GILBERTO RODRIGUES LISKA

CLASSIFICAÇÃO DE DADOS EM MODELOS COM RESPOSTA BINÁRIA VIA ALGORITMO BOOSTING E REGRESSÃO LOGÍSTICA

LAVRAS - MG

2012

GILBERTO RODRIGUES LISKA

CLASSIFICAÇÃO DE DADOS EM MODELOS COM RESPOSTA BINÁRIA VIA ALGORITMO BOOSTING E REGRESSÃO LOGÍSTICA

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-graduação em Estatística e Experimentação Agropecuária, área de concentração em Estatística e Experimentação Agropecuária, para a obtenção do título de Mestre.

Orientador
Dr. Fortunato Silva de Menezes

LAVRAS - MG
2012

Ficha Catalográfica Elaborada pela Divisão de Processos Técnicos da Biblioteca da UFLA

Liska, Gilberto Rodrigues.
Classificação de dados em modelos com resposta binária via algoritmo boosting e regressão logística / Gilberto Rodrigues Liska. - Lavras : UFLA, 2012.
105 p. : il.

Dissertação (mestrado) - Universidade Federal de Lavras, 2012.
Orientador: Fortunato Silva de Menezes.
Bibliografia.

1. Modelos de Regressão. 2. Métodos de Classificação. 3. Doença cardíaca coronariana. 4. Binomial Boosting. 5. Seleção de Modelos. I. Universidade Federal de Lavras. II. Título.

CDD-519.536

GILBERTO RODRIGUES LISKA

CLASSIFICAÇÃO DE DADOS EM MODELOS COM RESPOSTA BINÁRIA VIA ALGORITMO BOOSTING E REGRESSÃO LOGÍSTICA

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-graduação em Estatística e Experimentação Agropecuária, área de concentração em Estatística e Experimentação Agropecuária, para a obtenção do título de Mestre.

APROVADA em 20 de novembro de 2012.

Dr. Antônio Policarpo Souza Carneiro	UFV
Dr. João Domingos Scalon	UFLA
Dr. Marcelo Ângelo Cirillo	UFLA

Dr. Fortunato Silva de Menezes
Orientador

LAVRAS - MG
2012

Aos meus pais Lúcia e Istvan (*in memorian*),

pelo amor, carinho e educação.

Aos meus irmãos Estevan e Geraldo.

À minha namorada e futura esposa Grazielle.

DEDICO

AGRADECIMENTOS

A Deus. Muito obrigado por ter me dado a oportunidade de cursar um mestrado e de concluí-lo. Muito obrigado por ter me dado a oportunidade de ingressar no doutorado. Muito obrigado por ter me dado força, ânimo, paciência e comprometimento com os deveres dessa empreitada. Fico intrigado em me questionar se realmente existe um Deus que pode nos proporcionar coisas do tipo e logo fico emocionado em saber que sim, isso é possível, mas não sei explicar o porquê disso. Acredito que muitas pessoas que experimentam situações extremas, adversas e que exijam muito de si sabem o que quero dizer. Enfim, muito obrigado, meu Deus.

Aos meus pais, Istvan e Lúcia, que não estão mais entre nós, pelo amor infinito que criou todas as condições que me permitiram concluir mais esta etapa. Gostaria de agradecer pelas virtudes as quais admirava neles, como a sabedoria e inteligência em meu pai e coragem, força e garra em minha mãe.

Aos meus irmãos, Geraldo e Estevan. Cada um está seguindo sua vida hoje, mas gostaria de destacar aqui a contribuição que eles tiveram na época de minha graduação, principalmente no momento que ficamos sem nossos pais. Desejo a vocês que alcancem seus objetivos e que Deus os conduza da melhor maneira possível.

À minha namorada, Grazielle Aparecida Cassimiro, pela amizade, companheirismo, paciência e força que me sustentaram durante todo o tempo que estivemos e estamos juntos, mesmo que às vezes distante dela. Ficarei honrado em tê-la como minha esposa num futuro bem próximo.

Ao Professor Dr. Luiz Alberto Beijo, meu primeiro orientador, por ter me apresentado e conduzido na carreira como Estatístico.

Ao Professor Dr. Fortunato Silva de Menezes, pela orientação recebida no

mestrado, pela paciência nas explicações, pela sabedoria nas horas difíceis e pelo apoio e incentivo nos trabalhos.

Aos meus amigos do mestrado, Guido Gustavo Humada Gonzalez, Juraci Mendes Moreira, Juliano Bortolini, Rossicley Rangel Paiva e demais amigos que estiveram comigo nessa batalha. Obrigado a todos que me apoiaram nos momentos difíceis.

Ao Professor Dr. Marcelo Ângelo Cirillo, que aceitou com satisfação o convite para me orientar no doutorado.

À minha ex-chefe, Célia Pereira de Araújo, quando funcionário do Programa de Saúde da Família - Caensa, na função de Agente Comunitário de Saúde, pelo apoio, até então, na decisão mais difícil de minha vida: a saída desse serviço para a dedicação exclusiva aos estudos. Muito obrigado por acreditar no meu potencial.

A todos os brasileiros que pagam seus impostos honestamente e permitem que instituições como a UFLA, CAPES e CNPq mantenham cursos de alto nível e ofereçam bolsas aos alunos. Um agradecimento especial às agências de fomento CAPES e CNPq por ter me concedido bolsa de estudos durante meu mestrado.

Enfim, não posso deixar de agradecer a todos que torceram, incentivaram e que, diretamente ou indiretamente, contribuíram pelo sucesso desta empreitada. Muito obrigado.

"*Força!! ... Sangue!! ... Fibra!! ... Moral!! ... Ralação!! ... Vibração!! ...*
Ralação!! ... Vibração!! ..."

Canção cantada pelos atiradores durante os serviço militar nas corridas feitas pelas ruas da cidade de Alfenas-MG.

RESUMO

Classificar algo é uma tarefa natural do ser humano, mas existem situações em que o mesmo não é o mais indicado para desempenhar tal função. A necessidade de métodos automáticos de classificação surge em várias áreas, como por exemplo em reconhecimento de vozes, reconhecimento de tumores por meio de chapas de raio-x, na classificação de e-mail como legítimos ou spam, entre outros. Devido a importância e o aumento da complexidade de problemas do tipo, existe ainda a necessidade de métodos que forneçam maior precisão e interpretabilidade dos resultados. Entre eles, os métodos de Boosting, que funcionam aplicando-se sequencialmente um algoritmo de classificação a versões reponderadas do conjunto de dados de treinamento. Recentemente foi mostrado que Boosting pode ainda ser visto como um método para estimação funcional. Atualmente os modelos de regressão logística com seus parâmetros estimados via máxima verossimilhança (doravante chamado MRLMV) são muito utilizados para esse tipo de situação. Nesse sentido, o presente trabalho consistiu em comparar o modelo de regressão logística MRLMV e o estimado via algoritmo Boosting, mais especificamente algoritmo Binomial Boosting (doravante chamado MRLBB), e selecionar o modelo com melhor adequabilidade de ajuste e maior capacidade de discriminação na situação de presença/ausência de doença cardíaca coronariana (CHD) como função de várias variáveis biológicas, com vista a fornecer informações mais precisas para situações cuja resposta é binária. Para ajustar os modelos, o conjunto de dados foi particionado aleatoriamente em dois subconjuntos, sendo um subconjunto equivalente a 70% do conjunto original (denominado de amostra de treinamento) e o restante, denominado de conjunto de teste. Os resultados mostram valores menores de AIC e BIC para o MRLBB em comparação ao MRLMV e pelo teste de Hosmer-Lemeshow ambos modelos (MRLMV e MRLBB) não apresentaram evidências de mau ajuste. O modelo MRLBB apresentou maiores valores de AUC, sensibilidade, especificidade e acurácia e menores valores para a taxa de falsos positivos e falsos negativos, mostrando-se, portanto, um modelo mais adequado do que o MRLMV. Observando-se as razões de chances, o modelo MRLBB apresentou resultados mais confiáveis quanto à chance de um paciente possuir CHD. Diante dos resultados obtidos, o modelo MRLBB é o mais adequado para descrever o problema de presença/ausência de doença cardíaca coronariana em pacientes, pois fornece informações mais precisas acerca do problema exposto.

Palavras-chave: Métodos de Classificação, Binomial Boosting, Modelos de Regressão, Doença Cardíaca Coronariana (CHD), Seleção de Modelos.

ABSTRACT

Classify something is a natural human task, but there are situations where it is not best suited to perform this function. The need for automatic methods for classification arises in several areas, ranging from voice recognition, tumors recognition by x-ray films, email classification as spam or legitimate, among others. Due to the increasing complexity and importance of problems such as these, there is still a need for methods which provide greater accuracy and interpretability of the results. Among these methods Boosting, which operates sequentially applying a classification algorithm to reweighted versions of the training data set. Recently it was shown that Boosting may also be viewed as a method for estimating functional. Currently the logistic regression models with its parameters estimated by maximum likelihood (henceforth called LRMML) are very used to this kind of situation. In this sense, the present study was to compare the LRMML and Boosting algorithm, specifically Binomial Boosting algorithm (henceforth called LRMBB), logistic regression model, and select the model with the best fit and suitability of higher discrimination capacity in the situation of presence / absence of coronary heart disease (CHD) as a function of various biological variables in patients in order to provide the most accurate response to situations which is binary. To adjust the model, the data set was randomly partitioned into two subsets, one subset equivalent to 70 % of the original set (called training sample) and the remainder (called test set). The results show lower values of AIC and BIC for the LRMBB model compared to LRMML and the Hosmer-Lemeshow test shows both models (LRMLM and LRMBB) present no evidence of bad fit. The LRMBB model presented higher values of AUC, sensitivity, specificity and accuracy and lower values for the rate of false positives and false negatives, being therefore a model with better discrimination power in relation to the LRMML model. Observing the odds ratios, the LRMBB model showed more reliable results about the chance of a patient having CHD. Based on these results, the LRMBB model is best suited to describe the problem of presence / absence of coronary heart disease in patients because it provides more accurate information about the problem exposed.

Keywords: Classification Methods, Binomial Boosting, Regression Models, Coronary Heart Disease (CHD), Model Selection.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Em inúmeras situações o pesquisador se depara com a necessidade de realizar uma classificação nos dados, sobretudo, mediante ao tamanho amostral a ser considerado, bem como outras causas, por exemplo, se o modelo proposto ou os dados apresentarem algum tipo de perturbação, os métodos estatísticos convencionais podem apresentar taxas de erros de classificação incoerentes.

Tendo por base essa questão, uma alternativa plausível de ser utilizada é apontada na combinação de métodos computacionais e técnicas estatísticas. Nesse sentido surge a motivação para construir um classificador automático, o qual consiste em utilizar dados sobre o problema em mãos para se tentar criar uma regra que possa ser usada para classificar outros dados no futuro. A maneira com que essa regra é criada influi diretamente em aspectos como o desempenho e a interpretabilidade do classificador.

Convém ressaltar que as técnicas estatísticas Análise Discriminante e Regressão Logística, que são utilizadas em situações que envolvam classificação, a resposta a um determinado fenômeno não configura uma situação contínua, ou seja, admite-se a existência de categorias, podendo assumir dois ou mais valores. Nestes casos, a Regressão Logística tem sido aplicada com frequência e sua utilização permite obter a probabilidade de um determinado evento ocorrer. Contudo, a Análise Discriminante e a Regressão Logística, a priori, pressupõem a criação de regras bastante interpretáveis, mas com formas restritivas para a relação entre as respostas e as variáveis preditoras. Como método alternativo para classificação, existem as redes neurais, que notabilizaram-se por serem "caixas pretas" com alta precisão, mas com interpretabilidade pobre.

A necessidade de métodos automáticos de classificação é uma realidade

aos seres humanos, seja para executar uma tarefa que pareça maçante para um ser humano, como por exemplo o reconhecimento de códigos postais em cartas, reconhecimento de vozes de pessoas, ou até mesmo para classificar pacientes que tenham ou não uma determinada doença. Existem vários métodos que executam um mesmo processo de classificação, sendo que cada método tem sua peculiaridade, mas existe o interesse prático de se ter aquele classificador que erre o menos possível em uma determinada tarefa, pois, em várias situações, um erro de classificação pode trazer graves consequências ou até mesmo irreversíveis. Recentemente os métodos de Boosting têm recebido grande atenção por produzirem classificadores com alto poder de predição, entre eles os algoritmos capazes de estimar funções e, no nosso caso, estamos interessados em estimar um modelo adequado para resposta binária.

Mediante a conjectura de aprimorar a interpretabilidade e desempenho do uso de métodos classificadores aplicados em uma variedade de problemas, têm-se os algoritmos de Boosting, originados na área de computação, que em uma de suas versões funcionam aplicando-se sequencialmente um algoritmo de classificação a versões reponderadas do conjunto de dados de treinamento, dando maior peso às observações classificadas erroneamente no passo anterior. Eles foram introduzidos por Schapire (1990) e o algoritmo de Boosting mais famoso é o AdaBoost. Desde então, várias versões de algoritmos Boosting têm sido criadas.

Diante do exposto, esse trabalho objetiva estudar o desempenho de algoritmo Boosting em problemas de classificação que envolvam respostas binárias em comparação com o modelo de regressão logística estimado via máxima verossimilhança, e apresentar principais aspectos relacionados à abordagem estatística do algoritmo Boosting. Em adição, comparamos o modelo de regressão logística estimado via máxima verossimilhança (doravante chamado MRLMV) e o modelo

de regressão logística estimado via algoritmo Binomial Boosting (doravante chamado MRLBB) pelos critérios de informação de Akaike e Bayesiano. Verificamos também a acurácia, sensibilidade, especificidade, taxa de falso positivo e taxa de falso negativo dos modelos MRLMV e MRLBB.

Atualmente os modelos de regressão logística com seus parâmetros estimados via máxima verossimilhança (MRLMV) são os mais utilizados para esse tipo de situação. O presente trabalho consistirá em comparar o modelo MRLMV e o estimado via algoritmo Binomial Boosting (MRLBB) e selecionar o modelo com maior capacidade de discriminação na situação de presença/ausência de doença cardíaca coronariana como função de várias variáveis biológicas em pacientes, com vista a fornecer informações mais precisas acerca do problema exposto.

2 REFERENCIAL TEÓRICO

Esta seção abordará inicialmente a definição de um problema de classificação, bem como a definição formal de um classificador. Na sequência serão apresentadas as características dos classificadores utilizados neste trabalho. O presente trabalho utilizará o modelo logístico para classificação binária e serão discutidos duas formas de se estimar os parâmetros de um modelo logístico, sendo um no contexto de Boosting e o outro no contexto de máxima verossimilhança. Serão apresentadas os principais aspectos relacionados a abordagem estatística de ambas abordagens. Em seguida, serão discutidos os critérios de adequabilidade de ajuste que serão utilizados para efetuarmos as comparações do modelo ligístico estimado via Boosting e o mesmo estimado via máxima verossimilhança.

2.1 Classificação

O ato de classificar algo é uma tarefa natural à atividade humana. Utilizando informações que chegam aos seres humanos por meio de seus sentidos, eles designam objetos a classes. Em certo sentido, dadas as informações sobre um objeto, um ser humano toma uma decisão sobre a que classe (dentre um conjunto finito de classes) esse objeto pertence, ou tenta estabelecer a existência de classes nas quais os diversos tipos de objetos possam ser alocados (BISHOP, 1995). Exemplos de classificação feita pelos humanos são: reconhecer rostos e vozes de pessoas; identificar odores; reconhecer um alimento pelo sabor; etc.

Apesar de os seres humanos serem particularmente bons em muitas tarefas de classificação e as desempenharem naturalmente e sem esforço, existem certos motivos que tornam desejáveis que uma tarefa de classificação seja desempenhada

por uma máquina, um computador no caso. Uma tarefa pode ser muito repetitiva e maçante para que um ser humano a faça, ou custosa demais, ou ainda essa tarefa pode ser melhor desempenhada por uma máquina. Um exemplo de uma tarefa que pode ser considerada muito trabalhosa para um ser humano é a de reconhecimento de códigos postais em cartas.

A criação e uso de métodos para classificação automática despertou interesse em diversas áreas, e portanto muitos métodos foram criados independentemente, enquanto outros nasceram da união de esforços entre essas áreas. Existem praticamente duas abordagens: a estatística e a computacional. A estatística é a mais antiga e tradicional e por isso é as vezes tratada como se fosse menos automática (RIPLEY, 1996). A abordagem computacional é a feita pelas pessoas de uma comunidade a que se refere comumente como aprendizado de máquinas (*machine learning*). Nessa comunidade, encontram-se engenheiros, profissionais da computação e muitos outros. Devido a essa interdisciplinaridade, a linguagem utilizada varia bastante e utilizam-se termos possivelmente diferentes em cada área.

A abordagem estatística foi marcada inicialmente com as técnicas derivadas do trabalho de Fischer em discriminação linear, por volta de 1936. Mais adiante, por volta da década de 60, apareceram modelos com características mais flexíveis (aí se encaixa o modelo de regressão logística) e, em geral, o foco é obter uma estimativa da distribuição de probabilidade dos dados em cada classe, e obter com isso uma regra de classificação (MICHIE, 1994).

A comunidade de aprendizado de máquinas e das pessoas que trabalham com reconhecimento de padrões (*pattern recognittion*) foi motivada no início por tentativas de se modelar o modo pelo qual o ser humano aprende, influenciadas por ideias biológicas de como o cérebro funciona. Um exemplo particular são as chamadas redes neurais artificiais, que surgiram inicialmente como modelos simples

para explicar o funcionamento de agrupamento de neurônios. Logo, percebeu-se seu poder prático para reconhecer padrões (classificar) e, a partir daí, profissionais de muitas áreas as desenvolveram e utilizaram (FRIEDMAN; HASTIE; TIBSHIRANI, 2001).

Independentemente da abordagem, o objetivo da classificação é obter métodos automáticos que sejam tão bons classificadores quanto os seres humanos, e/ou que possam ser entendidos ou possam ser interpretados por seres humanos.

2.1.1 Abordagem Estatística em um problema de Classificação

Um procedimento de classificação, regra de classificação ou classificador é algum método que, possivelmente de maneira automática, separe objetos em classes. Em geral, um procedimento de classificação é construído com base na experiência passada, e o interesse é utilizá-lo para classificar objetos que não foram utilizados na construção desse classificador ou que possuem classificação desconhecida.

Existem basicamente dois tipos de classificação. No primeiro, temos informações sobre os objetos e o intuito é estabelecer a existência de classes. Esse tipo de classificação é conhecido na literatura estatística como agrupamento (*clustering*) e na comunidade de aprendizado de máquinas como aprendizagem não supervisionada. O segundo tipo de classificação supõe que existe um número conhecido de classes e o objetivo é estabelecer uma regra pela qual possam se alocar novos objetos a uma das classes. Na literatura estatística, isso é conhecido como discriminação, e fora dela como reconhecimento de padrões ou aprendizagem supervisionada (FRIEDMAN; HASTIE e TIBSHIRANI, 2001).

O presente trabalho será voltado para o segundo tipo de classificação, que

é generalizado por um classificador, considerando a seguinte notação: Seja um grupo de objetos que possam ser classificados em J classes, numeradas $1, 2, ..., J$ e seja $C = \{1,\ 2,\ \ldots,\ J\}$ o conjunto das classes. Considere também que são medidas p variáveis aleatórias em cada objeto e que essas p variáveis estão dispostas em um vetor $\boldsymbol{x} = (x_1,\ x_2,\ \ldots,\ x_p)$. Defina por Ω o espaço multidimensional contendo todos os possíveis vetores de $\boldsymbol{x}$.

Segundo Breiman (1984), um classificador é uma função

$$\begin{aligned} d:\ \Omega\ &\rightarrow C \\ \boldsymbol{x}\ &\mapsto d\left(\boldsymbol{x}\right)\ \in\ C \end{aligned}$$

ou seja, para cada objeto $\boldsymbol{x}$, o classificador d designa uma classe $d\left(\boldsymbol{x}\right) \in$
$\{1,\ 2,\ \ldots,\ J\}$.

Agora, defina $A_j = \{\boldsymbol{x}\ \in\ \Omega;\ d\left(\boldsymbol{x}\right) = j\}$, $j = 1,\ 2,\ \ldots,\ J$, ou seja, para cada j, A_j é o subconjunto de Ω no qual o classificador d prediz a classe j. Mais do que isso, os A_j definem uma partição de Ω, ou seja, $A_i \cap A_j\ =\ \emptyset$, $i\ \neq\ j$ e $\bigcup_{j=1}^{J} A_j = \Omega$. Assim, um classificador d induz uma partição $A_1,\ A_2,\ \ldots,\ A_j$ de Ω, com $\Omega = \bigcup_{j=1}^{J} A_j$, tal que para todo $\boldsymbol{x} \in A_j$ a classe predita é j. Para exemplificar, suponha um classificador d constituído por um modelo de regressão logística múltipla binária. Nesse caso, $J = 2$ e $C = \{0,\ 1\}$. Sabe-se que a predição desse modelo é a probabilidade de um determinado evento ocorrer e que dado um limiar obtém-se a predição de uma determinada classe. Suponha então que esse limiar é conhecido e com isso tem-se a classe predita. Portanto, o modelo em questão é um classificador d que relaciona cada $\boldsymbol{x}$ a uma classe correspondente em C. Observe

ainda que $A_0 = \{\boldsymbol{x} \in \Omega;\, d(\boldsymbol{x}) = 0\}$ e $A_1 = \{\boldsymbol{x} \in \Omega;\, d(\boldsymbol{x}) = 1\}$ definem o espaço Ω, uma vez que $A_0 \cup A_1 = \Omega$ e que $A_0 \cap A_1 = \emptyset$.

