis da variável categórica (*presente* ou *ausente*), quanto mais divergente forem as distribuições maior será a capacidade de discriminação da variável contínua.

Para o estudo de simulação geramos 1000 réplicas de amostras com 4 tamanhos distintos ($n=100,200,500,1000$). Cada uma destas amostras é constituída por duas variáveis: a variável que discrimina (ou marcador) e a variável binária que nos indica se o evento está presente ou não. Para gerarmos os dados destas observações assumimos que a população com o evento tem valores para o marcador que seguem uma distribuição normal de $N(20,8)$, e que a população em que o evento não ocorreu tem valores para o marcador que seguem uma distribuição normal $N(10,4)$. Consideramos também 4 proporções ($p=0.2,0.4,0.6,0.8$) de amostras onde o evento estava presente (gerados por uma distribuição de *bernoulli*).

Os intervalos de confiança para os três métodos (*Score*, *Wald* e *Bootstrap*) foram calculados em 4 pontos de corte ($k=5,10,15,20$).

Foram calculados os intervalos de confiança para a sensibilidade e especificidade pelos 3 métodos diferentes, *Wald*, *Score* e *Bootstrap*. Foi avaliada a probabilidade dos valores obtidos pertencerem ao intervalo de confiança de 95%, que se denomina de cobertura. Uma cobertura inferior significa que o intervalo de confiança é pouco exato e uma cobertura superior significa que o intervalo de confiança é pouco preciso. Uma cobertura perfeita deverá ter valores próximos de 95%.

5.1. Média dos níveis de cobertura dos Intervalos de confiança

Nós conduzimos uma simulação estatística para avaliar os intervalos de confiança da curva ROC perante um conjunto de diferentes cenários, de modo a avaliar a média dos níveis de cobertura de três diferentes métodos, acima enunciados, e que se espera que sejam próximos de 95%. De seguida são apresentadas tabelas, com a média dos níveis de cobertura dos intervalos de confiança, cada uma das tabelas faz referência a uma proporção diferente.

Tabela 9. Resultado do estudo de simulação para uma proporção de 0.2 e um nível de confiança de 0.95.

Método (Proporção = 0.2, p/nível de confiança=0.95)							
Ponto de Corte	Tamanho Amostra	Se Wald	Se Score	Se Boot	Sp Wald	Sp Score	Se Boot
5	100	0.459	0.460	0.458	0.920	0.926	0.934
	200	0.710	0.710	0.709	0.942	0.943	0.944
	500	0.874	0.777	0.928	0.939	0.945	0.941
	1000	0.922	0.840	0.928	0.943	0.950	0.941
10	100	0.848	0.626	0.848	0.949	0.922	0.953
	200	0.898	0.756	0.920	0.947	0.934	0.946
	500	0.931	0.861	0.930	0.949	0.947	0.951
	1000	0.948	0.909	0.946	0.960	0.952	0.959
15	100	0.903	0.721	0.923	0.935	0.857	0.942
	200	0.932	0.849	0.937	0.937	0.887	0.942
	500	0.936	0.906	0.942	0.938	0.926	0.942
	1000	0.929	0.910	0.933	0.947	0.938	0.943
20	100	0.931	0.820	0.947	0.381	0.381	0.380
	200	0.941	0.889	0.951	0.609	0.609	0.604
	500	0.948	0.918	0.953	0.904	0.679	0.899
	1000	0.945	0.931	0.943	0.881	0.811	0.934

Legenda: Se (Sensibilidade), Sp (Especificidade), Boot=método bootstrap.

Tabela 10. Resultado do estudo de simulação para uma proporção de 0.4 e um nível de confiança de 0.95.

Método (Proporção = 0.4, p/nível de confiança=0.95)							
Ponto de Corte	Tamanho Amostra	Se Wald	Se Score	Se Boot	Sp Wald	Sp Score	Se Boot
5	100	0.706	0.706	0.702	0.912	0.918	0.922
	200	0.914	0.695	0.912	0.932	0.929	0.934
	500	0.920	0.829	0.928	0.946	0.936	0.940
	1000	0.948	0.885	0.948	0.951	0.948	0.940
10	100	0.909	0.779	0.919	0.941	0.896	0.946
	200	0.912	0.822	0.926	0.933	0.913	0.937
	500	0.931	0.889	0.937	0.931	0.922	0.930
	1000	0.932	0.911	0.932	0.951	0.947	0.948
15	100	0.928	0.843	0.935	0.920	0.821	0.934
	200	0.943	0.899	0.942	0.924	0.883	0.930
	500	0.947	0.926	0.943	0.944	0.911	0.943
	1000	0.939	0.925	0.943	0.943	0.922	0.945
20	100	0.929	0.884	0.941	0.300	0.300	0.299
	200	0.947	0.907	0.952	0.527	0.527	0.520
	500	0.945	0.935	0.941	0.840	0.584	0.834
	1000	0.950	0.943	0.949	0.881	0.736	0.881

Legenda: Se(Sensibilidade), Sp (Especificidade), Boot=método bootstrap.

Tabela 11. Resultado do estudo de simulação para uma proporção de 0.6 e um nível de confiança de 0.95.

Método (Proporção = 0.6, p/nível de confiança=0.95)							
Ponto de Corte	Tamanho Amostra	Se Wald	Se Score	Se Boot	Sp Wald	Sp Score	Se Boot
5	100	0.840	0.583	0.834	0.911	0.896	0.924
	200	0.881	0.720	0.889	0.916	0.930	0.926
	500	0.929	0.864	0.931	0.936	0.948	0.939
	1000	0.945	0.915	0.947	0.942	0.940	0.939
10	100	0.925	0.829	0.934	0.933	0.882	0.940
	200	0.937	0.866	0.939	0.938	0.918	0.941
	500	0.951	0.912	0.948	0.951	0.944	0.950
	1000	0.946	0.931	0.946	0.944	0.939	0.945
15	100	0.941	0.880	0.940	0.929	0.794	0.925
	200	0.951	0.923	0.953	0.941	0.866	0.940
	500	0.943	0.936	0.945	0.942	0.912	0.936
	1000	0.951	0.948	0.943	0.946	0.920	0.943
20	100	0.941	0.910	0.944	0.211	0.211	0.211
	200	0.940	0.911	0.947	0.388	0.388	0.385
	500	0.950	0.944	0.937	0.719	0.720	0.719
	1000	0.953	0.946	0.947	0.921	0.710	0.919

Legenda: Se (Sensibilidade), Sp (Especificidade), Boot=método bootstrap.

Tabela 12. Resultado do estudo de simulação para uma proporção de 0.8 e um nível de confiança de 0.95.

Método (Proporção = 0.8, p/nível de confiança=0.95)							
Ponto de Corte	Tamanho Amostra	Se Wald	Se Score	Se Boot	Sp Wald	Sp Score	Se Boot
5	100	0.912	0.700	0.904	0.872	0.796	0.871
	200	0.893	0.801	0.934	0.913	0.893	0.927
	500	0.926	0.874	0.932	0.921	0.917	0.920
	1000	0.931	0.904	0.935	0.947	0.946	0.949
10	100	0.926	0.858	0.932	0.923	0.812	0.952
	200	0.955	0.917	0.955	0.942	0.871	0.954
	500	0.941	0.917	0.943	0.944	0.938	0.947
	1000	0.938	0.924	0.932	0.957	0.941	0.962
15	100	0.940	0.892	0.944	0.883	0.641	0.883
	200	0.950	0.929	0.951	0.908	0.757	0.916
	500	0.946	0.937	0.947	0.932	0.860	0.934
	1000	0.946	0.936	0.940	0.945	0.899	0.946
20	100	0.945	0.905	0.947	0.101	0.101	0.101
	200	0.947	0.932	0.948	0.237	0.237	0.236
	500	0.944	0.949	0.943	0.486	0.487	0.485
	1000	0.941	0.935	0.937	0.718	0.719	0.716

Legenda: Se (Sensibilidade), Sp (Especificidade), Boot=método bootstrap.

Nas tabelas acima apresentada pode-se verificar que em 64,8% dos casos, a média dos níveis de cobertura variaram entre os 90% e os 95%. A média dos níveis de cobertura foi perfeito, ou seja, de 95.0%, apenas em 1.4% dos casos e foi superior a 95%, ou seja, pouco preciso, em 5.7% das situações.

Para estes pontos de corte, se utilizarmos o método *Wald* podemos esperar uma cobertura superior a 90% em 82.8% dos casos, no entanto este nível de cobertura baixa para 64.1% no método *bootstrap* e para 47.6% dos casos no método *score*.

5.2. Avaliação gráfica dos níveis de cobertura

Uma vez que o número de valores obtidos para a cobertura é muito extenso, pois foram ponderadas diferentes variáveis, optámos por sumariar os resultados, apresentando graficamente a relação entre as médias dos níveis de cobertura, o método utilizado para calcular o nível de confiança, o ponto de corte, o tamanho da amostra e a proporção, de modo a compreender melhor a relação entre os níveis de cobertura e estas variáveis. Os valores de cobertura apresentados resultam da média das variáveis ocultas no gráfico, para as categorias em estudo.

5.2.1. Nível de Cobertura médio por método de cálculo dos intervalos de confiança e tamanho da amostra

O próximo gráfico sugere, que, o nível de cobertura se aproxima dos 95% à medida que o tamanho da amostra aumenta, para todos os métodos sem exceção, como era esperado.

Para os pontos de corte indicados, podemos esperar um nível médio de cobertura entre os 90 e os 95% se utilizarmos o método *Wald* e *Bootstrap* para o cálculo da sensibilidade, desde que o tamanho amostral seja igual ou superior a 200, ou no caso do método *score* para a sensibilidade se o tamanho da amostra for superior igual ou superior a 500.

Na especificidade podemos esperar uma cobertura superior a 90% para o método *Wald* e *Bootstrap* se a amostra for igual ou superior a 1000, para os pontos de corte utilizados. No método *score*, o nível mais alto de cobertura média, para estes pontos de corte na especificidade foi de 89%. Os métodos, *Wald* e *Bootstrap*, são aqueles que apresentam valores médios mais próximos do nível de cobertura ideal, entre os três métodos testados.

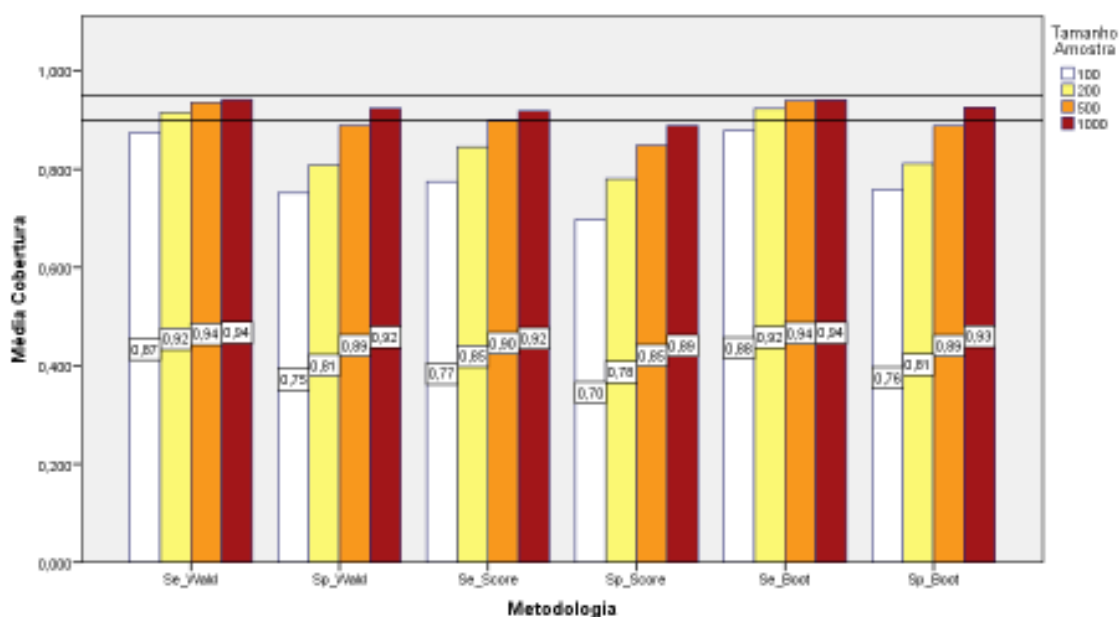


Figura 13. Níveis médios de cobertura por método e tamanho de amostra.

5.2.2. Nível de Cobertura médio por ponto de corte

5.2.2.1. Em relação ao método de cálculo dos intervalos de confiança

O próximo gráfico realça a presença de níveis médios de cobertura muito baixos para pontos de corte de 5 na sensibilidade (Se) e para pontos de corte de 20 na especificidade (Sp) para todos os métodos, sendo que o método *Wald* é aquele em que esse problema é mais atenuado.

Para valores de corte de 10 na especificidade e valores de 20 na sensibilidade no método *Bootstrap*, *Score* e no método *Wald*, os níveis médios de cobertura encontram-se entre os 92 e os 95%, e em particular no método *bootstrap* (Se Boot/Sp Boot) e no método *Wald* estes valores situam-se entre os 94-95%. O método *score* parece ser o método menos adequado.

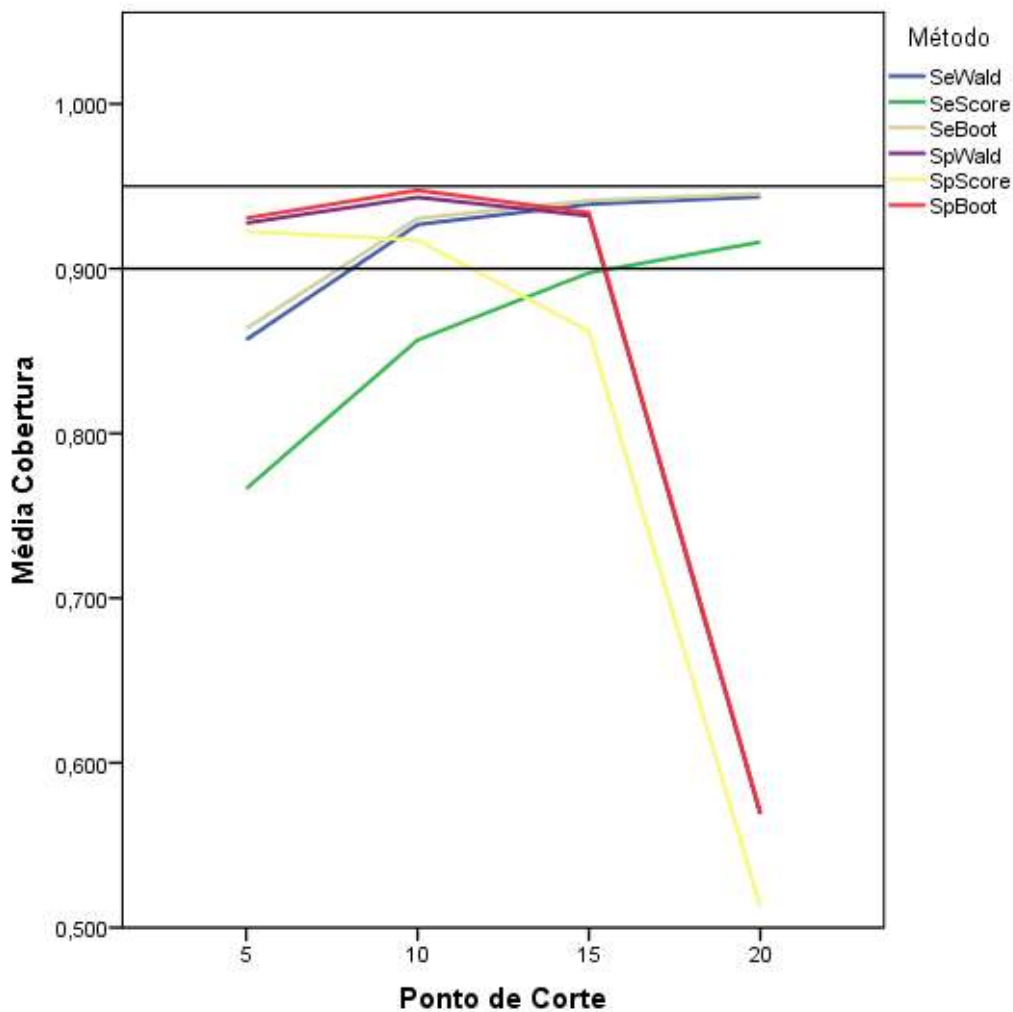


Figura 14. Níveis médios de cobertura por método e ponto de corte.

