Universidade Federal de São Carlos
Centro de Ciências Exatas e de tecnologia
Departamento de Estatística

Comparação de medidas de avaliação do poder preditivo em modelos com resposta binária

Claudio Henrique Leão de Almeida

Trabalho de Conclusão de Curso

Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Departamento de Estatística

Comparação de medidas de avaliação do poder preditivo em modelos com resposta binária

Claudio Henrique Leão de Almeida
Orientador: Gustavo Henrique de Araujo Pereira

Trabalho de Conclusão de Curso a ser apresentado como parte dos requisitos para obtenção do título de Bacharel em Estatística.

São Carlos
14 de Janeiro de 2021

Claudio Henrique Leão de Almeida

Comparação de medidas de avaliação do poder preditivo em modelos com resposta binária

Este exemplar corresponde à redação final do trabalho de conclusão de curso devidamente corrigido e defendido por Claudio Henrique Leão de Almeida e aprovado pela banca examinadora.

São Carlos, 14 de Janeiro de 2021.

Banca Examinadora

- Gustavo Henrique de Araújo Pereira
- Maria Sílvia de Assis Moura
- Pedro Ferreira Filho

Dedicatória

Dedico este trabalho aos meus pais, Maria Aparecida Leão de Almeida e José de Almeida.

Agradecimentos

Agradeço primeiramente à Deus, porque a fé com certeza sempre foi e é um conforto em todos os momentos. E também por ele me ter feito alcançar coisas que com sozinho não conseguiria.

Agradeço aos meus pais, pois sempre foram os meus maiores admiradores. Sempre fizeram dos meus sonhos, os deles. Me mostraram o que é um amor incondicional. Obrigado por me transformarem em quem sou hoje. Essa vitória foi muito mais fácil com vocês ao meu lado. Nós conseguimos!

Agradeço aos meus irmãos, pela união e amor recíproco que temos. Também sou grato à toda a minha família, por sempre torcerem pelo meu sucesso.

Agradeço ao meu orientador Prof. Gustavo Henrique de Araújo Pereira pela excelente parceira, por todo ensinamento e paciência. Maiormente agradeço por se tornar um grande conselheiro e amigo, e por ter se tornado uma grande inspiração como profissional para mim.

Agradeço ao Prof. Pedro Ferreira Filho e a Prof. Maria Sílvia de Assis Moura por aceitarem o convite para fazer parte da banca examinadora deste Trabalho e por todas as contribuições dadas, com certeza foram de grande importância. Agradeço também aos demais professores e funcionários do departamento de estatística da Universidade Federal de São Carlos que fizeram parte, de alguma forma, da minha graduação.

Agradeço a todos os professores que passaram pela minha vida desde o Ensino Fundamental. Muitos enxergaram esse capítulo na minha história, e me incentivaram a segui-lo. Em especial, sou grato a Prof. Daiane Aparecida Zuanetti, por ser uma grande amiga, conselheira, me arrisco a dizer psicóloga, e por me inspirar tanto.

Agradeço ao grupo de extensão que fiz parte, o PET, por ser a minha casa, minha família em São Carlos. Por ter me proporcionado tantos momentos felizes e inesquecíveis.

Agradeço aos meus amigos, por todos os momentos compartilhados. Minha graduação e minha vida foram muito melhores com a presença deles. Espero levá-los para minha vida inteira.

Resumo

Os Modelos Lineares Gereralizados (MLGs) foram propostos por Nelder e Wedderburn (1972) como uma extensão do modelo de regressão linear múltipla. Neste trabalho, utilizamos um caso particular dos MLGs, que é a Regressão Logística, empregada quando a variável resposta é binária. Nesse modelo frequentemente tem-se o interesse em saber se as observações estão sendo classificadas corretamente e se novas observações também serão classificadas perfeitamente, ou seja, saber o poder de predição dos modelos. Para isso, foram criadas medidas que verificam o desempenho desses modelos quanto a classificação das observações. Nesta monografia comparamos três dessas medidas: a área sob a curva ROC, a estatística KS e a medida H. Essa comparação foi feita a partir de estudos de simulação e também em bancos de dados reais, através de cálculos de algumas proporções e da construção de alguns boxplots. A principal conclusão que obtemos neste trabalho é que, para quando os coeficientes de regressão são pequenos (em torno do valor 1), o coeficiente de Gini apresenta melhor performance que as demais medidas na avaliação do poder preditivo de modelos de regressão para resposta binária. Já quando o ajuste de regressão apresenta coeficientes com altos valores (em torno do valor 3), a medida H também torna-se um medida com performance compatível ao coeficiente de Gini.

Palavras-chave: *área sob a curva ROC, estatística KS, medida H, modelos lineares generalizados, regressão logística.*

Sumário

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

A análise de regressão é utilizada quando queremos relacionar uma variável resposta com uma ou mais variáveis explicativas. Frequentemente encontramos situações em que a variável resposta é dicotômica ou binária, ou seja, que assume apenas dois resultados (sucesso e fracasso). Alguns exemplos são: (i) o resultado do diagnóstico de um exame de laboratório, positivo ou negativo; (ii) o resultado da inspeção de uma peça recém fabricada, defeituosa ou não defeituosa; (iii) a adimplência de uma empresa ou indivíduo, adimplência ou inadimplência; etc.

Os Modelos Lineares Generalizados (MLGs) são uma das classes principais de modelos de regressão que permitem utilizar outras distribuições para a variável resposta e uma função de ligação relacionando a média da variável resposta à combinação linear das variáveis explicativas, fato que contribui para a construção de modelos mais amplos. Neste trabalho será utilizado um caso particular de MLG, que é a regressão para variável resposta binária. Aqui, o interesse é estudar a probabilidade de ocorrência de um dos valores de uma variável binária em função das outras variáveis e para isso utilizamos a distribuição binomial (Paula, 2004).

Uma das questões comumente de interesse em modelos de regressão para dados binários é a classificação das observações, ou seja, se elas estão sendo classificadas como sucesso ou fracasso quando elas são verdadeiramente sucesso ou fracasso. A classificação tem por objetivo predizer características de dados futuros, mediante à informações já disponíveis (Nunes, 2011). Exemplificando, considere que o modelo que estamos ajustando precisa classificar um indivíduo como saudável ou doente, classificar erroneamente pode levar a sérias consequências. Como uma tentativa de contornar possíveis erros como o anteriormente citado, foram criadas diversas medidas que verificam o desempenho do

modelo quanto a essa classificação, ou seja, o quanto o modelo está prevendo corretamente. As medidas mais conhecidas são a área sob a curva ROC e a estatística KS (Kolmogorov-Smirnov) (Alves, 2008).

Existem outras medidas menos conhecidas, um exemplo é a medida H, proposta por Hand (2009), sendo uma alternativa à área sob a curva ROC que supre algumas incoerências da mesma. Ela será abordada, com mais detalhes posteriormente.

Diante das medidas citadas acima, surge o interesse em investigar se alguma dentre elas apresenta resultados mais eficientes do que as outras, ou seja, se alguma delas é mais adequada para avaliar o poder preditivo do modelo. Portanto, o principal objetivo deste trabalho é a comparação de algumas medidas que avaliam o poder preditivo de modelos de regressão para respostas binárias. Esta comparação será feita a partir de estudos de simulação e aplicações em dados reais.

Este trabalho está organizado da seguinte forma. No Capítulo 2, apresentamos a estrutura e os componentes de um modelo linear generalizado, com enfoque principal para dados binários, descrevendo as funções de ligação mais utilizadas, procedimentos para estimação dos parâmetros pelo método da máxima verossimilhança, análise de diagnóstico e também métodos de seleção de variáveis. No Capítulo 3, abordamos as medidas de avaliação do poder preditivo em modelos para dados binários, que serão comparadas neste trabalho. Em seguida, no Capítulo 4 é apresentado a implementação da metodologia estudada com aplicação em diferentes bancos de dados simulados. O Capítulo 5 é dedicado a implementação da metodologia estudada com aplicação em dois conjunto de dados reais. Ambas as implementações foram feitas pelo *software R* (R Core Team, 2020). Por fim, no Capítulo 6 concluímos este trabalho.

Capítulo 2

Modelos Lineares Generalizados

Na Seção 2.1 é visto a estrutura e os componentes de um modelo linear generalizado. Na Seção 2.2 aborda-se a distribuição binomial, apresentando sua densidade e logaritmo da função de verossimilhança em um MLG com resposta binária. Na Seção 2.3 apresentamos as funções de ligação mais utilizadas no caso binário. Discutimos a interpretação dos parâmetros, para o caso em que utilizamos a função de ligação canônica na Seção 2.4. Na Seção 2.5 vemos a estimação dos parâmetros do modelo. A análise de diagnóstico do modelo é apresentada na Seção 2.6. Por fim, na Seção 2.7 discutimos os métodos de seleção de variáveis Backward e Lasso.

2.1 Especificação do modelo

Nelder e Wedderburn (1972) mostraram que um conjunto de modelos estatísticos já existentes, estudados separadamente, podem ser agrupados como uma classe de modelos de regressão. A característica em comum de todos esses modelos, que nos permite uní-los, é que as distribuições das variáveis respostas dos mesmos pertencem à família exponencial linear. A essa classe, que é uma extensão dos modelos clássicos de regressão, foi dado o nome de modelos lineares generalizados, que a partir de agora representaremos pela sigla MLG. Uma das maiores vantagens do MLG então, é que a suposição de normalidade não é necessária.

Sejam $y_1, y_2, ..., y_n$ variáveis aleatórias independentes, assume-se que todos os y_i tem distribuição que pertence a família exponencial linear, na qual os parâmetros são θ_i, $i = 1, ..., n$ e ϕ e os mesmos também possuem função de probabilidade dada por

$$f(y,\theta,\phi) = \exp\left\{\phi[y\theta - b(\theta)] + c(y,\phi)\right\}, \tag{2.1}$$

em que $b(.)$ e $c(.)$ são funções conhecidas. Temos que $E(y_i) = b'(\theta)$ e $Var(y_i) = \phi^{-1}b''(\theta)$.

A especificação do MLG se faz através de um modelo de ligação entre a média μ_i e as covariáveis, ou seja,

$$g(\mu_i) = \boldsymbol{x_i}^\top \boldsymbol{\beta}, \tag{2.2}$$

onde $\mu_i = E(y_i) = b'(\theta_i)$, $\boldsymbol{\beta} = (\beta_0, \beta_1, ..., \beta_k)^\top$ e $\boldsymbol{x_i} = (1, x_{i1}, x_{i2}, ..., x_{ik})^\top$ são constantes que representam os valores das variáveis preditoras e g é uma função de ligação estritamente monótona e pelo menos duplamente diferenciável. Podemos representar também (2.2) como $g(\mu_i) = \eta_i$, em que $\eta_i = \boldsymbol{x_i}^\top \boldsymbol{\beta}$ é denominado preditor linear.

Os MLGs são caracterizados pela seguinte estrutura:

- Componente aleatório: representado por um conjunto de variáveis independentes $y_1, y_2, ..., y_n$ provenientes de uma mesma distribuição da família exponencial linear com parâmetros θ_i e ϕ, $i = 1, ..., n$.

- Componente sistemático: representado pelo termo $\eta_i = \boldsymbol{x_i}^\top \boldsymbol{\beta}$. O componente linear engloba tanto o vetor de parâmetros β como as variáveis preditoras e o componente sistemático é linear nos parâmetros.

- Função de ligação: relaciona o componente aleatório ao sistemático, sendo $g(.)$ uma função estritamente monótona e duplamente diferenciável.

Portanto, passos importantes no ajuste de um MLG são: (i) escolha de uma distribuição adequada para a variável resposta; (ii) escolha da matriz do modelo, ou seja, das variáveis preditoras mais importantes para o modelo e (iii) escolha da melhor função de ligação (Demétrio, 2002).

Neste trabalho será utilizado um caso particular de MLG, que é a regressão para variável resposta binária. Neste caso, o interesse é estudar a probabilidade de ocorrência de um dos valores de uma variável binária em função das outras variáveis e para isso utilizamos a distribuição binomial (Paula, 2004). Quando é usada uma função de ligação conhecida como Logito e que será definida na Seção 2.3, temos o conhecido modelo de regressão logística (Hosmer Jr e Lemeshow, 2004).

2.2 Distribuição Binomial

Considere uma variável aleatória Y definida como o número de sucessos em n ensaios independentes de Bernoulli (só assume dois resultados). Dizemos que nesse caso Y tem distribuição binomial, sendo μ a probabilidade de sucesso em cada ensaio de Bernoulli, a função de probabilidade da distribuição binomial é então dada por

$$P(Y = y) = \binom{n}{y} \mu^y (1-\mu)^{n-y} I_{\{0,1,\ldots,n\}}(y). \tag{2.3}$$

Em Magalhães (2011), por exemplo, mostra-se que $E(Y) = n\mu$ e $Var(Y) = n\mu(1-\mu)$.

Nesse trabalho, o interesse é quando a variável resposta é binária, portanto, $n = 1$, e nesse caso particular da Binomial temos a distribuição de Bernoulli. Nesse caso $Y \sim Bernoulli(\mu)$ e no formato da família exponencial apresentado em (Nelder e Wedderburn, 1972), sua função de probabilidade é dada por

$$P(Y = y) = \exp\left\{ y \log\left(\frac{\mu}{1-\mu}\right) + \log(1-\mu) \right\} I_{\{0,1\}}(y).$$

2.3 Função de ligação

Qualquer função $g(.)$, desde que seja estritamente monótona e duplamente diferenciável, pode ser considerada função de ligação. Se temos que $g(\mu_i) = \theta_i$, o preditor linear modela o parâmetro canônico θ_i e denominamos esta função de ligação como canônica (Paula, 2004). A escolha da função de ligação também depende do tipo de variável resposta utilizada.

Para o caso particular estudado nesse trabalho, ou seja, quando a distribuição da variável resposta é Bernoulli, a função de ligação canônica é a Logito, que é definida por,

$$g(\mu_i) = \log\left(\frac{\mu_i}{1-\mu_i}\right) = x_i^\top \beta, \tag{2.4}$$

e assim, aplicando exponencial em ambos os lados da igualdade da equação 2.4, e isolando

μ_i obtemos,

$$\mu_i = \frac{\exp(x_i^\top \beta)}{1 + \exp(x_i^\top \beta)}. \tag{2.5}$$

A função Logito é a mais utilizada para modelos com respostas binárias pois produz parâmetros facilmente interpretáveis (ver Seção 2.4), valores ajustados no intervalo $(0; 1)$ e também pelo fato de ser a ligação canônica.

As funções Probito e Complemento Log-Log, que são definidas respectivamente como (2.6) e (2.7), também são bastante utilizadas. Elas também produzem valores ajustados no intervalo (0;1), mas os seus parâmetros não são de fácil interpretação. Tem-se também que

$$g(\mu_i) = \Phi^{-1}(\mu_i) = x_i^\top \beta, \tag{2.6}$$

em que $\Phi(.)$ é a função de distribuição acumulada da normal padrão e assim $\mu_i = \Phi(\mu_i)$. Tem-se ainda que

$$g(\mu_i) = \log(-\log(1 - \mu_i)) = x_i^\top \beta, \tag{2.7}$$

e assim $\mu_i = 1 - \exp(-\exp(x_i^\top \beta))$.

2.4 Interpretação dos parâmetros - Ligação Canônica

Considerando-se o componente sistemático e a função de ligação Logito para o caso binário temos o seguinte modelo de regressão:

$$\log\left(\frac{\mu_i}{1 - \mu_i}\right) = x_i^\top \beta, \tag{2.8}$$

Para interpretarmos os parâmetros do modelo definido em (2.8) utilizamos a Razão de Chances (Agresti, 1990). Considere $x_{ij} = l$, fixando-se as outras variáveis, temos então que

$$\left.\frac{\mu_i}{1 - \mu_i}\right|_{x_{ij}=l} = \exp(\beta_0 + \beta_1 x_{i1} + \cdots + l \times \beta_j + \cdots + \beta_k x_{ik}). \tag{2.9}$$

Agora, se $x_{ij} = l + 1$,

$$\left.\frac{\mu_i}{1-\mu_i}\right|_{x_{ij}=l+1} = \exp(\beta_0 + \beta_1 x_{i1} + \cdots + (l+1) \times \beta_j + \cdots + \beta_k x_{ik}). \tag{2.10}$$

A Razão de Chances para este caso é definida por $\frac{(2.10)}{(2.9)}$. Portanto,

$$\frac{(2.10)}{(2.9)} = \frac{\exp(\beta_0 + \beta_1 x_{i1} + \cdots + (l+1) \times \beta_j + \cdots + \beta_k x_{ik})}{\exp(\beta_0 + \beta_1 x_{i1} + \cdots + l \times \beta_j + \cdots + \beta_k x_{ik})} = \exp(\beta_j), \tag{2.11}$$

em que $i = 1, \ldots, n$.

Sendo assim, podemos interpretar a resultante $\exp(\beta_j)$ como o valor pelo qual é multiplicado a chance de ocorrência de sucesso na variável resposta quando x_{ij} tem acréscimo de uma unidade, mantendo as demais variáveis preditoras constantes (Brolo, 2019).