Dependendo da estrutura da tarefa de classificação, podem ser de interesse não somente as classes preditas pelo classificador, mas também estimativas das probabilidades de um objeto pertencer a uma certa classe. Alguns métodos proporcionam essas probabilidades, como exemplificado acima, enquanto outros fornecem apenas a predição da classe, como por exemplo o método de Análise Discriminante, redes neurais, árvores de decisão e os métodos de Boosting (especificamente o algoritmo AdaBoost). Um procedimento comum em estatística aplicada é, dadas as estimativas das probabilidades de um objeto pertencer a cada classe, alocá-lo na classe com maior probabilidade de acerto (RUBESAM, 2004).

2.2 Introdução ao método de Boosting

O método conhecido como Boosting nasceu na comunidade de aprendizado de máquinas. Dentro dessa comunidade, foi proposto um problema teórico chamado de problema de Boosting, que pode ser informalmente exposto da seguinte maneira: "suponha que existe um método de classificação que é ligeiramente melhor do que uma escolha aleatória, para qualquer distribuição em Ω. Esse método é chamado de classificador fraco (*weak learner*). A existência de um classificador fraco implica a existência de um classificador forte (*strong learner*), com erro pequeno sobre todo o espaço Ω?"

Esse problema foi resolvido por Schapire (1990), que mostrou que era possível obter um classificador forte a partir de um fraco. A partir de então, foram desenvolvidos vários algoritmos dentro do contexto de Boosting. Um dos mais recentes e bem sucedidos deles é o algoritmo conhecido como AdaBoost (Adaptative Boosting), que funciona pertubando a amostra de treinamento gerando a cada

iteração (de forma determinística, mas adaptativa) uma distribuiuçao sobre as observações da amostra, dando maior peso (maior probabilidade de estar na amostra perturbada) às observações classificadas erroneamente no passo anterior. Existe um outro método de combinação de preditores, conhecido por *Bagging* (*Bootstrap Aggregating*), que funciona perturbando essa amostra de treinamento aleatoriamente por meio de re-amostragem, gerando a cada iteração um classificador e o classificador final é obtido pela agregação desses classificadores (SHAPIRE; FREUND, 2012).

Desde o seu desenvolvimento como uma resposta a um problema teórico, os algoritmos do tipo Boosting têm recebido grande atenção, tanto na comunidade estatística quanto na de *machine learning*. A comunidade estatística busca entender como e por que Boosting funciona, abordando aspectos como consistência, enquanto na comunidade de *machine learning* a abordagem é mais focada nos próprios algoritmos e em sua funcionalidade (RUBESAM, 2004).

O algoritmo AdaBoost é o mais famoso dos algoritmos de Boosting, e foi apresentado por Freund e Schapire (1996). Os autores fizeram uma análise do algoritmo em termos de limites para as probabilidades de erro na amostra de treinamento e nas amostras de teste (o erro de um classificador em casos novos é chamado na literatura de aprendizagem de máquinas de erro de generalização). Um dos limites teóricos mostrados implica que o erro na amostra de treinamento decai exponencialmente com o número de iterações do algoritmo. Empiricamente, observa-se que, após algumas iterações, o erro na amostra de treinamento cai a zero, confirmando o resultado teórico.

Inicialmente, observou-se que, quando se continua a executar o algoritmo AdaBoost, o erro na amostra teste continua a decrescer, indicando que o algoritmo é resistente a super ajuste (*overftting*) (BUHLMANN; HOTHORN, 2007). O su-

per ajuste é o problema que surge quando um modelo tem desempenho bom no conjunto de treinamento, mas em dados novos, que não foram usados no ajuste do modelo, tem desempenho ruim. Isso ocorre geralmente porque o modelo se torna complexo demais (ou seja, número excessivo de parâmetros) e passa a ajustar peculiaridades do conjunto de treinamento. Por exemplo, em regressão logística, a adição de variáveis sempre melhora o desempenho no conjunto usado para estimar o modelo, mas em algum ponto isso começa a se tornar prejudicial e o desempenho em um conjunto de teste é ruim. Em redes neurais, se o algoritmo de otimização é executado indefinidamente, o erro sempre diminui no conjunto de treinamento, mas em certo ponto ele começa a aumentar no conjunto de teste. Existem métodos para determinar o ponto de parada nesse caso, como por exemplo o método de parada precoce (*early stopping*), que cessa a otimização quando o erro começa a aumentar no conjunto de teste.

Friedman, Hastie e Tibshirani (2001) mudaram totalmente o modo como Boosting é visto, pelo menos na comunidade estatística. Eles colocaram Boosting como uma aproximação do ajuste de um modelo aditivo na escala logística, usando a máxima verossimilhança da Bernoulli como critério. Ademais, sugeriram uma aproximação mais direta, o que levou ao algoritmo LogitBoost, um algoritmo para ajustar regressão logística aditiva que dá resultados praticamente idênticos ao AdaBoost de Freund e Schapire.

Mais recentemente, notou-se que, se um algoritmo de Boosting for executado por um tempo (número de iterações) muito grande, da ordem de dezenas de milhares, isso ocasionará super ajuste. Friedman, Hastie e Tibshirani (2000) dá um exemplo em que isso ocorre. Algumas abordagens para esse problema foram tentadas. Jiang (2000) mostrou que, sob certas condições de regularidade, como o número ideal de iterações, o algoritmo AdaBoost é consistente em processo, no

sentido de que, durante o treinamento, ele gera uma sequencia de classificadores com erro que converge para o erro do classificador (regra) de Bayes.

2.2.1 Algoritmos Boosting utilizados na classificação binária

Serão apresentados a seguir dois algoritmos Boosting utilizados para classificação binária. O algoritmo AdaBoost não será utilizado neste trabalho, porém é necessário sua apresentação por ser um algoritmo precedente de outros algoritmos Boosting, inclusive o algoritmo utilizado neste trabalho, o algoritmo Gradiente Boosting de Friedman, e o entendimento de seu mecanismo funcional ajuda a compreender a funcionalidade do algoritmo Gradiente Boosting de Friedman.

2.2.1.1 AdaBoost para duas classes

O algoritmo AdaBoost para classificação binária é o algoritmo boosting mais conhecido. O classificador base (passo 2(a) do algoritmo a seguir) retorna valores em $\{-1,\ 1\}$ e pode ser, por exemplo, uma árvore de regressão ou uma rede neural. Será apresentado a seguir a versão desse algoritmo, dada em Friedman, Hastie e Tibshirani (2001).

Suponha que temos um conjunto de treinamento $L = (\boldsymbol{x}_1,\, y_1),\ \ldots,$ $(\boldsymbol{x}_N,\, y_N)$, onde as classes estão rotuladas $\{-1, 1\}$, ou seja, $C = \{-1, 1\}$. Defina $F(\boldsymbol{x}) = \sum_1^M c_m\, f_m(\boldsymbol{x})$, onde M é o número de vezes que o algoritmo é executado (iterações), f_m é um classificador base que retorna valores $\{-1, 1\}$, os valores c_m são constantes e a predição correspondente a cada valor de $\boldsymbol{x}$ é a função sinal de $F(\boldsymbol{x})$, ou seja, $sign\,(F(\boldsymbol{x}))$. A função $sign\,(\cdot)$ retorna 1 se $sign\,(\cdot) > 0$ e retorna -1 se $sign\,(\cdot) < 0$. O algoritmo AdaBoost ajusta classificadores base

f_m em amostras ponderadas do conjunto de treinamento, dando maior peso, ou ponderação, aos casos que são classificados erroneamente. Os pesos são ajustados adaptativamente em cada iteração e o classificador final é uma combinação linear dos classificadores f_m. A Figura 1 ilustra de maneira geral o funcionamento de um algoritmo Boosting para classificação binária, que segue a mesma ideia do algoritmo AdaBoost apresentado anteriormente.

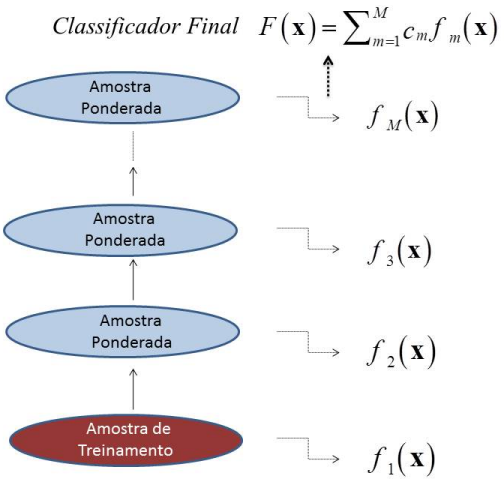

Figura 1 Algoritmo Boosting para classificação binária

O algoritmo AdaBoost consiste em três passos:

1. Dado $(x_1, y_1), \ldots, (x_N, y_N)$ em que $x_i \in \boldsymbol{X}$ e $y_i \in Y = \{-1, +1\}$. Inicialize os pesos $w_i^m = 1/N$, $i = 1, 2, \ldots, N$.

2. Repita para $m = 1, 2, \ldots, M$:

   (a) Ajuste o classificador $f_m(\boldsymbol{x}) \in \{-1, 1\}$ usando os pesos w_i e os dados de treinamento;

(b) Calcule

$$\varepsilon_m = \frac{\sum_{i=1}^{N} w_i^m I\left[Y_i \neq f_m\left(X_i\right)\right]}{\sum_{i=1}^{n} w_i^m}$$

$$c_m = \frac{1}{2} \ln\left(\frac{1-\varepsilon_m}{\varepsilon_m}\right)$$

(c) Faça

$$w_i^{m+1} = \frac{w_i^m}{z^m} \times \begin{cases} e^{-c_m} \, se \; y_i = f_m\left(x_i\right) \\ e^{c_m} \, se \; y_i \neq f_m\left(x_i\right) \end{cases}$$

em que z^m é um fator de normalização

$$z^m = \sum_{i=1}^{N} w_i^m e^{\left(-c_m \, y_i \, f_m(x_i)\right)}$$

3. A predição é dada por $sign\left(F\left(\boldsymbol{x}\right)\right) = sign\left(\sum_{m=1}^{M} c_m \, f_m\left(\boldsymbol{x}\right)\right)$.

No algoritmo acima, ε_m representa a média ponderada dos erros com pesos $w = (w_i, \ldots, w_N)$. Em cada iteração, o algoritmo aumenta os pesos w_i das observações classificadas erroneamente por um fator que depende dos erros ε_m das observações do conjunto de treinamento (passo 2 (c)).

No APÊDICE A é apresentada uma ilustração didática para melhor compreensão do algoritmo AdaBoost.

Friedman, Hastie e Tibshirani (2000) mostraram que o algoritmo AdaBoost pode ser derivado como algoritmo iterativo para ajustar um modelo aditivo logístico, otimizando um critério que até segunda ordem é equivalente à log-verossimilhança da binomial. A derivação do processo de atualização do algoritmo AdaBoost visto anteriormente, conforme descrito por Friedman, Hastie e Tibshirani (2000) encontra-se no ANEXO A.

2.2.1.2 Algoritmo Gradiente Boosting de Friedman

Breiman (1998, 1999) mostrou que o algoritmo AdaBoost pode ser representado como um algoritmo do gradiente no espaço funcional, o qual podemos denominar de Gradiente de Descida Funcional (FGD). Friedman, Hastie e Tibshirani (2000) e Friedman (2001) desenvolveram de forma mais geral uma estrutura estatística que leva a direta interpretação de Boosting como um método para estimação funcional. Na sua terminologia, trata-se de uma aproximação em modelagem aditiva *stagewise* (mas a palavra aditiva não implica o ajuste de um modelo que é aditivo nas variáveis independentes).

No contexto de Boosting, o objetivo é estimar uma função de predição ótima $f^{*}(\cdot)$, também chamada de minimizador populacional, que é definido por

$$f^{*}(\cdot) = \arg\min_{f} E_{Y,X}\left[\rho\left(Y, f\left(\boldsymbol{X}\right)\right)\right] \tag{2.1}$$

em que $\rho(\cdot, \cdot)$ é uma função perda que é assumida como sendo diferenciável e convexa com respeito a f. Na prática, trabalhamos com realizações $(y_i, \boldsymbol{x}_i^T)$, $i = 1, \ldots, n$, de $(\boldsymbol{y}, \boldsymbol{x}^T)$, e a esperança em 2.1 é, portanto, não conhecida. Por essa razão, em vez de minimizar o valor esperado dado em 2.1, os algoritmos Boosting minimizam a perda média observada, que é dada por $n^{-1}\sum_{i=1}^{n}\rho\left(Y_i,\ f\left(X_i\right)\right)$, perseguindo iterativamente no espaço funcional dos parâmetros de f.

Por exemplo, a perda *erro quadrática*

$$\rho(y,\ f) = (y - f)^2$$

leva ao bem conhecido minimizador populacional

$$f^{*}\left(\boldsymbol{x}\right) = E\left[\mathbf{Y}\left|\boldsymbol{X} = \boldsymbol{x}\right.\right]$$

De maneira geral, dada uma função perda $\rho\left(y,\, f\right)$ e um procedimento base, $g(x)$, que será visto nas seções a seguir, o seguinte algoritmo foi dado por Friedman (2001), também chamado de Algoritmo Gradiente Boosting de Friedman, e executando os seguintes passos:

1. Inicialize $\hat{f}^{(0)}\left(\cdot\right)$ com um valor inicial. Escolhas comuns são

   $$\hat{f}^{(0)}\left(\cdot\right) = \arg\min_{c} \frac{1}{n}\sum_{i=1}^{N} \rho\left(Y_i,\, c\right)$$

   ou $\hat{f}^{(0)}\left(\cdot\right) = 0$. Coloque $m = 0$.

2. Aumente m em 1. Calcule o gradiente negativo $-\frac{\partial}{\partial f}\rho\left(Y,\, f\right)$ e calcule em $\hat{f}^{(m-1)}\left(\boldsymbol{X}_i\right)$:

   $$z_i = -\frac{\partial}{\partial f\left(\boldsymbol{x}_i\right)}\rho\left(Y_i,\, f\left(\boldsymbol{x}_i\right)\right)\big|_{f(\boldsymbol{x}_i)=\hat{f}^{(m-1)}(\boldsymbol{x}_i)},\; i = 1,\, \ldots,\, n$$

3. Ajuste o vetor gradiente negativo $z_1,\, \ldots,\, z_n$ para $X_1,\, \ldots,\, X_n$ por um procedimento base $\hat{g}^{(m)}\left(\cdot\right)$ de valor real (por exemplo, regressão).

4. Atualize

   $$\hat{f}^{(m)}\left(\cdot\right) = \hat{f}^{(m-1)}\left(\cdot\right) + v\,\cdot\,\hat{g}^{(m)}\left(\cdot\right)$$

   em que $0 < v \leq 1$ é o fator *comprimento do passo*.

5. Continue o processo de iteração entre os passos 2 a 4 até $m = M$, para alguma iteração de parada M.

A iteração de parada, que é o principal parâmetro de controle, pode ser determinada via validação cruzada ou algum critério de informação. A escolha do *comprimento do passo* v no passo 4 é de menor importância, porém recomenda-se que seja pequeno, como $v = 0,1$. Um menor valor de v tipicamente requer um maior número de iterações boosting e, portanto, maior tempo de computação. Quando escolhendo v "suficientemente pequeno", resultados empíricos mostram que a acurácia preditiva do modelo é a melhor dentre outros valores de v (BUHLMANN; HOTHORN, 2007).

2.2.1.2.1 Função Perda e Algoritmos Boosting

Vários algoritmos Boosting podem ser definidos especificando diferentes funções perda $\rho(\cdot, \cdot)$ e serão mostrados a seguir os algoritmos derivados de diferentes funções perdas. Dado o fato de a aplicação proposta neste trabalho apresentar uma resposta binária, ou seja, para $Y \in \{0,1\}$ com $p(\boldsymbol{x}) = P[Y = 1 \,|\boldsymbol{X} = \boldsymbol{x}]$. Seguindo as recomendações de Buhlmann e Hothorn (2007) é conveniente codificar a resposta por $\tilde{Y} = 2Y - 1 \in \{-1, 1\}$ apenas por uma questão de eficiência computacional. Considere o negativo da log-verossimilança da binomial como função perda:

$$\rho(y, p(\boldsymbol{x})) = -[y \ln p(\boldsymbol{x}) + (1 - y) \ln(1 - p(\boldsymbol{x}))] \tag{2.2}$$

por simplificação, a perda 2.2 será chamada daqui em diante de *perda binomial* (Figura 2). Sendo $p(\boldsymbol{x})$ dado por

$$p(\boldsymbol{x}) = \frac{e^{f(\boldsymbol{x})}}{e^{f(\boldsymbol{x})} + e^{-f(\boldsymbol{x})}} \tag{2.3}$$

tal que

$$f(\boldsymbol{x}) = \frac{1}{2}\ln\left(\frac{p(\boldsymbol{x})}{1-p(\boldsymbol{x})}\right) \tag{2.4}$$

é igual a metade do log da chance (*log-odds*). O fator $1/2$ permitirá que o minimizador populacional da perda em 2.5 seja a mesma que a perda exponencial em 2.7 abaixo. Então, a perda binomial é dada por

$$\rho(y, f(\boldsymbol{x})) = \ln\left(1 + e^{-2\tilde{y}f}\right) \tag{2.5}$$

que se torna um limite superior do erro por mal classificação, também conhecida por *função degrau*. Convém ressaltar que a diferença entre as perdas 2.2 e 2.5 é que a perda 2.2 depende de $p(\boldsymbol{x})$ e ao substituir $p(\boldsymbol{x})$ (dado na equação 2.3) em 2.2 e substituir Y por $\hat{Y}$, obtém-se a perda em 2.5, que depende de f.

Pode-se mostrar que o minimizador populacional da perda binomial em 2.5 é dado por

$$f^*(\boldsymbol{x}) = \frac{1}{2}\ln\left(\frac{p(\boldsymbol{x})}{1-p(\boldsymbol{x})}\right) \tag{2.6}$$

em que $p(\boldsymbol{x})$ é como definido acima.

Uma função perda alternativa à binomial é a perda exponencial (Figura 2), dada pela expressão 2.7.

$$\rho(y,\ f) = e^{-\tilde{y}f} \tag{2.7}$$

cujo minimizador populacional pode ser mostrado como o mesmo para perda binomial (expressão 2.6) (FRIEDMAN; HASTIE; TIBSHIRANI, 2001).

Frente ao exposto, utilizar Boosting - FGD com diferentes funções perdas leva a diferentes algoritmos Boosting. Quando usando a perda binomial (2.5), ob-

temos o algoritmo Binomial Boosting e, com a perda exponencial em 2.7, obtemos o algoritmo AdaBoost para estimação funcional.

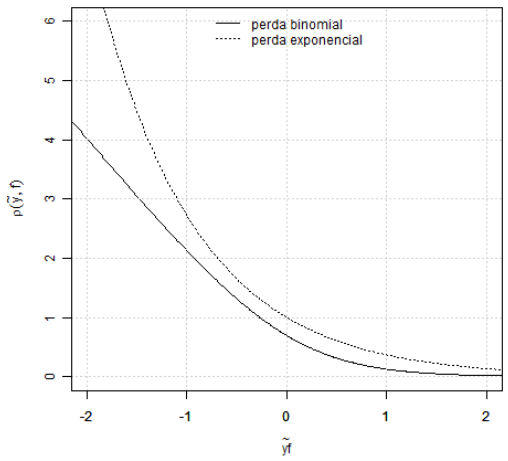

Figura 2 Funções perda binomial e exponencial como função do valor marginal $\tilde{y}f$

Importante ressaltar que a interpretação da estimativa Boosting $\hat{f}^{(m)}(\cdot)$ é feita como uma estimativa do minimizador populacional $f^*(\cdot)$. Dessa forma, os resultados do algoritmo Adaboost e Binomial Boosting correspondem as estimativas da metade do log da chance. Em particular, definimos as estimativas de probabilidade via

$$\hat{p}(\boldsymbol{x}) = \frac{e^{\hat{f}^{(m)}(\boldsymbol{x})}}{e^{\hat{f}^{(m)}(\boldsymbol{x})} + e^{-\hat{f}^{(m)}(\boldsymbol{x})}} \tag{2.8}$$

A razão da construção dessas estimativas de probabilidades estão baseadas no fato de que Boosting com iteração de parada razoável é consistente (BARTLETT; TRASKIN, 2007).

Para regressão com resposta $Y \in \mathbb{R}$, é conveniente utilizar a perda erro

quadrático, também conhecida como perda L_2 (Figura 3),

$$\rho(y,\, f) = (y - f)^2 \tag{2.9}$$

com minimizador populacional

$$f^*(\boldsymbol{x}) = E[Y \,|\boldsymbol{X} = \boldsymbol{x}] \tag{2.10}$$

O correspondente algoritmo Boosting é L_2Boosting. Uma função perda alternativa, que tem a propriedade de ser robusta, é a perda absoluta (perda L_1, Figura 3) e sua forma é

$$\rho(y,\, f) = |y - f| \tag{2.11}$$

e corresponde ao algoritmo L_1Boosting, cujo minimizador populacional é

$$f^*(\boldsymbol{x}) = mediana\,(Y \,|\boldsymbol{X} = \boldsymbol{x}) \tag{2.12}$$

Embora a perda L_1 seja não diferenciável no ponto $y = f$, podemos calcular derivadas parciais uma vez que o ponto $y = f$ tem probabilidade zero de ser realizado por um dado.