5.2.2.2. Em relação ao tamanho da amostra

A avaliação dos pontos de corte em relação ao tamanho da amostra informa que, o nível de cobertura aumenta substancialmente em função do tamanho da amostra e que para obter um nível de cobertura médio entre 90 e 95% para os pontos de corte, 5, 10 e 15, na sensibilidade e especificidade, é necessário que a amostra seja igual ou superior a 500. Se quisermos obter a cobertura máxima em todos os pontos de corte avaliados, para todas as opções, a solução ideal é a utilização de uma amostra de 1000 ou superior, com recurso ao método *Wald* ou *Bootstrap*.

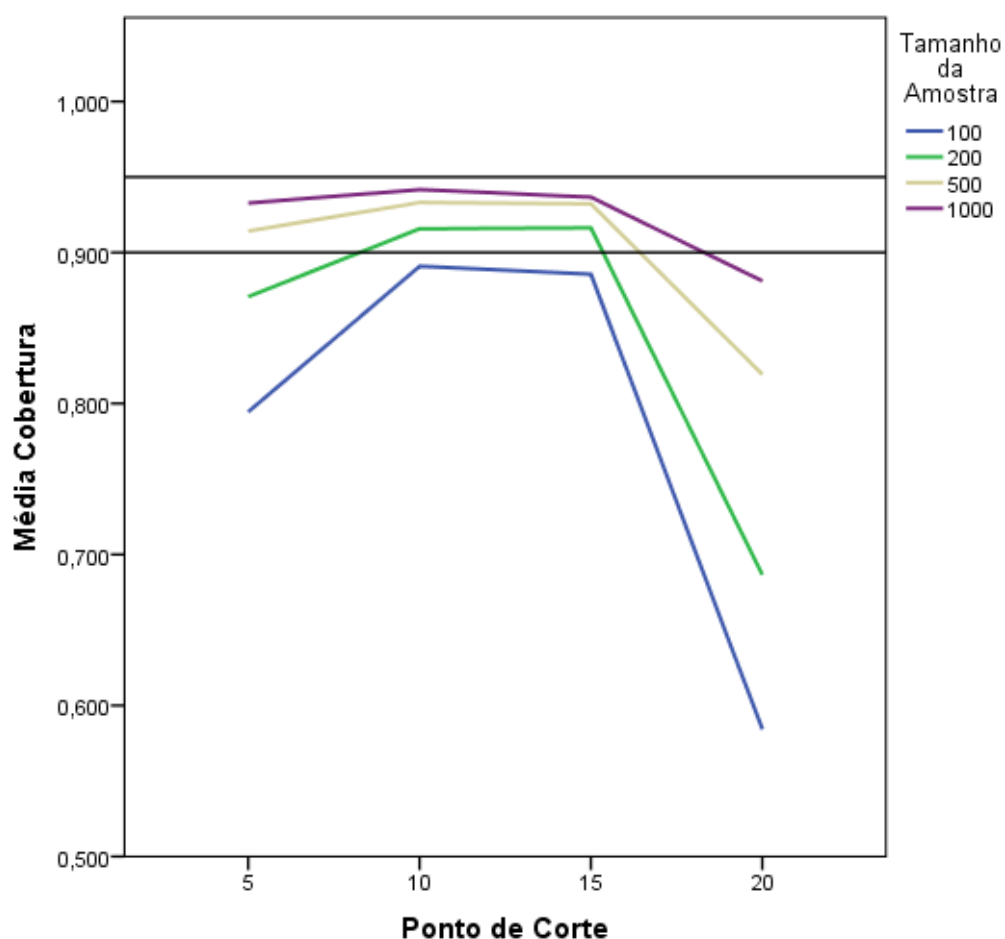


Figura 15. Níveis médios de cobertura por tamanho de amostra e ponto de corte.

5.2.2.3. Níveis médios de Cobertura por proporção

Os níveis médios de cobertura para cada um dos métodos e níveis de proporção variaram entre os 75% e os 95%. Os níveis mais elevados de cobertura foram obtidos pelos intervalos de confiança de sensibilidade e especificidade do método *Wald* e *Bootstrap*. Como seria de esperar, proporções maiores na sensibilidade e proporções menores na especificidade resultaram numa cobertura maior.

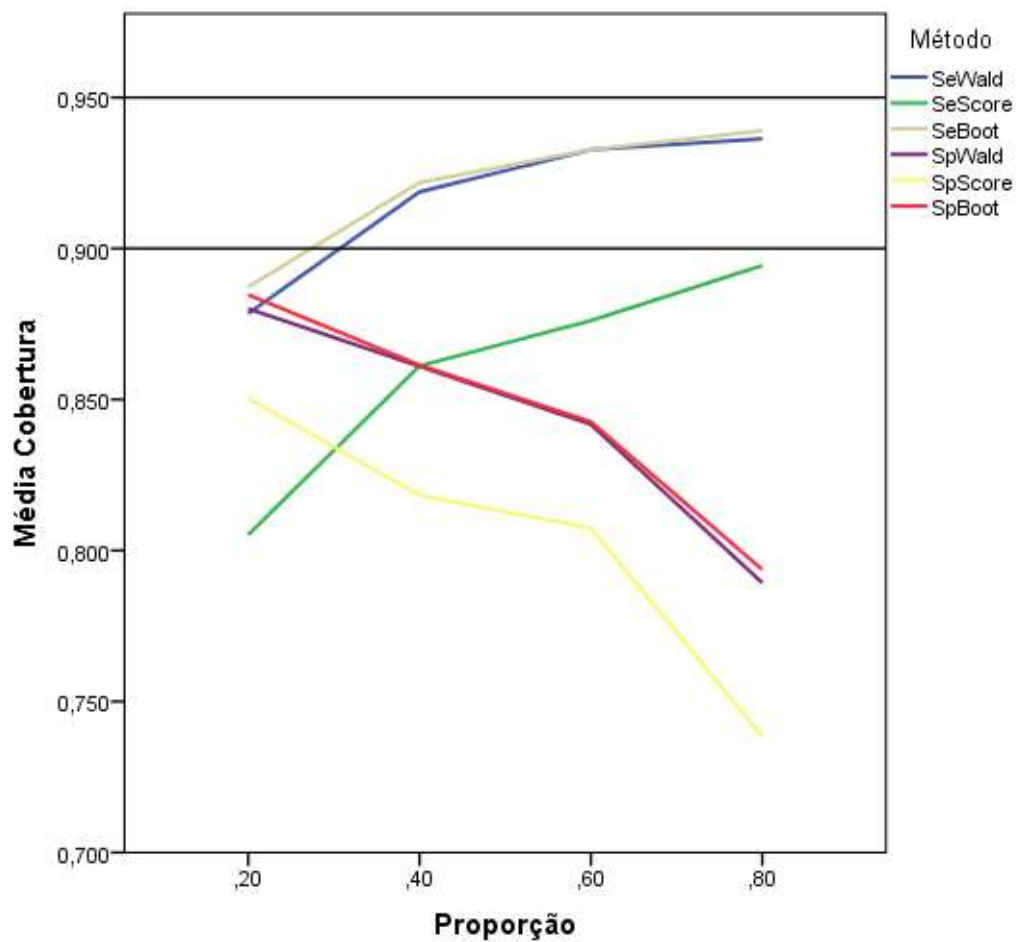


Figura 16. Níveis médios de cobertura por método e proporção.

5.3. Avaliação geral dos níveis de cobertura

Os níveis de cobertura são influenciados pelo tamanho amostral, pelo ponto de corte, pelo método e pela proporção. No caso da sensibilidade, proporções superiores e pontos de corte superiores e na especificidade, pontos de corte inferiores e proporções inferiores resultam em níveis médios de cobertura superiores.

O método *Wald* é aquele que apresenta melhor nível médio de cobertura, seguido do método *Bootstrap* e, por fim, do método *Score*. O tamanho amostral é uma característica importante, aumentando substancialmente o nível de cobertura. O próximo gráfico resume os aspetos aqui salientados, ressaltando a importância do método, tamanho da amostra e ponto de corte, para o nível de cobertura obtido.

Os nossos resultados não vão ao encontro aos resultados de Agresti e Coull (1998), uma vez que o método *Wald* teve um desempenho superior, mesmo nas amostras mais pequenas, quando comparado com o método *Score*.

Estes resultados salientam também o facto dos níveis de cobertura dos intervalos de confiança nem sempre serem os esperados, sobretudo em amostras mais pequenas e em valores extremos, reforçando a importância de se calcular os intervalos de confiança para as curvas ROC com amostras com o número mais elevado possível.

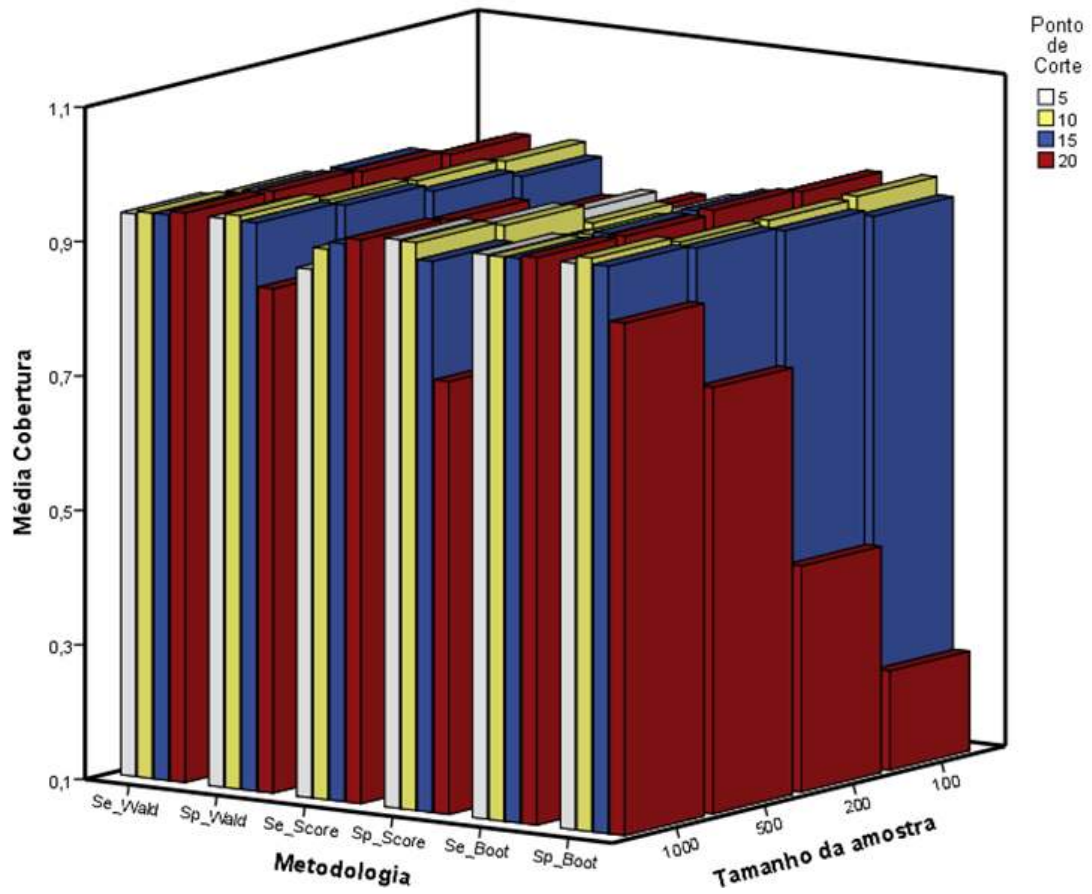


Figura 17. Níveis médios de cobertura por ponto de corte, tamanho da amostra e método.

Capítulo 6. Relação entre os Índices da Curva ROC e o Custo/Retorno Esperado

Um dos principais objetivos da análise ROC é determinar o ponto de corte ótimo para um determinado teste de diagnóstico [23-24].

A utilização do ponto de corte ótimo permite maximizar a relação custo-benefício e a relação custo-efetividade, no sentido em que o ponto de corte ótimo é aquele em que a sua utilidade é maximizada [23], no entanto esta relação depende de um conjunto alargado de fatores e deve ser analisada tendo em linha de conta a prevalência do evento em estudo na população abrangida, o custo monetário da intervenção e a capacidade de prevenção/promoção da ocorrência do evento (e.g. doença prevenida, utilização de contraceptivos, número de vidas salvas) [23,25].

A relação custo-benefício, na saúde em particular, resulta da razão entre os custos de diagnosticar um indivíduo que não é portador de uma doença e o benefício dessa avaliação ser realizada em doentes que são efetivamente portadores da doença.

Metz (1978) formula a média de custos através da seguinte equação para calcular o custo médio esperado. Nesta equação está presente o custo base C_0 e a probabilidade de ocorrência de cada um dos eventos (verdadeiros positivos (VP), verdadeiros negativos (VN), falsos positivos (FP) e falsos negativos (FN)) e os custos associados a cada um desses eventos.

$$\bar{C} = C_0 + C_{VP} \times P(VP) + C_{VN} \times P(VN) + C_{FP} \times P(FP) + C_{FN} \times P(FN) \quad (39)$$

Esta equação pode ser usada tanto para estimar os custos esperados como o retorno esperado. Podemos ainda calcular os custos/benefício esperados, excluindo da equação a incógnita C_0 , desde que o valor de C_0 seja um custo incluindo em cada um dos custos de cada um dos eventos possíveis (eg. C_{VN} , C_{FP}), uma vez que o custo de cada evento possível passa obrigatoriamente pelo custo do diagnóstico, o mesmo se verifica para as consequências associadas ao rastreio. Podendo assim a equação assumir a seguinte forma:

$$E(C) = C_{VP} \times P(I|D) \times P(D) + C_{FN} \times P(D) \times P(\bar{I}|D) + C_{FP} \times P(\bar{D}) \times P(I|\bar{D}) + C_{VN} \times P(\bar{I}|\bar{D}) \times P(\bar{D}) \quad (40)$$

Ressalva-se que nesta equação $P(I|D)$ representa a probabilidade condicionada de ser identificado doente sabendo-se que a pessoa está doente, ou seja, a sensibilidade, $P(\bar{I}|\bar{D})$ representa a especificidade, $P(I|\bar{D})$ é a probabilidade condicionada de um falso positivo $P(\bar{I}|D)$ e representa os falsos negativos.