2.5 Estimação dos parâmetros

A técnica de estimação para o vetor de parâmetros β, proposto por Nelder e Wedderburn (1972), utiliza o método de máxima verossimilhança, de forma para facilitar os cálculos matemáticos maximizamos o logaritmo da função de verossimilhança, já que ambas as formas levam ao mesmo estimador.

O logaritmo da função de verossimilhança em modelos de regressão para dados binários é dado por

$$l(\mu; y) = \sum_{i=1}^{n} \{y_i \log\left(\frac{\mu_i}{1-\mu_i}\right) + \log(1-\mu_i)\}. \tag{2.12}$$

Porém, ao maximizarmos a função de verossimilhança, percebemos que não há solução analítica e portanto a maximização é obtida por métodos de otimização numérica. Segundo McCullagh e Nelder (1989) a técnica de otimização mais utilizada é a de mínimos quadrados reponderados iterativos (Antonio, 2009).

2.6 Análise de Diagnóstico

A análise de diagnóstico é uma etapa fundamental no ajuste de modelos de regressão. No caso de MLG, ela está envolvida principalmente na verificação dos seguintes itens:

- Avaliação da distribuição proposta;
- Adequação da função de ligação;
- Identificação e avaliação do efeito de observações mal ajustadas;
- Identificação de pontos influentes e pontos de alavanca, e avaliação do impacto de tais observações no ajuste do modelo.

Como o objetivo principal deste trabalho é comparar medidas de avaliação do poder preditivo dos modelos, não entraremos em detalhes sobre a análise de diagnóstico. Paula (2004) expõe mais detalhes sobre o assunto.

2.7 Seleção de Variáveis

Os métodos de seleção automática de variáveis foram criados com o objetivo de encontrar um subconjunto de variáveis preditoras potencialmente importantes em um modelo. Neste trabalho trabalharemos com duas técnicas de seleção: o Backward (Neter *et al.*, 1996) e o Lasso (**?**).

2.7.1 Backward

O método Backward busca selecionar o subconjunto com as potenciais melhores variáveis preditoras a partir de algum critério (medida) ou a partir de resultados de testes de hipóteses. Algumas medidas possíveis são: AIC Akaike (1974), BIC (Schwarz, 1978; Akaike, 1978), entre outros. Neste trabalho utilizaremos o critério de seleção AIC.

O critério de informação de Akaike, também representado por AIC, é definido por:

$$AIC = -2l(\hat{\theta}) + 2k^*, \tag{2.13}$$

em que $l(\hat{\theta})$ é a log-verossimilhança maximizada do modelo (calculada com base nos EMVs dos parâmetros) e k^* o número de parâmetros. O componente $2k^*$, em (2.13), atua como termo de penalização atribuído ao número de parâmetros do modelo. O objetivo do termo de penalização é evitar o superajuste do modelo, ou seja, evitar que o modelo se ajuste muito bem nas bases de treinamento do modelo, mas se mostre ineficaz em uma base de teste.

O algoritmo do método de seleção de modelos Backward via AIC consite em, primeiramente, ajustar o modelo com todas as variáveis e em seguida ajustar todos os modelos possíveis, retirando-se apenas uma variável. Destes últimos modelos ajustados, escolhemos aquele modelo que apresenta menor valor do AIC, pois é o que possui o melhor ajuste. Posteriormente, a partir do modelo escolhido anteriormente, ajusta-se todos os modelos possíveis retirando-se uma variável novamente, e é escolhido mais uma vez o modelo que tiver o menor AIC. Repetimos estes processos até percebermos que os modelos com a retirada de uma variável não reduz o AIC em relação ao modelo escolhido no passo anterior.

2.7.2 Lasso

Proposto por Tibshirani (1996), o Lasso tem com objetivo principal encontrar um estimador dos parâmetros de um modelo de regressão que aumente a capacidade preditiva do modelo (Friedman *et al.*, 2001). O Lasso é bastante utilizado quando estamos trabalhando com problemas de alta dimensionalidade, ou seja, quando o número de variáveis preditoras é superior ao tamanho da amostra. Além disso, ele faz a estimação de parâmetros e a seleção de modelos, simultaneamente. Isto porque, durante o processo, o método faz com que as estimativas de vários parâmetros sejam iguais a zero, e portanto exclui essas variáveis, selecionando somente as restantes.

Quando estamos trabalhando com MLGs, ou seja, quando a estimação dos parâmetros é feita por máxima verossimilhança, a ideia do Lasso é adicionar uma penalização ao oposto do logaritmo da função de verossimilhança.

Para regressão logística, que é o tipo de MLG estudado neste trabalho, o logaritmo da função de verossimilhança é representado pela equação (2.12). Portanto, matematicamente, a ideia do Lasso é dada por,

$$-\, l(\mu;\beta;y) + \lambda \sum_{i=1}^{d} |\beta_i| \,, \tag{2.14}$$

em que λ é um *tuning parameter* (Izbicki e dos Santos, 2020).

Como já foi dito, o Lasso também funciona como um método de seleção automática de variáveis. Ele diminui de forma contínua os coeficientes estimados em direção a zero a medida que aumenta-se λ. Se λ é suficientemente grande, obtemos muitos coeficientes iguais a zero. Esta diminuição nos coeficientes à medida que λ cresce, melhora

frequentemente, a capacidade preditiva do modelo em virtude da redução da variância dos estimadores. Sendo assim, a escolha do valor de λ é muito importante, pois um valor muito alto de λ pode levar a exclusão de variáveis importantes do modelo (Brolo, 2019). Dessa forma, é importante escolher bem o valor de λ. O método mais comum de sua escolha é a validação cruzada (Izbicki e dos Santos, 2020).

Como nosso principal foco são as medidas de performance, não apresentaremos maiores detalhes sobre o Lasso. Ao leitor interessado no assunto, sugere-se a leitura de Hastie *et al.* (2015).

Capítulo 3

Medidas de avaliação do poder preditivo de modelos com resposta binária

Neste capítulo apresentamos as medidas que avaliam o poder preditivo de modelos para variáveis binárias, que são estudadas e comparadas neste trabalho. Na Seção 3.1 é abordada a área sob a curva ROC. Na Seção 3.2 é apresentado a estatística Kolmogorov-Smirnov (KS) e por fim, na Seção 3.3 a medida H é encontrada.

Em um modelo de regressão para variáveis resposta binárias, como já foi dito, se busca frequentemente classificar as observações em uma das categorias consideradas, ou seja, em sucessos ou fracassos e o objetivo é acertar, na maioria das vezes, nessas classificações.

A matriz de classificação é uma das formas utilizadas para relacionar as respostas preditas com as reais, ou seja, nessa matriz são encontrados cruzamentos entre a classificação feita pelo modelo e a condição real da observação. Na diagonal principal da mesma, são encontradas as classificações corretas, e os valores de fora da diagonal principal, correspondem as incorretas.

A seguir é apresentado a estrutura da matriz de classificação:

Tabela 3.1: Matriz de classificação.

Resultado		**Real**	
		Sucesso	**Fracasso**
Do modelo de classificação	**Sucesso**	VP	FP
	Fracasso	FN	VN

em que VP = verdadeiro positivo (sucessos que são classificados corretamente como sucessos), FP = falso positivo (fracassos que são classificados incorretamente como sucessos), FN = falso negativo (sucessos que são classificados incorretamente como fracassos) e VN = verdadeiro negativo (fracassos que são classificados corretamente como fracassos).

A sensibilidade é a estimativa da proporção de sucessos que são classificados verdadeiramente como sucesso. Ela é definida por

$$\text{Sensibilidade} = \frac{VP}{VP + FN}. \tag{3.1}$$

A especificidade é a estimativa da proporção de fracassos que são classificados verdadeiramente como fracasso. Ela é definida por

$$\text{Especificidade} = \frac{VN}{VN + FP}. \tag{3.2}$$

As medidas de especificidade e sensibilidade estimam a probabilidade de que um modelo classifique corretamente os sucessos e fracassos.

Ao ajustarmos um modelo de regressão para variáveis respostas binárias temos para cada observação uma estimativa da probabilidade da mesma ser um sucesso. Classificamos a observação como sucesso se ela for maior que um valor c (denominado de ponto de corte), e como fracasso, caso contrário. Normalmente a melhor forma de avaliar o poder preditivo do modelo é considerando diversos pontos de corte diferentes.

A seguir serão apresentadas algumas medidas de avaliação do poder preditivo de modelos de regressão para respostas binárias, construídas a partir das estimativas anteriores.

3.1 Área sob a curva ROC

A curva ROC tem sido muito utilizada na literatura. Por exemplo, Bradley (1997) usou a área sob a curva ROC na avaliação de algoritmos de aprendizado de máquina, Hajian-Tilaki (2013) usou para a avaliação de testes de diagnóstico médico e Delacour *et al.* (2013) a aplicou em um estudo na área da biologia.

Essa medida surgiu durante a Segunda Guerra Mundial, com o objetivo de

quantificar a capacidade dos radares distinguirem um sinal de ruído, ou seja, quando o radar detectava algo se aproximando, cabia ao operador decidir, por exemplo, se o ruído era derivado de um avião inimigo ou de uma nuvem de pássaros. Posteriormente, passou a ser usada em outras áreas científicas, como por exemplo, psicologia e medicina (Souza, 2019).

A curva ROC é construída utilizando diversos pontos de corte, em que, para cada um deles, teremos valores diferentes para a especificidade e sensibilidade. Ela é obtida ao considerarmos no eixo x o valor de $(1 - \text{especificidade})$ e no eixo y a sensibilidade para cada ponto de corte que foi escolhido.

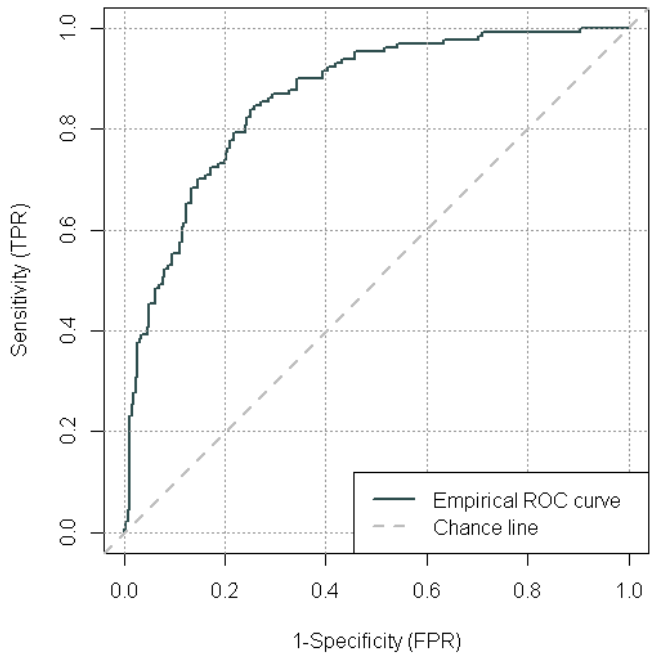
O número de pontos de corte escolhidos para a obtenção da curvas ROC varia de acordo com o problema. Porém, quanto maior a quantidade de pontos de corte, mais preciso é o resultado.

Definindo-se m como o número de sucessos da amostra e q como o número de fracassos, existem três tipos de curva ROC baseado no número de pontos de corte (Vaz, 2009);

- **curva ROC empírica:** obtida a partir de todos os possíveis pontos de corte que a amostra permite, sendo no máximo $m + q$ pontos de corte;
- **curva ROC aproximada:** determinada por um número (menor que $m + q$) de pontos de corte escolhido pelo usuário;
- **curva ROC teórica:** definida a partir do modelo estatístico que define a classificação da variável resposta;

Uma característica interessante é que a curva ROC não se altera se as observações amostrais forem submetidas a transformações, como logaritmo e raiz quadrada (Martinez *et al.*, 2003).

Na Figura 3.1 podemos observar a representação de uma curva ROC empírica que foi gerada usando o pacote do *software R* desenvolvido por Khan e Brandenburger (2020) chamado *ROCit*.



Fonte: Elaborada pelo autor.

Figura 3.1: Exemplo de curva ROC.

Chamamos a área que está situada entre a curva ROC, a reta $x = 1$ e o eixo (1-especificidade) de área sob a curva ROC (AUC). Ela é uma medida que resume o desempenho do modelo. Se o valor da AUC é f, então em $f \times 100\%$ dos $n_0 \times n_1$ pares de sucessos e fracassos, teremos $\hat{\mu}_i$ do sucesso superior ao $\hat{\mu}_i$ do fracasso (Fawcett, 2006).

Estimamos a AUC, considerando todas as especificidades e sensibilidades relativas a cada um dos pontos de corte utilizados. Um modelo péssimo, apresentará sua curva ROC próxima da reta $x = y$. Quanto mais próxima a curva for do canto superior esquerdo, ou seja, mais próxima do eixo y e da reta $y = 1$, mais próxima de 1 é a AUC, portanto melhor é o modelo.

À medida que c cresce, a sensibilidade diminui e a especificidade aumenta. O modelo ideal é aquele que, para pelo menos um ponto de corte, possui 100% de sensibilidade e especificidade, gerando assim uma área sob a curva ROC muito próxima de 1, já que neste caso estamos classificando corretamente 100% das observações. Entretanto, este modelo raramente existe na prática pois a tentativa de melhorar a sensibilidade geralmente tem o efeito de diminuir a especificidade (Brolo, 2019).

Paula (2004) ressalta que na prática a área sob a curva ROC (AUC) varia entre 0,5 e 1, sendo que quando o valor é exatamente 0,5, o modelo não possui poder

discriminante e, quanto maior seu valor, melhor o desempenho do modelo. Se o interesse é uma medida que varia entre 0 e 1, pode-se utilizar o coeficiente de Gini (Thomas *et al.*, 2002), que é dado por 2(AUC - 0,5). Para efeito de comparação com as outras medidas utilizadas nese estudo, que também variam entre 0 e 1, utilizaremos o coeficiente de Gini.

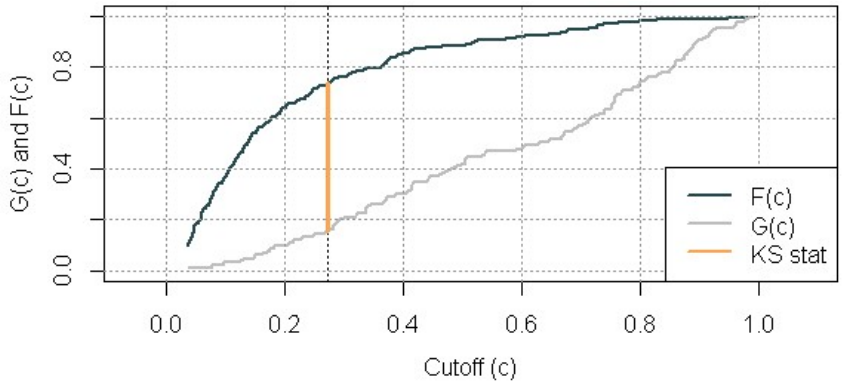
3.2 Estatística KS

A estatística de Kolmogorov-Smirnov (KS) é utilizada em um teste de hipóteses não paramétrico usado para testar se as funções de distribuição de dois grupos são iguais (Conover, 1999). Embora não tenha sido proposta com esse objetivo, é muito utilizada na área de *credit* scoring para avaliar o poder preditivo de um modelo de regressão para respostas binárias. Por exemplo, Forti (2019) estudou a aplicação de algumas técnicas de *machine learning* na recuperação de crédito do mercado brasileiro, enquanto Gouvêa *et al.* (2013) teve seu trabalho focado na aplicação de regressão logística e redes neurais na análise de risco de crédito. Ambos os autores utilizaram o KS para avaliar a qualidade do resultado das técnicas utilizadas por eles. O nome da estatística é em homenagem aos matemáticos russos Andrei Kolmogorov e Nikolai Smirnov (Barakat *et al.*, 2019). Ela é dada por

$$KS = max|F(c) - G(c)|, \tag{3.3}$$

em que $F(c)$ e $G(c)$ correspondem às frequências acumuladas de sucessos (1-sensibilidade) e fracassos (especificidade) respectivamente, no ponto de corte c. A estatística KS é obtida através da distância máxima entre as duas frequências acumuladas (Abreu, 2005). Portanto, quanto mais rápido o crescimento de $F(c)$ e mais lento o de $G(c)$, melhor é o modelo. A discriminação é considerada como aceitável quando a distância é maior que 25%. Acima dos 45% a discriminação é tida como excelentemente (Manfio, 2007). Assim como o coeficiente de Gini, o KS também varia de 0 a 1.

Na Figura 3.2 é apresentado um exemplo de cálculo do KS, onde a reta laranja representa o valor da estatística KS. Ela também foi gerada usando o pacote *ROCit* do *software R*.

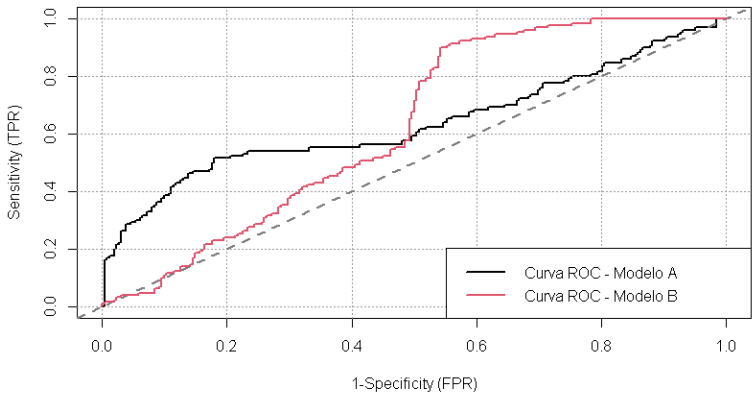


Fonte: Elaborada pelo autor.