As perdas L_1 e L_2 são funções não monotônicas de valor marginal $\tilde{y}f$. Um aspecto negativo é que elas penalizam valores marginais que são maiores do que 1 e penalizar altos valores marginais pode ser visto como um modo de estimular soluções $\hat{f} \in [-1, 1]$ que é o alcance dos minimizadores populacionais L_1 e L_2, respectivamente (BUHLMANN; HOTHORN, 2007).

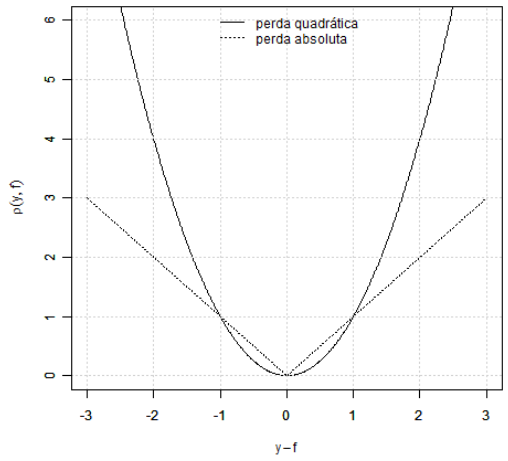

Figura 3 Função perda erro quadrático como função dos valores marginais de $y - f$

2.2.1.2.2 Mínimos Quadrados Linear Componente a Componente para Modelos Lineares

Boosting pode ser muito útil para ajustar modelos lineares generalizados em dimensões maiores. Considere o procedimento base

$$\hat{g}\left(x\right) = \hat{\beta}^{(\hat{\lambda})} x^{(\hat{\lambda})} \tag{2.13}$$

em que

$$\hat{\beta}^{(j)} = \frac{\sum_{i=1}^{n} X_i^{(j)} z_i}{\sum_{i=1}^{n} \left(X_i^{(j)}\right)^2} \tag{2.14}$$

e

$$\hat{\lambda} = \underset{1 \leq j \leq p}{\arg\min} \sum_{i=1}^{n} \left(z_i - \hat{\beta}^{(j)} X_i^{(j)} \right)^2 \tag{2.15}$$

Realizando esse procedimento, automaticamente é realizado o processo de seleção de variáveis em um modelo de regressão múltipla. Por essa razão e utilizando o procedimento base em 2.13, diz-se que o procedimento de seleção de variáveis está embutido no algoritmo Gradiente Boosting de Friedman (FRIEDMAN, 2001).

Quando utilizando L_2Boosting com esse procedimento base, selecionamos em cada iteração uma variável preditora, não necessariamente uma diferente para cada iteração, e atualizamos a função linearmente:

$$\hat{f}^{(m)}(x) = \hat{f}^{(m-1)}(x) + v \cdot \hat{\beta}^{(\hat{\lambda}_m)} x^{(\hat{\lambda}_m)} \tag{2.16}$$

em que $\hat{\lambda}_m$ denota o index da variável preditora na iteração m. Alternativamente, a atualização dos coeficientes estimados é

$$\hat{\beta}^{(m)} = \hat{\beta}^{(m-1)} + v\hat{\beta}^{(\hat{\lambda}_m)} \tag{2.17}$$

A notação deve ser lida que apenas o $\hat{\lambda}_m$-ésimo componente dos coeficientes estimados $\hat{\beta}^{(m)}$ (na iteração m) foi atualizado. Para cada iteração m, obtemos o ajuste de um modelo linear. Conforme m tende ao infinito, $\hat{f}^{(m)}(\cdot)$ converge para a solução de mínimos quadrados que é única se a matriz de delineamento tem posto completo.

No APÊNDICE B apresentamos uma ilustração didática do Algoritmo Gradiente Boosting de Friedman, bem como o processo de seleção de variáveis e a construção do modelo via Boosting.

Quando utilizando L_2Boosting com Mínimos Quadrados componente a

componente linear de 2.14, um valor inicial adequado é calcular a média da variável resposta Y. O vetor gradiente negativo é dado por

$$z_i = -\frac{\partial \rho\,(y,\, f)}{\partial f} = y - f \tag{2.18}$$

em que f é o procedimento base utilizado (APÊNDICE C).

Quando usando Binomial Boosting com Mínimos Quadrados componente a componente linear de 2.14, obtemos um ajuste, incluindo seleção de variáveis, de um modelo de regressão logística linear (BUHLMANN; HOTHORN, 2007). Um valor inicial adequado para esse algoritmo é calcular a frequência relativa de $Y = 1$ da amostra (APÊNDICE D). O vetor gradiente negativo para a perda binomial é dado por

$$z_i = -\frac{\partial \rho\,(y,\, f)}{\partial f} = y_i - \frac{1}{1 + e^{-f}} \tag{2.19}$$

em que f é o procedimento base utilizado (APÊNDICE D). Assim, o algoritmo Binomial Boosting utiliza a frequência relativa de $Y = 1$ e o vetor z_i para percorrer o espaço paramétrico do modelo proposto (BERK, 2008).

2.3 Regressão Logística

Nos modelos de regressão linear simples ou múltipla, a variável dependente Y é uma variável aleatória de natureza contínua. No entanto, em algumas situações, a variável dependente é qualitativa e expressa por duas ou mais categorias, ou seja, admite dois ou mais valores. Nesse caso, o método dos mínimos quadrados não oferece estimadores plausíveis. Uma boa aproximação é obtida pela regressão logística que permite o uso de um modelo de regessão para se calcular ou prever a probabilidade de um evento específico (PAULA; TUDER, 1986).

As categorias ou valores que a variável dependente assume podem ser de natureza nominal ou ordinal. Em caso de natureza ordinal, há uma ordem natural entre as possíveis categorias e, então, tem-se o contexto da Regressão Logística Ordinal. Quando essa ordem não existe entre as categorias da variável dependente assume-se o contexto da Regressão Logística Nominal.

O seguinte exemplo ilustra uma situação em que a variável dependente possui natureza nominal. Suponha que se deseja estudar a toxicidade de uma certa droga e as categorias são: o animal morreu após administração da dose x $(Y = 1)$ ou o animal não morreu após administração da dose x $(Y = 0)$. Nesse contexto, dosagens $x_1 < x_2 < \ldots < x_n$ são fixadas. A dosagem x_i geralmente é expressa como o logaritmo na base dez da concentração da droga em uma solução e é administrada em uma quantidade c_i de animais. Após esse procedimento, ocorre um número p_i de mortes para cada i, com $1 \leq i \leq n$. Assume-se que $\pi(x)$ é a probabilidade que um animal escolhido aleatoriamente sucumba com a dosagem x. Dessa forma, p_i, $1 \leq i$, são variáveis aleatórias independentes com distribuição binomial $Bin(c_i, \pi(x_i))$, com $i \in \{1, \ldots, n\}$. O objetivo aqui é encontrar um modelo no qual, para cada valor da variável independente x_i, é possível predizer a variável dependente $p(x_i)$, a qual é binomial com probabilidade de sucesso $\pi(x_i)$.

2.3.1 Regressão Logística Binária

Nesta seção apresenta-se o contexto em que a variável resposta possui apenas duas categorias, ou seja, natureza binária ou dicotômica.

Antes de se iniciar a discussão sobre a regressão logística, é interessante fazer um breve comentário sobre Modelos Lineares Generalizados (MLG). Um modelo linear generalizado é especifiado por três componentes: uma componente

aleatória, a qual identifica a distribuição de probabilidade da variável dependente, uma componente sistemática, que especifica uma função linear entre as variáveis independentes e uma função de ligação, que descreve a relação matemática entre a componente sistemática e o valor esperado da componente aleatória (HOSMER; LEMESHOW, 1989).

Em outras palavras, a componente aleatória de um MLG consiste nas observações da variável aleatória Y, ou seja, com o vetor $\mathbf{y} = (y_1, y_2, \ldots, y_n)$.

A componente sistemática do MLG é definida através de um vetor $\eta = (\eta_1, \eta_2, \ldots, \eta_n)$ que está associado ao conjunto das variáveis independentes por meio de um modelo linear $\eta = \boldsymbol{x}\boldsymbol{\beta}$, onde $\boldsymbol{x}$ é uma matriz que consiste nas variáveis independentes das n observações e $\boldsymbol{\beta}$ é um vetor de parâmetros do modelo.

A terceira componente do MLG é a função de ligação entre as componentes aleatória e sistemática. Seja $\mu_i = E\left[Y_i \,|\boldsymbol{x}_i\right]$, com $i \in \{1, \ldots, n\}$, então η_i é definida por $\eta_i = g\left(\mu_i\right)$, onde g é uma função monotônica e diferenciável.

Dessa forma, a função de ligação conecta os valores esperados das observações às variáveis explanatórias, para $i \in \{1, \ldots, n\}$, pela fórmula

$$g\left(\mu_i\right) = \sum_{j=1}^{p} \beta_j \; x_{i\,j} \tag{2.20}$$

em que p é o número de variáveis independentes no modelo.

É interessante comentar que, se a função g, dada por 2.20, for a função identidade, tem-se então o modelo de regressão linear.

Dependendo da natureza da componente aleatória de um MLG, existe um MLG adequado para cada situação. Se a componente aleatória for de natureza binária, os modelos *logit*, *probit* e *gompit* (*complemento log-log*) são adequados. Se a componente aleatória consiste do resultado de contagens, os modelos log-linear de Poisson e Binomial Negativo são candidatos. Para situações cuja resposta

é contínua e assimétrica, os modelos Gama são candidatos (PAULA; TUDER, 1986).

Na sequência, apresenta-se o modelo de regressão logística binário, que é um caso particular dos modelos lineares generalizados, mais especificamente dos modelos *logit*.

Para se analisar $\pi(\boldsymbol{x})$, tomam-se as observações independentes x_1, x_2, ..., x_n. Nesse contexto, é razoável assumir, como suposição inicial, que $\pi(\boldsymbol{x})$ é uma função monotônica com valores entre zero e um quando $\boldsymbol{x}$ varia na reta real, ou seja, $\pi(\boldsymbol{x})$ é uma função de distribuição de probabilidade.

Como $\pi(\cdot)$ varia entre zero e um, uma representação linear simples para π sobre todos os possíveis valores de $\boldsymbol{x}$ não é adequada, uma vez que os valores da forma linear estão no intervalo $(-\infty; +\infty)$. Nesse caso, uma transformação deve ser utilizada a fim de permitir que, para qualquer valor de $\boldsymbol{x}$, tenha-se um valor correspondente para $\pi(\cdot)$ no intervalo $[0; 1]$. Considere a transformação logística, também chamada de *logit*, logo

$$logit = \ln\left(\frac{\pi(\boldsymbol{x})}{1-\pi(\boldsymbol{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p \tag{2.21}$$

A razão $\frac{\pi(\boldsymbol{x})}{1-\pi(\boldsymbol{x})}$ é chamada de chance (*odds*). Seja A um evento de interesse, logo a chance do evento A é a relação entre probabilidade de ocorrência de A e a probabilidade de não ocorrência de A. Suponha que a probabilidade de ocorrência de A é de 80%, então a chance de ocorrência desse evento é de 4 para 1, ou em porcentagem, de 400% (400 ocorrências para 100 não ocorrências). Da mesma forma, se um evento A tem chance de 0,25 (25% ou 1 para 4) de ocorrer, então a probabilidade de ocorrência de A é de 20%.

A chance varia na escala de $(0; +\infty)$. Então o logaritmo neperiano da chance (*ln odds*) varia em $(-\infty; +\infty)$. Na expressão 2.21, se $\pi(\boldsymbol{x}) = 0,5$, então

$logit = 0$. Se $\pi(\boldsymbol{x}) < 0,5$, então $logit < 0$ e se $\pi(\boldsymbol{x}) > 0,5$, então $logit > 0$.

Exponenciando a expressão 2.21, tem-se que

$$\begin{aligned} e^{\log it} &= e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p} \\ \frac{\pi(\boldsymbol{x})}{1-\pi(\boldsymbol{x})} &= e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p} \end{aligned}$$

O inverso da função logit é a função logística, que é dada por

$$\pi(\boldsymbol{x}) = \frac{\exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)}{1 + \exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)} \tag{2.22}$$

em que $\pi(\boldsymbol{x})$ varia em $[0; 1]$. No caso de termos uma variável independente no modelo, x_1, se $\beta_1 > 0$, π é crescente e se $\beta_1 < 0$, π é decrescente. Quando x tende ao infinito, $\pi(x)$ tende a zero quando $\beta_1 < 0$ e tende a um quando $\beta_1 > 0$. Assim, dessa forma, define-se qualitativamente a função de ligação (vide Figura 4) necessária ao modelo, definido na equação 2.22. Caso $\beta_1 = 0$, a variável resposta Y é independente da variável X, logo $\pi(x)$ é constante. O caso $\beta_0 = 0$ e $\beta_1 = 0$ corresponde a $\pi(x) = 0,5$ (Figura 4).

2.3.2 Estimação dos parâmetros do modelo de Regressão Logística Binária

Seja $\boldsymbol{\beta} = (\beta_0,\ \beta_1)$ o vetor de parâmetros relacionado com a probabilidade condicional $P(Y_i = 1 \,|x_i) = \pi(x_i)$, com $\pi(x_i)$ dado por

$$\pi(x_i) = \frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)}$$

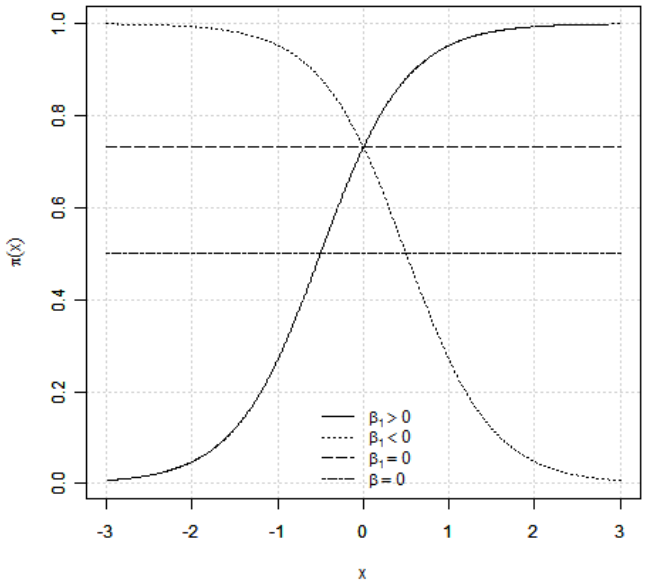

Figura 4 Ilustração do modelo logístico em uma variável independente

O método usual para estimar $\hat{\boldsymbol{\beta}}$ é via Máxima Verossimilhança. Sejam as probabilidades $P\left(y_i = 1 \left| x_i \right.\right) = \pi\left(x_i\right)$ e $P\left(y_i = 0 \left| x_i \right.\right) = 1 - \pi\left(x_i\right)$. Então, para os pares $\left(x_i,\, y_i\right)$ tais que $y_i = 1$, a contribuição para a função de verossimilhança é $\pi\left(x_i\right)$, e para os pares tais que $y_i = 0$, a contribuição para a função de verossimilhança é $1 - \pi\left(x_i\right)$, onde a quantidade $\pi\left(x_i\right)$ denota o valor de $\pi\left(x\right)$ avaliado em x_i. Como y_i tem distribuição Bernoulli, a contribuição de (x_i, y_i) à função de verossimilhança é dada por

$$\pi\left(x_i\right)^{y_i}\left[1 - \pi\left(x_i\right)\right]^{1-y_i}$$

para valores $y_i = 0$ ou $y_i = 1$, para todo $i \in \{1,\, \ldots,\, n\}$.

Como assume-se que as observações são independentes, a função de ve-

rossimilhança, L, obtida é dada por

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi(x_i)^{y_i} [1 - \pi(x_i)]^{1-y_i} \tag{2.23}$$

Aplicando-se logaritmo neperiano na equação 2.23 tem-se a expressão 2.24, ($l(\boldsymbol{\beta})$),

$$l(\boldsymbol{\beta}) = \ln [L(\boldsymbol{\beta})] = \sum_{i=1}^{n} [y_i \ln(\pi(x_i)) + (1 - y_i) \ln(1 - \pi(x_i))] \tag{2.24}$$

O princípio da máxima verossimilhança atesta que o estimador $\hat{\boldsymbol{\beta}}$ é o valor que maximiza a expressão 2.24. Assim, deriva-se $l(\boldsymbol{\beta})$ com respeito a β_0 e β_1 e igualam-se as expressões resultantes a zero, obtendo-se, respectivamente, as equações

$$\sum_{i=1}^{n} [y_i - \pi(x_i)] = 0 \tag{2.25}$$

$$\sum_{i=1}^{n} x_i [y_i - \pi(x_i)] = 0 \tag{2.26}$$

No modelo de regressão linear as equações de verossimilhança são facilmente resolvidas. Para o modelo de regressão logística, tais equações são não-lineares nos parâmetros e dessa forma, requer-se o uso de um procedimento iterativo conhecido como o método de Newton-Raphson.

Vamos fazer a derivação usando o método iterativo de Newton-Raphson considerando uma covariável no preditor linear, pois a forma múltipla é obtida de forma análoga ao caso simples com as devidas modificações. Como primeiro passo desse método, deve-se obter a matriz Hessiana ($I_{\boldsymbol{\beta}}$), cujos elementos da diagonal

principal são as derivadas de segunda ordem de 2.24 em relação a cada parâmetro, nesse caso, β_0 e β_1 e nos elementos fora da diagonal as derivadas parcias cruzadas de segunda ordem dos parâmetros.

A derivada parcial de primeira ordem, $\frac{\partial l(\boldsymbol{\beta})}{\partial \beta_0}$ e $\frac{\partial l(\boldsymbol{\beta})}{\partial \beta_1}$, são dadas por

$$\frac{\partial l\left(\boldsymbol{\beta}\right)}{\partial \beta_0} = \sum_{i=1}^{n}\left[y_i - \frac{\exp\left\{\beta_0 + \beta_1 x_i\right\}}{1+\exp\left\{\beta_0 + \beta_1 x_i\right\}}\right]$$

$$\frac{\partial l\left(\boldsymbol{\beta}\right)}{\partial \beta_1} = \sum_{i=1}^{n}\left[y_i x_i - \frac{\exp\left\{\beta_0 + \beta_1 x_i\right\}}{1+\exp\left\{\beta_0 + \beta_1 x_i\right\}}\right]$$

As derivadas de segunda ordem, $\frac{\partial^2 l(\boldsymbol{\beta})}{\partial^2 \beta_0}$ e $\frac{\partial^2 l(\boldsymbol{\beta})}{\partial^2 \beta_1}$, são dadas por

$$\frac{\partial^2 l\left(\boldsymbol{\beta}\right)}{\partial^2 \beta_0} = \sum_{i=1}^{n}\left[-\frac{\exp\left\{\beta_0 + \beta_1 x_i\right\}}{1+\exp\left\{\beta_0 + \beta_1 x_i\right\}}\right]$$

$$\frac{\partial^2 l\left(\boldsymbol{\beta}\right)}{\partial^2 \beta_1} = \sum_{i=1}^{n}\left[-x_i^2\frac{\exp\left\{\beta_0 + \beta_1 x_i\right\}}{1+\exp\left\{\beta_0 + \beta_1 x_i\right\}}\right]$$

A derivada parcial de segunda ordem $\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \beta_0 \partial \beta_1}$ é dada por

$$\frac{\partial^2 l\left(\boldsymbol{\beta}\right)}{\partial \beta_0 \partial \beta_1} = \sum_{i=1}^{n}\left[-x_i\frac{\exp\left\{\beta_0 + \beta_1 x_i\right\}}{1+\exp\left\{\beta_0 + \beta_1 x_i\right\}}\right]$$

Com o cálculo das derivadas, já temos condições de montarmos a regra de Newton Raphson

$$\boldsymbol{\beta}^{(i+1)} = \boldsymbol{\beta}^{(i)} - \left(I_{\boldsymbol{\beta}}\right)^{-1} U_{\boldsymbol{\beta}}\left(l\left(\boldsymbol{\beta}\right)\right) \tag{2.27}$$

em que $U_{\boldsymbol{\beta}}\left(l\left(\boldsymbol{\beta}\right)\right)$ é o vetor gradiente, cujos componentes são $\frac{\partial l(\boldsymbol{\beta})}{\partial \beta_0}$ e $\frac{\partial l(\boldsymbol{\beta})}{\partial \beta_1}$ e $\boldsymbol{\beta}^{(i)}$ representa um valor inicial para a primeira iteração do método de Newton-

Raphson.