Segundo Metz (1978), podemos obter a rede de benefícios através da equação:

$$\overline{\text{Rede de Benefício}} = \overline{\text{Custo Min}} (\text{não usar o teste}) - \overline{\text{Custo Min}} (\text{usar o teste}) \quad (41)$$

Com esta equação é possível estimar o valor não só de custos mas também da efetividade/retorno de um determinado teste de discriminação. Permite avaliar e comparar diferentes meios e identificar para um determinado custo máximo qual é o meio de discriminação/teste que melhor maximiza a obtenção dos objetivos propostos pelo investigador e qual o melhor ponto de corte.

Por exemplo, em medicina é tradicionalmente avaliado o custo de determinados procedimentos médicos por cada ano de vida proporcionado ao paciente, no entanto a efetividade/ retorno esperado pode ser avaliada com diferentes variáveis, tais como, o número de pacientes curados ou número de pessoas em que a doença foi prevenida [23, 27-28].

$$\text{Razão Custo efetividade} = \frac{\text{Custo da intervenção}}{\text{Efetividade da intervenção}} \quad (42)$$

Do ponto de vista político e social torna-se muito útil a comparação entre diferentes razões de custo efetividade que permitem comparar o incremento da razão de custo-efetividade (IRCE) da substituição de um determinado procedimento por outro [28-29].

$$IRCE = \frac{\text{Custo 1} - \text{Custo 2}}{\text{Efetividade 1} - \text{Efetividade 2}} \quad (43)$$

A relação custo-efetividade é bidimensional e neste sentido tem a vantagem de permitir avaliar o nível de execução dos ganhos que determinado custo permite atingir [29].

6.1 Função Programada: *slife*

A função *slife* calcula a sensibilidade, especificidade, custos e retorno esperado para cada ponto de corte. É útil não apenas em Ciências da Saúde, mas nas mais diversas áreas que englobem meios de diagnóstico. É uma ferramenta de apoio à decisão, estimação de custos e do retorno esperado e de calibração de meios de discriminação binária em função dos objetivos.

As próximas fórmulas, desenvolvidas com base na literatura foram programadas em *software* R, e a sua utilização tem por objetivo calcular o custo unitário esperado e o retorno unitário esperado para cada ponto de corte (k).

$$\begin{aligned} \text{Custo unitário}(k) = & (\text{Se}(k) \times P(D) \times \text{CSe}) + (\text{Sp}(k) \times P(\bar{D}) \times \text{CSp}) + (\text{FN}(k) \times P(\bar{D}) \times \text{CFn}) + \\ & (\text{FP}(k) \times P(D) \times \text{CFp}) \end{aligned} \quad (44)$$

$$\begin{aligned} \text{Retorno unitário} = & (\text{Se}(k) \times P(D) \times \text{RSe}) + (\text{Sp}(k) \times P(\bar{D}) \times \text{RSp}) + (\text{FN}(k) \times P(\bar{D}) \times \text{RFn}) + \\ & (\text{FP}(k) \times P(D) \times \text{RFp}) \end{aligned} \quad (45)$$

Os custos unitários e retornos unitários, são representados por “C” e “R” respetivamente, para cada uma dos acontecimentos possíveis, quando a doença está presente e o diagnóstico correto (CSe, Rse), quando a doença está ausente e o diagnóstico incorreto

(CSp,RSp), quando a doença está presente e o diagnóstico incorreto (CFn, RFn) e quando a doença está ausente e o diagnóstico está incorreto (CFp, RFp).

E onde $Se(k)$ é a sensibilidade, $F_n(k)$ ou $1-Se(k)$, os falsos negativos, $Sp(k)$ é a especificidade e $F_p(k)$ ou $1-Sp(k)$, reporta os falsos positivos, para determinado pontos de corte k . $P(D)$ é a probabilidade da doença estar presente e $1-P(D)$ a probabilidade de ausência da doença em estudo.

A equação de custo unitário pretende calcular o custo unitário esperado para cada ponto de corte. A equação de retorno unitário pretende determinar em que nível os objetivos podem ser alcançados, os resultados podem ser positivos ou negativos, e podem representar qualquer variável, por exemplo, a probabilidade de salvar uma vida, em função de cada ponto de corte.

A função *slife* é uma solução para a estimação dos custos e retornos unitários esperados para cada ponto de corte. Esta função tem a grande vantagem de permitir ao utilizador determinar o ponto de corte k em que o retorno é máximo para um determinado custo de rastreio ou determinar qual o custo mínimo para se obter um retorno unitário positivo. Permite ainda comparar estes valores com os demais métodos.

1.função

slife(y,x,pd,custse,retse,custfp,retfp,custsp,retsp,custfn,retfn)

2.argumentos:

1.x="variável quantitativa"

2.y="variável qualitativa"

3.pd="probabilidade de doença"

4. custse="custo da sensibilidade"

5. retse="retorno da sensibilidade"

6. custfp="custo dos falsos positivos"

7. retfp="retorno dos falsos positivos"

8. custsp="custo da especificidade"

9. retsp="retorno da especificidade"

10. custfn="custo dos falsos negativos"

11. retfn="retorno dos falsos negativos"

3. Esta função retorna uma lista com os seguintes itens:

1. *Lista de sensibilidade*
2. *Lista de especificidade*
3. *Lista com o custo unitário esperado*
4. *Lista com o retorno unitário esperado*
5. *Lista com os pontos de corte correspondentes*

Na exemplificação da função será utilizada a base de dados *Leukocyte Elastase Data* em que é avaliada a capacidade de discriminação do angiograma na avaliação da doença arterial coronária (DAC), vamos exemplificar a utilização das fórmulas acima apresentadas, *CostUnit* (custo unitário), *RetUnit* (retorno unitário), programadas em *software R*, na função *slife*.

No cálculo desta função foram considerados valores fictícios, apresentados a título exemplificativo, os resultados servem deste modo para conclusões do domínio estatístico e não do domínio médico.

Na função *slife*, foi considerada uma probabilidade de doença no grupo em estudo de 28%. Ao nível do retorno foi considerada a probabilidade de salvar uma vida, e nos custos, todos os custos associados.

Ao nível dos custos, considerou-se um preço de 5 quando um paciente era corretamente diagnosticado com DAC (rastreamento+tratamento), e uma probabilidade de salvar uma vida de 30%. Um custo de 20 se o doente é incorretamente diagnosticado com DAC (rastreamento+tratamento indevido), para um retorno de risco de vida de 10%. Considerou-se um custo de ser corretamente diagnosticado sem DAC, de 5 (rastreamento), para um retorno negativo de risco de rastreamento de 1% (riscos de rastreamento). Por fim, nos casos em que o doente de DAC não é diagnosticado, considerou-se um custo de 20 (rastreamento + preço de tratamento tardio), com o retorno negativo de 30%.

6.1.1. Custo e retorno unitário esperado por ponto de corte.

A próxima tabela apresenta, em versão reduzida, os resultados calculados pela função *slife*, com o custo unitário esperado (*CostUnit*) pelo retorno unitário esperado (*RetUnit*).

Tabela 13. Tabela reduzida com o custo e retorno unitário esperado para cada ponto de corte.

Pares	Ponto Corte	<i>Se</i>	<i>Sp</i>	<i>CustUnit</i>	<i>RetUnit</i>
8	16	0.969	0.267	9.040	0.024
9	18	0.969	0.289	8.960	0.025
10	20	0.969	0.311	8.880	0.027
11	21	0.958	0.356	8.720	0.028
12	22	0.938	0.378	8.640	0.026
13	24	0.927	0.378	8.640	0.024
14	25	0.896	0.378	8.640	0.019
15	26	0.854	0.378	8.640	0.012
16	27	0.813	0.400	8.560	0.006
17	29	0.802	0.467	8.320	0.009
18	30	0.771	0.489	8.240	0.005
19	31	0.760	0.489	8.240	0.003
20	32	0.740	0.511	8.160	0.001
21	33	0.729	0.533	8.080	0.001
22	34	0.698	0.556	8.000	-0.003
23	35	0.698	0.556	8.000	-0.003

Legenda: Se (Sensibilidade), Sp (Especificidade), Custunit (Custo unitário), RetUnit (Retorno Unitário)

Uma apreciação rápida da tabela permite-nos ver que *RetUnit* apresenta o seu máximo no par 11, no ponto de corte 21, que corresponde a uma sensibilidade de 0.958 e a uma

especificidade de 0.356, com um retorno de 0.028 e um custo associado (*CostUnit*) de 8.72.

Pontos de Corte: Custo e Retorno Esperado

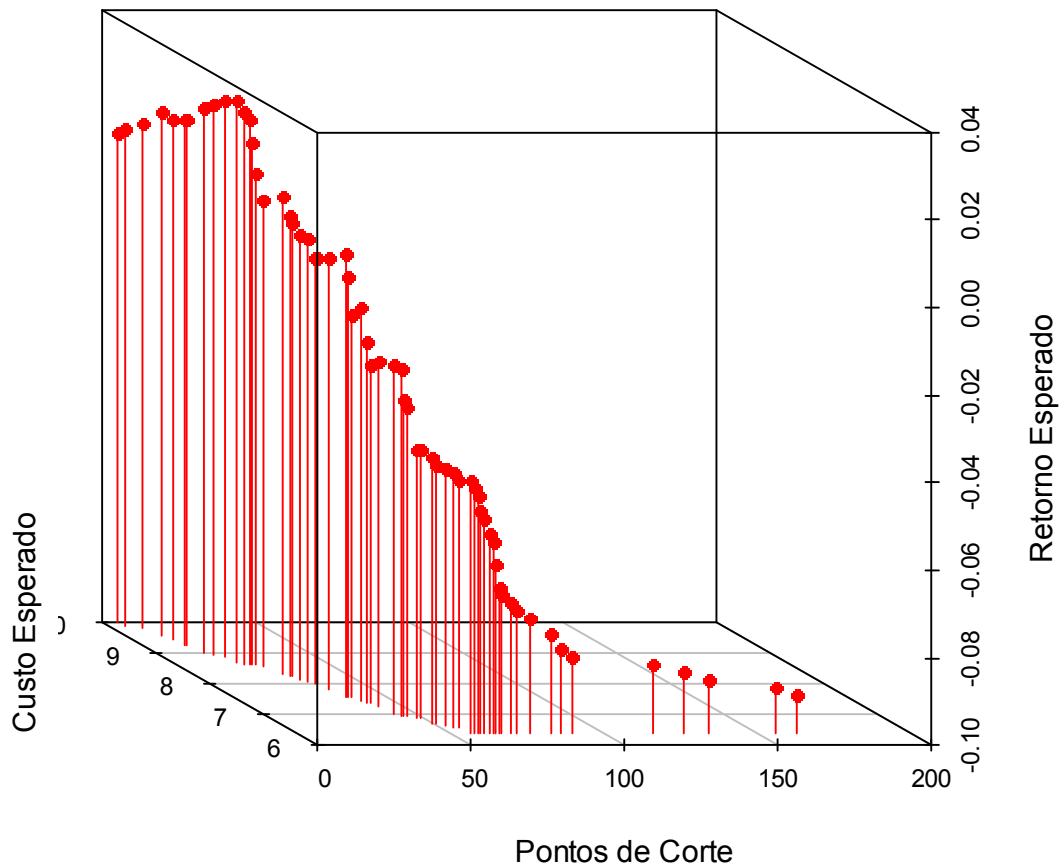


Figura 18. Gráfico 3D que analisa a relação entre o custo e o retorno unitário esperado em função de cada ponto de corte.

Como se pode ver no próximo gráfico, num rastreio populacional que envolvesse 10000 pessoas, o rastreio teria um custo de 87200 euros, esperando salvar-se 280 vidas em cada 10000 habitantes, como se pode ver nos próximos gráficos.

Custo-Retorno Esperado por 10000 hab

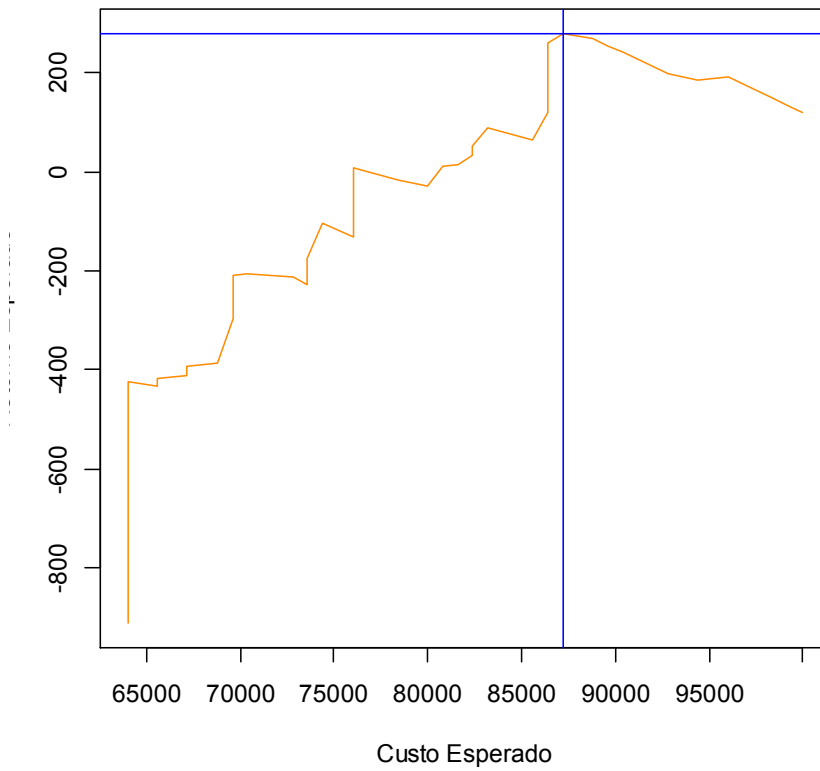


Figura 19. Gráfico que analisa a relação entre o custo e o retorno esperado em 10000 habitantes.

É também importante denotar que a partir do ponto de corte 34 o retorno começa a ser negativo, apesar de ter custos associados, estima-se que a partir deste ponto o rastreo seja prejudicial para a população.