Figura 3.2: Exemplo para representar a medida KS.

3.3 medida *H*

Segundo Hand (2009), a área sob a curva ROC possui algumas incoerências. Supondo dois modelos diferentes ajustados, e dado a AUC dos dois, a incoerência mais conhecida corresponde à quando queremos comparar curvas ROC que se cruzam, pois cada curva apresenta maior especificidade para uma região de valores de sensibilidade. Porém, mediante a diferença nos valores da AUC, temos uma falsa impressão de que uma curva domina a outra completamente. Portanto, podemos, nesses casos, tomar conclusões erradas. Na Figura 3.3 é apresentado uma representação em que as curvas ROC de dois modelos diferentes se cruzam. Essa representação também foi gerada usando o pacote *ROCit* do *software R*.



Fonte: Elaborada pelo autor.

Figura 3.3: Exemplo de situação em que duas curvas ROC se cruzam.

Uma outra incoerência está ligada ao custo de classificação incorreta. Este é um valor que representa a gravidade referente a classificar incorretamente as observações em relação ao valor da variável resposta. Em um modelo de regressão para variáveis respostas binárias temos dois custos de classificação incorreta. O primeiro consiste no custo de classificar sucessos como fracassos e outro se refere a classificar os fracassos como sucessos. A AUC é incoerente em relação aos custos de classificações incorretas, pois avalia diferentes métodos de classificação com diferentes métricas, ou seja, a AUC usa diferentes valores para os custos de classificação incorreta dos diferentes métodos de classificação. Em outras palavras, é como medir uma pessoa A com uma régua em centímetros e uma B com outra régua em polegadas, e comparar as duas. Isso não faz sentido, pois o custo das classificações incorretas deveria estar ligado ao problema e não ao método de classificação escolhido para resolvê-lo.

Diante dessas incoerências do AUC, Hand (2009) propôs a medida H, que é uma alternativa à AUC, na qual busca-se corrigir a última incoerência citada no parágrafo anterior. Para isso, essa medida leva em conta a escolha de uma distribuição, $w(.)$, para os custos de classificações incorretas dos diferentes métodos de classificação. Dessa forma, podemos utilizar diferentes métodos de classificação, com os mesmos custos. Assim, a medida H permite uma comparação justa entre os métodos de classificação.

Para obtermos a medida H, é necessário calcular a probabilidade dos elementos da amostra pertencerem a classe 0 ou 1, ou seja, ser fracasso ou sucesso. Essas probabilidades são chamadas de π_0 e π_1, respectivamente, em que $\pi_0 = \dfrac{q}{m+q}$ e $\pi_1 = \dfrac{m}{m+q}$, e m representa o número de sucessos da amostra e q o número de fracassos. Também chamaremos de v_0 o custo de classificação incorreta referente a classificação na classe 1 quando na verdade é da classe 0 e v_1 o contrário. Agora, podemos então definir uma função de custo Q, que representa as penalidades de todas as classificações incorretas.

$$Q(c, v_0, v_1) = v_0\pi_0(1 - G(c)) + v_1\pi_1 F(c). \tag{3.4}$$

O ponto de corte que minimiza a equação (3.4) pode ser denotado como

$$T(v_0, v_1) = \underset{c}{\operatorname{argmin}}(v_0\pi_0(1 - G(c)) + v_1\pi_1 F(c)). \tag{3.5}$$

Porém, analisando a equação (3.5), percebe-se que o ponto de corte c será o mesmo para qualquer par de valores na forma (kv_0, kv_1), com k sendo uma constante

positiva. Portanto, faz-se necessário um mudança de variáveis no par (v_0, v_1), fazendo com que agora a equação (3.5) dependa somente da razão entre v_1 e a soma dos custos (varia de 0 a 1). Fazendo $v = \dfrac{v_1}{v_0 + v_1}$ e $b = v_0 + v_1$, obtemos T^* dado por

$$T^*(v) = \underset{c}{\operatorname{argmin}}\big[((1-v)\pi_0(1-G(c)) + v\pi_1 F(c))b\big]. \tag{3.6}$$

Assim, a equação (3.4) passa a ser representada como

$$Q(c, b, v) = [(1-v)\pi_0(1-G(c)) + v\pi_1 F(c)]b \tag{3.7}$$

O valor de v precisa ser definido em termos do problema, e não em função do método de classificação utilizado, mas essa é uma tarefa muito difícil. Para resolver isso, na medida H, como já foi dito, escolhemos uma função densidade de probabilidade, $w(.)$ para representar v. Segundo Hand e Anagnostopoulos (2014) a melhor distribuição é a $Beta(\pi_0 + 1, \pi_1 + 1)$.

Com todas essas informações, conseguimos definir a perda mínima geral de classificação incorreta. Ela é dada por

$$L = \int_0^1 Q(T^*(v), b, v)w(v)dv. \tag{3.8}$$

Temos também o caso em que ocorre a perda máxima (Hand, 2009). Ela é dada por

$$L_{Max} = \pi_1 \int_0^{\pi_0} vw(v)dv + \pi_0 \int_{\pi_0}^1 (1-v)w(v)dv. \tag{3.9}$$

A partir das equações (3.8) e (3.9) definimos H como

$$H = 1 - \frac{L}{L_{Max}}. \tag{3.10}$$

Se verificarmos a equação (3.10), podemos ver que ela ainda está em função de v, que por sua vez é derivado dos valores de v_0 e v_1. Portanto, se o pesquisador não tiver conhecimento dos valores anteriormente citados, ainda temos problemas para efetuar o cálculo da medida H. Devido a isso, estimaremos o valor de v, e por consequência, o valor da medida H.

3.3.1 Estimação da medida H

Primeiramente, construiremos a curva ROC. A partir dela, chamaremos de s, o número de pontos que assume valores de escores (valores ajustados) diferentes (quando todas as observações possuírem escores diferentes, $s = n_0 + n_1$). Seja σ_{0i} o número de pontos da classe 0 que assumem o i-ésimo valor de escore. Isto vai ser 0 se o valor do i-ésimo escore for obtido apenas por ponto(s) da classe 1. Analogamente, σ_{1i} é o número de pontos da classe 1 que assumem o i-ésimo valor de escore. Seja (r_{00}, r_{10}) = (0,0) as coordenadas iniciais da curva ROC. As demais coordenadas são dadas por

$$(r_{0i}, r_{1i}) = (r_{0(i-1)}, r_{1(i-1)}) + (\frac{\sigma_{0i}}{n_0}, \frac{\sigma_{1i}}{n_1}), i = 1, ..., s. \tag{3.11}$$

O valor de v é então estimado por um vetor de tamanho $s + 1$, que pode ser expresso como

$$\hat{v}_{(j+1)} = \frac{\pi_0(r_{0(j+1)} - r_{0j})}{\pi_1(r_{1(j+1)} - r_{1j}) + \pi_0(r_{0(j+1)} - r_{0j})}, \tag{3.12}$$

em que $j \in \{0, ..., s\}$.

Agora a estimação da equação (3.8), já considerando a distribuição *Beta*, é dada por,

$$\begin{aligned}\hat{L}_\beta = \sum_{i=o}^{s} \Big[& \pi_1(1 - r_{1i})\{B(\hat{v}_{(i+1)}; 1 + \alpha, \beta) - (\hat{v}_{(i)}; 1 + \alpha, \beta)\}/B(1; \alpha, \beta) + \\ & \pi_0 r_{0i}\{B(\hat{v}_{(i+1)}; \alpha, 1 + \beta) - B(c_i; \alpha, 1 + \beta)\}/B(1, \alpha, \beta) \Big].\end{aligned} \tag{3.13}$$

em que $B(a; t, u)$ representa o valor da função densidade de probabilidade de uma Beta com parâmetros t e u no ponto a.

Enfim, temos todas as informações necessárias e a medida H pode ser estimada como

$$\hat{H} = 1 - \frac{\hat{L}_\beta B(1; \alpha, \beta)}{\pi_1 B(\pi_0; 1 + \alpha, \beta) + \pi_0 B(1; \alpha, 1 + \beta) - \pi_0 B(\pi_0; \alpha, 1 + \beta)}. \tag{3.14}$$

Capítulo 4

Estudos de simulação

No capítulo anterior apresentamos algumas medidas de avaliação do poder preditivo em modelos de regressão com resposta binária. Neste capítulo apresentamos e discutimos os estudos de simulação de Monte Carlo desenvolvidos para comparar as medidas anteriormente citadas.

Utilizamos o *software* R para desenvolvimento dos estudos de simulação. Em especial utilizamos a função *glm* para ajuste de MLGs. A seleção de variáveis via Backward foi feita utilizando o critério AIC. O pacote *glmnet* (Friedman *et al.*, 2009) foi utilizado para o ajuste via Lasso. Também utilizamos o pacote *hmeasure*, desenvolvido por Anagnostopoulos e Hand (2019), para cálculo das medidas de avaliação do poder preditivo do modelo.

Não foram utilizados códigos de simulação prontos. Os mesmos foram desenvolvidos ao decorrer do trabalho utilizando o *software R*. Consideramos nos estudos de simulação um tamanho amostral de 500 observações e 1000 réplicas de Monte Carlo. Inicialmente geramos independentemente quatro variáveis preditoras a partir de uma distribuição Uniforme com parâmetros 0 e 1. Estas permaneceram fixas, em todas as réplicas. Em seguida geramos a variável resposta para as 1000 réplicas. A forma como ela é gerada será explicada abaixo.

Dado $\beta = (1, \ell, -\ell, 0, 0)$, como estamos utilizando a função de ligação Logito, temos que $\mu_i = \frac{\exp(x_i^\top \beta)}{1+\exp(x_i^\top \beta)}$. A partir de μ_i, geramos 1000 vezes a variável resposta considerando

$$y_i = Bernoulli(\mu_i). \tag{4.1}$$

Podemos ver que teremos 1000 bancos de dados simulados iguais, exceto pela variável resposta. Por mais que os parâmetros μ_i sejam os mesmos para todos as 1000 vezes, como estamos gerando aleatoriamente, os valores de y_i serão diferentes. Portanto, podemos ajustar modelos em cada um dos bancos dados e calcular as medidas de avaliação do poder preditivo.

As 1000 bases de dados simuladas serão divididas em duas partes proporcionalmente, sendo uma parte utilizada para treinamento (70% das observações) e a outra para teste (30% das observações).

Inicialmente, compararemos as medidas e avaliação do modelo em dois cenários diferentes, 1 e 2. Para entender melhor esses cenários, considere o organograma apresentado na Figura 4.1

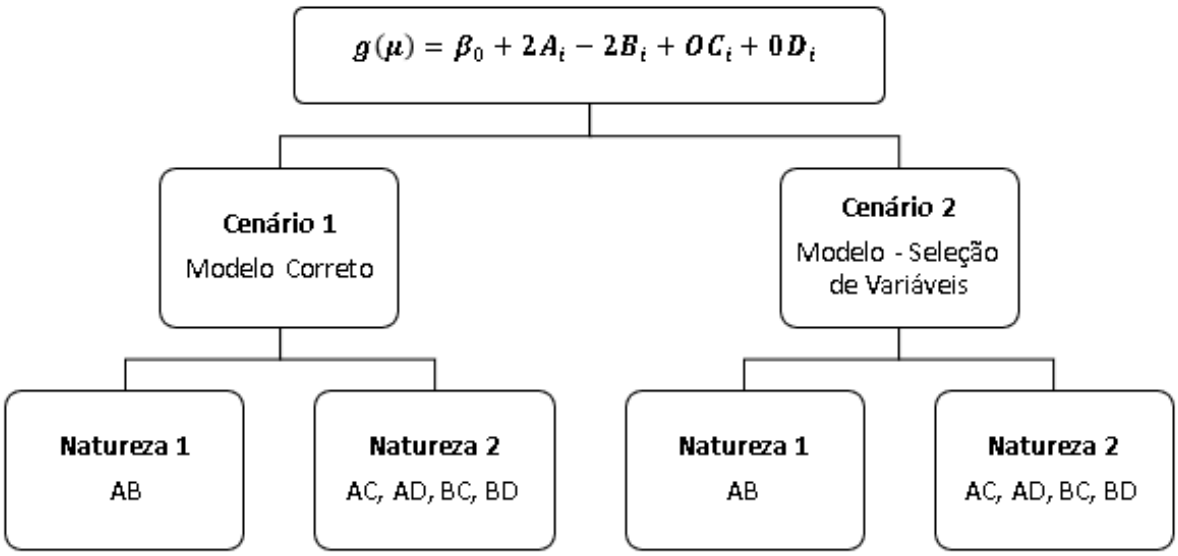


Figura 4.1: Organograma do funcionamento dos Cenários

Pela Figura 4.1, é possível ver que o cenário 1, possui modelos provenientes de duas naturezas, 1 e 2. A natureza 1 contém o modelo correto, ou seja, o modelo que contém somente as variáveis preditoras que apresentam valor do parâmetro diferente de zero associadas a elas na simulação. A outra é a natureza 2, que contém os modelos que chamaremos de modelos da substituição, na qual trocamos uma das variáveis do modelo de natureza 1 por uma que não foi adicionada ao mesmo. Portanto, considerando o modelo que se encontra no topo da Figura 4.1, na natureza 1 estará contido o modelo que contém as variáveis preditoras A e B e na natureza 2, estarão contidos os modelos AC (que contém as variáveis preditoras A e C), AD, BC e BD.

Já no cenário 2, a natureza 1 conterá o modelo selecionado por algum método

de seleção de variáveis e a outra, assim como no cenário 1, conterá os modelos da substituição.

A avaliação das medidas será feita inicialmente a partir do cálculo de algumas proporções. São elas:

- $p1$: a proporção de vezes que o melhor modelo, segundo as medidas de avaliação do poder preditivo nas bases de treinamento, está contido na natureza 1, analogamente nas bases de teste;

- $p2$: a proporção de vezes que o melhor modelo, segundo as medidas de avaliação do poder preditivo nas bases de treinamento, está contido na natureza 2, analogamente nas bases de teste;

- $p3$: a proporção de vezes que o modelo indicado como melhor pelas medidas de avaliação do poder preditivo foi o mesmo tanto nas bases de treinamento quanto nas de teste.

Para cada uma das 1000 bases de dados simuladas, espera-se que o modelo que está contido na natureza 1 apresente melhor poder preditivo do que os dos que estão contido na natureza 2, pois ele contém somente as variáveis preditoras com parâmetros associados diferente de zero. Portanto, é esperado que uma boa medida de avaliação do poder preditivo de modelos para dados binários apresente alto valor para $p1$. Também é desejável a observação de alto valor para $p3$, pois isso evidencia que em geral a medida considera o mesmo modelo como o melhor nas duas bases de dados.

4.1 Cenário 1

Nesta Seção apresentamos as proporções que serão utilizadas para avaliar o poder preditivo de modelos com resposta binária considerando o cenário 1. Também serão apresentados uma série de *boxplots* que relacionam as diferenças entre as naturezas com o valor (médio) das medidas. Para cada uma das bases simuladas temos que, a natureza 1 é composta por um único valor. Já a natureza 2, pode conter mais do que 1. Devido a isso, utilizamos a média dos valores contidos na natureza 2.

A seguir, apresentamos a Tabela 4.1 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando o modelo correto, ou seja, conside-

rando o cenário 1. Note que consideramos nos estudos de simulação três diferentes vetores β.

Tabela 4.1: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando a comparação com o modelo correto.

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,723	0,277	0,400	0,994	0,006	0,842	0,998	0,002	0,963
	Teste	0,500	0,500		0,846	0,154		0,963	0,037	
KS	**Treino**	0,523	0,477	0,271	0,895	0,105	0,618	0,985	0,015	0,858
	Teste	0,388	0,612		0,686	0,314		0,870	0,130	
H	**Treino**	0,617	0,383	0,332	0,987	0,013	0,794	0,999	0,001	0,958
	Teste	0,471	0,529		0,804	0,196		0,958	0,042	

Pela Tabela 4.1 podemos verificar que o coeficiente de Gini e a medida H, possuem as proporções $p1$ e $p3$ consideravelmente maiores que a estatística KS. Seja $\beta = (1, \ell, -\ell, 0, 0)$, quando $\ell = 1$, o coeficiente de Gini apresenta proporções $p1$ e $p3$ significativamente maiores que a medida H também. Já para $\ell = 2$, as proporções continuam sendo maiores, só que sutilmente. No caso em que $\ell = 3$, essas duas medidas apresentam valores semelhantes. Essas conclusões valem tanto para as bases de treinamento quanto para as de teste.

É interessante destacar que, conforme esperado, a medida que aumentamos o valor de ℓ, as proporções $p1$ e $p3$ aumentam. Isso já era esperado pois quanto mais distante de zero for os parâmetros associados as variáveis preditoras do modelo, melhor é o desempenho do mesmo.