As primeiras derivadas parciais da equação 2.24 são chamadas também de função *escore*. Vamos expressar agora essas equações e o método de Newton-Raphson para o caso múltiplo de variáveis independentes e, para isso, é conveniente escrever essas equações e a matriz Hessiana em notação matricial. Seja $\boldsymbol{y}$ denotado como o vetor dos valores y_i, $\boldsymbol{X}$ a matriz de ordem $\mathrm{n} \times (\mathrm{p}+1)$ dos valores x_i, $\boldsymbol{p}$ o vetor das probabilidades ajustadas com o i-ésimo elemento $\hat{\pi}_i$ e $\boldsymbol{W}$ a matriz diagonal $n \times n$ dos pesos com o i-ésimo elemento da diagonal dado por $\hat{\pi}_i\left(1-\hat{\pi}_i\right)$. Então temos

$$\frac{\partial l\left(\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} = \boldsymbol{X}^T\left(\boldsymbol{y}-\boldsymbol{p}\right) \tag{2.28}$$

$$\frac{\partial^2 l\left(\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} = -\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X} \tag{2.29}$$

Um problema que pode surgir para o método de Newton-Raphson é que a inversa da matriz Hessiana pode não existir. Em situações do tipo, o método de Escore-Fisher tem sido utilizado e esse método consiste em substituir a matriz Hessiana no método de Newton-Raphson pela matriz de informação de Fisher esperada. Pelo resultado de Wedderburn (1976), a matriz de informação de Fisher esperada é dada por

$$E\left[-\frac{\partial^2 l\left(\boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T}\right] = \phi \boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X} \tag{2.30}$$

em que ϕ é o parâmetro de dispersão do modelo e no caso do modelo de regressão logística $\phi = 1$. Logo, a atualização pelo método de Escore-Fisher é dada por

$$\boldsymbol{\beta}^{(i+1)} = \boldsymbol{\beta}^{(i)} + \left(\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X}\right)^{-1} U_{\boldsymbol{\beta}}\left(l\left(\boldsymbol{\beta}\right)\right)$$

$$\boldsymbol{\beta}^{(i+1)} = \boldsymbol{\beta}^{(i)} + \left(\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X}\right)^{-1} \boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{p}) \tag{2.31}$$

O método de Quasi-Newton consiste em substituir a matriz Hessiana por $U_{\boldsymbol{\beta}} (l(\boldsymbol{\beta})) U_{\boldsymbol{\beta}} (l(\boldsymbol{\beta}))^T$, logo a atualização é dada por

$$\boldsymbol{\beta}^{(i+1)} = \boldsymbol{\beta}^{(i)} + \left[U_{\boldsymbol{\beta}} (l(\boldsymbol{\beta})) U_{\boldsymbol{\beta}} (l(\boldsymbol{\beta}))^T\right]^{-1} U_{\boldsymbol{\beta}} (l(\boldsymbol{\beta})) \tag{2.32}$$

2.3.3 Técnicas de Diagnóstico

Com o objetivo de detectar observações que influenciam no processo inferencial do modelo, serão apresentadas aqui as técnicas utilizadas para diagnosticar possíveis pontos discrepantes. Estudos de simulação têm sugerido o resíduo padronizado t_{D_i} para as análises de diagnóstico em MLG, uma vez que o mesmo tem apresentado nesses estudos propriedades similares àquelas do resíduo da regressão normal linear (WILLIAMS, 1984). Em particular, para os modelos binomiais, esse resíduo é expresso, para $0 < y_i < n_i$, na forma

$$t_{D_i} = \pm \sqrt{\frac{2}{1 - \hat{h}_{ii}}} \left[y_i \ln\left(\frac{y_i}{n_i \hat{\pi}_i}\right) + (n_i - y_i) \ln\left(\frac{n_i - y_i}{n_i - n_i \hat{\pi}_i}\right) \right]^{\frac{1}{2}} \tag{2.33}$$

em que o sinal é o mesmo de $y_i - \hat{y}_i$. Se n_i for referente ao modelo binomial, y_i representa o número de sucessos ($Y = 1$) numa sequência de n_i tentativas independentes. Se n_i for referente ao modelo Bernoulli, Y_i representa o evento de interesse $Y = 1$ em um ensaio e, nesse caso, $n_i = 1$.

Para se medir a influência das observações nas estimativas dos coeficien-

tes, utilizamos a distância de Cook (LD) aproximada dada por

$$LD_i = \frac{1}{\left(1 - \hat{h}_{ii}\right)^2} \frac{\left(y_i - n_i\hat{\pi}_i\right)^2}{n_i\hat{\pi}_i\left(1 - \hat{\pi}_i\right)} \tag{2.34}$$

em que uma observação pode ser considerada como influente se $LD_i > 0,5$.

Hosmer e Lemeshow (1989) observam que $\hat{h}_{ii}$ depende das probabilidades ajustadas $\hat{\pi}_i$, $i = 1, \ldots, k$, e consequentemente os resíduos t_{D_i} e a medida de influência LD_i também dependem. O valor $\hat{h}_{ii}$, também denominado de *leverage*, é dado por

$$\hat{h}_{ii} = n_i\hat{\pi}_i\left(1 - \hat{\pi}_i\right)\boldsymbol{x}_i^T\left(\boldsymbol{X}^T\boldsymbol{V}\boldsymbol{X}\right)^{-1}\boldsymbol{x}_i \tag{2.35}$$

em que $\boldsymbol{V} = diag\left\{n_1\hat{\pi}_1\left(1 - \hat{\pi}_1\right), \ldots, n_n\hat{\pi}_n\left(1 - \hat{\pi}_n\right)\right\}$. Eles mostraram ainda por um estudo numérico que o comportamento de $\hat{h}_{ii}$ numa regressão logística pode ser muito diferente do comportamento de $\hat{h}_{ii}$ na regressão linear para uma mesma matriz $\boldsymbol{X}$. Os resultados de $\hat{h}_{ii}$, t_{D_i} e LD_i são apresentados em gráficos, que são informativos quanto ao posicionamento dos pontos aberrantes e influentes com relação às probabilidades ajustadas. Nesses gráficos, os pontos mais afastados dos demais são candidatos a serem aberrantes e/ou influentes (PAULA, 1995).

Outro gráfico utilizado para verificar a adequabilidade do modelo de regressão logística é o gráfico normal de probabilidades para o resíduo t_{D_i}, que indica se existem evidências de afastamento da suposição de distribuição binomial para a resposta. Consiste em gerar bandas de confiança por reamostragem, também chamado de envelope, e um ajuste adequado ocorre se todos os resíduos (ou grande parte deles) do modelo estiverem contidos nessas bandas de confiança. Mais detalhes sobre o envelope simulado podem ser vistos em Atkinson (1995).

2.3.4 Método *Stepwise* de Seleção de Variáveis

O método *stepwise* de seleção de variáveis consiste em eliminar do modelo variáveis que não contribuem de maneira significativa para o valor esperado da variável resposta, no caso do modelo logístico, para a probabilidade de ocorrência de um evento de interesse. A permanência de variáveis não significativas no modelo pode trazer problemas como a existência de multicolinearidade no mesmo, ou seja, as estimativas dos parâmetros do modelo podem não ser obtidas.

A ideia básica é selecionarmos um modelo que seja parcimonioso, ou, em outras palavras, que esteja bem ajustado e tenha um número reduzido de parâmetros. Para isso, utiliza-se algum critério para que sejam efetuadas as comparações. Nesse trabalho será utilizado o critério de informação de Akaike (AIC), logo, o método *stepwise* de seleção de variáveis consiste dos seguintes passos: (1) ajustamos o modelo completo com todas variáveis independentes; (2) retiramos uma variável independente por vez, ajustamos o modelo e calculamos o AIC; (3) retiramos do modelo completo a variável independente que produziu o maior AIC; (4) reajustamos o modelo sem a variável independente retirada no passo (3); (5) voltamos ao passo (2) e refazemos o processo até não haver variável independente para ser retirada. Após esse procedimento e com o modelo obtido, refazemos o processo inverso do passo (1), ou seja, incluiremos, se possível, variáveis independentes no modelo utilizando o AIC.

2.4 Critérios de Adequabilidade de Ajuste

Serão apresentados a seguir os critérios de adequabilidade de ajuste utilizados neste trabalho. Inicialmente, por meio da curva ROC pode-se avaliar o poder

de discriminação de um modelo e por ela pode-se também extrair os resultados da sensibilidade, especificidade, taxa de falsos positivos, taxa de falsos negativos e acurácia dos modelos ajustados. Em seguida será apresentado o teste de bondade de ajuste de Hosmer-Lemeshow, os critérios de informação de Akaike e Bayesiano. Para encerrar esta seção, será apresentado a medida de associação entre uma determinada variável independente e a variável resposta de um modelo logístico, conhecida por razão de chances.

2.4.1 A Curva ROC

Uma forma de avaliar o desempenho de modelos com resposta binária é verificar a quantidade de acertos do modelo. Esse sucesso do modelo pode ser avaliado com a curva ROC (*Receiver Operating Characteristic*). É aplicada em testes de classificação em visão computacional, assim como é utilizada em diagnóstico médico por imagens (HANLEY, 1989). Por extensão é aplicada em qualquer situação onde deseja-se avaliar a qualidade da classificação.

A curva ROC é um gráfico da *sensibilidade* (proporção de verdadeiros positivos) da predição do modelo contra o complemento de sua *especificidade* (proporção de falsos positivos), em uma série de limiares para um resultado positivo. Um limiar é um valor contido no intervalo $[0, 1]$ tal que converta uma probabilidade estimada em um valor binário, que pode ser 0 ou 1. Por exemplo, se $\hat{\pi}(x) \geq limiar \Rightarrow \hat{Y} = 1$ e se $\hat{\pi}(x) < limiar \Rightarrow \hat{Y} = 0$.

O modelo logístico retorna como resultado a probabilidade de um evento específico, no nosso caso, a probabilidade de uma pessoa ter uma doença coronária cardíaca (CHD). Essa probabilidade pode ser convertida para um resultado binário de acordo com a escolha de um limiar. Os valores correspondentes à conversão

das probabilidades em resultados binários, quando comparados com os valores observados, resultam nos valores de TP, TN, FP e FN (especificados a seguir) e podem ser organizados em uma tabela, chamada de Tabela de Confusão, como mostra a Tabela 1.

Tabela 1 Tabela de confusão.

Observado	**Predição do Modelo**	
	Positivo	Negativo
Positivo	TP	FN
Negativo	FP	TN

A Tabela 1 pode ser vista sob o ponto de vista da teoria de teste de hipóteses. Seja a hipótese nula H_0 definida como a situação em que um paciente não tem CHD e a hipótese alternativa H_1 relacionada com a condição de presença de CHD no paciente. A *taxa de falsos positivos* é equivalente à taxa de erro tipo I , denotado por α, ou seja, na verdade o paciente não tem CHD e o modelo estimou que o mesmo tem CHD. A taxa de *falso negativo* é equivalente à taxa de erro *tipo II*, denotado por β, ou seja, na verdade a pessoa tem CHD e o modelo decidiu que a mesma não tem CHD. O *poder do teste* é dado por $1 - \beta$ e representa a taxa de verdadeiros positivos do modelo (equivalente à *sensibilidade* do modelo), ou seja, a pessoa na verdade tem CHD e o modelo decidiu de maneira correta para essa condição. Por último, a taxa de verdadeiros negativos é dada por $1 - \alpha$ (equivalente à *especificidade* do modelo). A Tabela 2 resume as relações existentes para a hipótese nula.

Para qualquer limiar pode-se calcular a *sensibilidade* e a *especificidade* do modelo, comparando-se os valores preditos e os observados. A *sensibilidade* é definida como a habilidade do modelo encontrar as respostas positivas, isto é, as

Tabela 2 Representação tabular dos resultados possíveis em um teste de hipóteses e os erros e acertos que eles acarretam.

Verdade	**Decisão**	
	Aceita-se H_0	Rejeita-se H_0
H_0 é verdadeira	$1-\alpha$	α
H_0 é falsa	β	$1-\beta$

pessoas que realmente tem CHD, logo

$$sensibilidade = \frac{TP}{TP + FN} \tag{2.36}$$

em que TP é o número de verdadeiros positivos e FN o número de falsos negativos preditos pelo modelo.

A *especificidade* do modelo é definida como a proporção de verdadeiros negativos preditos pelo modelo, ou seja, a proporção de pessoas que realmente não têm CHD que o modelo preveu. Logo

$$especificidade = \frac{TN}{TN + FP} \tag{2.37}$$

em que TN é a quantidade de verdadeiros negativos e FP a quantidade de falsos positivos preditos pelo modelo.

Assim, pode-se obter a *acurácia* do modelo, que mede a capacidade do modelo em classificar corretamente pessoas que têm e que não têm problema no coração, e é dada por

$$acuracia = \frac{TP + TN}{TP + FN + TN + FP} \tag{2.38}$$

O complemento da *especificidade* é a *taxa de falsos positivos*, ou seja, a

proporção de predições incorretas de positivos (evento de interesse) em relação ao total de negativos (complementar do evento de interesse) observados. Similarmente, o complemento da *sensibilidade* é a *taxa de falsos negativos*, ou seja, a proporção de incorretas predições negativas em relação ao total de positivos. Note que a soma da *sensibilidade* e a *taxa de falsos negativos* deve ser 1. O mesmo ocorre somando-se a *especificidade* e a *taxa de falsos positivos*.

Assim, a curva ROC é um gráfico que relaciona a *sensitividade* (no eixo y) e a *taxa de falsos positivos* (no eixo x) em diferentes limiares (Figura 5). Idealmente, até mesmo em baixos limiares, o modelo prediziria mais verdadeiros positivos com poucos falsos positivos, então a curva se aproximaria rápido do ponto (0,0). Quanto mais próximo da borda do lado esquerdo, e em seguida da borda superior do gráfico, mais acurado é o modelo, ou seja, possui sensibilidade e especificidade elevadas, mesmo em baixos limiares. Quanto mais perto a curva vem para a diagonal, menos acurado é o modelo. A diagonal representa uma escolha ao acaso, ou seja, o modelo prediz ao acaso, então a probabilidade de um verdadeiro positivo é igual a de um falso positivo para qualquer limiar.

Uma característica da Figura 5 é que por ela é possível definir um limiar adequado para a situação, ou seja, um limiar que retorne valores para a sensibilidade e especificidade relativamente altos. Por exemplo, pela Figura 5, um limiar de 0,5 parece razoável, pois retornam valores de aproximadamente 80% e 90% para a sensibilidade e especificidade, respectivamente.

A área sob a curva ROC (AUC - *area under curve*) é calculada pela regra do trapézio, ou seja,

$$AUC = \sum_{i=1}^{n} (x_{i+1} - x_i) \left(\frac{y_{i+1} + y_i}{2} \right) \tag{2.39}$$

em que i ($i = 1, ..., n$) é o limiar onde a curva é calculada. Note que a área

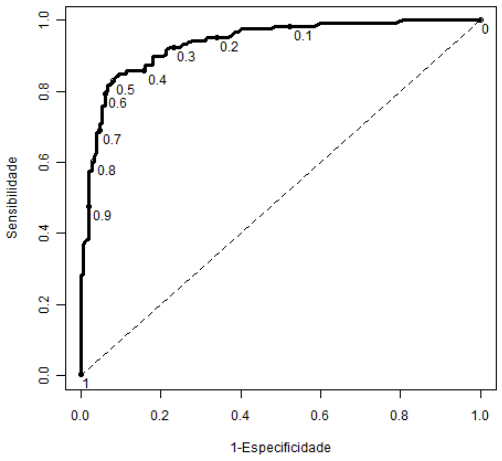

Figura 5 Ilustração da curva ROC

sob a diagonal é 0,5 unidades de área (u.a.), logo é desejável que a curva ROC defina uma área no mínimo maior do que essa diagonal. O AUC mede, portanto, o poder de discriminação do modelo, ou seja, o sucesso do modelo em classificar corretamente verdadeiros positivos e verdadeiros negativos.

Não existe um teste estatístico do AUC e seu valor depende do campo de aplicação. Como regra geral, uma discriminação é aceitável quando a área abaixo da curva ROC for maior que 0,7 u.a. e se for maior do que 0,8 u.a. a discriminação é dita excelente (FAVERO et al., 2009).

2.4.2 Teste de Hosmer-Lemeshow

Hosmer e Lemeshow (1989) propuseram dois diferentes tipos de agrupamentos baseados nas probabilidades estimadas. Suponha que $J = n$ em que teremos n probabilidades estimadas. Para fazer o teste, primeiramente ordenamos

as n probabilidades estimadas. Os dois agrupamentos são:

(a) Agrupamento 1: Baseado nos decis das probabilidades estimadas.

(b) Agrupamento 2: Pontos de corte são pré definidos.

Para o primeiro método, usamos g=10 grupos em que os primeiros $n_1^{'} = n/10$ são aqueles que contêm as menores probabilidades estimadas e $n_{10}^{'} = n/10$ são os com as maiores probabilidades estimadas. Para o segundo método, usamos g=10 com pontos de cortes definidos nos valores $k/10$, $k = 1, 2, ..., 9$, e os grupos contêm todos os indivíduos com probabilidades estimadas dentro dos limites do ponto de corte de cada grupo.

Antes do cálculo da estatística teste, é necessário estimar a frequência esperada. Para $Y = 1$, a frequência esperada estimada é dada pela soma das probabilidades estimadas de todos os indivíduos dentro daquele grupo. Para $Y = 0$, a frequência esperada estimada é dada pela soma de 1-probabilidade estimada de todos os indíviduos dentro daquele grupo.

Para cada estratégia de agrupamento, a estatística de Hosmer e Lemeshow, $\hat{C}$, é obtida da seguinte forma:

$$\hat{C} = \sum_{k=1}^{g} \frac{(o_k - n_k^{'}\bar{\pi}_k)^2}{n_k^{'}\bar{\pi}_k(1 - \bar{\pi}_k)}, \tag{2.40}$$

em que: $n_k^{'}$ é o número de indivíduos no k-ésimo grupo; $\bar{\pi_k} = \sum_{j=1}^{C_k} \frac{\bar{\pi}_j}{n_k^{'}}$; C_k é o número total de combinações de níveis dentro do k-ésimo decil; $O_k = \sum_{j=1}^{C_k} y_j$ é número total de respostas dentro do grupo k.

A estatística do teste de Hosmer e Lemeshow tem distribuição qui-quadrado com g-2 graus de liberdade. A hipótese nula do teste corresponde a um ajuste satisfatório do modelo. No ANEXO B é apresentado um exemplo do cálculo da estatística $\hat{C}$ utilizando o agrupamento 1.

2.4.3 Critérios de Informação de Akaike e Bayesiano

O Critério de informação de Akaike (AIC) proposto em Akaike (1974), é uma medida relativa da qualidade de ajuste de um modelo estatístico.

O AIC não é uma prova sobre o modelo, mas uma ferramenta útil na seleção de modelos. Para seu cálculo, não existe teste de hipóteses, significância e nem *valor-p*. É definido como:

$$AIC = -2\, l\,(\,\theta\,|y\,) + 2p \tag{2.41}$$

em que $l\,(\,\theta\,|y\,)$ é o logaritmo neperiano da função de verossimilhança do modelo em θ e p é o número de parâmetros do modelo.

Schwarz (1978) propôs um critério conhecido como Critério de Informação Bayesiano (BIC), que corresponde à troca do fator 2, que é o peso do número de parâmetros, em 2.41 por $\ln(n)$, logo o BIC é dado por:

$$BIC = -2\, l\,(\,\theta\,|y\,) + p\, \ln(n) \tag{2.42}$$

em que n é o número de observações da amostra.

Dado um conjunto de modelos ajustados aos dados, o modelo preferido é o que apresentar menor valor de AIC ou BIC, ou seja, quanto menor for o valor de AIC ou BIC melhor será o ajuste do modelo aos dados (AKAIKE, 1974).

2.5 Razão de Chances

Vamos considerar inicialmente o modelo logístico linear simples em que $\pi\,(x)$, a probabilidade de sucesso dado o valor x de uma variável independente

qualquer, é definida tal que

$$\ln\left\{\frac{\pi(x)}{1-\pi(x)}\right\} = \beta_0 + \beta_1 x$$

em que β_0 e β_1 são parâmetros desconhecidos. Esse modelo poderia ser aplicado, por exemplo, para analisar a associação entre uma determinada doença, no nosso caso a ocorrência de CHD, e a ocorrência ou não de um fator particular. Seriam então amostrados, independentemente, n_1 indivíduos com presença do fator ($x = 1$) e n_2 indivíduos com ausência do fator ($x = 0$) e $\pi(x)$ seria a probabilidade de ocorrência de CHD. Dessa forma, a chance (*odds*) de ocorrência de CHD para um indivíduo com presença ($x = 1$) do fator, denotada por OD_1, fica dada por

$$OD_1 = \frac{\pi(1)}{1-\pi(1)} = e^{\beta_0+\beta_1}$$

enquanto que a chance de ocorrência de CHD para um indivíduo com ausência ($x = 0$)do fator, denotado por OD_0, é

$$OD_0 = \frac{\pi(0)}{1-\pi(0)} = e^{\beta_0}$$

Logo, a razão de chances (*odds ratio*) de ocorrência de CHD entre indivíduos com presença e ausência do fator fica dada por

$$OR = \frac{\pi(1)[1-\pi(0)]}{\pi(0)[1-\pi(1)]} = e^{\beta_1} \tag{2.43}$$

dependendo apenas do parâmetro β_1. Esta é uma das grandes vantagens da regressão logística: a possibilidade de interpretação direta dos coeficientes como medidas de associação (HOSMER; LEMESHOW, 1989). Esse conceito pode ser extendido para o caso múltiplo de variáveis independentes, só que, nesse caso,

a razão de chances é feita entre a variável de interesse, como mencionado anteriormente, e mantendo-se fixas todas as outras variáveis explicativas, levando a equação 2.43.