6.1.2. Custo unitário esperado em função do retorno unitário esperado

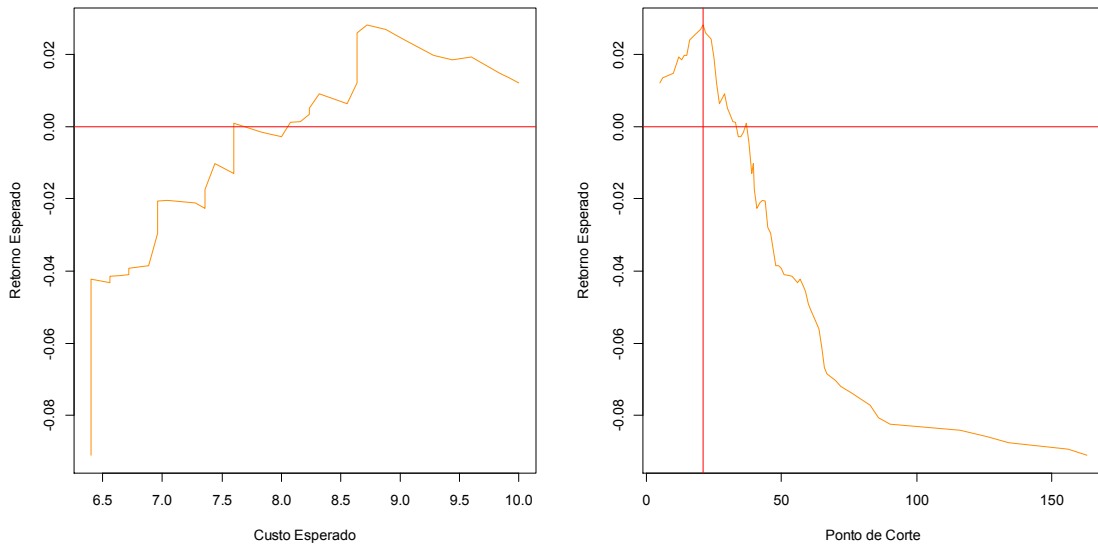


Figura 20. Custo unitário esperado em função do retorno unitário esperado e Retorno unitário esperado em função dos pontos de corte.

Outra aspeto que realça a importância deste tipo de funções é o facto de valores de ponto de corte inferiores a 21 apresentarem pior retorno apesar de terem custos associados mais elevados e de serem mais sensíveis. Como pode ser visto no próximo gráfico 3D que foca este aspeto específico.

Pontos de Corte: Custo e Retorno Esperado

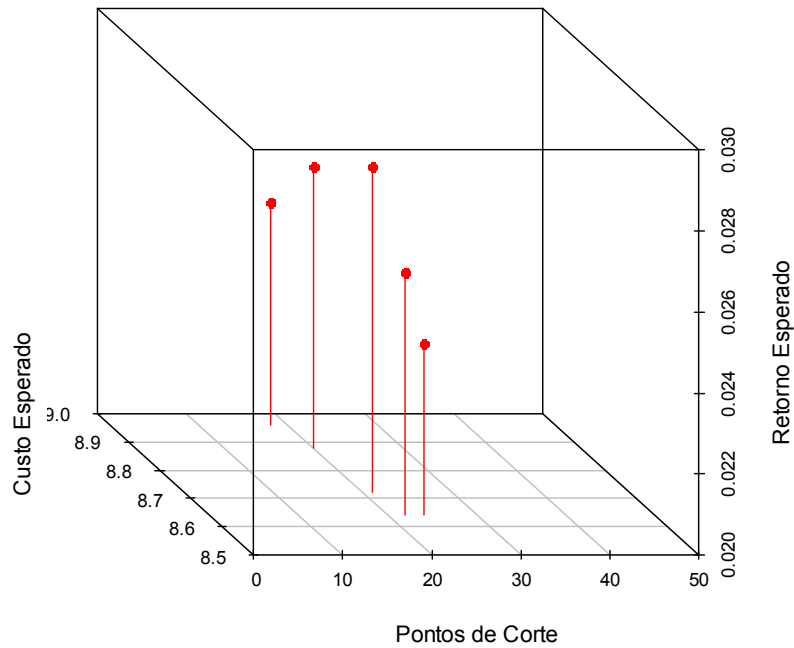


Figura 21. Representação gráfica com a relação entre os pontos de corte, o custo unitário esperado e o retorno unitário esperado (focado entre 0.020 e 0.030)

Apesar dos custos esperados aumentarem à medida que a sensibilidade aumenta, isto nem sempre acontece em relação ao retorno esperado uma vez que a partir do ponto de corte 21 a sensibilidade é maior, mas o aumento dos falsos positivos faz com que um rastreamento que tem uma sensibilidade superior e um custo esperado superior se repercuta numa desvantagem no retorno obtido.

6.1.3. Relação entre sensibilidade, especificidade e o Custo/ Retorno Unitário Esperado

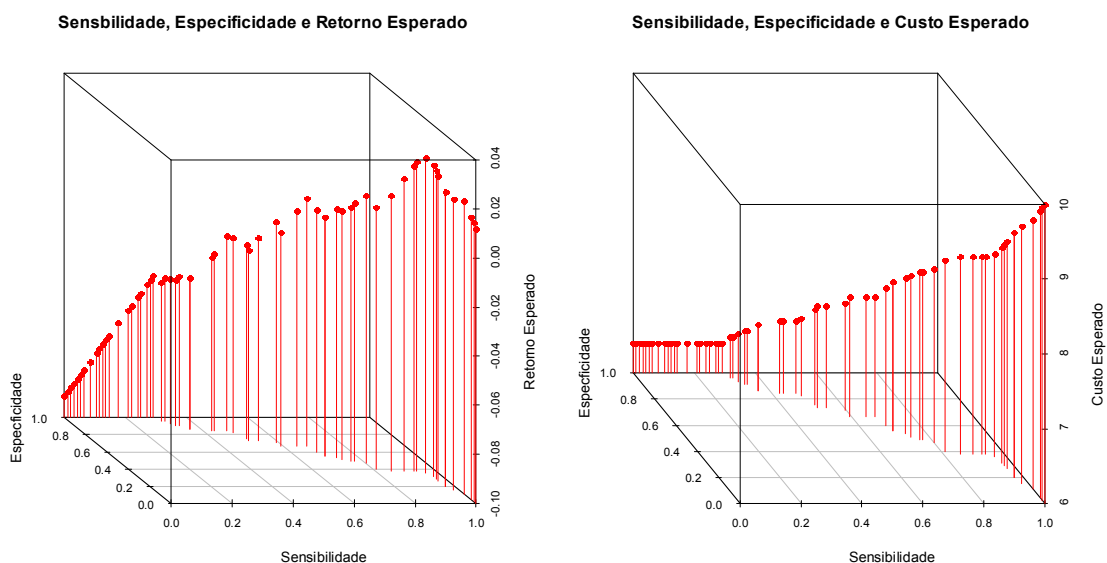


Figura 22. Representação gráfica com a relação entre a sensibilidade, especificidade em relação ao retorno unitário esperado e em relação ao custo unitário esperado.

Se o objetivo fosse obter um retorno positivo com o preço mínimo, em vez do retorno máximo possível, o ponto de corte seria de 37 em vez de 21, para uma sensibilidade de 0.67 e uma especificidade de 0.66, onde se esperaria salvar entre 9/10 vidas por 10000 habitantes, em vez de 280 vidas, mas reduzindo os custos absolutos de 87200 euros para o valor de 76000 euros, porém aumentando o custo por vida salva de 311,42 euros para 8000 euros por habitante.

Neste sentido, e uma vez que a tomada de decisão depende dos objetivos e dos recursos específicos de cada decisor, a avaliação de tabelas e gráficos com os dados que esta função gera poderá facilitar consideravelmente o processo de decisão.

Capítulo 7. Conclusões e projetos futuros

Ao longo desta dissertação pretendeu-se providenciar e avaliar diferentes métodos para estimar a sensibilidade e a especificidade, e conseqüentemente a curva ROC. Em todos os tópicos foram exemplificadas com análises estatísticas que visaram aprofundar as funções e a utilidade deste trabalho.

Em particular, no Capítulo 4 foram programadas e apresentadas 6 funções, *myroc*, *scut*, *plot.secut*, *plot.spcut*, *SpSe* e por fim, o *ypindex*. Estas funções permitem estimar e representar a sensibilidade, a especificidade e a curva ROC, bem como os limites de confiança, através de 3 métodos diferentes, *Wald*, *Score* e *Bootstrap*, providenciando representações gráficas.

Muitas das funções retornam variáveis que lhes são comuns, como a sensibilidade e a especificidade, isto poderia ter sido evitado se criássemos uma só função que devolve-se toda a informação programada, mas optámos por ir criando funções à medida que aprofundávamos conceitos teóricos relevantes na dissertação, de modo a permitir uma abordagem concisa e direcionada.

No Capítulo 5 foram avaliados os níveis médios de cobertura, a 95%, para os intervalos de confiança calculados pelos métodos acima referidos. Os melhores níveis de cobertura médios foram obtidos com recurso ao método *Wald* e *Bootstrap*, sendo que o método *score* é o menos recomendado. Conclui-se ainda que o tamanho da amostra é um fator muito relevante nos níveis de cobertura, pelo que é recomendado a utilização de amostras de número elevado.

O sexto Capítulo disponibiliza uma função, *slife*, que permitirá auxiliar na tomada de decisão em situações em que os custos e as consequências dos índices que determinam a curva ROC divergem entre si, situação que é muito frequente.

Demonstramos ainda, que sob determinadas situações, como as apresentadas neste capítulo, apenas uma função deste tipo permitirá determinar o ponto de corte que melhor vai de encontro aos objetivos do investigador ou agente político.

Por fim, pretendemos deixar uma nota sobre os projetos futuros, que resultarão desta dissertação, nomeadamente, a submissão de um pacote para o R, que resulte em quatro funções, *myroc*, *slife*, *ypindex* e uma função que agregue todas as outras funções apresentadas no Capítulo 4 (*scut*, *plot.secut*, *plot.spcut*, *SpSe*).

Pretendemos ainda, apresentar, no âmbito científico, os resultados do estudo de simulação, na expectativa de resultar num diálogo estimulante e frutuoso sobre o tema que suscite interesse na realização de novos estudos científicos.

Bibliografia

1. Azevedo L.F., & Pereira, A.C. (2010). Avaliação Crítica e Implementação Prática de Estudos Sobre a Validade de Testes Diagnósticos: Parte II. *Nascer e Crescer*, 40(190): 265-277.
2. Arian, R., van Erkel, P., & Pattynama, M.T. (1998). Receiver operating characteristic (ROC) analysis: Basic principles and applications in radiology. *European Journal of Radiology*, 27 (2): 88–94
3. Pepe M.S. (2003) *The Statistical Evaluation of Medical Tests for Classification and Prediction*. Oxford: Oxford University Press.
4. Gastwirth, J. L. (1972). The Estimation of the Lorenz Curve and Gini Index. *The Review of Economics and Statistics*, 54 (3): 306–316.
5. Min, Y., Kryjov, VN & Chung-Kyu, P. (2009). A Probabilistic Multimodel Ensemble Approach to Seasonal Prediction. *Wea. Forecasting.*, 24: 812–828.
6. Braga, A.C. (2000). *Curvas ROC: Aspectos funcionais e aplicações* (Tese de Doutorado). Braga: Universidade do Minho.
7. Marini, A. (2007) *Análise de Estratégias de Rejeição para Problemas com Múltiplas Classes Utilizando Curvas ROC*. (Tese de Mestrado). Paraná: Pontifícia Universidade Católica do Paraná.
8. Kamphuis, J.H., & Noordhof, A. (2009). On categorical diagnoses in DSM-V: Cutting dimensions at useful points?. *Psychological Assessment*, 21: 294-301.
9. Fawcett, T. (2006). *An introduction to ROC analysis*. *Pattern Recognition Letters*, 27: 861–874.

10. Swets, J.A. (1996). Signal detection theory and ROC analysis in psychology and diagnostics. Collected papers. New Jersey: Hillsdale.
11. Milan L.A., Neto F. & Curi, M. (2009). Regiões de incerteza para a curva ROC em testes diagnósticos. (Tese de Mestrado). São Carlos: Universidade Federal de São Carlos.
12. McCarthy W.F. & Guo N.(2006). The estimation of sensitivity and specificity of clustered binary data. Papper number: 206-310. SUGI 31 Proceedings.
13. McSkassy, S. & Provost, F., Rosset, S. (2005). ROC Confidence Bands: An Empirical Study *NYU Working Paper No. 2451/14107*
14. Regehr, G. & Colliver, J. (2003). On the equivalence of classic ROC analysis and the loss-function model to set cut points in sequential testing. *Academic Medicine*, 78(4): 361-4.
15. Agresti, A. & Coull, B. (1998). Approximate is better than “exact” for interval estimation of binomial proportions. *Am. Stat.*, 52: 119-126.
16. Smits. N. (2010). A note on Youden’s J and its cost ratio. *BMC Medical Research Methodology*, (10):89: 1-4.
17. Kumar R, Indrayan A. (2010). Receiver operating characteristic [ROC] curve for medical researchers. *Indian Pediatr.*, 48(4): 277-87.
18. Amaro, A., Gude, F., Gonzalez-Juanetey, R., Iglesias, C., Fernández-Vazquez, F., Garcia-Acuna, J. & Gil, M. (1995). Plasma leukocyte elastase concentration in angiographically diagnosed coronary artery disease. *European Heart Journal*, 16: 615-622.
19. Efron, B. & Tibshirani, R., J. (1998). *An Introduction to the Bootstrap*. London: Chapman & Hall/CRC.

20. Abreu, A.L.E., & Neto, A.C. (2011). Método computacionalmente intensivo para o cálculo dos intervalos de confiança para valores da função de confiabilidade. XVI Simpósio de Pesquisa operacional & Logística da Marinha.
21. Prati R, Batista G & Monard M. (2008). Curvas ROC para a avaliação de classificadores. *Revista IEEE América Latina*, 6 (2): 215-222.
22. Pestana, M.H., & Gageiro (2005). *Análise de Dados para Ciências Sociais: A Complementaridade do SPSS*. Lisboa: Sílabo
23. Cantor, S.B., Sun, C.C., Tortolero-Luna, G., Richards-Kortum, R. & Follen, M. (1999). A comparison of C/B ratios from studies using receiver operating characteristic curve analysis. *Journal of Clinical Epidemiology*, 52(9): 885-892.
24. Metz, C.E. (1978) Basic principles of ROC analysis. *Semin Nucl Med*, 8: 283–298.
25. Chen, L.S., Liao, C.S., Chang, S.H., Lai, H.C., & Chen, H.H. (2007) Cost-effectiveness analysis for determining optimal cut-off of immunochemical faecal occult blood test for population-based colorectal cancer screening (KCIS 16). *J Med Screen.*, 323 (1-2) :205-15.
26. Rousson, V. & Zumbunn, T. (2011) Decision curve analysis revisited: overall net benefit, relationships to ROC curve analysis, and application to case-control studies. 2011. *BMC Medical Informatics and Decision Making*, 11: 45-54.
27. Chang K-G, Lee H-C, Huang Y-C, Hung, *et al.* (2012). Cost-effectiveness analysis of stroke management under a universal health insurance system. *J Neurol Sci*, 323: 205-215.
28. Bell, C.M., Urbach, D.R., Ray, J.G., *et al.* (2006) Bias in published cost effectiveness studies: systematic review. *BMJ*, 332 (7543): 699–703.