Temos evidências então, que quando ℓ é próximo de 1 o coeficiente de Gini apresenta um melhor desempenho, já quando ℓ for maior ou igual a 2, a medida H acaba se tornando uma boa opção, assim como o coeficiente de Gini.

4.1.1 Boxplots

A seguir, apresentamos *boxplots* representando as diferenças entre os valores das naturezas 1 e 2 de todas as bases simuladas. É esperado que esses *boxplots* estejam concentrados acima de 0, pois isso evidencia que os valores das medidas estudadas que os modelos contidos na natureza 1 assumem são maiores que os contidos na natureza 2.

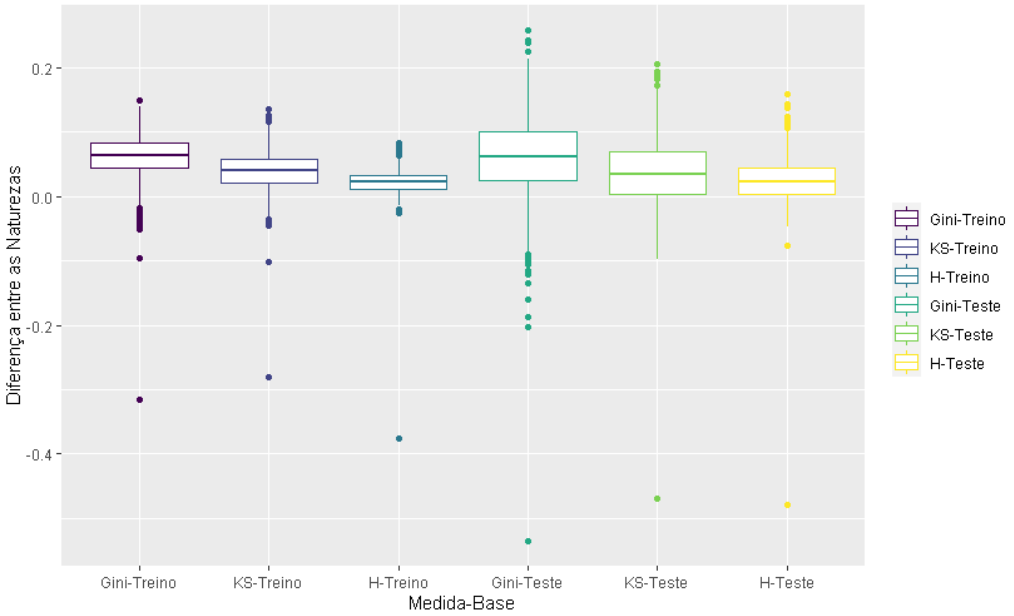


Figura 4.2: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

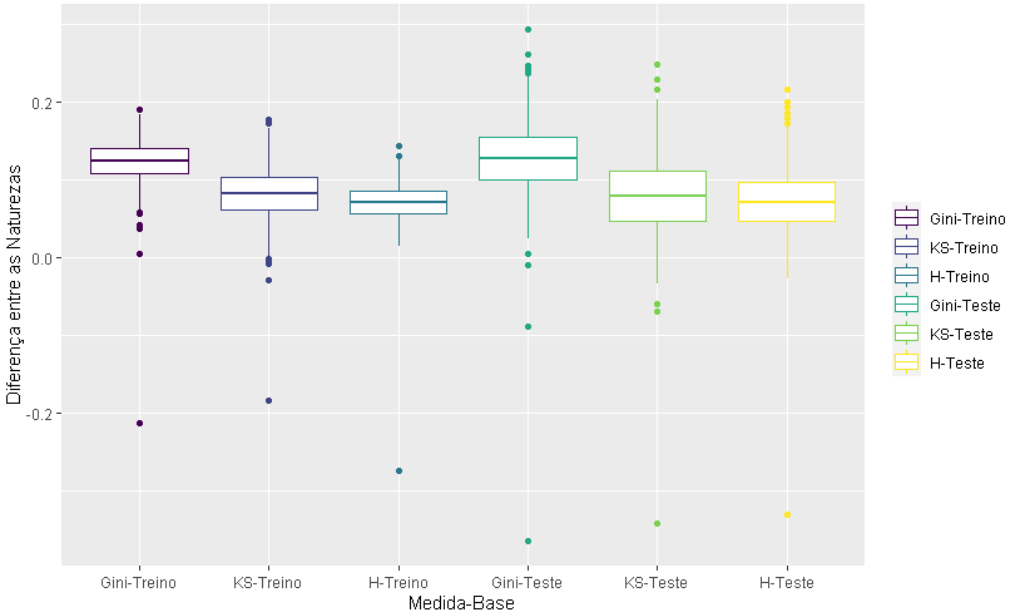


Figura 4.3: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

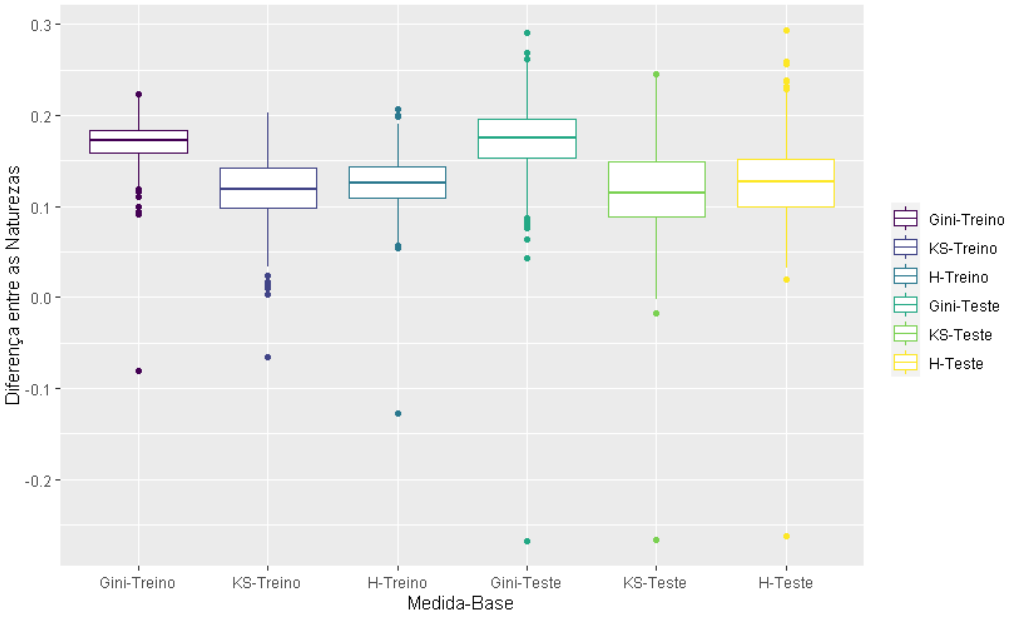


Figura 4.4: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

Analisando conjuntamente as Figuras 4.2, 4.3 e 4.4 temos que, na maioria dos casos, as diferenças entre as naturezas são positivas. Portanto, os valores das medidas estudadas que os modelos corretos assumem são maiores que os que encontramos nas médias das medidas assumidas pelos modelos da substituição em quase todas as bases simuladas.

O coeficiente de Gini parece assumir maiores diferenças entre as naturezas, nos dando indícios de que ela seja a medida que tem o melhor desempenho. À medida que ℓ aumenta, as conclusões anteriores se tornam mais evidentes.

Os *boxplots* que representam as bases de teste apresentam maiores variabilidades do que os das bases de treino. Isso já era esperado, pois os modelos utilizados não foram treinados utilizando essas bases.

Há um número considerável de *outliers* nas três figuras. Embora, a mediana das diferenças sejam positivas para todos os valores de ℓ, medidas e bases de dados e, na maioria dos casos, o primeiro quartil também ser positivo, para todos os valores de ℓ, medidas e bases de dados, há *outliers* bem inferiores a zero.

4.2 Cenário 2

Nesta Seção apresentamos as proporções que serão utilizadas para avaliar o poder preditivo de modelos com resposta binária considerando o cenário 2. Também serão

apresentados uma série de *boxplots* que relacionam as diferenças entre as naturezas com os valores (médios) das medidas.

4.2.1 Backward

Abaixo, apresentamos a Tabela 4.2 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando o método de seleção de variáveis Backward.

Tabela 4.2: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando o método de seleção de variáveis Backward.

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,975	0,025	0,402	0,986	0,014	0,693	0,988	0,012	0,799
	Teste	0,399	0,601		0,692	0,308		0,795	0,205	
KS	**Treino**	0,705	0,295	0,309	0,844	0,156	0,548	0,883	0,117	0,759
	Teste	0,361	0,639		0,589	0,411		0,740	0,261	
H	Treino	0,821	0,179	0,361	0,938	0,062	0,684	0,943	0,057	0,797
	Teste	0,390	0,610		0,668	0,332		0,739	0,212	

A análise da Tabela 4.2 é análoga à que encontramos na Tabela 4.1. Pode-se notar que, nesse caso, quando utilizamos o método de seleção de variáveis Backward, todas as proporções $p1$ e $p3$ referentes ao coeficiente de Gini são maiores que as demais medidas. Portanto, o coeficiente de Gini considera de forma mais frequente o modelo selecionado pelo método Backward como o modelo de maior poder preditivo do que o KS e a medida H. E, de modo geral, a medida que ℓ cresce, as proporções $p1$ e $p3$ aumentam e consequentemente, a de $p2$ diminui.

Boxplots

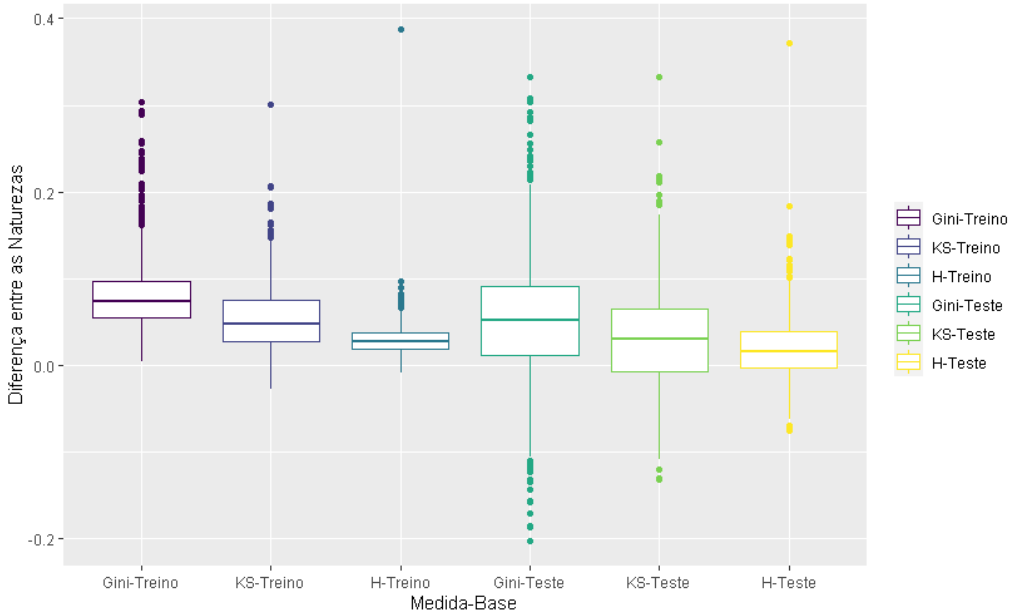


Figura 4.5: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

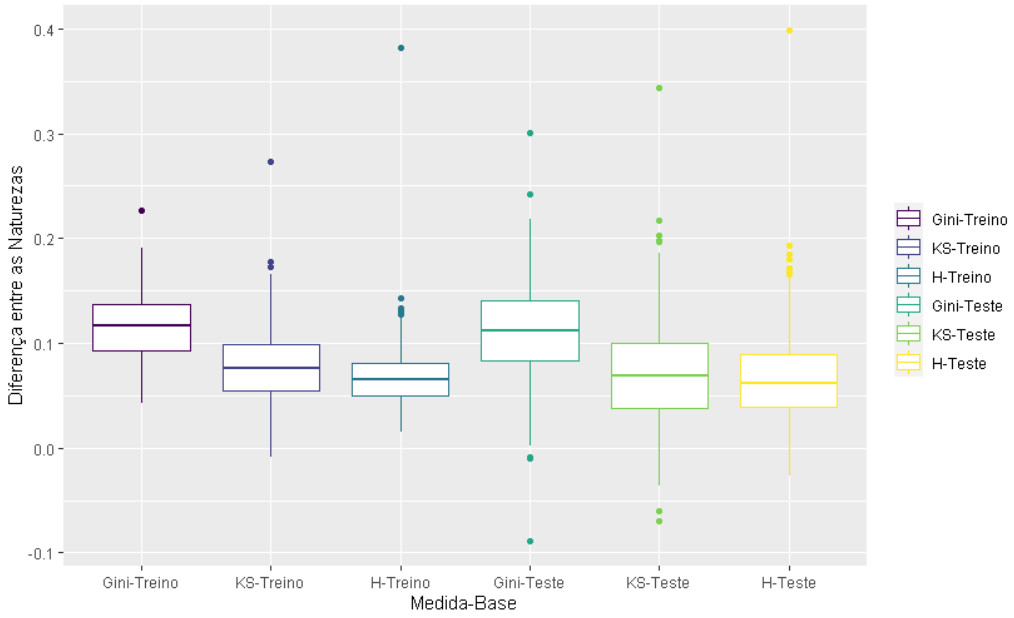


Figura 4.6: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

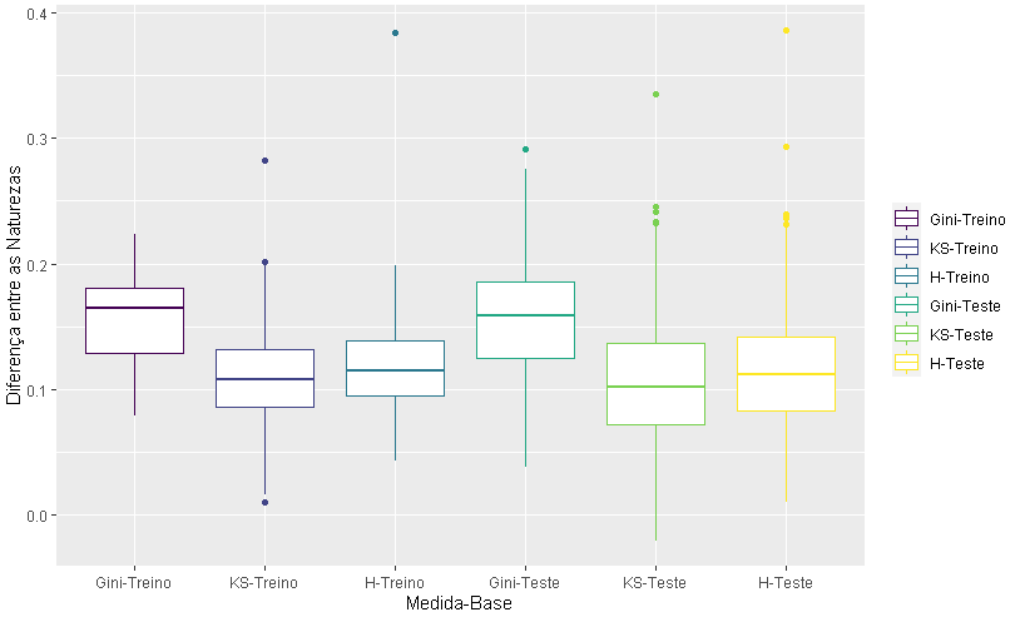


Figura 4.7: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

Pelas Figuras 4.5, 4.6 e 4.7, observamos que os *boxplots* estão alocados, em sua maioria, acima do valor 0. Dessa forma, os modelos selecionados pelo método Backward em cada uma das bases simuladas possuem medidas com valores mais altos do que as médias dos modelos da substituição.

O coeficiente de Gini parece ter o melhor desempenho para avaliar o poder preditivo de modelos para dados binários, pois apresenta as maiores diferenças entre as naturezas.

Há um número considerável de *outliers* nas três figuras. À medida que ℓ aumenta, esse número diminui.

4.2.2 Lasso

A seguir, apresentamos a Tabela 4.3 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando o método de seleção de variáveis Lasso.

Tabela 4.3: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando o método de seleção de variáveis Lasso.

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,970	0,030	0,448	0,986	0,014	0,828	0,996	0,004	0,957
	Teste	0,450	0,550		0,829	0,171		0,960	0,040	
KS	**Treino**	0,804	0,196	0,391	0,911	0,089	0,646	0,976	0,024	0,839
	Teste	0,414	0,586		0,684	0,316		0,854	0,146	
H	**Treino**	0,867	0,133	0,416	0,977	0,023	0,788	0,997	0,003	0,945
	Teste	0,440	0,560		0,795	0,205		0,945	0,055	

Analisando a Tabela 4.3 temos que, quando ℓ é menor ou igual a 2, o coeficiente de Gini apresenta as melhores proporções $p1$ e $p3$. Quando $\ell = 3$, as proporções relacionadas a medida H se tornam muito próximas as associadas ao coeficiente de Gini.

Portanto, a medida que ℓ aumenta, a medida H também se torna uma boa escolha de medida para avaliar o poder preditivo de modelos para dados binários, mas no geral, o coeficiente de Gini continua sendo a melhor escolha.