O valor observado da variável independente no modelo logístico pode representar o valor de alguma variável quantitativa qualquer, como, por exemplo, o nível de colesterol sérico de um paciente. Nesse caso, faz sentido calcularmos a razão de chances de um indivíduo ser diagnosticado com CHD a cada incremento, que pode ser de uma unidade ou mais, no seu resultado de colesterol sérico. A razão de chances de diagnóstico de CHD para um incremento c, tal que $c = x^* - x$, fica dada por

$$OR_{(x^*-x)} = \frac{\pi\left(x^*\right)\left[1-\pi\left(x\right)\right]}{\pi\left(x\right)\left[1-\pi\left(x^*\right)\right]} = e^{\beta_1(x^*-x)}$$

Uma vez estimado $\hat{OR} = e^{\hat{\beta}_1}$, um intervalo assintótico de confiança para OR com coeficiente $(1-\alpha)$ é dado por

$$\left(\hat{OR}_I; \hat{OR}_S\right) = e^{\hat{\beta}_1 \pm z_{(1-\alpha/2)}\sqrt{Var\left(\hat{\beta}_1\right)}}$$

em que $Var\left(\hat{\beta}_1\right)$ é a variância da estimativa de $\hat{\beta}_1$. Num modelo de regressão logística com seus parâmetros estimados via Máxima Verossimilhança, $Var\left(\hat{\beta}_1\right)$ é obtido a partir da matriz de variâncias e covariâncias do modelo. No contexto de Boosting, como não é conhecida a distribuição amostral de $\hat{\beta}_1$, não é possível obter o intervalo de confiança para $\hat{OR}$.

3 MATERIAL E MÉTODOS

A seguir a metodologia proposta nesta dissertação. Inicialmente serão apresentados os dados que serão utilizados para ajustar os modelos logísticos. Em seguida o procedimento para estimar o modelo de regressão logística via algoritmo Boosting e pelo método da máxima verossimilhança, bem como os critérios para selecionar o melhor modelo para situação binária frente ao problema exposto.

3.1 Dados

Foram utilizados os dados disponibilizados por *UCI Machine Learning Repository* (FRANK; ASUNCION, 2010). Os dados são referentes a 270 pacientes com presença ou não de doença coronariana cardíaca (Coronary Heart Disease - CHD) e essa condição está em função de 13 variáveis independentes. Na Tabela 3 estão reunidas essas variáveis, bem como a natureza de cada uma e os possíveis valores que elas podem assumir.

A resposta que se pretende modelar é a condição presença/ausência de doença cardíaca coronariana (CHD), cuja representação é dada pela sigla *DIS*. Se $DIS = 1$ corresponde à presença de CHD no paciente e se $DIS = 0$ o paciente não possui CHD. Além da resposta, existem três variáveis de natureza binária, que são as variáveis independentes *SEX*, *SUG* e *EXE*. A variável *SEX* diz respeito ao sexo da pessoa (0: feminino; 1: masculino), a variável *SUG* está relacionada ao nível da glicemia no sangue da pessoa (0: $\leq$ 120 mg/dL; 1: $>$ 120 mg/dL) e a variável *EXE* está relacionada com a situação de angina induzida, que é a condição de que a pessoa pode sentir dor no peito mesmo quando em repouso (0: não; 1: sim). Existem três variáveis explicativas de natureza nominal, são elas: *PAIN*

Tabela 3 Relação das variáveis presentes no problema do diagnóstico de doença cardíaca coronariana (CHD).

Variável	**Natureza**	**Descrição**
AGE	contínua	em anos
SEX	binária	0: feminino
		1: masculino
PAIN	nominal; 4 níveis	1: angina típica
		2: angina atípica
		3: sem dor anginosa
		4: assintomático
PRESS	contínua	em mm/Hg
COL	contínua	em mg/dl
SUG	binária	0: $\leq$ 120mg/dL
		1: $>$ 120mg/dL
ELE	nominal; 3 níveis	1: normal
		2: com onda ST-T anormal
		3: mostrando provável hipertrofia do ventrículo esquerdo
HEART	contínua	em bpm
EXE	binária	0: não
		1: sim
ST	contínua	em milímetros
SLOPE	ordinária; 3 níveis	1: inclinação Ascendente
		2: inclinação horizontal
		3: inclinação Descendente
VES	discreta	0, 1, 2, ou 3
THAL	nominal; 3 níveis	3: normal
		6: defeito
		7 : defeito reversível
DIS	binária	0: ausente para CHD
		1: presente para CHD

refere-se ao tipo de dor no peito que pode ser classificada em quatro formas diferentes (1: angina típica; 2: angina atípica; 3: sem dor anginosa; 4: assintomático); a variável *ELE* está relacionada com o comportamento do segmento ST no eletrocardiograma, em que seus níveis 2 e 3 acusam anormalidade no resultado e o nível 3 é um indicativo de CHD (1: normal; 2: com onda ST-T anormal; 3: mostrando provável hipertrofia do ventrículo esquerdo); a variável THAL representa a Talassemia, que é uma doença hereditária que afeta o sangue da pessoa (3: normal; 6: defeito; 7: defeito reversível). A variável *SLOPE* está relacionada com a inclinação do segmento ST, que é o segmento do eletrocardiograma utilizado para diagnosticar eventos isquêmicos agudos e, por ser uma variável ordinária, seus três níveis levam à condição mais provável de isquemia (1: inclinação ascendente; 2: inclinação horizontal; 3: inclinação descendente). A variável VES, cuja natureza é discreta, representa o número de grandes vasos coloridos por fluoroscopia (0, 1, 2 ou 3). As outras variáveis são de natureza contínua e representam a idade do paciente (*AGE*) em anos, a pressão arterial em repouso (*PRESS*) em mm/Hg, o nível de colesterol sérico no sangue (*COL*) em mg/dL, a frequência cardíaca máxima atingida (*HEART*) em batimentos por minuto (bpm) e o comprimento do segmento ST do eletrocardiograma em milímetros (*ST*).

3.2 Ajuste do Modelo de Regressão Logística via Algoritmo Boosting

De acordo com a Tabela 3, ao todo são 13 variáveis independentes para serem ajustadas. Para estimar os parâmetros do modelo de regressão logística via algoritmo Boosting será utilizado o algoritmo Binomial Boosting (MRLBB). Para executar o algoritmo Binomial Boosting é necessário que sejam definidas duas componentes, sendo uma função perda (definido na seção 2.2.1.2.1) e um procedimento base (definido na seção 2.2.1.2.2). O algoritmo Binomial Boosting utiliza

a função perda binomial e o procedimento base mínimos quadrados componente a componente, uma vez que a resposta DIS configura uma situação binária e estamos interessados em ajustar um modelo linear generalizado.

O algoritmo Binomial Boosting, durante o processo de estimação paramétrica, já realiza seleção de variáveis, retornando, portanto, aquelas variáveis independentes que minimizam a função perda utilizada, levando ao modelo com as variáveis independentes que contribuem significativamente no modelo.

3.3 Ajuste do Modelo de Regressão Logística via Máxima Verossimilhança

De acordo com a Tabela 3, ao todo são 13 variáveis independentes para serem ajustadas. Para estimar os parâmetros do modelo de regressão logística via máxima verossimilhança (MRLMV), foi utilizado o método descrito na seção 2.3.2. Em seguida, foi utilizado o método *stepwise* de seleção de variáveis via AIC, com o objetivo de eliminar as variáveis independentes que não contribuem de forma significativa para a probabilidade de ocorrência de doença cardíaca coronariana em pacientes.

3.4 Comparação dos Modelos MRLBB e MRLMV

Para avaliar o desempenho dos modelos obtidos pelos dois métodos, o conjunto de dados foi separado em duas partes, sendo uma parte de treinamento, que será destinada à estimação dos parâmetros dos modelos MRLBB e MRLMV, e a parte de teste, que será destinada à validação dos modelos MRLBB e MRLMV. O conjunto de treinamento será constituído pelas partições de 30%, 40%, 50%, 60%, 70%, 80% e 90% da amostra original, que é de 270 pacientes. O complementar das partições constituirá o conjunto de teste. A validação será feita comparando-

se os critérios de informação de Akaike (AIC) e Bayesiano (BIC) dos modelos obtidos após processo de seleção de variáveis e o modelo preferido será aquele cujos critérios são menores.

Será utilizado o Teste de Hosmer-Lemeshow para verficar a existência de problemas de ajuste dos modelos MRLBB e MRLMV. A escolha da partição ideal (conjunto de treinamento e teste) será feita para a partição cujo resultado do Teste de Hosmer-Lemeshow, for não significativa para os modelos MRLBB e MRLMV.

Para determinar o limiar adequado a fim de classificar um paciente quanto à presença ou não de CHD, será utilizada a curva ROC em ambos modelos MRLB e MRLMV.

Em seguida, para os modelos MRLBB e MRLMV estimados com a partição ideal serão calculados a sensibilidade, especificidade, acurácia, taxa de falsos negativos, taxa de falsos positivos e AUC. Será julgado o modelo que apresentar os melhores valores para essas quantidades.

Serão calculadas as razões de chances de ocorrência de CHD para todas as variáveis independentes ajustadas pelos modelos MRLBB e MRLMV.

Finalizando a metodologia proposta, para obtenção dos resultados serão utilizados os pacotes estatísticos *mboost*, *ROCR* e *MKmisc* do Sistema Computacional Estatístico R (R DEVELOPMENT CORE TEAM, 2011), para realização das análises.

4 RESULTADOS E DISCUSSÃO

Inicilamente serão apresentados os resultados de treinamento e teste, a fim de obter a melhor partição para o conjunto de treinamento e teste para a situação presença/ausência de CHD. Na sequência, uma vez determinado o melhor corte no conjunto de dados, será proposto o modelo logístico estimado via algoritmo Boosting (MRLBB) e via máxima verossimilhança (MRLMV). Frente aos critérios de adequabilidade de ajuste, será selecionado o melhor modelo para explicar a ocorrência de CHD em pacientes. Por fim, serão apresentados os resultados das razões de chances estimadas, a fim de verificar as relações existentes entre as diversas variáveis independentes selecionadas para o modelo proposto e a ocorrência de doença cardíaca coronariana em pacientes.

4.1 Treinamento e Teste

A variável *DIS* representa uma situação de sucesso ou fracasso de um evento, logo pode ser associada a uma variável aleatória Bernoulli. O modelo completo para essa situação é dado por

$$P\left(DIS = 1 \,|\boldsymbol{X} = \boldsymbol{x}\right) = \pi\left(\boldsymbol{x}\right) = \frac{e^{g(\boldsymbol{x})}}{1 + e^{g(\boldsymbol{x})}} \tag{4.1}$$

em que

$$
\begin{aligned}
g(\boldsymbol{x}) =& \beta_0 + \beta_1 AGE + \beta_{21} SEX_1 + \beta_{22} SEX_2 \\
&+ \beta_{31} PAIN_1 + \beta_{32} PAIN_2 + \beta_{33} PAIN_3 + \beta_{34} PAIN_4 \\
&+ \beta_4 PRESS + \beta_5 COL + \beta_{61} SUG_1 + \beta_{62} SUG_2 \\
&+ \beta_{71} ELE_1 + \beta_{72} ELE_2 + \beta_{73} ELE_3 + \beta_8 HEART \\
&+ \beta_{91} EXE_1 + \beta_{92} EXE_2 + \beta_{10} ST \\
&+ \beta_{111} SLOPE_1 + \beta_{112} SLOPE_2 + \beta_{113} SLOPE_3 \\
&+ \beta_{12} VES + \beta_{131} THAL_1 + \beta_{132} THAL_2 + \beta_{133} THAL_3
\end{aligned}
$$

em que as variáveis independentes categóricas *SEX*, *PAIN*, *SUG*, *ELE*, *EXE*, *SLOPE* e *THAL* são do tipo dummy (assumem níveis de fatores) e assume-se que os primeiros níveis de cada um dessas variáveis independentes é zero, reportando, portanto, ao modelo condizente com o primeiro nível de cada fator.

A Tabela 4 apresenta os resultados dos critérios de informação de Akaike e Bayesiano para diversos cortes no conjunto de dados, bem como a quantidade de dados resultantes de cada corte para o conjunto de treinamento e teste. Observa-se que, para todos os cortes, o modelo MRLBB apresentou menores valores de AIC e BIC, sendo, portanto, o mais adequado.

Para avaliar o ajuste do modelo obtido por cada método, foi utilizado também o teste de Hosmer-Lemeshow e seus resultados (*valor-p*) são apresentados na Tabela 5. Observa-se que, considerando-se 5% como nível de significância, o modelo MRLBB para cada corte, em todos os casos, foi maior do que o nível de significância adotado, indicando que o ajuste do modelo é adequado. O mesmo não ocorre para o modelo MRLMV nos cortes de 30%, 40%, 50 % e 60% para os

Tabela 4 Resultados dos critérios de Informação de Akaike (AIC) e Bayesiano (BIC) em diversos conjuntos de treinamento e teste.

corte		**AIC**		**BIC**	
trein. (%)	**teste (%)**	**MRLBB**	**MRLMV**	**MRLBB**	**MRLMV**
30,00	70,00	62,4032	76,7070	73,6052	96,1570
40,00	60,00	77,8526	90,2510	91,2616	118,0397
50,00	50,00	98,1087	107,7300	113,2539	139,6903
60,00	40,00	123,3942	130,7300	137,8766	160,2329
70,00	30,00	144,1786	154,4300	160,4283	180,7688
80,00	20,00	160,2569	171,7100	177,5306	195,5802
90,00	10,00	174,1674	189,1100	193,1337	203,3674

conjuntos de treinamento, em que a hipótese nula de adequabilidade de ajuste é rejeitada ao nível de 5% de significância.

Tabela 5 Resultados do teste de Hosmer-Lemeshow (*valor-p*) em diversos conjuntos de treinamento e teste.

corte		**N**		**Hosmer-Lemeshow**	
treinamento (%)	**teste (%)**	**treinamento**	**teste**	**MRLBB**	**MRLMV**
30,00	70,00	81	189	0,4326	0,0030
40,00	60,00	108	162	0,5758	0,0001
50,00	50,00	135	135	0,5101	0,0001
60,00	40,00	162	108	0,5574	0,0265
70,00	30,00	189	81	0,1596	0,0506
80,00	20,00	216	54	0,2341	0,6549
90,00	10,00	243	27	0,7017	0,3996

4.2 Modelo Proposto

Tomando-se como referência os resultados encontrados nas Tabelas 4 e 5, vamos especificar agora o modelo proposto via algoritmo Binomial Boosting e o mesmo obtido via máxima verossimilhança, cujas estimativas dos parâmetros

são apresentadas na Tabela 6. Como visto nas Tabelas 4 e 5, foi feito um estudo do comportamento do modelo estimado por ambos métodos em diferentes cortes no conjunto de dados, no entanto, é aconselhável que o corte determine uma quantidade maior de dados no conjunto de treinamento e ficando o restante para o conjunto de teste, a fim de diminuir o viés proveniente desse processo. A literatura recomenda ainda que o conjunto de teste tenha observações o suficiente para representar o conjunto de treinamento. Sendo assim, um corte de 70% para o conjunto de treinamento, ficando 30% para oconjunto de teste, parece razoável e será o escolhido daqui em diante.

O modelo MRLBB é o que minimiza a função perda como mostrado na seção 2.2.1.2. Como trata-se de um método iterativo, a cada iteração do algoritmo é estimado um modelo e desse modelo é calculado o seu critério de informação de Akaike, logo, o modelo que minimiza a função perda nesse caso é também o que fornece o menor valor de AIC (da mesma forma o BIC). A Figura 6 mostra a evolução do AIC conforme aumenta-se o número de iterações do algoritmo.

Dessa forma, a Figura 6 ilustra que o número ótimo de iterações do algoritmo Binomial Boosting é 146 iterações, cujo AIC é de 144,1786 (Tabela 4). Observa-se ainda na Figura 6 a necessidade de que o algoritmo não seja executado indefinidamente, pois isso, além de aumentar o AIC, forçaria a inclusão de váriáveis não importantes no modelo. Logo, a probabilidade via MRLBB de um indivíduo $\boldsymbol{x}_i$ ter uma doença coronariana é estimada pela expressão $\pi_{Boost}\left(\boldsymbol{x}_i\right)$ em 4.2.

$$P\left(DIS_i = 1 \,|\boldsymbol{X}_i = \boldsymbol{x}_i\right) = \pi_{Boost}\left(\boldsymbol{x}_i\right) = \frac{e^{g_{Boost}(\boldsymbol{x}_i)}}{1 + e^{g_{Boost}(\boldsymbol{x}_i)}} \tag{4.2}$$

em que

$$
\begin{aligned}
g_{Boost}\left(\boldsymbol{x}_i\right) = & -4,6268 + 0,7979\,SEX_{2i} + 1,5284\,PAIN_{4i} + 0,0107\,PRESS_i \\
& +0,0028\,COL_i + 0,2730\,ELE_{3i} - 0,0053\,HEART_i + 0,5502\,EXE_{2i} \\
& +0,3762\,ST_i + 0,6459\,SLOPE_{2i} + 0,922\,VES_i + 0,325\,THAL_{3i}
\end{aligned}
$$

Observe que o algoritmo Binomial Boosting selecionou 11 das 13 variáveis independentes para o modelo final. Portanto, a probabilidade de ocorrência de CHD não é influenciada pela idade (AGE) das pessoas nem pelo nível de sua glicemia (SUG). Esse modelo explica ainda que, se a pessoa for do sexo masculino, a probabilidade de doença cardíaca coronariana é aumentada e essa ideia de

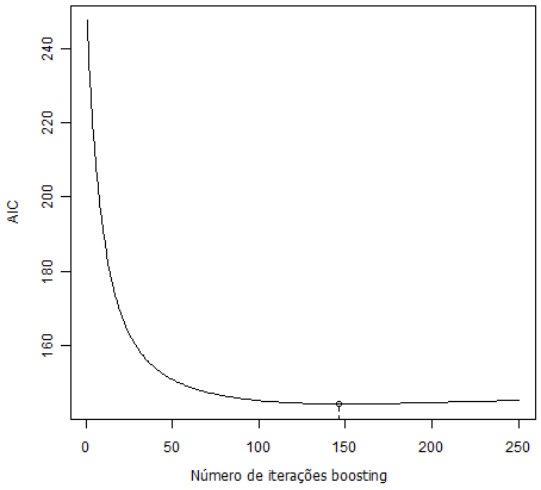

Figura 6 Gráfico da evolução do Critério de Informação de Akaike ao longo do número de iterações do algoritmo Binomial Boosting

aumento será melhor explicada pela razão de chances, cujos resultados estão reservados para a seção seguinte. O modelo explica ainda que a probabilidade de uma pessoa ter CHD sofre incremento se a pessoa apresentar dor no peito assintomática ($PAIN_4$), o resultado do eletrocardiograma em repouso ser classificado como alto (ELE_3), se o paciente tiver resultado positivo para angina induzida (EXE), a inclinação do segmento ST ao pico de exercício apresentar inclinação horizontal ($SLOPE_2$) e defeito reversível para a Talassemia ($THAL_3$). Essa probabilidade é incrementada ainda com as variáveis contínuas relacionadas à pressão arterial ($PRESS$), nível de colesterol (COL), frequência cardíaca ($HEART$), comprimento do segmento ST (ST) e número de grandes vasos coloridos por fluoroscopia (VES).

O modelo proposto via regressão logística utilizando o método da máxima verossimilhança (MRLMV), com a aplicação do método *stepwise* encontra-se com as estimativas descritas na Tabela 6.

A probabilidade de um indivíduo $\boldsymbol{x}_i$ ter uma doença coronariana é estimada pela expressão $\pi_{RL}(\boldsymbol{x}_i)$ em 4.3.

$$P(DIS_i = 1 \,|\boldsymbol{X}_i = \boldsymbol{x}_i) = \pi_{RL}(\boldsymbol{x}_i) = \frac{e^{g_{RL}(\boldsymbol{x}_i)}}{1 + e^{g_{RL}(\boldsymbol{x}_i)}} \tag{4.3}$$

em que

$$\begin{aligned} g_{RL}(\boldsymbol{x}_i) = &-10,7509 + 1,5066\, SEX_{2i} + 2,0030\, PAIN_{3i} + 3,9293\, PAIN_{4i} \\ &+0,0336\, PRESS_i + 1,7693\, SLOPE_{2i} + 1,9773\, SLOPE_{3i} \\ &+1,0290\, VES_i + 1,4205\, THAL_{3i} \end{aligned}$$

Tabela 6 Estimativas dos parâmetros referentes ao modelo logístico ajustado aos dados sobre doença coronariana.

Variável	**Parâmetro**	**MRLBB**	**MRLMV**	
		Estimativa	**Estimativa**	**Erro padrão**
Constante	β_0	-4,6268	-10,7509	2,5400
AGE	β_1	NA	NA	NA
SEX	β_{21}	0	0	-
SEX	β_{22}	0,7979	1,5066	0,5991
PAIN	β_{31}	0	0	-
PAIN	β_{32}	NA	NA	-
PAIN	β_{33}	NA	2,0030	1,0524
PAIN	β_{34}	1,5284	3,9293	1,0390
PRESS	β_4	0,0107	0,0336	0,0131
COL	β_5	0,0028	NA	-
SUG	β_{61}	0	0	-
SUG	β_{62}	NA	NA	-
ELE	β_{71}	0	0	-
ELE	β_{72}	NA	NA	-
ELE	β_{73}	0,2730	NA	-
HEART	β_8	-0,0053	NA	-
EXE	β_{91}	0	0	-
EXE	β_{92}	0,5502	NA	-
ST	β_{10}	0,3762	NA	-
SLOPE	β_{111}	0	0	-
SLOPE	β_{112}	0,6459	1,7693	0,4991
SLOPE	β_{113}	NA	1,9773	1,0305
VES	β_{12}	0,6967	1,0290	0,3270
THAL	β_{131}	NA	NA	-
THAL	β_{132}	NA	NA	-
THAL	β_{133}	1,1746	1,4205	0,4984

NA: Não Ajustado.