29. Warner, K.E. & Luce, B.R. (1982). Cost-Benefit Analysis and Cost-Effectiveness Analysis in Health Care: Principles, Practice, and Potential. Ann Arbor, Mich.: Health Administration Press.

ANEXO A. Código de Sintaxe e Programação

```
library(pROC)
library(OptimalCutpoints)
library(scatterplot3d)
data(elas)
attach(elas)
elas$status2<-as.factor(elas$status)
```

#1.função myroc: Sensibilidade e Especificidade

```
library(pROC)
library(OptimalCutpoints)
data(elas)
attach(elas)
elas$status2<-as.factor(elas$status)
```

```
myroc<-function(y,x){

  if(!is.factor(y)) stop("The variable must be of class factor")
  if(nlevels(y)!=2) stop("The variable must be dichotomous")

  x1<-sort(unique(x))

  n1<-length(x1)

  res_se<-rep(0,n1)
  res_sp<-rep(0,n1)

  for (k in 1:n1){
```

```

cutp<-x1[k]
p1<-which(x > cutp & y == levels(y)[2] )
p2<-which( y == levels(y)[2] )
res_se[k]<-length(p1)/length(p2)

p3<-which(x < cutp & y == levels(y)[1] )
p4<-which( y == levels(y)[1] )
res_sp[k]<-length(p3)/length(p4)}

auc<-sum(diff(1-res_sp[length(res_sp):1])*res_se[length(res_se):1][-1])

#res<-cbind(res_se,res_sp)
#return(res)
#return(list(Se=res_se,Sp=res_sp,SeSp=data.frame(res_se,res_sp),auc=auc))
return(list(SeSp=data.frame(res_se,res_sp),auc=auc))
}

res<-myroc(elas$status2,elas$elas)
res

plot(1-res$SeSp[,2],res$SeSp[,1],type="s",main="Curva ROC",xlab="1-
Especificidade",ylab="Sensibilidad")
abline(0,1)
legend(x=0.6,y=0.3,legend=c("AUC = ",round(res$auc,2)))

#Bootstrap código

boot<-function(y,x,nboot,conf.level){
dat<-cbind(y,x)
m<-length(y)
cut1<-sort(unique(x))

```



```

n1<-length(cut1)

#mat<-matrix(NA,n1,6)
mat<-array(NA,dim=c(nboot,n1,2))

for (i in 1:nboot){
val<-sample.int(n=m, size = m, replace = TRUE)
dat1<-dat[val,]

for (j in 1:n1){
cut<-cut1[j]

p1<-which(dat1[,2] > cut & dat1[,1] == levels(factor(dat1[,1]))[2] )
p2<-which( dat1[,1] == levels(factor(dat1[,1]))[2] )
mat[i,j,1]<-length(p1)/length(p2)      #Sensitivity
p3<-which(dat1[,2] < cut & dat1[,1] == levels(factor(dat1[,1]))[1] )
p4<-which( dat1[,1] == levels(factor(dat1[,1]))[1] )
mat[i,j,2]<-length(p3)/length(p4)     #specificity
}
}

res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)
res_lise<-rep(0,n1)
res_lsse<-rep(0,n1)

for (j in 1:n1){
    res_lise[j]<-quantile(mat[,j,1],(1-conf.level)/2)
    res_lsse[j]<-quantile(mat[,j,1],(1-conf.level)/2+conf.level)}
res_lisp[j]<-quantile(mat[,j,2],(1-conf.level)/2)
    res_lssp[j]<-quantile(mat[,j,2],(1-conf.level)/2+conf.level)

```

```

return(list(li_Sp=res_lisp,ls_Sp=res_lssp,li_Se=res_lise,ls_Se=res_lsse))
}

```

```

boot2<-function(y,x,nboot,conf.level,cut1){
dat<-cbind(y,x)
m<-length(y)
n1<-length(cut1)

mat<-array(NA,dim=c(nboot,n1,2))

for (i in 1:nboot){
val<-sample.int(n=m, size = m, replace = TRUE)
dat1<-dat[val,]

for (j in 1:n1){
cut<-cut1[j]

p1<-which(dat1[,2] > cut & dat1[,1] == levels(factor(dat1[,1]))[2] )
p2<-which( dat1[,1] == levels(factor(dat1[,1]))[2] )
mat[i,j,1]<-length(p1)/length(p2)      #Sensitivity
p3<-which(dat1[,2] < cut & dat1[,1] == levels(factor(dat1[,1]))[1] )
p4<-which( dat1[,1] == levels(factor(dat1[,1]))[1] )
mat[i,j,2]<-length(p3)/length(p4)      #specificity
}
}

res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)
res_lise<-rep(0,n1)

```

```

res_lsse<-rep(0,n1)

for (j in 1:n1){
  res_lise[j]<-quantile(mat[,j,1],(1-conf.level)/2)
  res_lsse[j]<-quantile(mat[,j,1],(1-conf.level)/2+conf.level)}
res_lisp[j]<-quantile(mat[,j,2],(1-conf.level)/2)
  res_lssp[j]<-quantile(mat[,j,2],(1-conf.level)/2+conf.level)

return(list(li_Sp=res_lisp,ls_Sp=res_lssp,li_Se=res_lise,ls_Se=res_lsse))
}

```

#função secut e função plot.secut. Cálculo da sensibilidade para um ponto de corte escolhido, cálculo da sensibilidade para todos os pontos de corte e representação gráfica

```

secut<-function(y,x,z,cut,conf.level=0.95,conf=FALSE, method="wald", nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

p1<-which(x > cut & y == levels(y)[2] )
p2<-which( y == levels(y)[2] )
res<-length(p1)/length(p2)

if (method=="wald" & conf==TRUE){
li<-res-(qnorm((1+conf.level)/2)*sqrt(res*(1-res)/length(p2)))
ls<-res+(qnorm((1+conf.level)/2)*sqrt(res*(1-res)/length(p2)))

}

if (method=="score" & conf==TRUE){

```

```

li<-(res+(qnorm((1+conf.level)/2)^2)/(2*length(p2))-qnorm((1+conf.level)/2)*sqrt((res*(1-
res)+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2)))/(1+(qnorm((1+conf.level)/2)^2)/
(length(p2))
ls<-(res+(qnorm((1+conf.level)/2)^2)/(2*length(p2))+qnorm((1+conf.level)/2)*sqrt((res*(1-
res)+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2)))/(1+(qnorm((1+conf.level)/2)^2)/
(length(p2))
}

```

```

if (method=="bootstrap" & conf==TRUE){
res2<-boot2(y=y,x=x,nboot=nboot,conf.level=conf.level,cut1=cut)
li<-res2$li_Se
ls<-res2$ls_Se
}

```

```

return(c(res,li,ls,cut))
}

```

```

res11<-secut(y=elas$status2,x=elas$elas,cut=30,conf=TRUE, method="bootstrap",nboot=5)
res11

```

```

plot.secut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

```

```

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

```

```

x1<-sort(unique(x))

```

```

n1<-length(x1)

```

```

res_se<-rep(0,n1)

```

```

res_li<-rep(0,n1)

```

```

res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-secut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)

  res_se[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

res<-cbind(res_se,res_li,res_ls)
rescutwse<-cbind(x1,res)
print(rescutwse)
matplot(x=x1,y=res,type="l",col=c(1,2,2),main="Bandas de Confiança para a Sensibilidade-
Método Bootstrap",xlab="Pontos de Corte",ylab="Sensibilidade")

}

#par(mfrow=c(1,3))
#plot.secure(y=elas$status2,x=elas$elas,conf=TRUE,method="wald")
#plot.secure(y=elas$status2,x=elas$elas,conf=TRUE,method="score")
plot.secure(y=elas$status2,x=elas$elas,conf=TRUE,method="bootstrap",nboot=500)

#função spcut e função plot.spcut. Cálculo da especificidade para um ponto de corte escolhido,
cálculo da especificidade para todos os pontos de corte e representação gráfica

spcut<-function(y,x,z,cut,conf.level=0.95,conf=FALSE, method="wald", nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

p3<-which(x < cut & y == levels(y)[1] )

```

```

p4<-which( y == levels(y)[1] )
ressp<-length(p3)/length(p4)

if (method=="wald" & conf==TRUE){
li<-ressp-(qnorm((1+conf.level)/2)*sqrt(ressp*(1-ressp)/length(p4)))
ls<-ressp+(qnorm((1+conf.level)/2)*sqrt(ressp*(1-ressp)/length(p4)))
}

if (method=="score" & conf==TRUE){
li<-(ressp+(qnorm((1+conf.level)/2)^2)/(2*length(p4))-qnorm((1+conf.level)/2)*sqrt((ressp*(1-
ressp)+(qnorm((1+conf.level)/2)^2)/(4*length(p4)))/length(p4)))/(1+(qnorm((1+conf.level)/2)^2
)/(length(p4)))
ls<-(ressp+(qnorm((1+conf.level)/2)^2)/(2*length(p4))+qnorm((1+conf.level)/2)*sqrt((ressp*(1-
ressp)+(qnorm((1+conf.level)/2)^2)/(4*length(p4)))/length(p4)))/(1+(qnorm((1+conf.level)/2)^2
)/(length(p4)))
}

if (method=="bootstrap" & conf==TRUE){
res<-boot2(y=y,x=x,nboot=nboot,conf.level=conf.level,cut1=cut)
li<-res$li_Sp
ls<-res$ls_Sp
}

return(c(ressp,li,ls,cut))
}

res<-spscut(y=elas$status2,x=elas$elas,z=cut,cut=30,conf=TRUE, method="score")
res2<-spscut(y=elas$status2,x=elas$elas,z=cut,cut=30,conf=TRUE, method="bootstrap",nboot=5)
res

```

```

plot.spcut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

x1<-sort(unique(x))

n1<-length(x1)

res_sp<-rep(0,n1)
res_li<-rep(0,n1)
res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-spcut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)
  res_sp[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

res<-cbind(res_sp,res_li,res_ls)
rescutwsp<-cbind(x1,res)
print(rescutwsp)

matplot(x=x1,y=res,type="l",col=c(1,2,2), main="Bandas de Confiança para a
Especificidade",xlab="Pontos de Corte",ylab="Especificidade")

}

#par(mfrow=c(1,3))
plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="wald")
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="score")
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="bootstrap",nboot=500)

```

função scut: Cálculo da sensibilidade, especificidade e limites de confiança para um ponto de corte escolhido

```
scut<-function(y,x,z,cut,conf.level=0.95,conf=FALSE,method="wald", nboot){
```

```
  if(!is.factor(y)) stop("The variable must be of class factor")
```

```
  if(nlevels(y)!=2) stop("The variable must be dichotomous")
```

```
  p1<-which(x > cut & y == levels(y)[2] )
```

```
  p2<-which( y == levels(y)[2] )
```

```
  res1<-length(p1)/length(p2)
```

```
  p3<-which(x < cut & y == levels(y)[1] )
```

```
  p4<-which( y == levels(y)[1] )
```

```
  ressp<-length(p3)/length(p4)
```

```
  if (method=="wald" & conf==TRUE){
```

```
    lise<-res1-qnorm((1+conf.level)/2)*sqrt(res1*(1-res1)/length(p2))
```

```
    lsse<-res1+qnorm((1+conf.level)/2)*sqrt(res*(1-res1)/length(p2))
```

```
    lisp<-ressp-qnorm((1+conf.level)/2)*sqrt(ressp*(1-ressp)/length(p4))
```

```
    lssp<-ressp+qnorm((1+conf.level)/2)*sqrt(ressp*(1-ressp)/length(p4))
```

```
  }
```

```
  if (method=="score" & conf==TRUE){
```

```
    lise<-(-res1+(qnorm((1+conf.level)/2)^2)/(2*length(p2))-qnorm((1+conf.level)/2)*sqrt((res1*(1-  
res1)+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2)))/(1+(qnorm((1+conf.level)/2)^2)/  
length(p2))
```

```
    lsse<-(-res1+(qnorm((1+conf.level)/2)^2)/(2*length(p2))+qnorm((1+conf.level)/2)*sqrt((res1*(1-  
res1)+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2)))/(1+(qnorm((1+conf.level)/2)^2)/  
length(p2))
```



```

lisp<-(ressp+(qnorm((1+conf.level)/2)^2)/(2*length(p4))-qnorm((1+conf.level)/2)*sqrt((ressp*(1-
ressp)+(qnorm((1+conf.level)/2)^2)/(4*length(p4)))/length(p4)))/(1+(qnorm((1+conf.level)/2)^2
)/ (length(p4)))
lssp<-
(ressp+(qnorm((1+conf.level)/2)^2)/(2*length(p4))+qnorm((1+conf.level)/2)*sqrt((ressp*(1-
ressp)+(qnorm((1+conf.level)/2)^2)/(4*length(p4)))/length(p4)))/(1+(qnorm((1+conf.level)/2)^2
)/ (length(p4)))
}

```

```

if (method=="bootstrap" & conf==TRUE){
res<-boot2(y=y,x=x,nboot=nboot,conf.level=conf.level,cut1=cut)
lise<-res$li_Se
lsse<-res$ls_Se
lisp<-res$li_Sp
lssp<-res$ls_Sp
}

```

```

return(c(cut=cut,sensibilidade=res1,lisensibilidade=lise,lssensibilidade=lsse,especificidade=ressp,
liespecificidade=lisp,lsespecificidade=lssp))
}

```

```

#res<-scut(y=elas$status2,x=elas$elas,cut=30,conf=TRUE, method="score")
res2<-scut(y=elas$status2,x=elas$elas,cut=30,conf=TRUE, method="wald")
res<-scut(y=elas$status2,x=elas$elas,cut=30,conf=TRUE, method="bootstrap",nboot=500)
res
res2

```

```

boot<-function(y,x,nboot,conf.level){
dat<-cbind(y,x)
m<-length(y)

```

```

cut1<-sort(unique(x))
n1<-length(cut1)

#mat<-matrix(NA,n1,6)
mat<-array(NA,dim=c(nboot,n1,2))

for (i in 1:nboot){
val<-sample.int(n=m, size = m, replace = TRUE)
dat1<-dat[val,]

for (j in 1:n1){
cut<-cut1[j]

p1<-which(dat1[,2] > cut & dat1[,1] == levels(factor(dat1[,1]))[2] )
p2<-which( dat1[,1] == levels(factor(dat1[,1]))[2] )
mat[i,j,1]<-length(p1)/length(p2)      #Sensitivity
p3<-which(dat1[,2] < cut & dat1[,1] == levels(factor(dat1[,1]))[1] )
p4<-which( dat1[,1] == levels(factor(dat1[,1]))[1] )
mat[i,j,2]<-length(p3)/length(p4)      #specificity
}
}

res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)
res_lise<-rep(0,n1)
res_lsse<-rep(0,n1)

for (j in 1:n1){
res_lisp[j]<-quantile(mat[,j,2],(1-conf.level)/2)
res_lssp[j]<-quantile(mat[,j,2],(1-conf.level)/2+conf.level)
res_lise[j]<-quantile(mat[,j,1],(1-conf.level)/2)
}