Para efeito de informação adicional, de modo geral, as proporções encontradas pela medida Lasso foram maiores que as obtidas utilizando o modelo correto ou o Backward. Isso era esperado porque o Lasso é uma método de seleção de variáveis que busca identificar o modelo com maior poder preditivo.

Boxplots

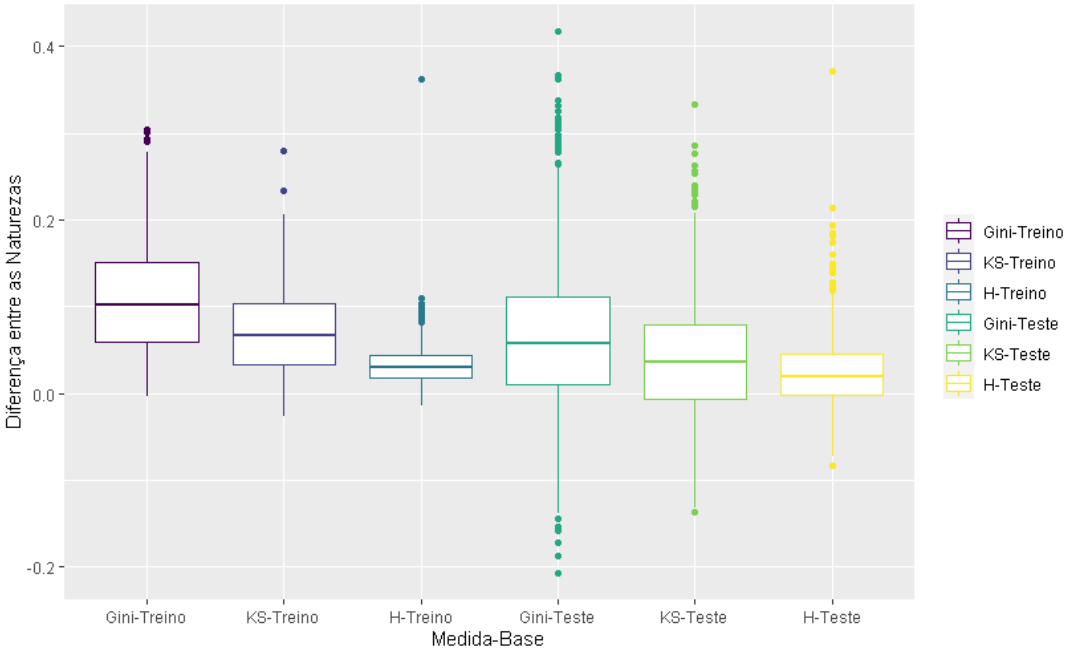


Figura 4.8: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

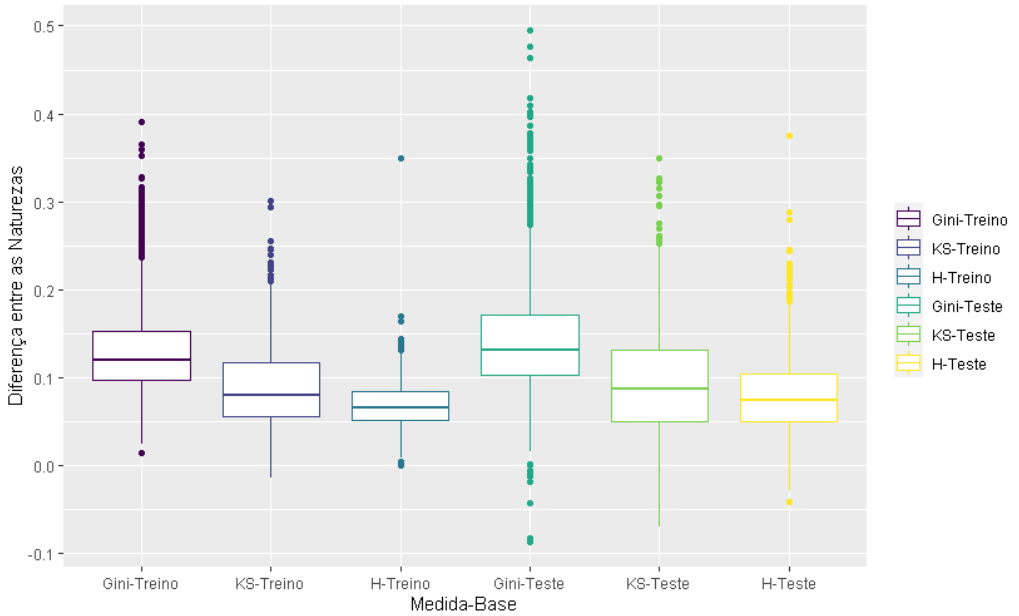


Figura 4.9: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

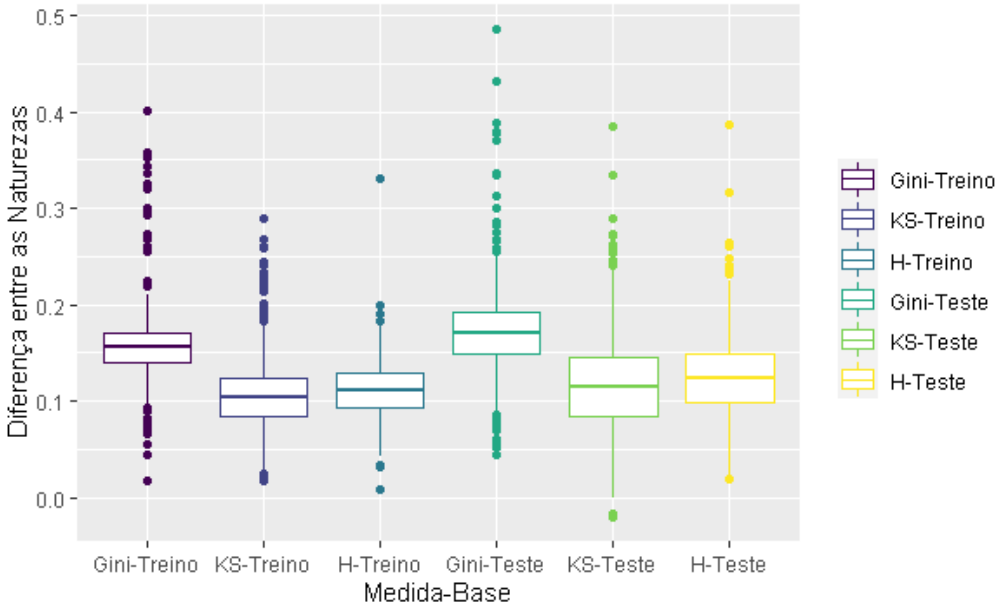


Figura 4.10: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

As conclusões para as Figuras 4.8, 4.9 e 4.10 são similares as de quando estamos usando o método de seleção de variáveis Backward. Assim sendo, pelas figuras vemos que o coeficiente de Gini parece ser a medida que apresenta maiores diferenças entre as naturezas, tanto para as bases de treino quanto para as de teste, ou seja, aparenta ser a medida com o melhor desempenho.

Os *boxplots* acima, na qual as variáveis dos modelos contidos na natureza 1 foram selecionadas pelo Lasso, apresentam maiores variabilidades do que os que utilizamos o modelo correto ou Backward. Eles também apresentam maior presença de *outliers* do que os demais.

4.3 Novos Cenários

Para verificar se as conclusões discutidas na Seção 4.1 valem quando alteramos algum aspecto do modelo, outros cenários de simulação foram considerados. Os resultados desses novos cenários, serão avaliados somente utilizando o modelo correto, e não mais utilizando métodos de seleção de variáveis. Como nos estudos de simulação sabemos qual é o modelo correto (ao contrário do que ocorre quando trabalhamos com dados reais), essa análise é mais importante do que a realizada considerando métodos de seleção de variáveis.

4.3.1 Proporções considerando a função de ligação Probito

A seguir, apresentamos a Tabela 4.4 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando a função de ligação Probito, considerando três diferentes vetores β.

Tabela 4.4: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando a função de ligação Probito.

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,958	0,042	0,718	0,999	0,001	0,975	0,999	0,001	0,997
	Teste	0,746	0,254		0,975	0,025		0,997	0,003	
KS	**Treino**	0,807	0,193	0,483	0,994	0,06	0,907	0,998	0,002	0,982
	Teste	0,584	0,416		0,911	0,089		0,983	0,017	
H	**Treino**	0,837	0,163	0,533	0,998	0,002	0,936	0,999	0,001	0,993
	Teste	0,618	0,382		0,937	0,063		0,993	0,007	

Pela Tabela 4.4, é possível verificar que, considerando $\beta = (1, \ell, -\ell, 0, 0)$, quando $\ell = 1$, o coeficiente de Gini apresenta proporções $p1$ e $p3$ significativamente maiores que a medida H e estatística KS. No caso em que $\ell = 3$, as proporções referentes ao coeficiente de Gini e a medida H se tornam muito próximas com o KS apresentando proporções um pouco menores. Essas conclusões são as mesmas que as encontradas na Tabela 4.1, ou seja, quando mudamos a função de ligação Logito para Probito, as conclusões não se alteram. Contudo, as proporções $p1$ e $p2$ referentes as três medidas, principalmente quando $\ell = 1$, tornam-se consideravelmente maiores, tanto para as bases de treinamento quanto para as de validação.

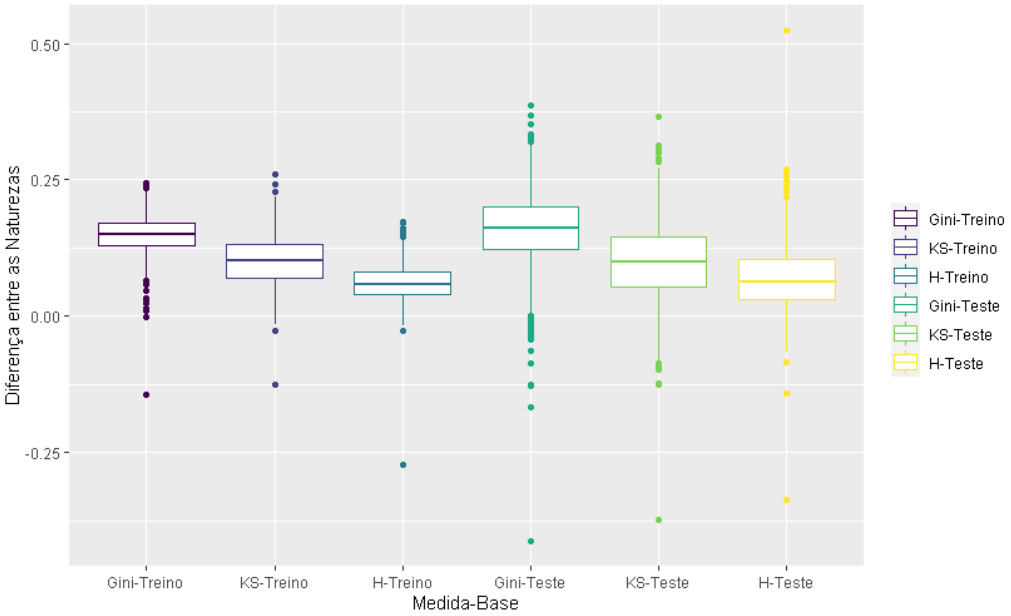


Figura 4.11: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

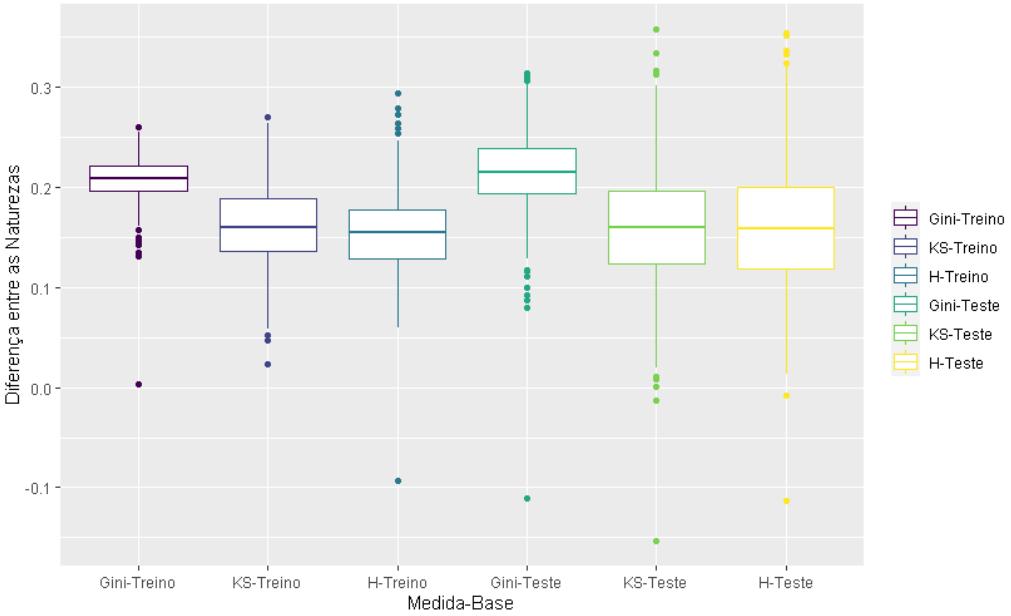


Figura 4.12: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

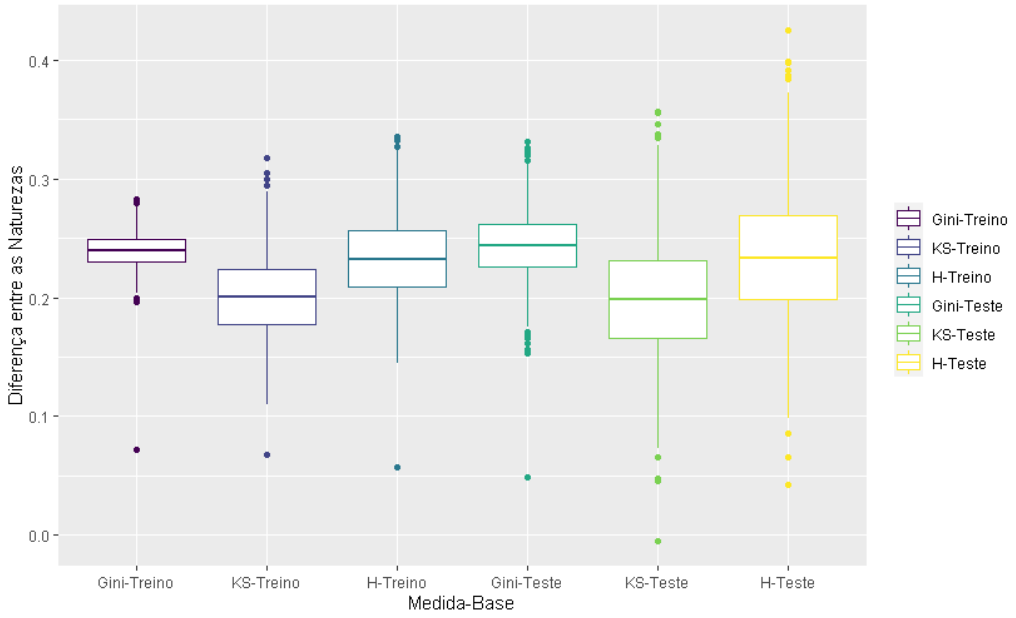


Figura 4.13: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

Analisando conjuntamente as Figuras 4.11, 4.12 e 4.13 temos que, na maioria dos casos, as diferenças entre as naturezas são positivas. Portanto, os valores das medidas estudadas que os modelos corretos assumem são maiores que os que encontramos nas médias das medidas assumidas pelos modelos da substituição em quase todas as bases simuladas. A medida que o valor de ℓ aumenta, o número de vezes que a diferença é negativa fica cada vez mais próximo de zero.

A medida que aparenta possuir as maiores diferenças entre as naturezas é o coeficiente de Gini, tanto para as bases de treinamento quanto para as de validação. Isso nos dá indícios de que ela seja a medida que tem o melhor desempenho. À medida que ℓ aumenta, as conclusões anteriores se tornam mais evidentes.

Vemos que, no geral, os *boxplots* apresentados anteriormente apresentam maiores variabilidades do que os encontrados na Subseção 4.1.1, que se referem a quando estávamos utilizando a função de ligação Logito. Quando $\ell = 3$, as diferenças medianas entre as naturezas referentes ao coeficiente de Gini e medida H são muito próximas, tanto para as bases de treinamento quanto para as de validação, diferente do que acontece quando estamos utilizando a função de ligação Logito, na qual o coeficiente de Gini apresenta medianas visivelmente maiores que a medida H.

4.3.2 Proporções considerando a função de ligação Complemento Log-Log

A seguir, apresentamos a Tabela 4.5 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando a função de ligação Complemento Log-Log, considerando três diferentes vetores β.

Tabela 4.5: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando a função de ligação Complemento Log-Log.