Observe que o método *stepwise* selecionou 6 das 13 variáveis independentes para o modelo final. Logo, a probabilidade de ocorrência de CHD não é influenciada pelas seguintes variáveis independentes: idade (AGE), nível de sua glicemia (SUG), nível de colesterol, resultado do eletrocardiograma, frequência cardíaca, ocorrência de angina induzida e comprimento do segmento ST, uma vez que essas variáveis independentes não foram selecionadas para o modelo final (expressão 4.3) após procedimento *stepwise*. Esse modelo explica ainda que, se a pessoa for do sexo masculino, a probabilidade de doença coronariana é aumentada. O modelo explica ainda que a probabilidade de ocorrência de CHD sofre incremento se a pessoa apresentar dor no peito provavelmente não anginosa ($PAIN_3$) ou assintomática ($PAIN_4$), a inclinação do segmento ST ao pico de exercício apresentar inclinação horizontal ($SLOPE_2$) ou descendente ($SLOPE_3$) e defeito reversível para a Talassemia ($THAL_3$). Essa probabilidade é incrementada ainda com as variáveis contínuas relacionadas à pressão arterial ($PRESS$) e número de grandes vasos coloridos por fluoroscopia (VES).

A Figura 7 apresenta quatro gráficos de diagnóstico do modelo MRLMV. Na Figura 7 (a) temos o gráfico de $\hat{h}_{ii}$ contra os valores ajustados e notamos dois pontos com maior destaque, $\#265$ (índice 55) e $\#88$ (índice 129). No gráfico dos resíduos t_{D_i}, Figura 7 (c), a maioria dos pontos cai dentro do intervalo $[-2, 2]$, com exceção das observações $\#235, \#4$ e $\#188$ (índices 50, 73 e 44, respectivamente) e algumas outras que estão próximas dos limites do intervalo. O gráfico de influência, Figura (b), destaca novamente as observações $\#265, \#88, \#235$ e $\#188$. O paciente $\#88$ tem 59 anos, é do sexo masculino, pressão arterial de 178 mm/Hg, nível de colesterol igual 270 mg/dL, frequência cardíaca de 145 bpm, comprimento do segmento ST igual a 4.2 mm e não apresenta doença cardíaca coronariana (Tabela 7). Na prática, pacientes com perfil semelhante a esse é espe-

rado que tenha CHD, como foi predito pelo modelo MRLBB e MRLMV. Situação semelhante ocorreu com o paciente #235, mas esse apresenta ainda inclinação horizontal do segmento ST, três grandes vasos coloridos por fluoroscopia e não

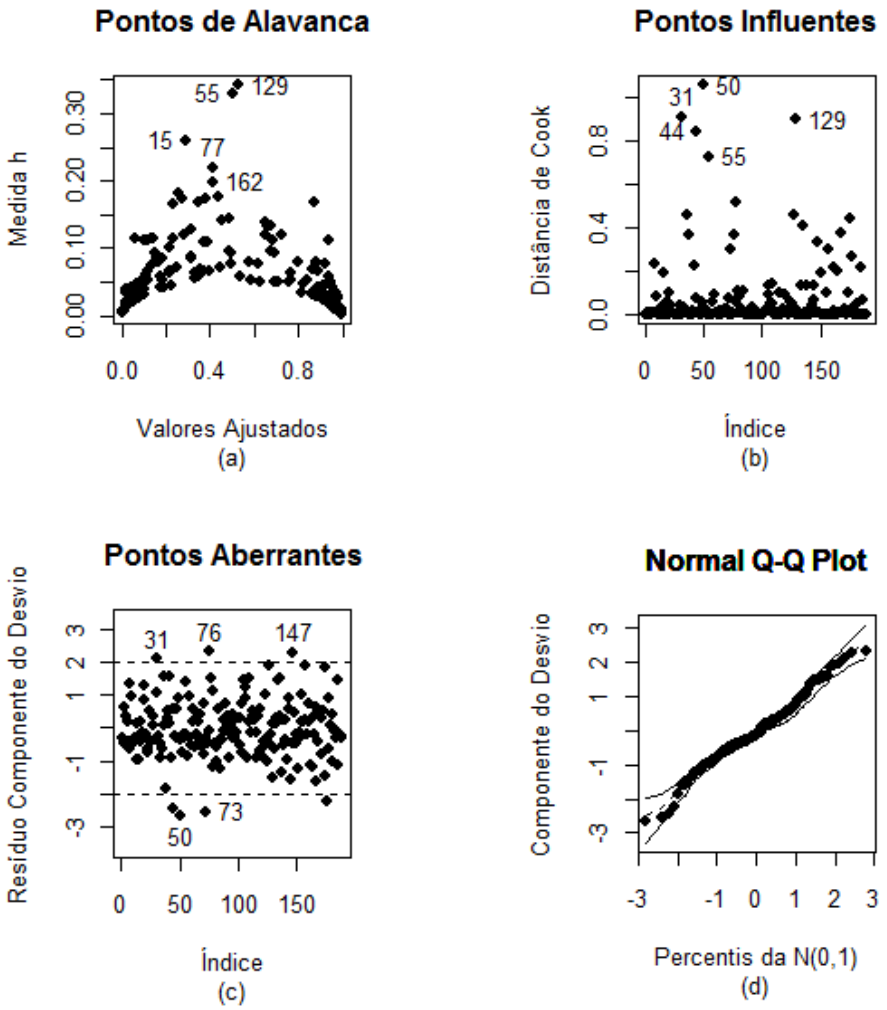

Figura 7 Gráficos de diagnóstico referente ao modelo MRLMV ajustado aos dados sobre doença cardíaca coronariana

possui CHD, mas o modelo o classificou como positivo para a presença de CHD pelo dois métodos. O mesmo ocorreu para o paciente $\#188$. O paciente $\#265$ tem 48 anos, pressão arterial 110 mm/Hg, nível de colesterol de 229 mg/dL, frequência cardíaca de 168 bpm, segmento ST de 1 mm com inclinação ascendente, não possui grandes vasos coloridos por fluoroscopia e possui CHD, porém os modelos MRLBB e MRLMV os classificaram como ausente e presente, respectivamente. Apesar da presença dessas observações, no gráfico normal de probabilidades para o resíduo t_{D_i}, Figura (d), também chamada de envelope simulado, não se observa nenhum indício de que a distribuição utilizada seja inadequada, uma vez que todos os pontos estão dentro das bandas de confiança.

Tabela 7 Relação e Predição pelos modelos MRLBB e MRLMV das observações consideradas discrepantes pelo gráfico de diagnóstico.

Variável	**Paciente**			
	88	235	265	188
linha	129	50	55	44
AGE	59	62	48	52
SEX	1	1	1	1
PAIN	1	3	2	4
PRESS	178	130	110	108
COL	270	231	229	233
SUG	0	0	0	1
ELE	3	1	1	1
HEART	145	146	168	147
EXE	0	0	0	0
ST	4.2	1.8	1	0.1
SLOPE	3	2	3	1
VES	0	3	0	3
THAL	7	7	7	7
DIS	0	0	1	0
MRLBB	1	1	0	1
MRLMV	1	1	1	1

*"linha" não é variável e corresponde à i-ésima linha do conjunto de treinamento.

Uma vez obtido o modelo para explicar a ocorrência de doença cardíaca coronariana, pode-se verificar o poder de discriminação desse modelo, ou seja, a capacidade do modelo em classificar corretamente indivíduos que têm CHD e os que não têm. As Figuras 8 e 9 mostram a curva ROC do modelo MRLBB e MRLMV e observa-se que os dois modelos apresentam alto poder de discriminação, uma vez que a área abaixo de cada curva ROC é de $0,947u.a.$ e $0,905u.a.$, respectivamente.

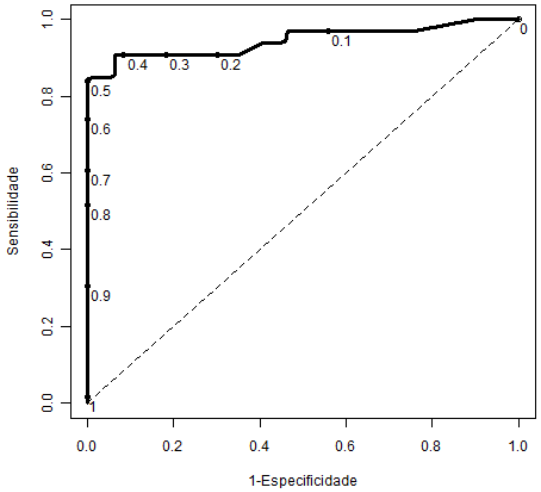

Figura 8 Curva ROC do modelo MRLBB

Diante do exposto na seção 2.4.1, uma outra vantagem da curva ROC é a possibilidade de escolher um limiar adequado para a classificação de pacientes quanto a presença ou não de CHD. As Figuras 8 e 9 evidenciam que um limiar adequado seria $0,5$ em ambos modelos. Portanto, para avaliar a *sensibilidade* e *especificidade* do modelo, será utilizado o seguinte critério para classificar um paciente como positivo para presença de CHD ($Y = 1$): se a probabilidade de

ocorrência de CHD for maior do que 0,5 (50%). Caso contrário, será classificado como ausente para CHD ($Y = 0$). A predição dos modelos MRLBB e MRLMV mostrada na Tabela 7 refere-se a esse limiar de 0,5.

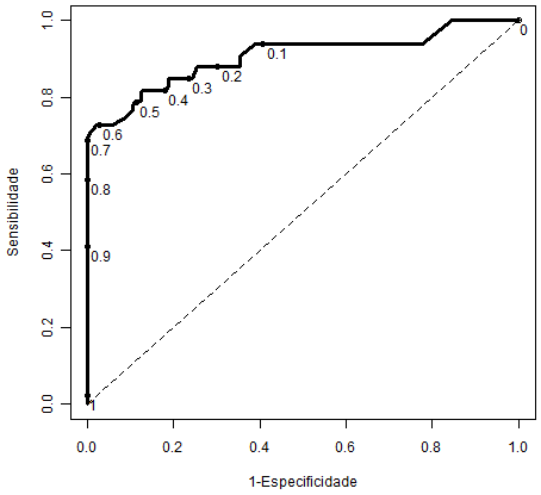

Figura 9 Curva ROC do modelo MRLMV

As Tabelas 8 e 9 resumem o poder de discriminação de cada modelo obtido. Na Tabela 8 observa-se que a *sensibilidade* do modelo MRLBB é de 82%, ou seja, 82% dos pacientes que têm CHD o modelo os classificaram como positivo para essa característica. A taxa de falsos negativos do modelo foi de 18%, ou seja, 18% das pessoas que têm CHD o modelo acusou como falso para essa característica. A taxa de falsos positivos foi de 0%, logo, dos pacientes que não têm CHD o modelo não classificou nenhum paciente como positivo para CHD e, como consequência, a *especificidade* do modelo foi de 100%. A acurácia do modelo foi de 92,59%.

Tabela 8 Tabela de confusão do modelo MRLBB ajustado aos dados sobre doença arterial coronariana.

Observado	**Modelo**	
	presença	ausência
presença	27	6
ausência	0	48

De maneira análoga, observa-se que a *sensibilidade* do modelo MRLMV foi de 79%. A taxa de falsos negativos e de falsos positivos do modelo foram 21% e 10%, respectivamente. A *especificidade* do modelo foi de 90%, ou seja, dos pacientes que não têm CHD, 90% foram classificados nessa condição. A acurácia do modelo foi de 85,18%. (Tabela 9).

Tabela 9 Tabela de confusão do modelo MRLMV ajustado aos dados sobre doença arterial coronariana.

Observado	**Modelo**	
	presença	ausência
presença	26	7
ausência	5	43

4.3 Razão de Chances

Uma das vantagens de se utilizar um modelo de regressão logística é a de se obter a relação entre probabilidades de ocorrência de CHD com a uma determinada variável independente. Essa relação é chamada de razão de chances e a Tabela 10 sintetiza esses valores para cada estimativa dos parâmetros dos modelos MRLBB e MRLMV finais (entenda como modelos finais os que contêm apenas as variáveis independentes selecionadas apresentadas na Tabela 6).

Denote por $OR_{SEX,Boost}$ e $OR_{SEX,RL}$ a razão de chances de doença coronariana cardíaca com relação ao sexo dos pacientes, obtida via modelos MRLBB e MRLMV, respectivamente. Logo, a razão de chances de CHD positivo entre paciente do sexo masculino e feminino é estimada por

$$\hat{OR}_{SEX,Boost} = \exp\{\beta_{22,Boost}\} = \exp\{0,7979\} = 2,2210$$
$$\hat{OR}_{SEX,RL} = \exp\{\beta_{22,RL}\} = \exp\{1,5066\} = 4,5112$$

Assim, $\hat{OR}_{SEX,Boost}$ indica que um paciente do sexo masculino tem uma chance de 122,1% maior em ter doença cardíaca coronariana em relação ao paciente do sexo feminino via modelo MRLBB, ao passo que esse mesmo evento ocorre com chance de 351,12% maior via modelo MRLMV.

Denote por $OR_{PAIN4,Boost}$ e $OR_{PAIN4,RL}$ as razões de chances de um paciente ter CHD positivo e dor no peito do tipo 4 em relação a um paciente ter CHD positivo e ter dor no peito do tipo 1 obtidas via modelos MRLBB e MRLMV, respectivamente. Então, via algoritmo Binomial Boosting, a chance do paciente que tem dor no peito do tipo 4 ter CHD é de 361,09% (quase 4 vezes) maior do que um paciente que apresentar dor do tipo 1. De forma análoga, via MRLMV, essa chance é de 4987,01% (quase 50 vezes!) maior.

$$\hat{OR}_{PAIN4,Boost} = \exp\{\beta_{34,\,\mathrm{Boost}}\} = \exp\{1,5284\} = 4,6109$$
$$\hat{OR}_{PAIN4,RL} = \exp\{\beta_{34,\,RL}\} = \exp\{3,9293\} = 50,8701$$

Tabela 10 Razões de chance (OR) estimados para as variáveis independentes **selecionadas** pelos modelos MRLBB e MRLMV e intervalos de confiança assintótico de OR para MRLMV, referentes aos dados sobre doença cardíaca coronariana.

Variável	**Parâmetro**	**MRLBB**	**MRLMV**		
		$\hat{OR}$	$\hat{OR}$	**LI (95%)**	**LS (95%)**
SEX	β_{22}	2,2210	4,5112	1,4481	15,5089
PAIN	β_{33}	NA	7,4111	1,0977	75,8004
	β_{34}	4,6109	50,8701	8,1312	524,6194
PRESS	β_4	1,1133 *	1,3986 *	1,0818	1,8082
COL	β_5	1,0280 *	NA	-	-
ELE	β_{73}	1,3139	NA	-	-
HEART	β_8	0,9480 *	NA	-	-
EXE	β_{92}	1,7336	NA	-	-
ST	β_{10}	1,4568	NA	-	-
SLOPE	β_{112}	1,9076	5,8667	2,2765	16,3721
	β_{113}	1,9692	7,2230	1,0572	63,3568
VES	β_{12}	2,0071	2,7982	1,5475	5,6173
THAL7	β_{22}	3,2369	4,1393	1,5755	11,2750

* Razão de Chances correspondente ao incremento de 10 unidades.

NA: Não Ajustado.

No caso das variáveis contínuas, existe uma ligeira diferença na interpretação das razões das chances, desse modo, a cada incremento de uma unidade nesse tipo de variável acarreta um aumento correspondente na chance de um paciente ser diagnosticado com CHD. No caso da variável associada ao comprimento do segmento ST, mantendo as outras variáveis independentes fixas, um aumento de 1 mm nesse segmento implicará um aumento de 45,68% na chance de um paciente ser classificado com doença coronariana cardíaca.

No entanto, o aumento de uma unidade em algumas variáveis independentes não tem muito sentido prático, como é o caso da covariável associada à pressão arterial do paciente ($PRESS$). Logo, via MRLBB, para um incremento

de 10 mm/Hg dessa covariável implica um aumento de 11,33% na chance do paciente ser diagnosticado com CHD. Da mesma forma, via MRLMV, essa chance aumenta para 39,86%.

Para a covariável associada à frequência cardíaca máxima atingida (HEART), um aumento de 10 bpm acarreta um decréscimo de 5,2% na chance do paciente ter CHD, via MRLBB.

Foi apresentado também na Tabela 10 o intervalo de 95% de confiança assintótico para cada razão de chances estimada para o modelo MRLMV e observa-se que, como cada intervalo não contém a estimativa pontual 1 de razão de chances, a estimativa da razão de chances é significativa e valem as considerações feitas anteriormente.

4.4 Discussão

O presente trabalho apresentou uma comparação do modelo de regressão logística estimado via algoritmo Binomial Boosting (MRLBB) e pelo método da máxima verossimilhança (MRLMV). Na literatura, não foram encontrados trabalhos que fizeram esse tipo de comparação, então serão dicutidos nesta seção alguns trabalhos que utilizaram algum tipo de algoritmo Boosting e compararam seu desempenho com outros tipos de classificadores.

Cai et al. (2006) utilizaram o algoritmo LogitBoost para classificar diversas estruturas de proteínas em biologia molecular. Os autores compararam a eficiência do algoritmo LogitBoost com um outro método bastante conhecido na comunidade de aprendizado de máquinas, o método de Máquinas de Vetor Suporte (*Support Vector Machines*), observando um desempenho superior de quase 9% do algoritmo Boosting na predição de classes estruturais para um dado conjunto de dados.

Cao et al. (2010) compararam o algoritmo Gradiente Boosting de Friedman Estocástico, que é uma versão do algoritmo Boosting - FGD com árvores de decisão e *bagging*, com dois métodos comumente usados em quimiometria, o método de análise discriminante parcial mínimos quadrados (PLS-DA) e *bagging*. Utilizaram o conjunto de dados de CHD (o mesmo utilizado nessa dissertação) obtido no grupo UCI Machine Learning. A taxa de erro obtida pelos métodos gradiente Boosting estocástico, *bagging* e PLS-DA foi de 14,7%, 18,6% e 16,2%, respectivamente, mostrando superioridade do algoritmo Boosting.

Em um estudo com dados simulados de expressão gênica, Dettling e Buhlmann (2003) mostraram que o algoritmo LogitBoost apresentou resultados mais acurados quando comparados com os métodos Vizinhos mais Próximos e Árvore de Classificação, da ordem de 12,37% e 10,21%, respectivamente. Além disso, comparou os resultados obtidos via algoritmo LogitBoost e com o algoritmo AdaBoost em seis conjuntos de dados públicos relacionados a tipos de câncer e mostrou uma ligeira melhora nos resultados obtidos pelo LogitBoost.

Estudando a situação de presença/ausência de doença cardíaca coronariana em um conjunto de 297 pacientes, Rodrigues, Macrini e Monteiro (2008) ajustaram uma rede neural a esse conjunto de dados e obtiveram uma taxa de acerto de 91%. Compararam ainda esse reultado com os métodos de Análise Discriminante e algoritmo C4.5, que apresentaram taxa de acerto de 87,1% e 82,3%, respectivamente. Embora sejam conjuntos de dados um pouco diferentes, mas de mesma natureza, a taxa de acerto (acurácia) obtida pelo algoritmo Binomial Boosting nesta dissertação foi de 92,59%.

Schonlau (2005) apresenta a implementação de Boosting no software *Stata* e faz uma aplicação de Boosting em duas situações no contexto de regressão, uma com dados simulados de um modelo normal e uma outra com dados simu-

lados de um modelo logístico. Na primeira situação, o modelo ajustado obteve $R^2 = 21,3\%$ e aplicando Boosting obteve-se $R^2 = 93,8\%$. A taxa de acerto do modelo logístico ajustado foi de 54,1% e com Boosting foi de 76,0%.

5 CONCLUSÕES

Os modelos de regressão logística estimados via algoritmo Binomial Boosting (MRLBB) e pelo método da máxima verossimilhança (MRLMV) apresentaram ajuste satisfatório ao problema presença/ausência de doença cardíaca coronariana (CHD).

O método de Boosting, mais especificamente o algoritmo Binomial Boosting, ajustou um modelo com melhor adequabilidade na situação presença/ausência de CHD, uma vez que a acurácia, sensibilidade, especificidade, taxa de falsos positivos e taxa de falsos negativos desse modelo foram melhores.

O modelo estimado via algoritmo Binomial Boosting (MRLBB) apresentou-se mais adequado com relação às razões de chances estimadas ($\hat{OR}$), ou seja, seus valores são menores quando comparados com as razões de chances obtidas via método de máxima verossimilhança (MRLMV).

O algoritmo Binomial Boosting constitui-se, portanto, numa alternativa poderosa para a análise de situações cuja resposta é binária.

REFERÊNCIAS

AKAIKE, H. A new look at the statistical model identification, **IEEE Transactions on Automatic Control**, Boston, v. 19, n. 6, p. 716-723, 1974.

ATKINSON, A. C. Plots, Transformations and Regression, **Oxford University Press**, Oxford, 1985.

BARTLETT, P.; TRASKIN, M. AdaBoost is consistent, **Journal of Machine Learning Resources**, v. 8, p. 2347 - 2368, 2007.