```

```

        res_lsse[j]<-quantile(mat[,j,1],(1-conf.level)/2+conf.level)}

return(list(li_Sp=res_lisp,ls_Sp=res_lssp,li_Se=res_lise,ls_Se=res_lsse))
}

#res<-boot(y=elas$status2,x=elas$elas,nboot=5,conf.level=0.95)

#função SpSe: Lista da sensibilidade e especificidade e dos limites de confiança para cada ponto
de corte

SpSe<-function(y,x,conf.level=0.95,conf=FALSE, method="wald",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

cut1<-sort(unique(x))

n1<-length(cut1)

res_sp<-rep(0,n1)
res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)

res_se<-rep(0,n1)
res_lise<-rep(0,n1)
res_lsse<-rep(0,n1)

for (k in 1:n1){
cut<-cut1[k]

```

```

p1<-which(x > cut & y == levels(y)[2] )
p2<-which( y == levels(y)[2] )
res_se[k]<-length(p1)/length(p2)
p3<-which(x < cut & y == levels(y)[1] )
p4<-which( y == levels(y)[1] )
res_sp[k]<-length(p3)/length(p4)

if (method=="wald" & conf==TRUE){
res_lise[k]<-res_se[k]-qnorm((1+conf.level)/2)*sqrt(res_se[k]*(1-res_se[k])/length(p2))
res_lsse[k]<-res_se[k]+qnorm((1+conf.level)/2)*sqrt(res_se[k]*(1-res_se[k])/length(p2))
res_lisp[k]<-res_sp[k]-qnorm((1+conf.level)/2)*sqrt(res_sp[k]*(1-res_sp[k])/length(p4))
res_lssp[k]<-res_sp[k]+qnorm((1+conf.level)/2)*sqrt(res_sp[k]*(1-res_sp[k])/length(p4))
}

if (method=="score" & conf==TRUE){
res_lise[k]<-res_se[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p2))-
qnorm((1+conf.level)/2)*sqrt((res_se[k]*(1-
res_se[k])+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2))/(1+(qnorm((1+conf.level)/2)^
2)/ (length(p2)))
res_lsse[k]<-
res_se[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p2))+qnorm((1+conf.level)/2)*sqrt((res_se[k]*(
1-
res_se[k])+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2))/(1+(qnorm((1+conf.level)/2)^
2)/ (length(p2)))
res_lisp[k]<-res_sp[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p4))-
qnorm((1+conf.level)/2)*sqrt((res_se[k]*(1-
res_se[k])+qnorm((1+conf.level)/2)^2/(4*length(p4)))/length(p4))/(1+(qnorm((1+conf.level)/2)^
2)/ (length(p4)))
res_lssp[k]<-
res_sp[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p4))+qnorm((1+conf.level)/2)*sqrt((res_se[k]*(

```

```

1-
res_se[k]+qnorm((1+conf.level)/2)^2/(4*length(p4))/length(p4)/(1+(qnorm((1+conf.level)/2)^
2)/ (length(p4)))

}

if (method=="bootstrap" & conf==TRUE){
res<-boot(y=y,x=x,nboot=nboot,conf.level=conf.level)
res_lise<-res$li_Se
res_lsse<-res$ls_Se
res_lisp<-res$li_Sp
res_lssp<-res$ls_Sp
}
}

return(list(Se=res_se, Sp=res_sp, li_Se=res_lise, ls_Se=res_lsse, li_Sp=res_lisp,
ls_Sp=res_lssp, cutval=cut1))

}

res2<-SpSe(y=elas$status2,x=elas$elas,conf.level=0.95,conf=TRUE,
method="bootstrap",nboot=500)

res4<-SpSe(y=elas$status2,x=elas$elas,conf.level=0.95,conf=TRUE, method="wald",nboot=5)

res5<-SpSe(y=elas$status2,x=elas$elas,conf.level=0.95,conf=TRUE, method="score",nboot=5)

plot(1-res2$Sp,res2$Se,type="s",xlim=c(0,1),ylim=c(0,1),xlab="1-Especificidade",
ylab="Sensitivity")
abline(0,1)

```

```

matplot(cbind(1-res2$Sp,1-res2$li_Sp,1-res2$ls_Sp),cbind(res2$Se,res2$li_Se,res2$ls_Se),
type=c("s","s","s"), lty=c(1,2,2), col=c(1,2,2), main="Curva ROC com Bandas de Confiança 2d -
Bootstrap",xlab="Falsos Positivo",ylab="Sensibilidade")
spe1<-scatterplot3d(x=1-res2$Sp,y=res2$cutval, z=res2$Se, type = "h", color = "dimgray",angle
= 230, scale.y = 0.7, pch = 16,xlab="Falsos Positivos", ylab="Pontos
Corte",zlab="Sensibilidade", main="Gráfico 3d Curva ROC - Método Bootstrap")
spe1$points3d(x=1-res2$li_Sp,y=res2$cutval, z=res2$li_Se,type="h",col="red")
spe1$points3d(x=1-res2$res.ls_Sp,y=res2$res.cutval, z=res2$res.ls_Se,type="h",col="red")

```

```

par(mfrow=c(1,2))
matplot(cbind(1-res4$Sp,1-res4$li_Sp,1-res4$ls_Sp),cbind(res4$Se,res4$li_Se,res4$ls_Se),
type=c("s","s","s"), lty=c(1,2,2), col=c(1,2,2), main="Curva ROC com Bandas de Confiança 2d -
Wald",xlab="Falsos Positivos",ylab="Sensibilidade")
spe2<-scatterplot3d(x=1-res4$Sp,y=res4$cutval, z=res4$Se, type = "h", color = "dimgray",angle
= 230, scale.y = 0.7, pch = 16,xlab="Falsos Positivos", ylab="Pontos
Corte",zlab="Sensibilidade", main="Gráfico 3d Curva ROC - Método Wald")
spe2$points3d(x=1-res4$li_Sp,y=res4$cutval, z=res4$li_Se,type="h",col="red")
spe2$points3d(x=1-res4$ls_Sp,y=res4$cutval, z=res4$ls_Se,type="h",col="red")

```

```

par(mfrow=c(1,2))
matplot(cbind(1-res5$Sp,1-res5$li_Sp,1-res5$ls_Sp),cbind(res5$Se,res5$li_Se,res5$ls_Se),
type=c("s","s","s"), lty=c(1,2,2), col=c(1,2,2), main="Curva ROC com Bandas de Confiança 2d -
Score",xlab="Falsos Positivos",ylab="Sensibilidade")
spe3<-scatterplot3d(x=1-res5$Sp,y=res5$cutval, z=res5$Se, type = "h", color = "dimgray",angle
= 230, scale.y = 0.7, pch = 16,xlab="Falsos Positivos", ylab="Pontos
Corte",zlab="Sensibilidade", main="Gráfico 3d Curva ROC - Método Score")
spe3$points3d(x=1-res5$li_Sp,y=res5$cutval, z=res5$li_Se,type="h",col="red")
spe3$points3d(x=1-res5$ls_Sp,y=res5$cutval, z=res5$ls_Se,type="h",col="red")

```

#####Gráfico como se encontra na tese do plot.secut e do plot.spcut

```

plot.secut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

x1<-sort(unique(x))

n1<-length(x1)

res_se<-rep(0,n1)
res_li<-rep(0,n1)
res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-secut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)

  res_se[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

res<-cbind(res_se,res_li,res_ls)
rescutwse<-cbind(x1,res)
print(rescutwse)
matplot(x=x1,y=res,type="",col=c(1,2,2),main="BC Sensibilidade: Método Wald",xlab="Pontos
de Corte",ylab="Sensibilidade")

}

layout(matrix(c(2,3,1,1), 2, 2,byrow=TRUE), respect=TRUE)

```

```

plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="wald")
#plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="score")
#plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="bootstrap",nboot=500)

plot.secut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

x1<-sort(unique(x))

n1<-length(x1)

res_se<-rep(0,n1)
res_li<-rep(0,n1)
res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-secut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)

  res_se[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

res<-cbind(res_se,res_li,res_ls)
rescutwse<-cbind(x1,res)
print(rescutwse)
matplot(x=x1,y=res,type="l",col=c(1,2,2),main="BC Sensibilidade: Método Score",xlab="Pontos
de Corte",ylab="Sensibilidade")

}

```



```

#par(mfrow=c(1,3))
#plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="wald")
plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="score")
#plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="Score",nboot=500)

plot.secut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

x1<-sort(unique(x))

n1<-length(x1)

res_se<-rep(0,n1)
res_li<-rep(0,n1)
res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-secut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)

  res_se[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

res<-cbind(res_se,res_li,res_ls)
rescutwse<-cbind(x1,res)
print(rescutwse)
matplot(x=x1,y=res,type="",col=c(1,2,2),main="BC Sensibilidade: Método
Bootstrap",xlab="Pontos de Corte",ylab="Sensibilidade")

```

```
}
```

```
#par(mfrow=c(1,3))
```

```
#plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="wald")
```

```
#plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="score")
```

```
plot.secut(y=elas$status2,x=elas$elas,conf=TRUE,method="bootstrap",nboot=500)
```

```
#####
```

```
plot.spcut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){
```

```
  if(!is.factor(y)) stop("The variable must be of class factor")
```

```
  if(nlevels(y)!=2) stop("The variable must be dichotomous")
```

```
  x1<-sort(unique(x))
```

```
  n1<-length(x1)
```

```
  res_sp<-rep(0,n1)
```

```
  res_li<-rep(0,n1)
```

```
  res_ls<-rep(0,n1)
```

```
  for(k in 1:n1){
```

```
    myres<-spcut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)
```

```
    res_sp[k]<-myres[1]
```

```
    res_li[k]<-myres[2]
```

```
    res_ls[k]<-myres[3]}
```

```
  res<-cbind(res_sp,res_li,res_ls)
```

```
  rescutwsp<-cbind(x1,res)
```

```

print(rescutwsp)

matplot(x=x1,y=res,type="",col=c(1,2,2), main="BC Especificidade: Método Wald",xlab="Pontos
de Corte",ylab="Especificidade")

}

layout(matrix(c(2,3,1,1), 2, 2,byrow=TRUE), respect=TRUE)
plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="wald")
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="score")
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="bootstrap",nboot=500)

plot.spcut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

x1<-sort(unique(x))

n1<-length(x1)

res_sp<-rep(0,n1)
res_li<-rep(0,n1)
res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-spcut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)
  res_sp[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

```

```

res<-cbind(res_sp,res_li,res_ls)
rescutwsp<-cbind(x1,res)
print(rescutwsp)

matplot(x=x1,y=res,type="l",col=c(1,2,2), main="BC Especificidade: Método Score",xlab="Pontos
de Corte",ylab="Especificidade")

}
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="wald")
plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="score")
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="bootstrap",nboot=500)

plot.spcut<-function(y,x,z,conf.level=0.95,conf=FALSE, method="score",nboot){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

x1<-sort(unique(x))

n1<-length(x1)

res_sp<-rep(0,n1)
res_li<-rep(0,n1)
res_ls<-rep(0,n1)

for(k in 1:n1){
  myres<-spcut(y,x,z,cut=x1[k],conf=conf,conf.level=conf.level,method=method,nboot)
  res_sp[k]<-myres[1]
  res_li[k]<-myres[2]
  res_ls[k]<-myres[3]}

```

```

res<-cbind(res_sp,res_li,res_ls)
rescutwsp<-cbind(x1,res)
print(rescutwsp)

matplot(x=x1,y=res,type="l",col=c(1,2,2), main="BC Especificidade: Método
Bootstrap",xlab="Pontos de Corte",ylab="Especificidade")

}
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="wald")
#plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="score")
plot.spcut(y=elas$status2,x=elas$elas,conf=TRUE, method="bootstrap",nboot=500)

```

#ypindex: Lista com os valores do Índice de Youden, da precisão, da sensibilidade e da especificidade para cada ponto de corte

```
ypindex<-function(y,x){
```

```
cut1<-sort(unique(x))
```

```
n1<-length(cut1)
```

```
res_sp<-rep(0,n1)
```

```
res_se<-rep(0,n1)
```

```
yd<-rep(0,n1)
```

```
yli<-rep(0,n1)
```

```
yls<-rep(0,n1)
```

```
cc<-rep(0,n1)
```

```
total<-rep(0,n1)
```

```
accuracy<-rep(0,n1)
```

```
a<-rep(0,n1)
```

```
b<-rep(0,n1)
```

```

for (k in 1:n1){
cut<-cut1[k]

p1<-which(x > cut & y == levels(y)[2])
p2<-which(y == levels(y)[2])
res_se[k]<-length(p1)/length(p2)
p3<-which(x < cut & y == levels(y)[1])
p4<-which(y == levels(y)[1])
res_sp[k]<-length(p3)/length(p4)
yd[k]<-(res_se[k]+res_sp[k])-1
a[k]<-length(p1)+length(p3)
b[k]<-length(p2)+length(p4)
accuracy[k]<-a[k]/b[k]

}
return(list(res_se=res_se,res_sp=res_sp,accuracy=accuracy,yd=yd,cutval=cut1))

}

alfa3<-ypindex(y=elas$status2,x=elas$elas)
alfa3

library(scatterplot3d)

par(mfrow=c(1,2))
scatterplot3d(x=1-alfa3$res_sp,y=alfa3$res_se,z=alfa3$accuracy, type = "h", color =
"blue",angle = 50, scale.y = 0.7, pch = 16,xlab="1-Especficidade",
ylab="Sensibilidade",zlab="Precisão", main="Precisão x Curva ROC")

```

```

scatterplot3d(x=1-alfa3$res_sp,y=alfa3$res_se,z=alfa3$yd, type = "h", color = "orange",angle =
50, scale.y = 0.7, pch = 16,xlab="1-Especificidade", ylab="Sensibilidade",zlab="Índice de
Youden", main="Índice de Youden x Curva ROC")
alfa2<-ypindex(y=elas$status2,x=elas$elas)
alfa2<-data.frame(alfa2$res_se,alfa2$res_sp,alfa2$accuracy,alfa2$yd,alfa2$cutval)
alfa2

```

#função slife: Custo unitário esperado e Retorno unitário esperado para cada nível de sensibilidade/especificidade/ponto de corte

```

slife<-function(y,x,pd,custse,retse,custsp,retsp,custfp,retfp,custfn,retfn){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

cut1<-sort(unique(x))

n1<-length(cut1)

res_sp<-rep(0,n1)
res_se<-rep(0,n1)
Costunit<-rep(0,n1)
Retunit<-rep(0,n1)
pd<-pd
custse<-custse
retse<-retse
custsp<-custsp
retsp<-retsp
custfp<-custfp
retfp<-retfp
custfn<-custfn