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,956	0,044	0,714	0,999	0,001	0,978	0,999	0,001	0,997
	Teste	0,744	0,256		0,978	0,022		0,997	0,003	
KS	**Treino**	0,807	0,193	0,476	0,996	0,004	0,906	0,998	0,002	0,981
	Teste	0,575	0,425		0,909	0,091		0,982	0,018	
H	**Treino**	0,830	0,170	0,533	0,998	0,002	0,939	0,999	0,001	0,993
	Teste	0,622	0,378		0,940	0,060		0,993	0,007	

A análise da Tabela 4.5 é análoga à que encontramos na Tabela 4.4. Portanto, temos evidências de que o coeficiente de Gini parece ser a medida que apresenta a melhor performance na avaliação do poder preditivo de modelos com variável resposta binária. Por consequência, essas conclusões são as mesmas que as encontradas na Tabela 4.1, ou seja, para quando usamos a função de ligação Logito. No entanto, as proporções $p1$ e $p3$ referentes as três medidas, principalmente quando $\ell = 1$, tornam-se consideravelmente maiores, tanto para as bases de treinamento quanto para as de validação.

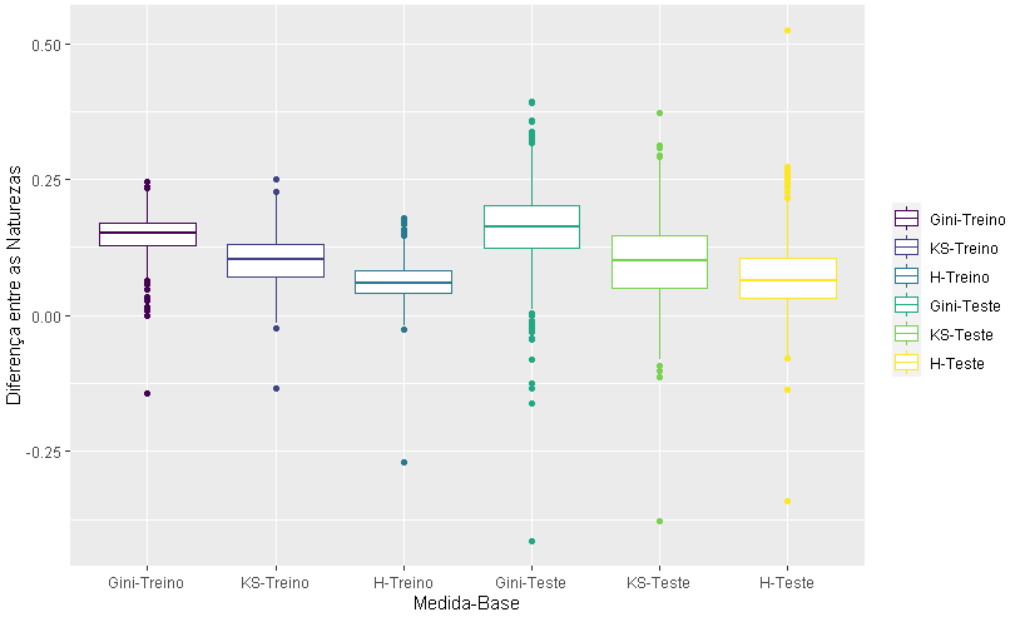


Figura 4.14: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

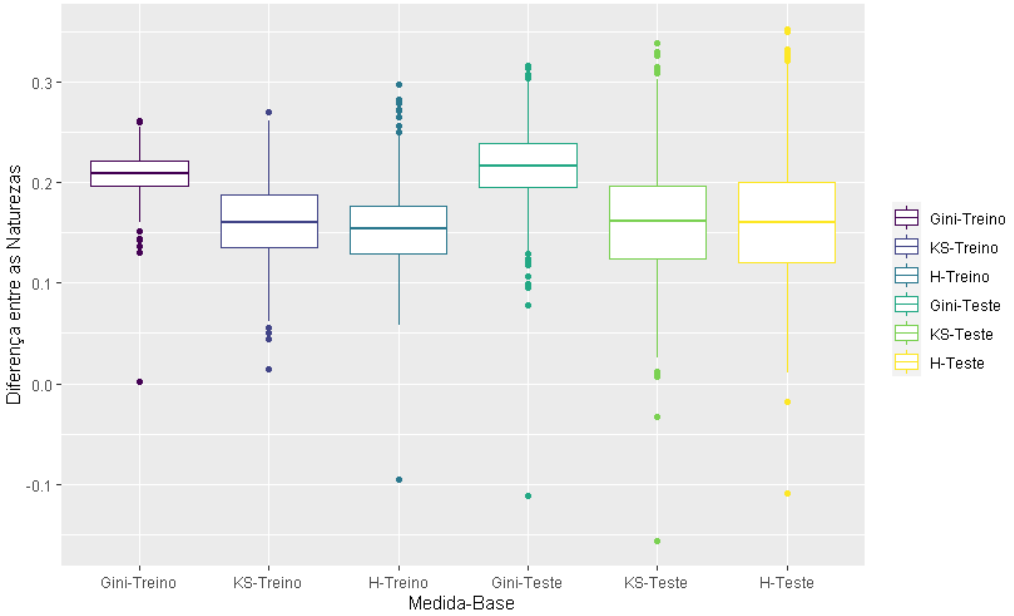


Figura 4.15: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

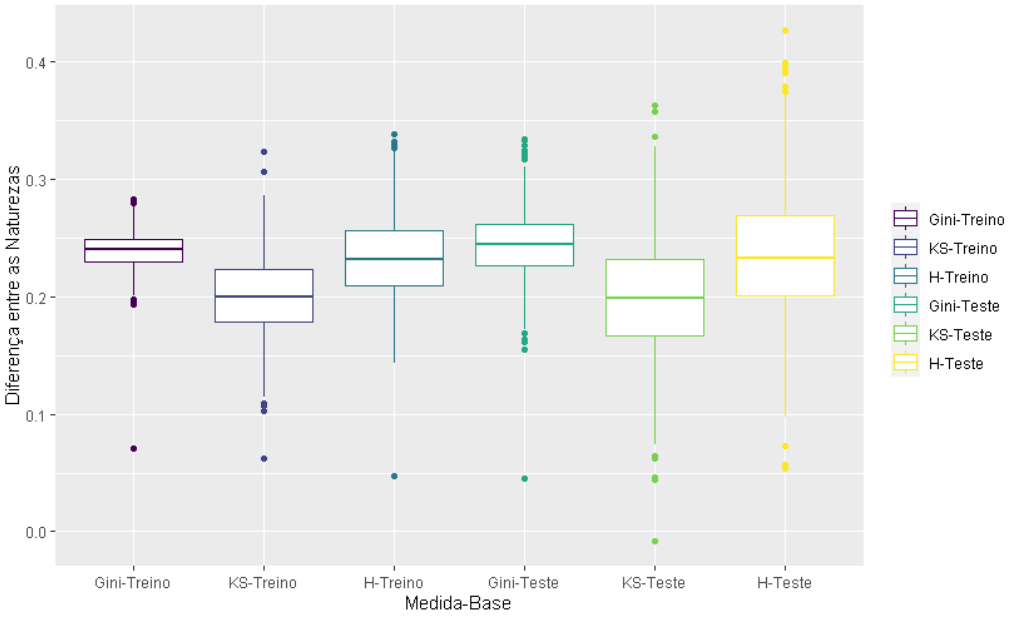


Figura 4.16: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

As conclusões e comportamentos das Figuras 4.14, 4.15 e 4.16 são similares às encontradas na Subseção 4.3.1, ou seja, majoritariamente, as diferenças entre as naturezas são positivas e essas diferenças são mais expressivas para o coeficiente de Gini, o que evidencia que ela parece ser medida com a melhor performance.

4.3.3 Proporções considerando variáveis preditoras com distribuições diferentes

A seguir, apresentamos a Tabela 4.6 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, onde duas variáveis preditoras foram geradas de uma distribuição Normal e as outras duas de uma distribuição Gama. Consideramos valores dos parâmetros para essas distribuições de forma que elas apresentem mesma média e variância populacionais que a distribuição Uniforme utilizada nas simulações anteriores, ou seja, média 1/2 e variância 1/12.

Tabela 4.6: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando variáveis preditoras com distribuições diferentes

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,957	0,043	0,726	1,000	0,000	0,966	1,000	0,000	0,994
	Teste	0,758	0,242		0,966	0,034		0,994	0,006	
KS	**Treino**	0,793	0,207	0,479	0,990	0,010	0,868	1,000	0,000	0,977
	Teste	0,590	0,410		0,875	0,125		0,977	0,023	
H	Treino	0,864	0,136	0,554	0,999	0,001	0,948	1,000	0,000	0,990
	Teste	0,647	0,353		0,949	0,051		0,999	0,010	

Analisando a Tabela 4.6 temos que, quando ℓ é igual a 1, o coeficiente de Gini apresenta as melhores proporções $p1$ e $p3$. Quando ℓ é maior ou igual a 2, as proporções relacionadas as outras duas medidas se tornam muito próximas as associadas ao coeficiente de Gini, tanto nas bases de treinamento quanto às de validação.

Deste modo, ao passo que ℓ aumenta, todas as medidas se tornam boas escolhas para avaliar o poder preditivo de modelos para dados binários, mas no geral, o coeficiente de Gini continua sendo a melhor escolha.

Nota-se que, quando passamos a simular os dados através das distribuições Normal e Gama ao invés da Uniforme (cujo os resultados são apresentados na Tabela 4.1), as proporções $p1$ e $p3$ tornam-se consideravelmente maiores para todas as medidas, principalmente quando ℓ é igual a 1. Contudo, as conclusões são as mesmas.

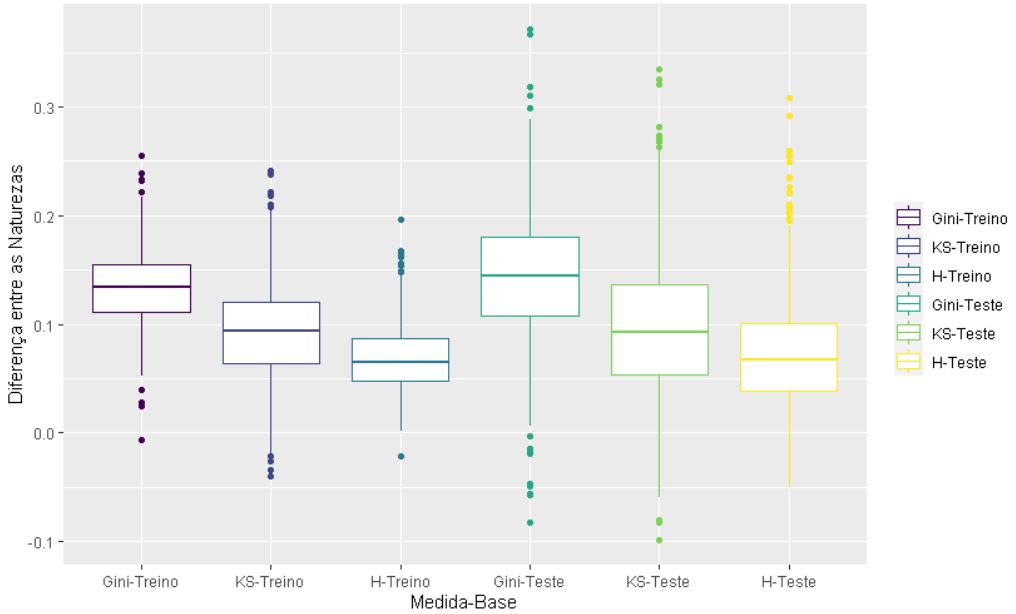


Figura 4.17: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

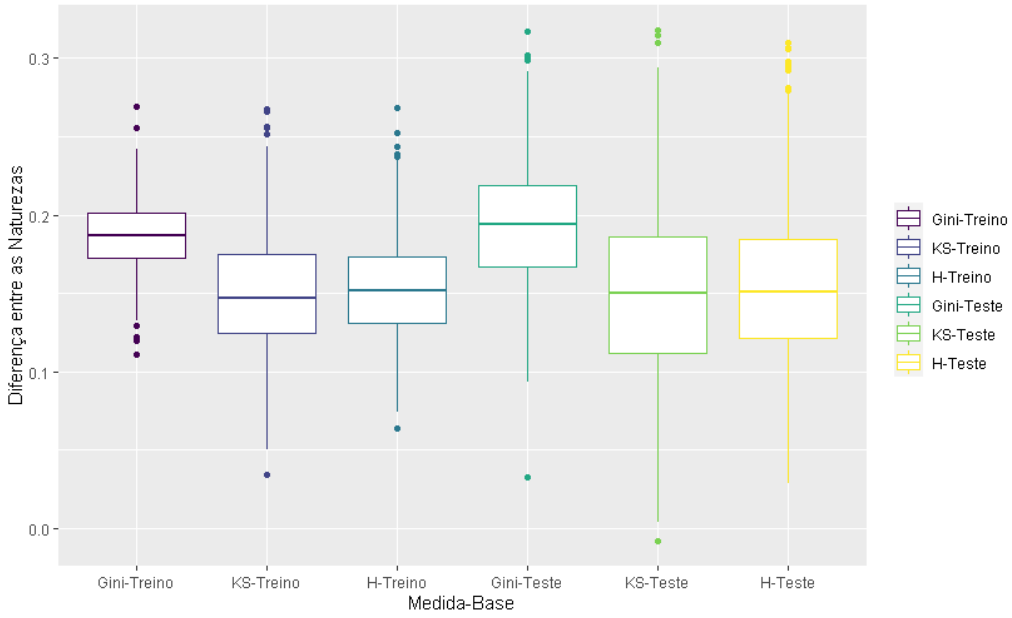


Figura 4.18: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

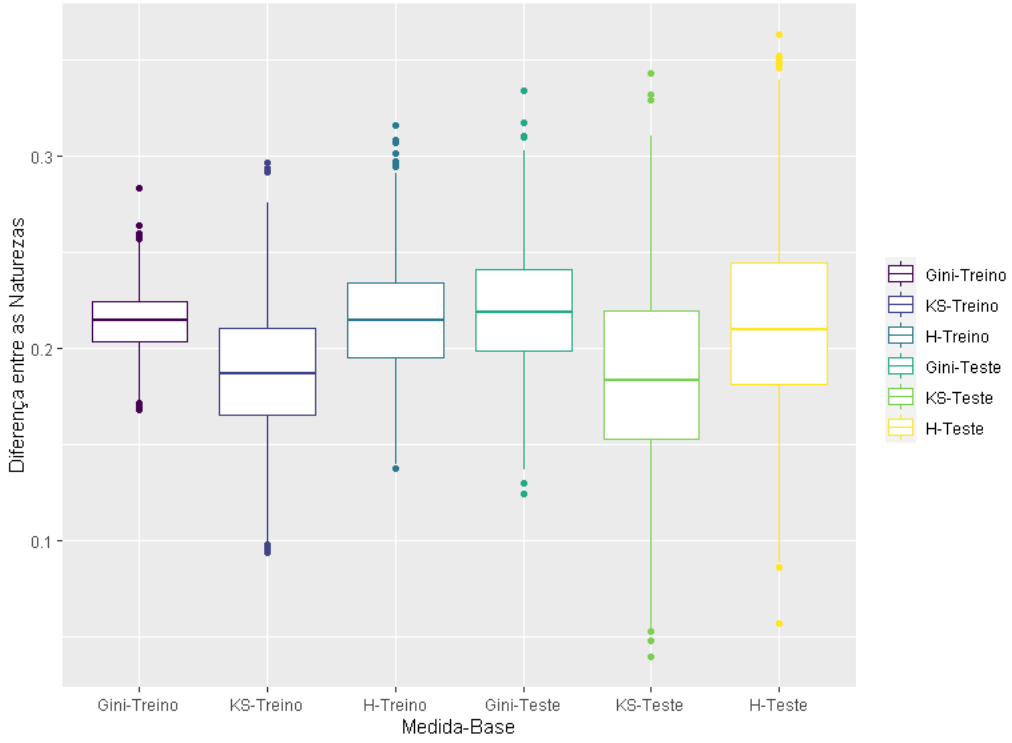


Figura 4.19: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

Pelas Figuras 4.17, 4.18 e 4.19, observamos que os *boxplots* estão alocados, em sua maioria, acima do valor 0. Isso fica mais evidente conforme aumenta o valor de ℓ. Dessa forma, os modelos selecionados pelo modelo correto em cada uma das bases simuladas possui medidas com valores mais altos do que as médias dos modelos da substituição. Para as duas primeiras Figuras, o coeficiente de Gini, parece ser a medida que

apresenta as diferenças mais notáveis, dando fortes indícios de que ela possui a melhor performance na avaliação do poder predito de modelos logísticos. Já para última, nota-se que as medianas das diferenças entre naturezas 1 e 2 relacionadas ao coeficiente de Gini e medida H são muito parecidas, o que nos dá evidências que para esse caso, ambas as medidas parecem ter a mesma performance.

4.3.4 Proporções considerando variáveis preditoras correlacionadas

Nesse cenário, usamos os coeficientes de correlação linear de Pearson apresentados na Tabela 4.7 para gerar as variáveis preditoras com o auxílio do pacote *MultiRNG: Multivariate Pseudo-Random Number Generation* (Demirtas *et al.*, 2020) do *software R*. Desse modo, definimos três pares correlacionados, sendo o primeiro formado pelas variáveis preditoras 1 e 2. O segundo composto pelas covariáveis 1 e 3; e o último pelas covariáveis 2 e 3. As proporções são apresentadas na Tabela 4.8

Tabela 4.7: Coeficientes de correlação linear de Pearson utilizados para gerar as variáveis preditoras.