BERK, R. A. **Statistical Learning from a Regression Perspective**, Springer Series in Statistics, 373 p., 2008.

BISHOP, C. M. **Neural Networks for Pattern Recognition**, Oxford University Press, 504 p., 1995.

BREIMAN, L. et al. **Classification and regression Trees**, Chapman and Hall/CRC, 368 p., 1º ed., 1984.

BREIMAN, L. Arcing classifiers (with discussion), **The Annals of Statistics**, v. 26, n.3, p. 801 - 849, 1998.

BREIMAN, L. Prediction games and arcing algorithms, **Neural Computation**, v. 11, p. 1463 - 1517, 1999.

BUHLMANN, P.; HOTHORN, T. Boosting Algorithms: Regularization, Prediction and Model Fitting, **Statistical Science**, v. 22, n. 4, p. 477-505, 2007.

CAI, Y. D. et al. Using LogitBoost classifier to predict protein structural classes, **Journal of Theoretical Biology**, v. 238, p. 172-176, 2006.

CAO, D. S. et al. The Boosting: A new idea of building models, **Chemometrics and Intelligent Laboratory Systems**, v. 100, p. 1-11, 2010.

DETTLING, M.; BUHLMANN, P. Boosting for tumor classification with gene expression data, **Bioinformatics**, v. 19, n. 9, p. 1061-1069, 2003.

FÁVERO, L. P. et al. **Análise de Dados: modelagem multivariada para tomada de decisõe**, Rio de Janeiro: Elsevier, 646 p., 2009.

FRANK, A.; ASUNCION, A. Machine Learning Repository, Irvine, CA: University of California, **School of Information and Computer Science**, [http://archive.ics.uci.edu/ml], 2010.

FREUND, Y.; SCHAPIRE, R. E. Experiments with a new Boosting algorithm, **In: International Conference on Machine Learning.**, p. 148-156, 1996.

FRIEDMAN, J. Greedy function aproximation: A gradient boosting machine, **The Annals of Statistics**, v. 29, p. 1189 - 1232, 2001.

FRIEDMAN, J. H.; HASTIE, T. J.; TIBSHIRANI, R. J. **The Elements of Statistical Learning**, Basel: Springer Verlag, 2001.

FRIEDMAN, J. H.; HASTIE, T. J.; TIBSHIRANI, R. J. Additive logistic regression: A statistical view of Boosting (with discussion), **The Annals of Statistics**, v. 28, p. 337 - 407, 2000.

HANLEY, J. A. Receiver operating characteristic (ROC) methodology: the state of the art, **Critical Reviews in Diagnostic Imaging**, v. 29(3), p. 307 - 335, 1989.

HOSMER, D. W.; LEMESHOW, S. **Applied Logistic Regression**, 2º ed., John Wiley, New York, 1989.

JIANG, L. **Process consistency for adaboost**, Technical Report 05, Department of Statistics, Northwestern University, 2000.

KEARNS, M.; VALIANT, L. Crystographic limitations on learning Boolean formulae and finite automata, **Journal Assoc. Comput. Machinery**, v. 41, p. 67 - 95, 1994.

MICHIE, D.; SPIEGELHALTER, D. J.; TAYLOR, C. C. **Machine Learning: Neural and Statistical Classification**, Ellis Horwood Series in Artificial Intelligence, 290 p., 1994.

PAULA, G. A. Influence and residuals in restricted generalized linear models, **Journal of Statistical Computation and Simulation**, v. 51, p. 315 - 352, 1995.

PAULA, G. A.; TUDER, R. M. Utilização da regressão logística para aperfeiçoar o diagnóstico de processo infeccioso pulmonar, **Revista Ciência e Cultura**, v. 40, p. 1046-1050, 1986.

R DEVELOPMENT CORE TEAM (2011). **R: A language and environment for statistical computing**, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

RIPLEY, B. D. **Pattern Recognittion and Neural Networks**, Cambridge University Press, ISBN 0 521 46086 7, 416 p., 1996.

RODRIGUES, T. B.; MACRINI, J. L. R.; MONTEIRO, E. C. Seleção de Variáveis e Classificação de Padrões por Redes Neurais como auxílio ao diagnóstico de Cardiopatia Isquêmica, **Pesquisa Operacional**, v. 28, n. 2, p. 285-302, 2008.

RUBESAM, A. **Estimação Não Paramétrica Aplicada a Problemas de Classificação via Bagging e Boosting**, Dissertação de mestrado do Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas, Campinas, 127 p., 2004.

SCHAPIRE, R. E. The strength of weak learnability, **Machine learning**, v. 5, p. 197-227, 1990.

SCHAPIRE, R. E.; FREUND, Y. **Boosting: Foundations and Algorithms**, Massachusetts Institute of Technology, 526p., 2012.

SCHWARZ, G. Estimating the dimensional of a model, **The Annals of Statistics**,

Hayward, v. 6, n. 2, p. 461-464, 1978.

SCHONLAU, M. Boosted regression (Boosting): An introductory tutorial and a Stata plugin, **The Stata Journal**, v. 5, n. 3, p. 330-354, 2005.

WEDDERBURN, R. W. M. On the existence and uniqueness of the maximum likelihood estimates for certain generalized linear models, **Biometrika**, v. 68, p. 27-32, 1976.

WILLIAMS, D. A. Residuals in generalized linear models, **In: Proceedings of the 12th. International Biometrics Conference**, Tokyo, p. 59-68, 1984.

ANEXOS

ANEXO A - Por que $e^{-yF(\boldsymbol{x})}$

ANEXO B - Exemplo do cálculo da estatística $\hat{C}$ de Hosmer-Lemeshow

ANEXO A - Por que $e^{-yF(\boldsymbol{x})}$

Considere o seguinte critério para ajuste de um modelo:

$$C(F) = E\left[e^{-yF(\boldsymbol{x})}\right] \tag{5.1}$$

Esse critério, que pode ser pensado como uma medida de bondade de ajuste, é minimizado em

$$F(\boldsymbol{x}) = \frac{1}{2}\log\frac{P(y=1\,|\boldsymbol{x})}{P(y=-1\,|\boldsymbol{x})} \tag{5.2}$$

que é a transformação logística simétrica.

Para minimizar o critério acima iterativamente, considere que temos uma estimativa $F(\boldsymbol{x})$, e queremos uma atualização $F(\boldsymbol{x}) + cf(\boldsymbol{x})$, onde c é um escalar e f é uma atualização fornecida por um algoritmo. A atualização é baseada na versão populacional do critério. Para c e $\boldsymbol{x}$ fixos, a expansão de Taylor até segunda ordem de $C(F(\boldsymbol{x}) + cf(\boldsymbol{x}))$ ao redor de $f(\boldsymbol{x}) = 0$ é

$$
\begin{aligned}
C\left(F\left(\boldsymbol{x}\right)+cf\left(\boldsymbol{x}\right)\right) =& E\left[e^{-y(F(\boldsymbol{x})+cf(\boldsymbol{x}))}\right] \\
\approx& E\left[e^{-yF(\boldsymbol{x})}-cyf\left(\boldsymbol{x}\right)e^{-yF(\boldsymbol{x})}+\frac{c^2\,y^2}{2}\,e^{-yF(\mathbf{x})}\,f^2\left(\boldsymbol{x}\right)\right] \\
=& E\left[e^{-yF(\boldsymbol{x})(1-cyf(\boldsymbol{x}))}+\frac{c^2\,y^2}{2}f^2\left(\boldsymbol{x}\right)\right] \\
=& E\left[e^{-yF(\boldsymbol{x})}\left(1-cyf\left(\boldsymbol{x}\right)+\frac{c^2}{2}\right)\right]
\end{aligned}
$$

Acima foram usadas as derivadas

$$
\frac{\partial}{\partial f}\,e^{-y(F(\boldsymbol{x})+cf(\boldsymbol{x}))}=-cye^{-yF(\boldsymbol{x})}
$$

$$
\frac{\partial^2}{\partial f^2}\,e^{-y(F(\boldsymbol{x})+cf(\boldsymbol{x}))}=c^2y^2e^{-yF(\boldsymbol{x})}
$$

Na equação acima, usamos o fato de que $y^2=f^2\left(\boldsymbol{x}\right)=1$.

Minimizando essa expansão pontulamente com respeito a $f\left(\boldsymbol{x}\right)\in\{-1,1\}$, escrevemos:

$$
f\left(\boldsymbol{x}\right)=\arg\min_{f}E_w\left[1-cyf\left(\boldsymbol{x}\right)+\frac{c^2}{2}\left|\boldsymbol{x}\right.\right] \tag{5.3}
$$

A notação $E_w\left[\left|\boldsymbol{x}\right.\right]$ refere-se à esperança condicional ponderada (quando populacional) ou média ponderada, numa amostra. Denotando por $w=w\left(\boldsymbol{x},y\right)=e^{-yF(\boldsymbol{x})}$, define-se:

$$
E_w\left[g\left(\boldsymbol{x},y\right)\left|\boldsymbol{x}\right.\right]=\frac{E_w\left[w\left(\boldsymbol{x},y\right)g\left(\boldsymbol{x},y\right)\left|\boldsymbol{x}\right.\right]}{E_w\left[w\left(\boldsymbol{x},y\right)\left|\mathbf{x}\right.\right]} \tag{5.4}
$$

Assim, a esperança em 5.4 é igual à expansão de Taylor acima.

Para $c>0$, minimizar a expansão de Taylor acima é equivalente a maxi-

mizar

$$\begin{aligned} E_w\left[yf\left(\boldsymbol{x}\right)\right] =& f\left(\boldsymbol{x}\right)P_w\left(y=1\left|\boldsymbol{x}\right.\right) - f\left(\boldsymbol{x}\right)P_w\left(y=-1\left|\boldsymbol{x}\right.\right) \\ =& f\left(\boldsymbol{x}\right)\left[P_w\left(y=1\left|\boldsymbol{x}\right.\right) - P_w\left(y=-1\left|\boldsymbol{x}\right.\right)\right] \end{aligned}$$

Há dois casos:

- $P_w\left(y=1\left|\boldsymbol{x}\right.\right) - P_w\left(y=-1\left|\boldsymbol{x}\right.\right) > 0$
- $P_w\left(y=1\left|\boldsymbol{x}\right.\right) - P_w\left(y=-1\left|\boldsymbol{x}\right.\right) < 0$

Como $f\left(\boldsymbol{x}\right)$ só assume os valores $\{-1,1\}$, o máximo da equação acima é em $f\left(\boldsymbol{x}\right)=1$, no primeiro caso, e em $f\left(\boldsymbol{x}\right)=-1$, no segundo caso.

Usando novamente que $y^2 = f^2\left(\boldsymbol{x}\right) = 1$, note que

$$-E_w\left[yf\left(\boldsymbol{x}\right)\right] = \frac{E_w\left[y-f\left(\boldsymbol{x}\right)\right]^2}{2} - 1 \tag{5.5}$$

ou seja, partindo de uma aproximação quadrática (expansão de 2º ordem) do critério, chegamos ao problema equivalente de maximizar a aquação acima.

Agora, dada $f\left(\boldsymbol{x}\right) \in \{-1,1\}$, podemos minimizar diretamente

$$C\left(F\left(\boldsymbol{x}\right) + cf\left(\boldsymbol{x}\right)\right)$$

para determinar c:

$$c = \arg\min_c C\left(F\left(\boldsymbol{x}\right) + cf\left(\boldsymbol{x}\right)\right) = \arg\min_c E_w\left[e^{-cyf(\boldsymbol{x})}\right]$$

(a igualdade acima é válida, pois $C\left(F\left(\boldsymbol{x}\right) + cf\left(\boldsymbol{x}\right)\right) = E\left[e^{-y(F(\boldsymbol{x})+cf(\boldsymbol{x}))}\right] =$

$E\left[e^{-yF(\boldsymbol{x})}\ e^{-cf(\boldsymbol{x})}\right] = E\left[\,e^{-cyf(\boldsymbol{x})}\right]$). Para fazer essa minimização, considere a variável aleatória

$$yf\left(\boldsymbol{x}\right) = \begin{cases} 1\ \ se\ y = f\left(\boldsymbol{x}\right) \\ -1\ se\, y \neq f\left(\boldsymbol{x}\right) \end{cases} \tag{5.6}$$

Temos

$$\begin{aligned} E\left[\,e^{-cyf(\boldsymbol{x})}\right] =& e^{-c}P_w\left(y = f\left(\boldsymbol{x}\right)\right) + e^{c}P_w\left(y \neq f\left(\boldsymbol{x}\right)\right) \\ =& e^{-c}\left[1 - P_w\left(y \neq f\left(\boldsymbol{x}\right)\right) + e^{c}P_w\left(y \neq f\left(\boldsymbol{x}\right)\right)\right] \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial c}E_w\left[\,e^{-cyf(\mathbf{x})}\right] =& -\,e^{-c}\left[1 - P_w\left(y \neq f\left(\boldsymbol{x}\right)\right) + e^{c}P_w\left(y \neq f\left(\boldsymbol{x}\right)\right)\right] = 0 \\ \Rightarrow& e^{c}P_w\left(y \neq f\left(\boldsymbol{x}\right)\right) = e^{-c}\left(1 - P_w\left(y \neq f\left(\mathbf{x}\right)\right)\right) \\ \Rightarrow& e^{2c} = \frac{1 - P_w\left(y \neq f\left(\boldsymbol{x}\right)\right)}{P_w\left(y \neq f\left(\boldsymbol{x}\right)\right)} \\ \Rightarrow& c = \frac{1}{2}\log\frac{1 - P_w\left(y \neq f\left(\boldsymbol{x}\right)\right)}{P_w\left(y \neq f\left(\boldsymbol{x}\right)\right)} \end{aligned}$$

Como $P_w\left(y \neq f\left(\boldsymbol{x}\right)\right) = E_w\left[I\left(y \neq f\left(\boldsymbol{x}\right)\right)\right]$, temos

$$c = \frac{1}{2}\log\frac{1 - E_w\left[I\left(y \neq f\left(\boldsymbol{x}\right)\right)\right]}{E_w\left[I\left(y \neq f\left(\boldsymbol{x}\right)\right)\right]} = \frac{1}{2}\log\frac{1-\varepsilon}{\varepsilon}$$

onde $\varepsilon = E_w\left[I\left(y \neq f\left(\mathbf{x}\right)\right)\right]$.

Combinando os passos acima, a atualização de $F\left(\boldsymbol{x}\right)$ é

$$f(\boldsymbol{x}) \leftarrow F(\boldsymbol{x}) + \frac{1}{2}\log\frac{1-\varepsilon}{\varepsilon} f(\boldsymbol{x}) \tag{5.7}$$

Na próxima iteração, os pesos são aumentados, pois o algoritmo é adaptativo:

$$w(\boldsymbol{x}, y) \leftarrow w(\boldsymbol{x}, y)\ e^{-cyf(\boldsymbol{x})} \tag{5.8}$$

Como $-yf(\boldsymbol{x}) = 2I(y \neq f(\mathbf{x})) - 1$, a atualização é equivalente a

$$w(\boldsymbol{x}, y) \leftarrow w(\boldsymbol{x}, y)\ \exp\left\{\log\frac{1-\varepsilon}{\varepsilon} I(y \neq f(\boldsymbol{x}))\right\} \tag{5.9}$$

As atualizações e a função obtida pelo desenvolvimento apresentado acima são idênticas as usadas no algoritmo AdaBoost discreto.

Uma questão que surge naturalmente é por que usar $E\left[e^{-yF(\boldsymbol{x})}\right]$? Note que o termo $\sum_i \exp(-y_i F(\boldsymbol{x}_i))$ é um limite superior para a probabilidade de erro no conjunto de treinamento. Friedman, Hastie e Tibshirani (2000) usaram o seguinte modelo para explicar alguns fatos sobre essa escolha. Considere

$$\tilde{y} = \frac{y+1}{2} \in \{0, 1\}$$

e a parametrização das probabilidades binomiais dada por

$$p(\boldsymbol{x}) = \frac{e^{F(\boldsymbol{x})}}{e^{F(\boldsymbol{x})} + e^{-F(\boldsymbol{x})}} \tag{5.10}$$

Essa parametrização é dada pelo minimizador $F(\boldsymbol{x})$ da equação 5.2. A log-verossimilhança da binomial é então

$$\begin{aligned} l\left(\tilde{y}, p\left(\boldsymbol{x}\right)\right) =& \tilde{y} \log p\left(\boldsymbol{x}\right) + \left(1 - \tilde{y}\right) \log\left(1 - p\left(\boldsymbol{x}\right)\right) \\ =& - \log\left(1 + e^{-2yF(\boldsymbol{x})}\right) \end{aligned}$$

O modelo dado acima é equivalente a um modelo logístico, a menos de um fator 2. Para notar isso, basta multiplicar a equação 5.10 por $e^{F(\boldsymbol{x})}$, obtendo-se

$$p\left(\boldsymbol{x}\right) = \frac{e^{2F(\boldsymbol{x})}}{1 + e^{2F(\boldsymbol{x})}} \tag{5.11}$$

No modelo logístico usual, temos

$$p\left(\boldsymbol{x}\right) = \frac{e^{F(\boldsymbol{x})}}{1 + e^{F(\boldsymbol{x})}}$$

Os seguintes fatos podem ser notados:

- $yF\left(\boldsymbol{x}\right)$ é negativo se e somente se a classificação dada por F é errada, ou seja, $I\left(yF\left(\boldsymbol{x}\right) < 0\right)$ indica um erro.
- em expansão de Taylor até 2º ordem ao redor de $F = 0$, o critério exponencial e (menos) a log-verossimilhança da binomial são equivalentes.
- o mínimo populacional de $-E\left[l\left(\tilde{y}, p\left(\boldsymbol{x}\right)\right)\right]$ e $E\left[e^{-yF(\boldsymbol{x})}\right]$ coincidem. A log-verossimilhança é maximizada em $p\left(\boldsymbol{x}\right) = P\left(\tilde{y} = 1 \left| \boldsymbol{x} \right.\right)$, a probabilidade a posteriori verdadeira, que define a função logito, e o mínimo de $E\left[e^{-yF(\boldsymbol{x})}\right]$ é o dado na equação 5.2.

Assim, $e^{-yF(\boldsymbol{x})}$ é uma aproximação da log-verossimilhança da binomial para ajustar um modelo aditivo logístico.

ANEXO B - Exemplo do cálculo da estatística $\hat{C}$ de Hosmer-Lemeshow

Para o cálculo da estatística $\hat{C}$ de Hosmer-Lemeshow, considere as seguintes quantidades resumidas na Tabela 11. Nesssa tabela, os valores de O_i indicam a quantidade de eventos $Y = 1$ no grupo k, n_i indica a quantidade de elementos no grupo k e $\hat{\pi}_i$ e calculado por $\bar{\pi_k} = \sum_{j=1}^{C_k} \frac{\bar{\pi}_j}{n_k}$

Tabela 11 Quantidades usadas para o cálculo da estatística $\hat{C}$ de Hosmer-Lemeshow referente ao modelo logístico.

Grupo	O_i	n_i	$\hat{\pi}_i$
1	0	5	0,0024
2	2	5	0,0459
3	0	5	0,2737
4	1	5	0,5113
5	3	5	0,6728
6	5	5	0,7956
7	5	5	0,8974
8	4	4	0,9766

Foram considerados sete grupos com cinco observações cada e um grupo com quatro observações. Os termos para o cálculo de $\hat{C}$ são dados abaixo

$$\begin{aligned}\hat{C} =& 0,0120 + 14,3157 + 1,8842 + 1,9391 \\ &+ 0,1203 + 1,2846 + 0,5716 + 0,0958 \\ =& 20,2233\end{aligned}$$

cuja estatística do teste qui-quadrado com $g - 2 = 6$ graus de liberdade é dado por

valor p=0,0025, indicando que o ajuste não é adequado.

APÊNDICES

APÊNDICE A - Ilustração Didática do Algoritmo AdaBoost

Para ajudar a fixar a ideia do algoritmo, vamos apresentar um exemplo numérico com um conjunto de dados muito simples. Considere cinco observações com valores para a variável resposta para $i = 1, 2, 3, 4, 5$ de 1, 1, 1, -1, -1, respectivamente. Considere conhecido o classificador $f_m(\boldsymbol{x})$ e estamos interessados apenas nos seus resultados. Para esse exemplo, temos o seguinte algoritmo.