```

```

retfn<-retfn

for (k in 1:n1){
cut<-cut1[k]

p1<-which(x > cut & y == levels(y)[2] )
p2<-which( y == levels(y)[2] )
res_se[k]<-length(p1)/length(p2)
p3<-which(x < cut & y == levels(y)[1] )
p4<-which( y == levels(y)[1] )
res_sp[k]<-length(p3)/length(p4)
Costunit[k]<-(((res_se[k]*pd*custse)+(res_sp[k]*((1-pd)*(custsp)))+((1-res_sp[k])*((1-
pd)*(custfp)))+(1-res_se[k])*(pd)*(custfn)))
Retunit[k]<-(((res_se[k]*pd*retse)+(res_sp[k]*((1-pd)*(retsp)))+((1-res_sp[k])*((1-
pd)*(retfp)))+(1-res_se[k])*(pd)*(retfn)))
}

return(list(Se=res_se, Sp=res_sp, cutval=cut1, Costunit=Costunit, Retunit=Retunit))

}

res<-slife(y=elas$status2,x=elas$elas,pd=0.28,custse=10,retse=0.30,custfp=10,retfp=-
0.1,custsp=5,retsp=-0.01,custfn=10,retfn=-0.3)

scatterplot3d(x=res$cutval,y=res$Costunit, z=res$Retunit, type = "h", color = "red",angle = 135,
scale.y = 0.7, pch = 16,xlab="Pontos de Corte", ylab="Custo Esperado",zlab="Retorno
Esperado",xlim=c(0,50),ylim=c(8.5,9),zlim=c(0.02,0.03), main="Pontos de Corte: Custo e
Retorno Esperado")

```



```
scatterplot3d(x=res$cutval,y=res$Costunit, z=res$Retunit, type = "h", color = "red",angle = 135,
scale.y = 0.7, pch = 16,xlab="Pontos de Corte", ylab="Custo Esperado",zlab="Retorno
Esperado", main="Pontos de Corte: Custo e Retorno Esperado")
```

```
par(mfrow=c(2,2))
```

```
plot(res$Costunit,res$Retunit,type="l",col="dark orange",xlab="Custo Esperado", ylab="Retorno
Esperado")
```

```
abline(h=0,col="red")
```

```
abline(h=0.028,col="red")
```

```
abline(v=7.62,col="red")
```

```
abline(v=8.72,col="red")
```

```
plot(res$Costunit,res$Retunit,type="l",col="dark orange",ylim=c(-
0.02,0.03),xlim=c(7.6,9.5),xlab="Custo Esperado", ylab="Retorno Esperado")
```

```
abline(h=0,col="red")
```

```
abline(h=0.028,col="red")
```

```
abline(v=7.62,col="red")
```

```
abline(v=8.72,col="red")
```

```
scatterplot3d(x=res$cutval,y=res$Costunit, z=res$Retunit, type = "h", color = "red",angle = 135,
scale.y = 0.7, pch = 16,xlab="Pontos de Corte", ylab="Custo Esperado",zlab="Retorno
Esperado", main="Pontos de Corte: Custo e Retorno Esperado")
```

```
scatterplot3d(x=res$cutval,y=res$Costunit, z=res$Retunit, type = "h", color = "red",angle = 135,
scale.y = 0.7, pch = 16,xlab="Pontos de Corte", ylab="Custo Esperado",zlab="Retorno
Esperado",xlim=c(0,50),ylim=c(8.5,9),zlim=c(0.02,0.03), main="Pontos de Corte: Custo e
Retorno Esperado")
```

```
#####
```

```
#Estudo de simulação ROC
```

```
#Objectivo: comparar metodologias para obter os IC para a Sensibilidade (Se)
```

```

#e a Especificidade (Sp)
#Métodos: Score, Wald e Bootstrap
#-----

SpSe<-function(y,x,conf.level=0.95,conf=FALSE, method="wald",nboot,cut1){

if(!is.factor(y)) stop("The variable must be of class factor")
if(nlevels(y)!=2) stop("The variable must be dichotomous")

n1<-length(cut1)

res_sp<-rep(0,n1)
res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)

res_se<-rep(0,n1)
res_lise<-rep(0,n1)
res_lsse<-rep(0,n1)

for (k in 1:n1){
cut<-cut1[k]

p1<-which(x > cut & y == levels(y)[2] )
p2<-which( y == levels(y)[2] )
res_se[k]<-length(p1)/length(p2)
p3<-which(x < cut & y == levels(y)[1] )
p4<-which( y == levels(y)[1] )
res_sp[k]<-length(p3)/length(p4)

if (method=="wald" & conf==TRUE){
res_lise[k]<-res_se[k]-qnorm((1+conf.level)/2)*sqrt(res_se[k]*(1-res_se[k])/length(p2))

```

```

res_lsse[k]<-res_se[k]+qnorm((1+conf.level)/2)*sqrt(res_se[k]*(1-res_se[k])/length(p2))
res_lisp[k]<-res_sp[k]-qnorm((1+conf.level)/2)*sqrt(res_sp[k]*(1-res_sp[k])/length(p4))
res_lssp[k]<-res_sp[k]+qnorm((1+conf.level)/2)*sqrt(res_sp[k]*(1-res_sp[k])/length(p4))
}

```

```

if (method=="score" & conf==TRUE){
res_lise[k]<-res_se[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p2))-
qnorm((1+conf.level)/2)*sqrt((res_se[k]*(1-
res_se[k])+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2))/(1+(qnorm((1
+conf.level)/2)^2)/ (length(p2)))
res_lsse[k]<-
res_se[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p2))+qnorm((1+conf.level)/2)*sqrt((res_se[k]*(
1-res_se[k])+qnorm((1+conf.level)/2)^2/(4*length(p2)))/length(p2))/(1+(qnorm((1
+conf.level)/2)^2)/ (length(p2)))
res_lisp[k]<-res_sp[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p4))-
qnorm((1+conf.level)/2)*sqrt((res_sp[k]*(1-
res_sp[k])+qnorm((1+conf.level)/2)^2/(4*length(p4)))/length(p4))/(1+(qnorm((1
+conf.level)/2)^2)/ (length(p4)))
res_lssp[k]<-
res_sp[k]+(qnorm((1+conf.level)/2)^2)/(2*length(p4))+qnorm((1+conf.level)/2)*sqrt((res_sp[k]*(
1-res_sp[k])+qnorm((1+conf.level)/2)^2/(4*length(p4)))/length(p4))/(1+(qnorm((1
+conf.level)/2)^2)/ (length(p4)))
}

```

```

if (method=="bootstrap" & conf==TRUE){
res<-boot2(y=y,x=x,nboot=nboot,conf.level=conf.level,cut1=cut)
res_lise<-res$li_Se
res_lsse<-res$ls_Se
res_lisp<-res$li_Sp

```

```

res_lssp<-res$ls_Sp
}
}
return(list(Se=res_se, Sp=res_sp, li_Se=res_lise, ls_Se=res_lsse, li_Sp=res_lisp,
ls_Sp=res_lssp, cutval=cut1))

}

```

```

boot<-function(y,x,nboot,conf.level){
dat<-cbind(y,x)
m<-length(y)
cut1<-sort(unique(x))
n1<-length(cut1)

#mat<-matrix(NA,n1,6)
mat<-array(NA,dim=c(nboot,n1,2))

for (i in 1:nboot){
val<-sample.int(n=m, size = m, replace = TRUE)
dat1<-dat[val,]

for (j in 1:n1){
cut<-cut1[j]

p1<-which(dat1[,2] > cut & dat1[,1] == levels(factor(dat1[,1]))[2] )
p2<-which( dat1[,1] == levels(factor(dat1[,1]))[2] )
mat[i,j,1]<-length(p1)/length(p2)      #Sensitivity
p3<-which(dat1[,2] < cut & dat1[,1] == levels(factor(dat1[,1]))[1] )
p4<-which( dat1[,1] == levels(factor(dat1[,1]))[1] )
mat[i,j,2]<-length(p3)/length(p4)     #specificity

```

```

}
}

res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)
res_lise<-rep(0,n1)
res_lsse<-rep(0,n1)

for (j in 1:n1){
  res_lise[j]<-quantile(mat[,j,1],(1-conf.level)/2)
  res_lsse[j]<-quantile(mat[,j,1],(1-conf.level)/2+conf.level)}
res_lisp[j]<-quantile(mat[,j,2],(1-conf.level)/2)
  res_lssp[j]<-quantile(mat[,j,2],(1-conf.level)/2+conf.level)

return(list(li_Sp=res_lisp,ls_Sp=res_lssp,li_Se=res_lise,ls_Se=res_lsse))
}

```

```

boot2<-function(y,x,nboot,conf.level,cut1){
dat<-cbind(y,x)
m<-length(y)
n1<-length(cut1)

#mat<-matrix(NA,n1,6)
mat<-array(NA,dim=c(nboot,n1,2))

for (i in 1:nboot){
val<-sample.int(n=m, size = m, replace = TRUE)
dat1<-dat[val,]

```

```

for (j in 1:n1){
cut<-cut1[j]

p1<-which(dat1[,2] > cut & dat1[,1] == levels(factor(dat1[,1]))[2] )
p2<-which( dat1[,1] == levels(factor(dat1[,1]))[2] )
mat[i,j,1]<-length(p1)/length(p2)      #Sensitivity
p3<-which(dat1[,2] < cut & dat1[,1] == levels(factor(dat1[,1]))[1] )
p4<-which( dat1[,1] == levels(factor(dat1[,1]))[1] )
mat[i,j,2]<-length(p3)/length(p4)      #specificity
}
}

res_lisp<-rep(0,n1)
res_lssp<-rep(0,n1)
res_lise<-rep(0,n1)
res_lsse<-rep(0,n1)

for (j in 1:n1){
  res_lise[j]<-quantile(mat[,j,1],(1-conf.level)/2)
  res_lsse[j]<-quantile(mat[,j,1],(1-conf.level)/2+conf.level)}
res_lisp[j]<-quantile(mat[,j,2],(1-conf.level)/2)
  res_lssp[j]<-quantile(mat[,j,2],(1-conf.level)/2+conf.level)

return(list(li_Sp=res_lisp,ls_Sp=res_lssp,li_Se=res_lise,ls_Se=res_lsse))
}

#-----
n<-100/200/500/1000  #tamanho amostral
M<-1000  #n° de réplicas
p<-0.2/0.4/0.6/0.8  #proporção de doentes
cut<-c(5,10,15,20)

```

```

nc<-length(cut)
Se<-rep(1,nc)
Sp<-rep(1,nc)

for (k in 1:nc){
  Se[k]<- 1-pnorm(cut[k],20,8) #true Se
  Sp[k]<-pnorm(cut[k],10,4) #true Sp

mydata <- matrix(NA,n,2)
res_score_Se <- matrix(0,M,nc)
res_score_Sp <- matrix(0,M,nc)
res_wald_Se <- matrix(0,M,nc)
res_wald_Sp <- matrix(0,M,nc)
res_boot_Se <- matrix(0,M,nc)
res_boot_Sp <- matrix(0,M,nc)

for (k in 1:M){

for (j in 1:n){
  pp<-rbinom(1,1,p)
  if (pp==0) {mydata[j,1]<-rnorm(1,10,4); mydata[j,2]<-0}
  else {mydata[j,1]<-rnorm(1,20,8); mydata[j,2]<-1}
  }

for (i in 1:nc){
  res1<-SpSe(y=as.factor(mydata[,2]),x=mydata[,1],conf.level=0.95,conf=TRUE,
method="wald",cut1=cut[i])
  if (Se[i]>=res1$li_Se & Se[i]<=res1$ls_Se) res_wald_Se[k,i]<-1
  if (Sp[i]>=res1$li_Sp & Sp[i]<=res1$ls_Sp) res_wald_Sp[k,i]<-1

```

```

res2<-SpSe(y=as.factor(mydata[,2]),x=mydata[,1],conf.level=0.95,conf=TRUE,
method="score",cut1=cut[i])
if (Se[i]>=res2$li_Se & Se[i]<=res2$ls_Se) res_score_Se[k,i]<-1
if (Sp[i]>=res2$li_Sp & Sp[i]<=res2$ls_Sp) res_score_Sp[k,i]<-1

res3<-SpSe(y=as.factor(mydata[,2]),x=mydata[,1],conf.level=0.95,conf=TRUE,
cut1=cut[i],method="bootstrap",nboot=500)
if (Se[i]>=res3$li_Se & Se[i]<=res3$ls_Se) res_boot_Se[k,i]<-1
if (Sp[i]>=res3$li_Sp & Sp[i]<=res3$ls_Sp) res_boot_Sp[k,i]<-1
    }
}

```

#Os valores médios das matrizes em baixos indicam as coberturas para cada ponto de corte

```

summary(res_wald_Se)
summary(res_wald_Sp)
summary(res_score_Se)
summary(res_score_Sp)
summary(res_boot_Se)
summary(res_boot_Sp)

```


ANEXO B. Versão não reduzida das tabelas com os valores de Sensibilidade, Especificidade, limites de confiança e pontos de corte

Tabela B1.1. Tabela com a informação completa das tabelas 2, 5,6 e 7.

		Limites de Confiança													
		Método Wald					Score				Bootstrap (nboot=500)				
PC	Se	Sp	li_Se	ls_Se	li_Sp	ls_Sp	li_Se	ls_Se	li_Sp	ls_Sp	li_Se	ls_Se	li_Sp	ls_Sp	
5	1.00	0.00	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.08	1.00	1.00	0.00	0.00	
6	1.00	0.02	1.00	1.00	0.00	0.07	1.00	1.00	0.03	0.10	1.00	1.00	0.00	0.07	
10	1.00	0.04	1.00	1.00	0.00	0.10	1.00	1.00	0.05	0.13	1.00	1.00	0.00	0.11	
12	1.00	0.11	1.00	1.00	0.00	0.20	1.00	1.00	0.11	0.19	1.00	1.00	0.03	0.21	
13	0.98	0.16	0.95	1.00	0.05	0.26	0.97	1.00	0.14	0.25	0.95	1.00	0.06	0.27	
14	0.97	0.20	0.93	1.00	0.08	0.32	0.95	1.00	0.18	0.30	0.93	1.00	0.10	0.32	
15	0.97	0.20	0.93	1.00	0.08	0.32	0.95	1.00	0.18	0.30	0.93	1.00	0.09	0.32	
16	0.97	0.27	0.93	1.00	0.14	0.40	0.95	1.00	0.25	0.37	0.93	1.00	0.15	0.41	
18	0.97	0.29	0.93	1.00	0.16	0.42	0.95	1.00	0.27	0.39	0.94	1.00	0.15	0.43	
20	0.97	0.31	0.93	1.00	0.18	0.45	0.95	1.00	0.29	0.41	0.93	1.00	0.17	0.44	
21	0.96	0.36	0.92	1.00	0.22	0.50	0.94	1.00	0.33	0.46	0.91	0.99	0.21	0.50	
22	0.94	0.38	0.89	0.99	0.24	0.52	0.91	1.00	0.34	0.50	0.88	0.98	0.25	0.51	
24	0.93	0.38	0.88	0.98	0.24	0.52	0.89	1.00	0.34	0.50	0.88	0.98	0.24	0.52	
25	0.90	0.38	0.83	0.96	0.24	0.52	0.85	0.98	0.33	0.51	0.84	0.96	0.24	0.51	
26	0.85	0.38	0.78	0.92	0.24	0.52	0.80	0.94	0.32	0.52	0.78	0.91	0.25	0.53	
27	0.81	0.40	0.73	0.89	0.26	0.54	0.76	0.91	0.33	0.55	0.72	0.89	0.26	0.54	
29	0.80	0.47	0.72	0.88	0.32	0.61	0.74	0.90	0.40	0.62	0.71	0.87	0.33	0.61	
30	0.77	0.49	0.69	0.85	0.34	0.63	0.71	0.87	0.41	0.65	0.68	0.86	0.34	0.63	
31	0.76	0.49	0.68	0.85	0.34	0.63	0.70	0.86	0.41	0.65	0.66	0.84	0.33	0.65	
32	0.74	0.51	0.65	0.83	0.37	0.66	0.67	0.85	0.43	0.68	0.66	0.82	0.35	0.66	
33	0.73	0.53	0.64	0.82	0.39	0.68	0.66	0.84	0.45	0.70	0.64	0.81	0.40	0.69	

Legenda: Esta tabela retorna a lista completa da sensibilidade (Se), especificidade (Sp), limite de confiança superior (ls) e inferior (li) para os três métodos e ponto de corte (PC).