Variáveis	V_1	V_2	V_3	V_4
V_1	1,0			
V_2	**0,7**	1,0		
V_3	**0,9**	**0,7**	1,0	
V_4	0,1	0,1	0,1	1,0

Tabela 4.8: Proporções encontradas a partir dos 1000 bancos de dados simulados, utilizando variáveis preditoras correlacionadas.

Medida	**Base**	**Proporções**								
		$\beta = (1, 1, -1, 0, 0)$			$\beta = (1, 2, -2, 0, 0)$			$\beta = (1, 3, -3, 0, 0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,443	0,557	0,222	0,796	0,204	0,479	0,916	0,084	0,697
	Teste	0,305	0,695		0,560	0,440		0,752	0,248	
KS	**Treino**	0,353	0,647	0,217	0,653	0,347	0,390	0,798	0,202	0,521
	Teste	0,271	0,729		0,468	0,532		0,622	0,378	
H	Treino	0,397	0,603	0,202	0,755	0,245	0,452	0,898	0,102	0,667
	Teste	0,273	0,727		0,537	0,463		0,721	0,279	

Pela Tabela 4.8, podemos constatar que o coeficiente de Gini e a medida H, possuem proporções $p1$ e $p3$ maiores que a estatística KS. Porém, independente do valor de ℓ, o coeficiente de Gini ainda possui $p1$ e $p3$ maiores que a medida H também, dando

indícios que a medida que possui a melhor performance para avaliar o poder preditivo de modelos com variáveis respostas binárias é o coeficiente de Gini.

Note que este é o cenário que apresentou as proporções p_1 e p_3 mais baixas dentre as apresentadas até o momento. Isso é coerente com o que era esperado, pois a multicolinearidade dificulta o ajuste de modelos de regressão.

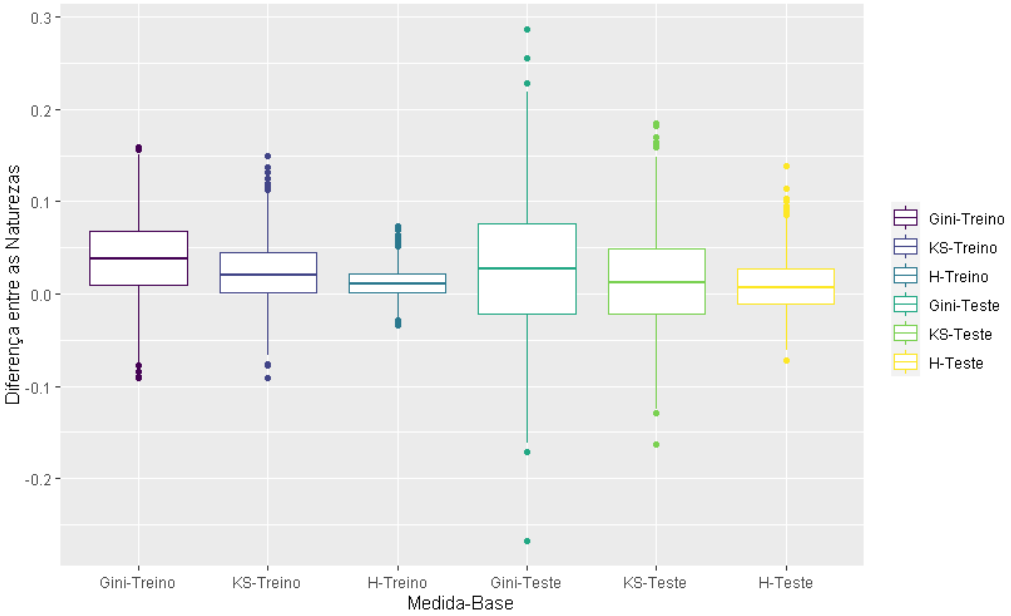


Figura 4.20: Boxplot considerando $\beta = (1, 1, -1, 0, 0)$.

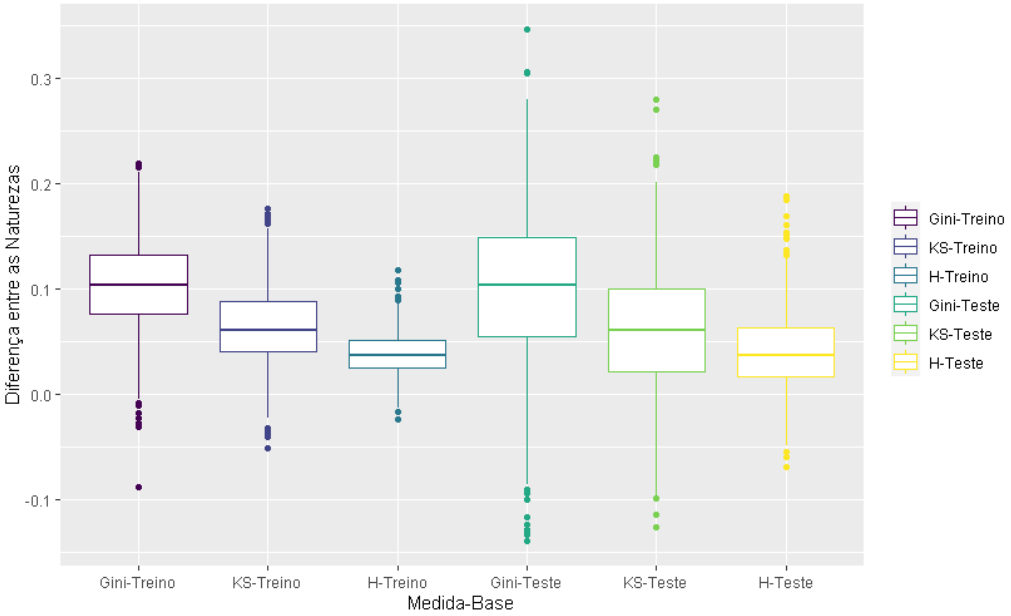


Figura 4.21: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

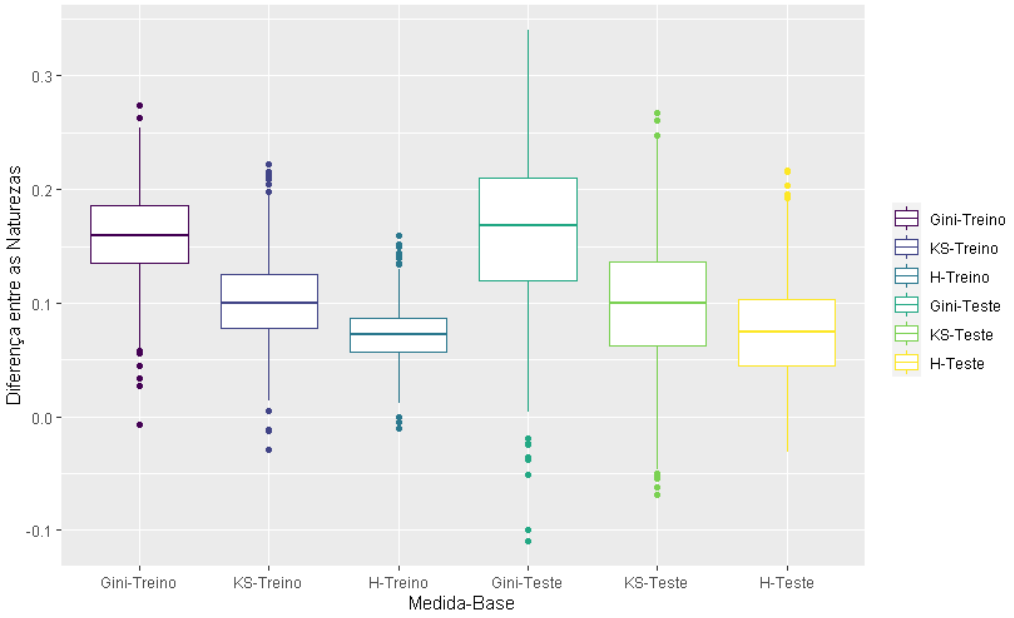


Figura 4.22: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

Pelas Figuras 4.20, 4.21 e 4.22, podemos concluir que o coeficiente de Gini parece ser a medida que apresenta o melhor desempenho na avaliação do poder preditivo dos modelos logísticos, pois as caixas dos *boxplots* se encontram, quase que unanimemente, acima de 0.

Também vemos que a medida que parece apresentar maiores diferenças entre as naturezas é o coeficiente de Gini, tanto para as bases de treinamento quanto para as de validação, independentemente do ℓ utilizado.

É possível perceber que, a estatística KS apresenta maiores diferenças entre as naturezas do que a medida H, independentemente de qual ℓ foi considerado. Isso difere do que a maioria dos cenários anteriores apresentaram, que no caso, para ℓ iguais a 2 e 3, a medida H apresentava diferenças entre as naturezas maiores.

4.3.5 Proporções considerando aumento na quantidade de covariáveis

A seguir, apresentamos a Tabela 4.9 contendo as proporções encontradas a partir dos 1000 bancos de dados simulados, onde foi gerado 8 variáveis preditoras, considerando $\beta = (1, \ell, \ell, -\ell, -\ell, 0, 0, 0, 0)$.

Tabela 4.9: Proporções encontradas a partir dos 1000 bancos de dados simulados, aumentando a quantidade de variáveis preditoras

Medida	**Base**	**Proporções**								
		$\beta = (1,1,1,-1,-1,0,0,0,0)$			$\beta = (1,2,2,-2,-2,0,0,0,0)$			$\beta = (1,3,3,-3,-3,0,0,0,0)$		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,380	0,620	0,104	0,961	0,039	0,568	0,999	0,001	0,861
	Teste	0,167	0,833		0,584	0,416		0,862	0,138	
KS	**Treino**	0,170	0,830	0,056	0,541	0,459	0,202	0,859	0,141	0,492
	Teste	0,118	0,882		0,318	0,682		0,575	0,425	
H	Treino	0,286	0,714	0,077	0,884	0,116	0,430	0,998	0,002	0,783
	Teste	0,139	0,861		0,473	0,527		0,785	0,215	

Considerando a Tabela 4.9 é possível verificar que quando $\ell = 1$, as proporções $p1$ e $p3$ são bem mais baixas do que quando havia apenas quatro variáveis preditoras. Isso era esperado porque, quanto maior for o número de variáveis preditoras, maior a quantidade de modelos existente na natureza 2, consequentemente, maior é a chance do melhor modelo estar contido na mesma. À medida que ℓ aumenta, as proporções vão melhorando. Independentemente do ℓ utilizado, as proporções $p1$ e $p3$ apresentadas pelo coeficiente de Gini e pela medida H, são consideravelmente maiores que as apresentadas pela estatística KS. Contudo, o coeficiente de Gini ainda possui um desempenho melhor que o da medida H.

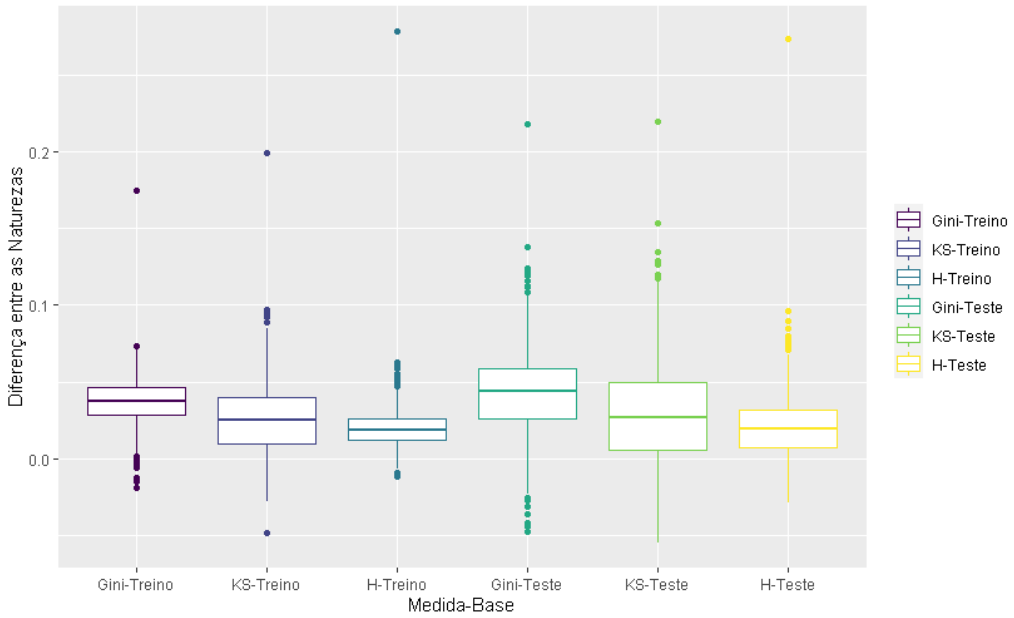


Figura 4.23: Boxplot considerando $\beta = (1,1,-1,0,0)$.

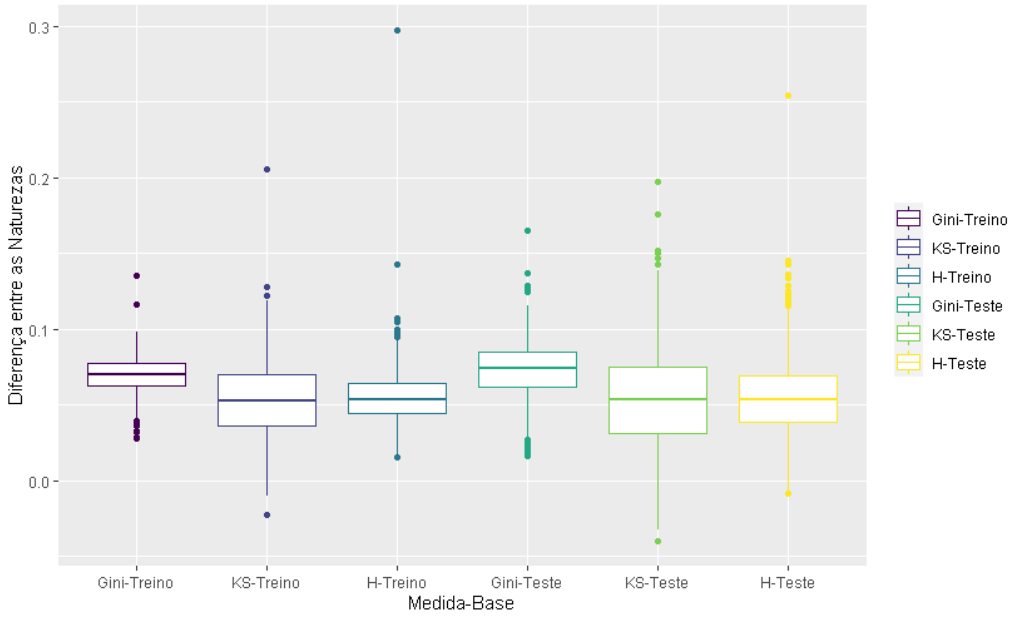


Figura 4.24: Boxplot considerando $\beta = (1, 2, -2, 0, 0)$.

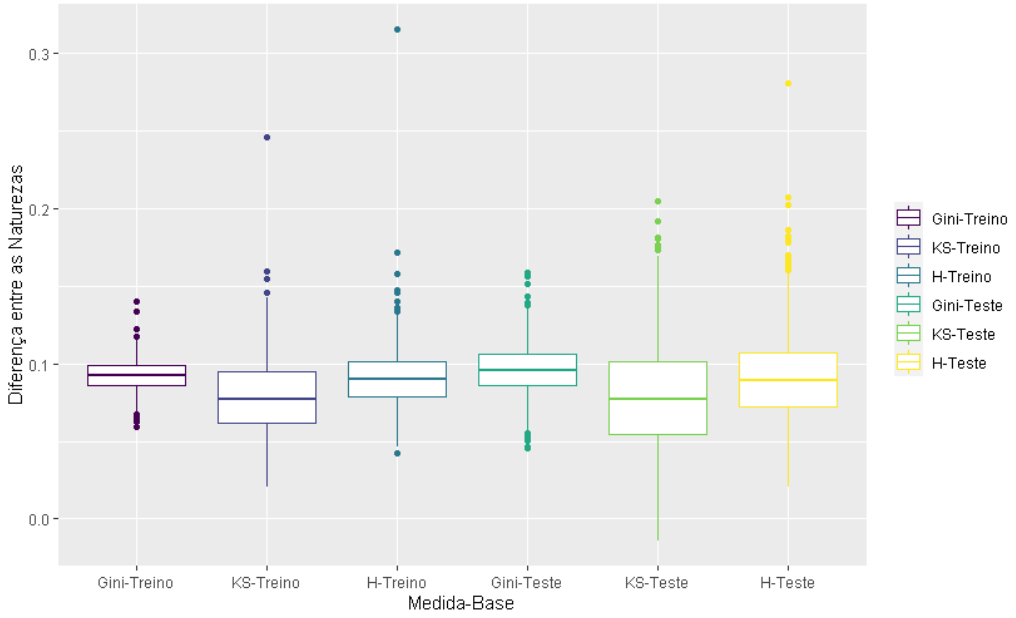


Figura 4.25: Boxplot considerando $\beta = (1, 3, -3, 0, 0)$.