1. Inicie as observações com peso $w_i^1 = 1/5$.

2. Para a primeira iteração use os pesos iguais, suponha que os valores ajustados para as observações $i = 1, 2, 3, 4, 5$ são 1, 1, 1, 1, 1 (valores retornados pelo classificador $f_m(\boldsymbol{x})$). As primeiras três respostas estão corretas e as últimas duas estão incorretas. O erro para essa iteração é:

$$\varepsilon_1 = \frac{(0,20 \times 0) + (0,20 \times 0) + (0,20 \times 0) + (0,20 \times 1) + (0,20 \times 1)}{1} = 0,40$$

3. Os pesos que serão dados a essa iteração são

$$c_1 = \frac{1}{2} \ln\left(\frac{1 - 0,40}{0,40}\right) = \frac{1}{2} \ln\left(\frac{0,60}{0,40}\right) = \frac{1}{2} \ln(1,50) = 0,20$$

4. Para as observações que foram classificadas de forma correta e errada, respectivamente, os novos pesos serão

$$y_i = f_1(x_i) \Rightarrow w_i^2 = 0,20 \times e^{-0,20} = 0,16$$
$$y_i \neq f_1(x_i) \Rightarrow w_i^2 = 0,20 \times e^{0,20} = 0,24$$

e renormalizando para que a soma não passe de 1

$$z^1 = (0,16 \times 3) + (0,24 \times 2) = 0,96$$

Logo, os pesos para a segunda iteração serão

$$w_1^2 = \frac{0,16}{0,96} = 0,17$$
$$w_2^2 = \frac{0,16}{0,96} = 0,17$$
$$w_3^2 = \frac{0,16}{0,96} = 0,17$$
$$w_4^2 = \frac{0,24}{0,96} = 0,25$$
$$w_5^2 = \frac{0,24}{0,96} = 0,25$$

5. Agora começamos a segunda iteração. Ajustamos o classificador $f_m(\boldsymbol{x})$ novamente e para $i = 1, 2, 3, 4, 5$, obtemos 1, 1, 1, 1, -1. Somente a penúltima resposta está incorreta. O erro para a segunda iteração é

$$\varepsilon_2 = \frac{(0,17 \times 0) + (0,17 \times 0) + (0,17 \times 0) + (0,25 \times 1) + (0,25 \times 0)}{(0,17 \times 3) + (0,25 \times 2)} = 0,25$$

6. O peso para ser dado a essa iteração é

$$c_2 = \frac{1}{2} \ln \left(\frac{1 - 0,25}{0,25} \right) = \frac{1}{2} \ln \left(\frac{0,75}{0,25} \right) = \frac{1}{2} \ln (3) = 0,55$$

7. Nós normalmente manteríamos o processo de iteração, começando com o cálculo de um terceiro conjunto de pesos. Mas suponha que o processo de iteração termine agora. As classes estimadas são:

$$\begin{aligned}
\hat{y}_1 &= sign\left[(1 \times 0,20) + (1 \times 0,55)\right] > 0 \Rightarrow 1 \\
\hat{y}_2 &= sign\left[(1 \times 0,20) + (1 \times 0,55)\right] > 0 \Rightarrow 1 \\
\hat{y}_3 &= sign\left[(1 \times 0,20) + (1 \times 0,55)\right] > 0 \Rightarrow 1 \\
\hat{y}_4 &= sign\left[(1 \times 0,20) + (-1 \times 0,55)\right] < 0 \Rightarrow -1 \\
\hat{y}_5 &= sign\left[(1 \times 0,20) + (1 \times 0,55)\right] > 0 \Rightarrow 1
\end{aligned}$$

Como pode-se ver nesse exemplo, as observações mal classificadas receberam relativamente maior peso. A classe estimada é apenas uma média ponderada das classes estimadas em cada iteração. A segunda iteração tinha menos observações mal classificadas e então foi dado maior peso nessa iteração. Essa ideia é aplicada até mesmo em conjunto de dados muito grandes e para milhares de iterações.

O exemplo ilustra também a ideia de que o algoritmo "tenta" minimizar o valor de ε a cada iteração e, como descrito na seção 2.2, o erro na amostra de treinamento cai exponencialmente. Uma forma de determinar o número de iterações para que o algoritmo não seja executado indefinidamente é observar o decréscimo do erro na amostra de teste (erro de generalização) e quando esse estabilizar ou aumentar, fica aí definido o número ideal de iterações.

APÊNDICE B - Ilustração Didática do Algoritmo Gradiente Boosting de Friedman

Para exemplificar a ideia do algoritmo Gradiente Boosting de Friedman, considere o seguinte exemplo com variável resposta assumida como cotínua, três variáveis preditoras $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_3$, três bases aprendizes lineares com coeficientes $\hat{\beta}_j^{(m)}$, $j = 1, 2, 3$. Considere o seguinte conjunto de dados:

Y_i	X_{1i}	X_{2i}	X_{3i}
8	2	1	4
10	-1	2	1
9	1	-3	4
6	2	1	2
12	1	4	6

1. Como primeiro passo do algoritmo, um valor inicial $\hat{f}^{(0)}(\cdot)$ considerando a função perda erro quadrático (2.9) é a média da resposta Y. Essa derivação será feita depois desse exemplo. Logo

$$\hat{f}^{(0)} = \bar{Y} = 9$$

2. Aumentamos m em 1 e calculamos o vetor gradiente negativo referente a

perda 2.9, cuja derivação será feita também ao final do exemplo. Assim,

$$\begin{aligned} z_i = -\frac{\partial \rho\left(Y_i,\, f\right)}{\partial f} = Y_i - \hat{f}^{(0)} &\Rightarrow z_1 = 8 - 9 = -1 \\ &\Rightarrow z_2 = 10 - 9 = 1 \\ &\Rightarrow z_3 = 9 - 9 = 0 \\ &\Rightarrow z_4 = 6 - 9 = -3 \\ &\Rightarrow z_5 = 12 - 9 = 3 \end{aligned}$$

3. No caso de ajuste de modelos lineares generalizados, o procedimento base adequado é o da equação 2.13 com parâmetros estimados por 2.14. Logo

$$\begin{aligned} \hat{\beta}_{(j=1)} =& \frac{2 \times (-1) + (-1) \times 1 + 1 \times 0 + 2 \times (-3) + 1 \times 3}{2^2 + (-1)^2 + 1^2 + 2^2 + 1^2} = -0,5454 \\ \hat{\beta}_{(j=2)} =& \frac{1 \times (-1) + 2 \times 1 + (-3) \times 0 + 1 \times (-3) + 4 \times 3}{1^2 + 2^2 + (-3)^2 + 1^2 + 4^2} = 0,3226 \\ \hat{\beta}_{(j=3)} =& \frac{4 \times (-1) + 1 \times 1 + 4 \times 0 + 2 \times (-3) + 6 \times 3}{4^2 + 1^2 + 4^2 + 2^2 + 6^2} = 0,1233 \end{aligned}$$

queremos o $\hat{\beta}$ que retorna a menor soma de quadrados do resíduo, que é

dado resolvendo-se a expressão 2.15, daí

$$\begin{aligned}
\hat{\lambda}^{(j=1)} &= [(-1) - (-0,5454) \times 2]^2 + [1 - (-0,5454) \times (-1)]^2 \\
&\quad + [0 - (-0,5454) \times 1]^2 + [-3 - (-0,5454) \times 2]^2 \\
&\quad + [3 - (-0,5454) \times 1]^2 = 16,7273 \\
\hat{\lambda}^{(j=2)} &= [(-1) - 0,3226 \times 1]^2 + [1 - 0,3226 \times 2]^2 + [0 - 0,3226 \times (-3)]^2 \\
&\quad + [-3 - 0,3226 \times 1]^2 + [3 - 0,3226 \times 4]^2 = 16,7742 \\
\hat{\lambda}^{(j=3)} &= [(-1) - 0,1233 \times 4]^2 + [1 - 0,1233 \times 1]^2 + [0 - 0,1233 \times 4]^2 \\
&\quad + [-3 - 0,1233 \times 2]^2 + [3 - 0,1233 \times 6]^2 = 18,8904
\end{aligned}$$

Portanto, a variável escolhida nessa iteração é X_1, uma vez que produziu menor valor para $\hat{\lambda}$.

4. Então a atualização é dada por

$$\begin{aligned}
\hat{f}^{(1)}(x) &= \hat{f}^{(0)} + v \times \hat{g}^{(1)}(x) \\
&= \hat{f}^{(0)} + v \times \hat{\beta}^{(\hat{\lambda}_1)} x^{(\hat{\lambda}_1)} \\
&= 9 + 0,1 \times (-0,5454) \times X_1 \\
\hat{f}^{(1)}(x) &= 9 - 0,0545 \times X_1
\end{aligned}$$

Agora procedemos à segunda iteração. Retornemos ao passo dois do algoritmo.

1. o vetor gradiente negativo é dado por

$$z_i = Y_i - \underbrace{(9 - 0,0545 \times X_1)}_{\hat{f}^{(1)}} \Rightarrow z_1 = 8 - 9 + 0,0545 \times 2 = -0,8909$$

$$\Rightarrow z_2 = 10 - 9 + 0,0545 \times (-1) = 0,9454$$

$$\Rightarrow z_3 = 9 - 9 + 0,0545 \times 1 = 0,0545$$

$$\Rightarrow z_4 = 6 - 9 + 0,0545 \times 2 = -2,8909$$

$$\Rightarrow z_5 = 12 - 9 + 0,0545 \times 1 = 3,0545$$

2. Ajustando o vetor gradiente ao procedimento base, a fim de obter $\hat{g}^{(2)}(x)$, daí

$$\hat{\beta}_{(j=1)} = \frac{2 \times z_1 + (-1) \times z_2 + 1 \times z_3 + 2 \times z_4 + 1 \times z_5}{2^2 + (-1)^2 + 1^2 + 2^2 + 1^2} = -0,4909$$

$$\hat{\beta}_{(j=2)} = \frac{1 \times z_1 + 2 \times z_2 + (-3) \times z_3 + 1 \times z_4 + 4 \times z_5}{1^2 + 2^2 + (-3)^2 + 1^2 + 4^2} = 0,3279$$

$$\hat{\beta}_{(j=3)} = \frac{4 \times z_1 + 1 \times z_2 + 4 \times z_3 + 2 \times z_4 + 6 \times z_5}{4^2 + 1^2 + 4^2 + 2^2 + 6^2} = 0,1390$$

queremos o $\hat{\beta}$ que retorna menor $\hat{\lambda}$

$$\begin{aligned}
\hat{\lambda}^{(j=1)} =& [-0,8909 + 0,4909 \times 2]^2 + [0,9454 + 0,4909 \times (-1)]^2 \\
&+ [0,0545 + 0,4909 \times 1]^2 + [-2,8909 + 0,4909 \times 2]^2 \\
&+ [3,0545 + 0,4909 \times 1]^2 = 16,7273 \\
\hat{\lambda}^{(j=2)} =& [-0,8909 - 0,3279 \times 1]^2 + [0,9454 - 0,3279 \times 2]^2 \\
&+ [0,0545 - 0,3279 \times (-3)]^2 + [-2,8909 - 0,3279 \times 1]^2 \\
&+ [3,0545 - 0,3279 \times 4]^2 = 16,0460 \\
\hat{\lambda}^{(j=3)} =& [-0,8909 - 0,1390 \times 4]^2 + [0,9454 - 0,1390 \times 1]^2 \\
&+ [0,0545 - 0,1390 \times 4]^2 + [-2,8909 - 0,1390 \times 2]^2 \\
&+ [3,0545 - 0,1390 \times 6]^2 = 17,9682
\end{aligned}$$

3. A atualização de $\hat{f}^{(2)}(x)$ é dada por

$$\begin{aligned}
\hat{f}^{(2)}(x) =& \hat{f}^{(1)}(x) + v \times \hat{g}^{(2)}(x) \\
=& \hat{f}^{(1)}(x) + v \times \hat{\beta}^{(\hat{\lambda}_2)} x^{(\hat{\lambda}_2)} \\
=& \underbrace{9 - 0,0545 \times X_1}_{\hat{f}^{(1)}(x)} + 0,1 \times 0,3279 \times X_2 \\
\hat{f}^{(2)}(x) =& 9 - 0,0545 \times X_1 + 0,0328 \times X_2
\end{aligned}$$

A terceira iteração é feita de forma análoga ás iterações anteriores.

1. voltando para o passo 2 do algoritmo, calculamos o vetor gradiente negativo

$$
\begin{aligned}
z_1 &= -\,0,9237 \\
z_2 &= 0,8799 \\
z_3 &= 0,1529 \\
z_4 &= -\,2,9237 \\
z_5 &= 2,9234
\end{aligned}
$$

2. Obtendo $\hat{g}^{(3)}(x)$

$$
\begin{aligned}
\hat{\beta}^{(j=1)} &= -\,0,4998 \qquad \hat{\lambda}^{(j=1)} = 15,9967 \\
\hat{\beta}^{(j=2)} &= 0,2951 \qquad \hat{\lambda}^{(j=1)} = 15,9967 \\
\hat{\beta}^{(j=3)} &= 0,1299 \qquad \hat{\lambda}^{(j=3)} = 19,5007
\end{aligned}
$$

3. Logo, a atualização é dada por

$$
\begin{aligned}
\hat{f}^{(3)}(x) &= \hat{f}^{(2)}(x) + v \times \hat{g}^{(3)}(x) \\
&= \hat{f}^{(2)}(x) + v \times \hat{\beta}^{(\hat{\lambda}_3^1)} x^{(\hat{\lambda}_3^1)} \\
&= \underbrace{9 - 0,0545 \times X_1 + 0,0328 \times X_2}_{\hat{f}^{(2)}(x)} + 0,1 \times (-0,4998) \times X_1 \\
&= 9 - (0,0545 + 0,0500) \times X_1 + 0,0328 \times X_2 \\
\hat{f}^{(3)}(x) &= 9 - 0,1045 \times X_1 + 0,0328 \times X_2
\end{aligned}
$$

O algoritmo poderia continuar a ser executado por várias iterações e até mesmo por um número muito grande de iterações. Como dito anteriormente, executar o algoritmo de forma indefinida pode acarretar problemas no modelo, como por exemplo, forçar a escolha de uma variável não significativa ao modelo. Uma forma de determinar o número ideal de iterações é plotar em um gráfico o AIC resultante do modelo a cada iteração e quando o AIC atingir seu valor mínimo, este representará o número ótimo de iterações do algoritmo.

Como visto no exemplo, uma mesma variável pode ser escolhida não apenas em uma iteração, mas em várias iterações, aumentando sua contribuição individual no modelo final. Desse processo pode ocorrer também de alguma variável não estar no modelo final, caracterizando portanto, o sistema de seleção de variáveis, que está embutido no algoritmo.

APÊNDICE C - Demonstração 1

Vamos fazer agora a derivação do valor inicial $\hat{f}^{(0)}(\cdot)$ do passo um do algoritmo Gradiente Boosting de Friedman considerando a **perda quadrática**. Devemos obter o valor de c que minimiza a perda média, ou seja

$$\begin{aligned}\hat{f}^{(0)}(\cdot) &= \arg\min_{c} \frac{1}{n}\sum_{i=1}^{N} \rho\left(Y_i,\, c\right) \\ &= \arg\min_{c} \frac{1}{n}\sum_{i=1}^{N} \frac{1}{2}\left(y_i - c\right)^2 \\ &= \frac{1}{2n}\sum_{i=1}^{N}\left(y_i^2 - 2y_i c + c^2\right)\end{aligned}$$

em seguida derivamos essa expressão em relação a c e igualamos a zero

$$\begin{aligned}\frac{\partial \rho\left(y,\, c\right)}{\partial c} = 0 \Rightarrow \frac{1}{2n}\sum_{i=1}^{N}\left(-2y_i + 2c\right) = 0 \\ \Rightarrow -2\sum\nolimits_{i=1}^{N} y_i + \sum\nolimits_{i=1}^{N} 2c = 0 \\ \Rightarrow -2\sum\nolimits_{i=1}^{N} y_i = -2nc \\ \Rightarrow -2\sum\nolimits_{i=1}^{N} y_i = -2nc \\ \Rightarrow c = \sum\nolimits_{i=1}^{N} y_i \Big/ n = \bar{Y}\end{aligned}$$

Portanto, o valor de c que minimiza a perda quadrática é a média da variável resposta Y, ou seja, um valor inicial adequado do primeiro passo do algoritmo seria $\bar{Y}$. Note que a perda quadrática assume valores positivos para quaisquer valores de y e f, logo c é realmente um valor de mínimo.

Com um simples cálculo, pode-se obter o vetor gradiente negativo z_i para a perda quadrática

$$\begin{aligned}z_i = -\frac{\partial \rho\left(y,\, f\right)}{\partial f} &= -\frac{\partial}{\partial f}\left[\frac{1}{2}\left(y - f\right)^2\right] \\ &= -\left[\left(y - f\right)\left(-1\right)\right] \\ z_i = -\frac{\partial \rho\left(y,\, f\right)}{\partial f} &= y - f\end{aligned}$$

APÊNDICE D - Demonstração 2

De forma análoga ao feito no APÊNDICE C, pode-se fazer a derivação do valor inicial $\hat{f}^{(0)}(\cdot)$ do passo um do algoritmo Gradiente Boosting de Friedman considerando a **perda binomial** (2.5). Primeiramente, devemos obter o valor de c, logo

$$
\begin{aligned}
\hat{f}^{(0)}(\cdot) &= \arg\min_{c} \frac{1}{n}\sum_{i=1}^{N} \rho\left(Y_i,\, c\right) \\
&= \arg\min_{c} \frac{1}{n}\sum_{i=1}^{N} -\left\{y_i\, \ln c + (1-y_i)\ln\left(1-c\right)\right\} \\
&= -\frac{1}{n}\sum\nolimits_{i=1}^{N} y_i\, \ln c - \frac{1}{n}\sum\nolimits_{i=1}^{N} (1-y_i)\ln\left(1-c\right) \\
&= -\frac{\ln c}{n}\sum\nolimits_{i=1}^{N} y_i\, - \frac{1}{n}\sum\nolimits_{i=1}^{N} \left[\ln\left(1-c\right) - y_i \ln\left(1-c\right)\right] \\
&= -\frac{\ln c}{n}\sum\nolimits_{i=1}^{N} y_i\, - \frac{1}{n}\sum\nolimits_{i=1}^{N} \ln\left(1-c\right) + \frac{\ln\left(1-c\right)}{n}\sum\nolimits_{i=1}^{N} y_i \\
&= -\frac{\ln c}{n}\sum\nolimits_{i=1}^{N} y_i\, - \ln\left(1-c\right) + \frac{\ln\left(1-c\right)}{n}\sum\nolimits_{i=1}^{N} y_i
\end{aligned}
$$

em seguida derivamos essa expressão em relação a c e igualamos a zero

$$
\begin{aligned}
\frac{\partial \rho\left(y,\, c\right)}{\partial c} = 0 \Rightarrow & -\frac{1}{nc}\sum_{i=1}^{N} y_i + \frac{1}{1-c} - \frac{1}{n\left(1-c\right)}\sum_{i=1}^{N} y_i = 0 \\
& \Rightarrow -\sum_{i=1}^{N} y_i \left(\frac{1}{nc} + \frac{1}{n-nc}\right) = -\frac{1}{1-c} \\
& \Rightarrow -\sum_{i=1}^{N} y_i \left(\frac{n-nc+nc}{nc\left(n-nc\right)}\right) = -\frac{1}{1-c} \\
& \Rightarrow \sum_{i=1}^{N} y_i = \frac{1}{1-c}\frac{nc\left(n-nc\right)}{n} \\
& \Rightarrow \sum_{i=1}^{N} y_i = \frac{1}{1-c}\frac{nc-nc^2}{1} \\
& \Rightarrow \left(1-c\right)\sum_{i=1}^{N} y_i = nc - nc^2 \\
& \Rightarrow \left(1-c\right)\frac{\sum_{i=1}^{N} y_i}{n} = c - c^2 \\
& \Rightarrow \left(1-c\right)\bar{Y} = c\left(1-c\right) \\
& \Rightarrow c = \bar{Y}
\end{aligned}
$$

Portanto, o valor de c que minimiza a perda 2.5 é a frequência relativa de $Y = 1$. Note que essa perda assume valores estritamente positivos, logo c é realmente um valor de mínimo.

Vamos obter agora o vetor gradiente negativo z_i para a perda 2.5. O cálculo é feito de forma análoga ao feito para a perda quadrática.

$$
\begin{aligned}
z_i = -\frac{\partial \rho\left(y,\, f\right)}{\partial f} &= -\frac{\partial}{\partial f}\left\{-\left[y_i\, \ln p\left(f\right) + \left(1-y_i\right)\ln\left(1-p\left(f\right)\right)\right]\right\} \\
&= -\frac{\partial}{\partial f}\left\{-\left[y_i\, \ln\left(\frac{e^f}{1+e^f}\right) + \left(1-y_i\right)\ln\left(1-\frac{e^f}{1+e^f}\right)\right]\right\} \\
&= -\frac{\partial}{\partial f}\left\{-\left[y_i\, \ln\left(\frac{e^f}{1+e^f}\right) - \ln\left(\frac{1}{1+e^f}\right) + y_i \ln\left(\frac{e^f}{1+e^f}\right)\right]\right\} \\
&= -\left\{-y_i \frac{1+e^f}{e^f}\frac{e^f}{\left(1+e^f\right)^2} - \frac{1+e^f}{1}\frac{\left(-e^f\right)}{\left(1+e^f\right)^2} + y_i \frac{1+e^f}{1}\frac{\left(-e^f\right)}{\left(1+e^f\right)^2}\right\} \\
&= -\left\{-y_i \frac{1}{1+e^f} + \frac{e^f}{1+e^f} - y_i \frac{e^f}{1+e^f}\right\} \\
&= -\left\{-y_i \left[\frac{1}{1+e^f} + \frac{e^f}{1+e^f}\right] + \frac{e^f}{1+e^f}\right\} \\
&= -\left\{-y_i \left[\frac{1+e^f}{1+e^f}\right] + \frac{e^f}{1+e^f}\right\} \\
&= -\left\{-y_i \left[\frac{1+e^f}{1+e^f}\right] + \frac{\frac{e^f}{e^f}}{\frac{1+e^f}{e^f}}\right\} \\
&= -\left\{-y_i + \frac{1}{1+e^{-f}}\right\} \\
z_i = -\frac{\partial \rho\left(y,\, f\right)}{\partial f} &= y_i - \frac{1}{1+e^{-f}}
\end{aligned}
$$