Tabela B1.2. Tabela com a informação completa das tabelas 2, 5,6 e 7.

Limites de Confiança														
PC	Método Wald						Score				Bootstrap (nboot=500)			
	Se	Sp	li_Se	ls_Se	li_Sp	ls_Sp	li_Se	ls_Se	li_Sp	ls_Sp	li_Se	ls_Se	li_Sp	ls_Sp
34	0.70	0.56	0.61	0.79	0.41	0.70	0.63	0.81	0.47	0.73	0.61	0.79	0.40	0.70
35	0.70	0.56	0.61	0.79	0.41	0.70	0.63	0.81	0.47	0.73	0.60	0.80	0.41	0.70
36	0.69	0.60	0.59	0.78	0.46	0.74	0.62	0.80	0.51	0.77	0.59	0.78	0.45	0.74
37	0.68	0.67	0.58	0.77	0.53	0.80	0.61	0.79	0.58	0.84	0.59	0.77	0.52	0.82
38	0.65	0.67	0.55	0.74	0.53	0.80	0.57	0.76	0.57	0.84	0.54	0.73	0.53	0.80
39	0.59	0.67	0.50	0.69	0.53	0.80	0.52	0.71	0.57	0.85	0.49	0.69	0.53	0.80
39.5	0.59	0.71	0.50	0.69	0.58	0.84	0.52	0.71	0.62	0.89	0.48	0.69	0.57	0.84
40	0.54	0.73	0.44	0.64	0.60	0.86	0.46	0.66	0.64	0.92	0.44	0.64	0.60	0.87
41	0.51	0.73	0.41	0.61	0.60	0.86	0.43	0.63	0.64	0.92	0.41	0.61	0.60	0.86
42	0.51	0.76	0.41	0.61	0.63	0.88	0.43	0.63	0.66	0.94	0.41	0.60	0.62	0.87
43	0.49	0.82	0.39	0.59	0.71	0.93	0.41	0.61	0.72	1.00	0.38	0.60	0.70	0.93
44	0.48	0.84	0.38	0.58	0.74	0.95	0.40	0.60	0.75	1.00	0.39	0.56	0.74	0.94
45	0.44	0.84	0.34	0.54	0.74	0.95	0.36	0.55	0.75	1.00	0.35	0.54	0.73	0.93
46	0.43	0.84	0.33	0.53	0.74	0.95	0.35	0.54	0.75	1.00	0.33	0.53	0.73	0.94
48	0.36	0.87	0.27	0.46	0.77	0.97	0.29	0.48	0.77	1.00	0.26	0.46	0.77	0.96
49	0.36	0.87	0.27	0.46	0.77	0.97	0.29	0.48	0.77	1.00	0.26	0.46	0.76	0.95
50	0.34	0.91	0.25	0.44	0.83	0.99	0.27	0.46	0.82	1.00	0.24	0.43	0.81	0.98
51	0.33	0.91	0.24	0.43	0.83	0.99	0.26	0.45	0.82	1.00	0.24	0.43	0.83	0.98
53	0.32	0.93	0.23	0.42	0.86	1.00	0.25	0.43	0.84	1.00	0.23	0.42	0.86	1.00
54	0.31	0.96	0.22	0.41	0.90	1.00	0.24	0.42	0.87	1.00	0.22	0.41	0.88	1.00
56	0.30	0.96	0.21	0.39	0.90	1.00	0.23	0.41	0.87	1.00	0.22	0.39	0.88	1.00
57	0.29	1.00	0.20	0.38	1.00	1.00	0.22	0.40	0.91	1.00	0.20	0.38	1.00	1.00
58	0.28	1.00	0.19	0.37	1.00	1.00	0.21	0.39	0.92	1.00	0.20	0.36	1.00	1.00
59	0.27	1.00	0.18	0.36	1.00	1.00	0.20	0.38	0.92	1.00	0.19	0.35	1.00	1.00
60	0.25	1.00	0.16	0.34	1.00	1.00	0.18	0.36	0.92	1.00	0.17	0.34	1.00	1.00

Legenda: Esta tabela retorna a lista completa da sensibilidade (Se), especificidade (Sp), limite de confiança superior (ls) e inferior (li) para os três métodos e ponto de corte (PC).

Tabela B1.3. Tabela com a informação completa das tabelas 2, 5,6 e 7.

Limites de Confiança														
PC	Método Wald						Score				Bootstrap (nboot=500)			
	Se	Sp	li_Se	ls_Se	li_Sp	ls_Sp	li_Se	ls_Se	li_Sp	ls_Sp	li_Se	ls_Se	li_Sp	ls_Sp
61	0.24	1.00	0.15	0.32	1.00	1.00	0.18	0.34	0.92	1.00	0.16	0.32	1.00	1.00
63	0.22	1.00	0.14	0.30	1.00	1.00	0.16	0.32	0.92	1.00	0.14	0.30	1.00	1.00
64	0.21	1.00	0.13	0.29	1.00	1.00	0.15	0.31	0.93	1.00	0.14	0.29	1.00	1.00
65	0.18	1.00	0.10	0.25	1.00	1.00	0.12	0.27	0.93	1.00	0.10	0.25	1.00	1.00
66	0.15	1.00	0.08	0.22	1.00	1.00	0.10	0.24	0.94	1.00	0.08	0.22	1.00	1.00
67	0.14	1.00	0.07	0.20	1.00	1.00	0.09	0.22	0.94	1.00	0.08	0.21	1.00	1.00
70	0.13	1.00	0.06	0.19	1.00	1.00	0.08	0.21	0.95	1.00	0.06	0.19	1.00	1.00
72	0.11	1.00	0.05	0.18	1.00	1.00	0.07	0.20	0.95	1.00	0.06	0.18	1.00	1.00
76	0.10	1.00	0.04	0.17	1.00	1.00	0.06	0.19	0.95	1.00	0.04	0.18	1.00	1.00
83	0.08	1.00	0.03	0.14	1.00	1.00	0.05	0.16	0.96	1.00	0.03	0.14	1.00	1.00
86	0.06	1.00	0.01	0.11	1.00	1.00	0.03	0.13	0.97	1.00	0.01	0.11	1.00	1.00
90	0.05	1.00	0.01	0.10	1.00	1.00	0.03	0.12	0.97	1.00	0.01	0.10	1.00	1.00
116	0.04	1.00	0.00	0.08	1.00	1.00	0.02	0.10	0.98	1.00	0.01	0.08	1.00	1.00
126	0.03	1.00	0.00	0.07	1.00	1.00	0.01	0.09	0.98	1.00	0.00	0.06	1.00	1.00
134	0.02	1.00	0.00	0.05	1.00	1.00	0.01	0.07	0.99	1.00	0.00	0.05	1.00	1.00
156	0.01	1.00	0.00	0.03	1.00	1.00	0.00	0.06	0.99	1.00	0.00	0.03	1.00	1.00
163	0.00	1.00	0.00	0.00	1.00	1.00	0.00	0.04	1.00	1.00	0.00	0.00	1.00	1.00

Legenda: Esta tabela retorna a lista completa da sensibilidade (Se), especificidade (Sp), limite de confiança superior (ls) e inferior (li) para os três métodos e ponto de corte (PC).

ANEXO C. Versão não reduzida da tabela com os valores da Precisão e do Índice de Youden

Tabela C.1.1 Tabela com a informação completa da tabela 3.

Ponto de Corte	Sensibilidade	Especificidade	Precisão	Índice de Youden
5	1.00	0.00	0.68	0.00
6	1.00	0.02	0.69	0.02
10	1.00	0.04	0.70	0.04
12	1.00	0.11	0.72	0.11
13	0.98	0.16	0.72	0.13
14	0.97	0.20	0.72	0.17
15	0.97	0.20	0.72	0.17
16	0.97	0.27	0.74	0.24
18	0.97	0.29	0.75	0.26
20	0.97	0.31	0.76	0.28
21	0.96	0.36	0.77	0.31
22	0.94	0.38	0.76	0.32
24	0.93	0.38	0.75	0.30
25	0.90	0.38	0.73	0.27
26	0.85	0.38	0.70	0.23
27	0.81	0.40	0.68	0.21
29	0.80	0.47	0.70	0.27
30	0.77	0.49	0.68	0.26
31	0.76	0.49	0.67	0.25
32	0.74	0.51	0.67	0.25
33	0.73	0.53	0.67	0.26
34	0.70	0.56	0.65	0.25
35	0.70	0.56	0.65	0.25

Tabela C.1.2 Tabela com a informação completa da tabela 3.

Ponto de Corte	Sensibilidade	Especificidade	Precisão	Índice de Youden
36	0.69	0.60	0.66	0.29
37	0.68	0.67	0.67	0.34
38	0.65	0.67	0.65	0.31
39	0.59	0.67	0.62	0.26
39.50	0.59	0.71	0.63	0.30
40	0.54	0.73	0.60	0.28
41	0.51	0.73	0.58	0.24
42	0.51	0.76	0.59	0.27
43	0.49	0.82	0.60	0.31
44	0.48	0.84	0.60	0.32
45	0.44	0.84	0.57	0.28
46	0.43	0.84	0.56	0.27
48	0.36	0.87	0.52	0.23
49	0.36	0.87	0.52	0.23
50	0.34	0.91	0.52	0.25
51	0.33	0.91	0.52	0.24
53	0.32	0.93	0.52	0.26
54	0.31	0.96	0.52	0.27
56	0.30	0.96	0.51	0.26
57	0.29	1.00	0.52	0.29
58	0.28	1.00	0.51	0.28
59	0.27	1.00	0.50	0.27
60	0.25	1.00	0.49	0.25
61	0.24	1.00	0.48	0.24
63	0.22	1.00	0.47	0.22
64	0.21	1.00	0.46	0.21
65	0.18	1.00	0.44	0.18

Tabela C.1.3 Tabela com a informação completa da tabela 3.

Ponto de Corte	Sensibilidade	Especificidade	Precisão	Índice de Youden
66	0.15	1.00	0.42	0.15
67	0.14	1.00	0.41	0.14
83	0.08	1.00	0.38	0.08
86	0.06	1.00	0.36	0.06
90	0.05	1.00	0.35	0.05
116	0.04	1.00	0.35	0.04
126	0.03	1.00	0.34	0.03
134	0.02	1.00	0.33	0.02
156	0.01	1.00	0.33	0.01
163	0.00	1.00	0.32	0.00

ANEXO D. Versão não reduzida da tabela com os valores do Custo Unitário Esperado e Retorno Unitário Esperado

Tabela D.1.1 Tabela com a informação completa da tabela 12.

Ponto de Corte	Sensibilidade	Especificidade	Custo Unitário Esperado	Retorno Unitário Esperado
5	1.00	0.00	10.00	0.01
6	1.00	0.02	9.92	0.01
10	1.00	0.04	9.84	0.01
12	1.00	0.11	9.60	0.02
13	0.98	0.16	9.44	0.02
14	0.97	0.20	9.28	0.02
15	0.97	0.20	9.28	0.02
16	0.97	0.27	9.04	0.02
18	0.97	0.29	8.96	0.03
20	0.97	0.31	8.88	0.03
21	0.96	0.36	8.72	0.03
22	0.94	0.38	8.64	0.03
24	0.93	0.38	8.64	0.02
25	0.90	0.38	8.64	0.02
26	0.85	0.38	8.64	0.01
27	0.81	0.40	8.56	0.01
29	0.80	0.47	8.32	0.01
30	0.77	0.49	8.24	0.01
31	0.76	0.49	8.24	0.00
32	0.74	0.51	8.16	0.00
33	0.73	0.53	8.08	0.00
34	0.70	0.56	8.00	0.00
35	0.70	0.56	8.00	0.00

Tabela D.1.2 Tabela com a informação completa da tabela 12.

Ponto de Corte	Sensibilidade	Especificidade	Custo Unitário Esperado	Retorno Unitário Esperado
36	0.69	0.60	7.84	0.00
37	0.68	0.67	7.60	0.00
38	0.65	0.67	7.60	0.00
39	0.59	0.67	7.60	-0.01
39.5	0.59	0.71	7.44	-0.01
40	0.54	0.73	7.36	-0.02
41	0.51	0.73	7.36	-0.02
42	0.51	0.76	7.28	-0.02
43	0.49	0.82	7.04	-0.02
44	0.48	0.84	6.96	-0.02
45	0.44	0.84	6.96	-0.03
46	0.43	0.84	6.96	-0.03
48	0.36	0.87	6.88	-0.04
49	0.36	0.87	6.88	-0.04
50	0.34	0.91	6.72	-0.04
51	0.33	0.91	6.72	-0.04
53	0.32	0.93	6.64	-0.04
54	0.31	0.96	6.56	-0.04
56	0.30	0.96	6.56	-0.04
57	0.29	1.00	6.40	-0.04
58	0.28	1.00	6.40	-0.04
59	0.27	1.00	6.40	-0.05
60	0.25	1.00	6.40	-0.05
61	0.24	1.00	6.40	-0.05
63	0.22	1.00	6.40	-0.05
64	0.21	1.00	6.40	-0.06
65	0.18	1.00	6.40	-0.06

Tabela D.1.3 Tabela com a informação completa da tabela 12.

Ponto de Corte	Sensibilidade	Especificidade	Custo Unitário Esperado	Retorno Unitário Esperado
66	0.15	1.00	6.40	-0.07
67	0.14	1.00	6.40	-0.07
83	0.08	1.00	6.40	-0.08
86	0.06	1.00	6.40	-0.08
90	0.05	1.00	6.40	-0.08
116	0.04	1.00	6.40	-0.08
126	0.03	1.00	6.40	-0.09
134	0.02	1.00	6.40	-0.09
156	0.01	1.00	6.40	-0.09
163	0.00	1.00	6.40	-0.09