As conclusões para as Figuras 4.23, 4.24 e 4.25 são análogas as que encontramos na Subseção 4.1.1, exceto pela escala do eixo Y. Desse modo, pelas figuras vemos que o coeficiente de Gini parece ser a medida que apresenta maiores diferenças entre as naturezas, tanto para as bases de treino quanto para as de teste, ou seja, aparenta ser a medida com o melhor desempenho.

4.4 Resumo dos resultados

No decorrer deste capítulo, pelas tabelas contendo as proporções encontradas a partir dos 1000 bancos de dados simulados e também pelos *boxplots* das diferenças entre as naturezas foi possível verificar que o coeficiente de Gini é a medida que apresenta o melhor desempenho para avaliar o poder preditivo de modelos para dados binários. A medida H possui o segundo melhor desempenho. Na maioria dos *boxplots* apresentados neste capítulo, é notável que existe um número expressivo de *outliers*. Isso sugere que as diferenças entre as medidas nas naturezas 1 e 2 apresentam uma variabilidade considerável, mesmo que os intervalos interquartis não sejam tão grandes.

Capítulo 5

Aplicação

Neste capítulo comparamos as medidas apresentadas neste trabalho em dois bancos de dados reais, chamados Haberman e Pima.te.

5.1 Haberman

Este conjunto de dados contém casos de um estudo realizado entre 1958 e 1970 no Hospital Billings da Universidade de Chicago sobre a sobrevivência de pacientes que se submeteram a cirurgia de câncer de mama. O banco de dados está disponível no site UCI (UCI, 1999) e as variáveis envolvidas neste estudo são:

- Idade: representa a idade do paciente no momento da operação.
- Ano: representa o ano de operação do paciente
- Nódulos: representa o número de nódulos axilares positivos detectados.
- Teste: assume apenas 2 valores (0 ou 1) e representa o status de sobrevivência do paciente, em que 0 indica que o paciente sobreviveu 5 anos ou mais e 1 que o paciente morreu dentro de 5 anos após a cirurgia.

Este banco de dados possui 306 observações, das quais 225 pacientes sobreviveram 5 anos ou mais após a cirurgia e 81 morreram dentro de 5 anos.

5.1.1 Análise Descritiva

Nesta Subseção apresentaremos uma análise descritiva dos dados descritos na Seção 5.1.

Boxplots

A seguir, apresentamos *boxplots* para cada variável preditora de acordo com o o status de sobrevivência do paciente, em que 0 indica que o paciente sobreviveu 5 anos ou mais e 1 que o paciente morreu dentro de 5 anos após a cirurgia.

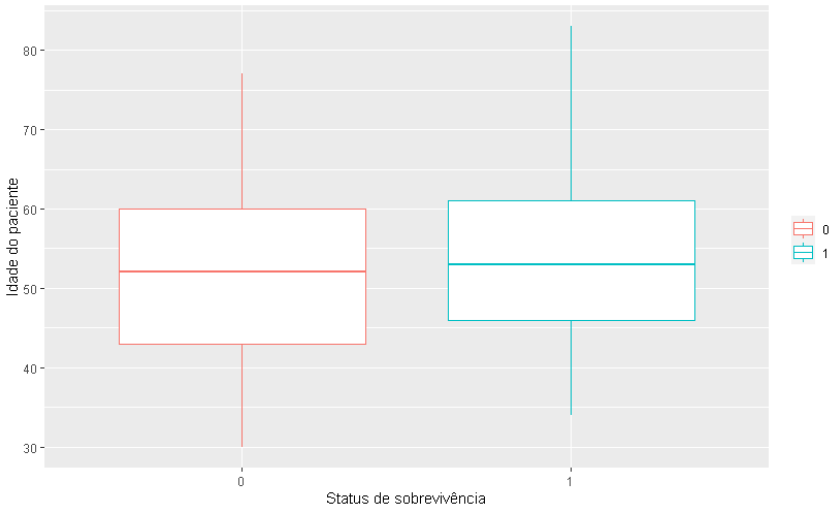


Figura 5.1: Boxplot para a variável idade do paciente

Na Figura 5.1 podemos notar que a idade que os pacientes que morreram dentro de cinco anos após a operação tinham na data da cirurgia parece ser muito similar as dos pacientes que sobreviveram cinco anos ou mais após a cirurgia. A variabilidade da idade em ambos os grupos também parece ser muito parecida.

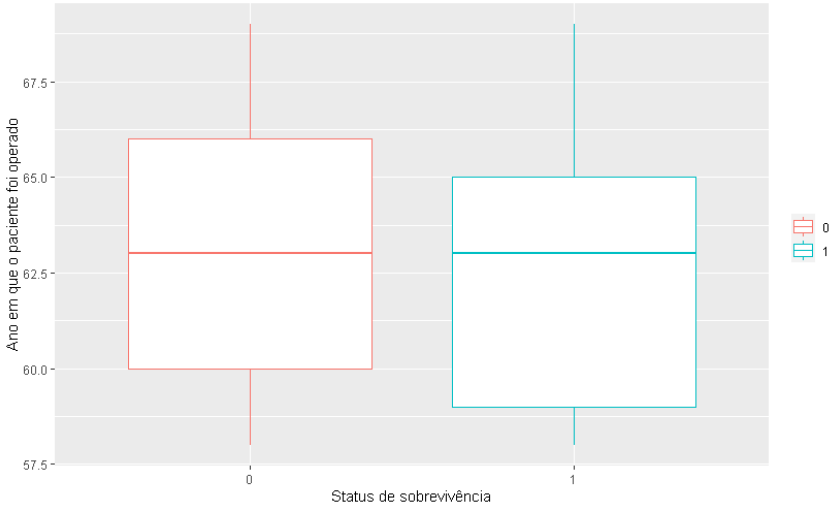


Figura 5.2: Boxplot para a variável ano da operação do paciente

Na Figura 5.2 vemos que a mediana e a variabilidade dos anos em que ocorreram as cirurgias dos pacientes parece ser semelhante nos 2 grupos. Entretanto,os pacientes

que sobreviveram mais de cinco anos após a cirurgia apresentam mediana amostral e terceiro quartil mais elevados dos anos em que ocorreram as operações se comparado ao outro grupo.

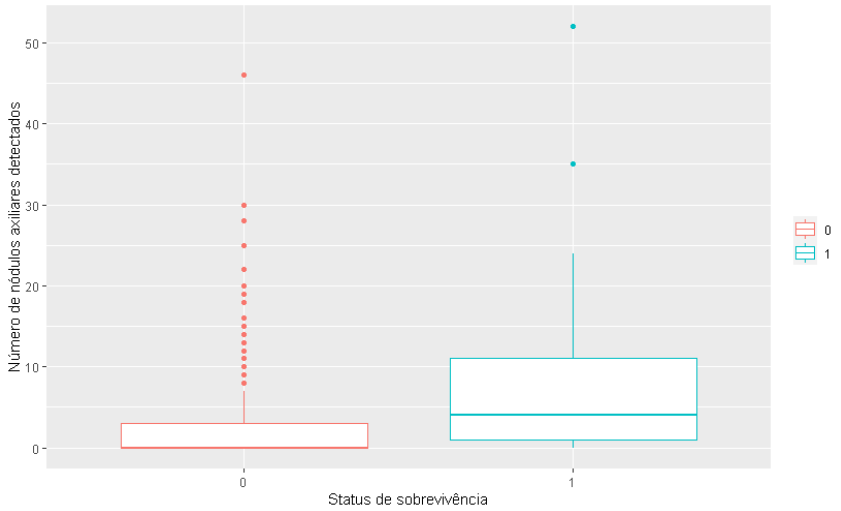


Figura 5.3: Boxplot para a variável número de nódulos axiliares

Na Figura 5.3 vemos que a distância interquartílica para o grupo que morreu dentro de cinco anos após a cirurgia parece ser maior comparativamente ao grupo que sobreviveu mais de cinco anos, porém há muitos pontos discrepantes no último grupo citado. O número mediano de nódulos axilares positivos detectados nos pacientes que morreram dentro de cinco anos após a cirurgia é bem maior que o do outro grupo.

Portanto, resumidamente, podemos observar que quanto maior o número de nódulos axilares positivos forem detectados nos pacientes, menor parece ser a probabilidade do paciente sobreviver mais de 5 anos após a cirurgia. Já as outras duas variáveis parecem apresentar pouca associação com a variável resposta.

5.1.2 Proporções

Para calcular as proporções neste Capítulo, dividimos o banco de dados em duas partes proporcionalmente, sendo uma parte utilizada para treinamento (70% das observações) e a outra parte para validação (30% das observações). Essa divisão foi feita 1000 vezes, utilizando o mesmo banco de dados. Ao final, temos 1000 réplicas diferentes, pois as divisões são feitas aleatoriamente. As proporções utilizadas na aplicação usam as mesmas definições discutidas nos estudos de simulação.

Tabela 5.1: Proporções encontradas a partir do banco de dados Haberman

Medida	**Base**	**Proporções**					
		Backward			**Lasso**		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,825	0,175	0,687	0,785	0,215	0,650
	Teste	0,716	0,284		0,696	0,304	
KS	**Treino**	0,793	0,207	0,724	0,909	0,227	0,701
	Teste	0,708	0,292		0,696	0,304	
H	Treino	0,941	0,059	0,694	0,909	0,091	0,430
	Teste	0,713	0,287		0,708	0,292	

A Tabela 5.1 apresenta as proporções de interesse encontradas a partir do banco de dados Haberman. Pode-se notar que, quando estamos trabalhando com a base de treinamento, utilizando o método de seleção de variáveis Backward, a medida H apresenta a proporção $p1$ consideravelmente maior que as demais medidas. E quando utilizamos o Lasso, ela tem o mesmo desempenho que a estatística KS, apresentando a proporção $p1$ notavelmente maior que a do coeficiente de Gini. Já quando estamos trabalhando com a base de validação, que no caso é a mais importante para responder o objetivo deste trabalho, temos que no geral, as três medidas apresentam proporções $p1$ próximas, tanto utilizando o Backward quanto o Lasso, sugerindo que o desempenho das medidas é parecido.

A estatística KS é a medida que apresenta as melhores proporções de $p3$, tanto quando estamos utilizando o Backward como quando utilizamos o Lasso. Isso significa que para esse caso, a estatística KS parece ser a medida que mais frequentemente considera o mesmo modelo como melhor nas bases de treinamento e validação.

Abaixo apresentamos os *boxplots* que representam as diferenças entre as natureza para as três medidas, utilizando os métodos de seleção de variáveis Backward e Lasso.

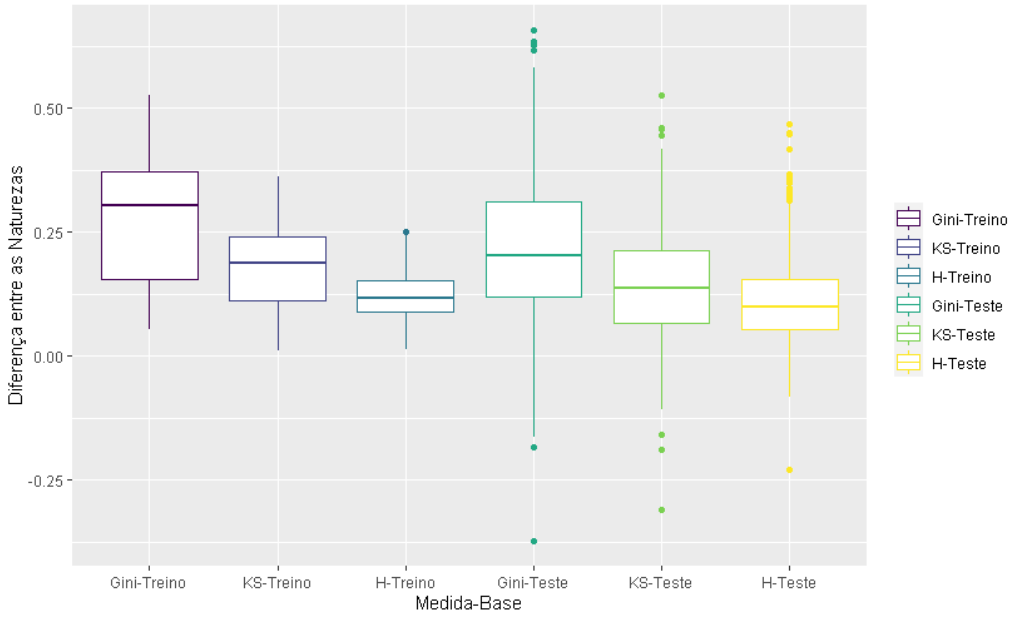


Figura 5.4: Boxplot utilizando o Backward.

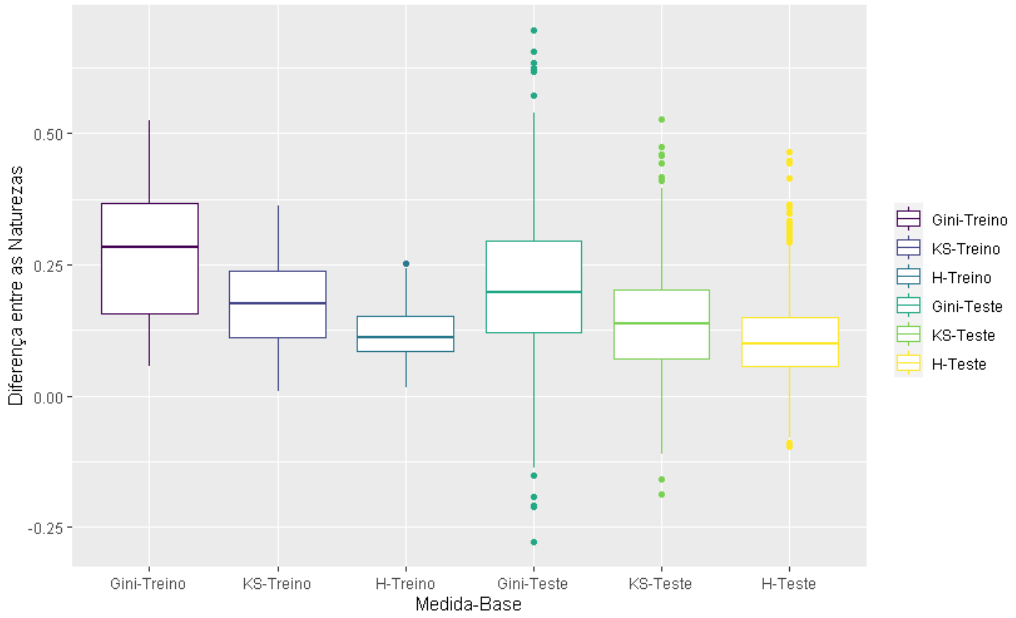


Figura 5.5: Boxplot utilizando o Lasso.

Através das Figuras 5.4 e 5.5, observamos que os *boxplots* estão alocados, em sua maioria, acima do valor 0. Dessa forma, os modelos selecionados pelo Backward e Lasso possuem medidas com valores mais altos do que as médias dos modelos da substituição.

Também é possível ver que o coeficiente de Gini parece ser a medida que apresenta diferenças mais consideráveis entre as naturezas, assim como as medianas dessas diferenças. Devido a isso, há indícios de que, nesta base de dados, ela seja a medida que

oferece o melhor desempenho na avaliação do poder preditivo.

A variabilidade dos *boxplots* que representam o coeficiente de Gini, são consideravelmente maiores do que os das demais medidas, para ambas as bases. Também verificamos que os *boxplots* referentes as bases de validação apresentam uma quantidade de *outliers* notável.

5.2 Pima.te

Este banco de dados é contido de informações de 332 mulheres com pelo menos 21 anos, de herança indígena Pima que vivem perto de Phoenix, Arizona. Os índios Pima, como são conhecidos, apresentam a maior taxa de obesidade e diabetes já registrada e o National Institute of Diabetes and Digestive and Kidney coletou dados sobre mulheres Pima. O banco é proveniente do pacote *MASS* (Venables e Ripley, 2002), do softaware R e as variáveis envolvidas neste estudo são:

- Gravidez: representa o número de vezes que cada mulher engravidou durante sua vida.
- Glicose: representa a concentração de glicoseno plasma em um teste de tolerância à glicose oral.
- Diastólico: representa a pressão sanguínea diastólica em mm/ Hg.
- Tríceps: representa a espessura da prega cutânea do tríceps (mm).
- IMC: representa o Índice de Massa Corporal (peso em kg / altura em metros quadrados).
- Diabetes: é um indicador quantitativo da história de diabetes na família.
- Idade: representa a idade em anos da mulher.
- Teste: pode levar apenas 2 valores (0 ou 1) e representa se o paciente possui sinais de diabetes, em que 0 significa que o paciente não apresenta sinais de diabetes e 1 que o paciente apresenta sinais de diabetes.

Este banco de dados possui 332 observações, das quais 223 mulheres não apresentam sinais de diabetes e 109 apresentam sinais de diabetes.

5.2.1 Análise descritiva

Nesta Subseção apresentaremos uma análise descritiva dos dados descritos na Subseção anterior.

Boxplots

A seguir, apresentamos *boxplots* para cada uma das variáveis preditoras contidas neste banco, relacionando-as com o diagnóstico que a mulher possui, em que 0 ela não possui sinais de diabetes e 1 ela possui.

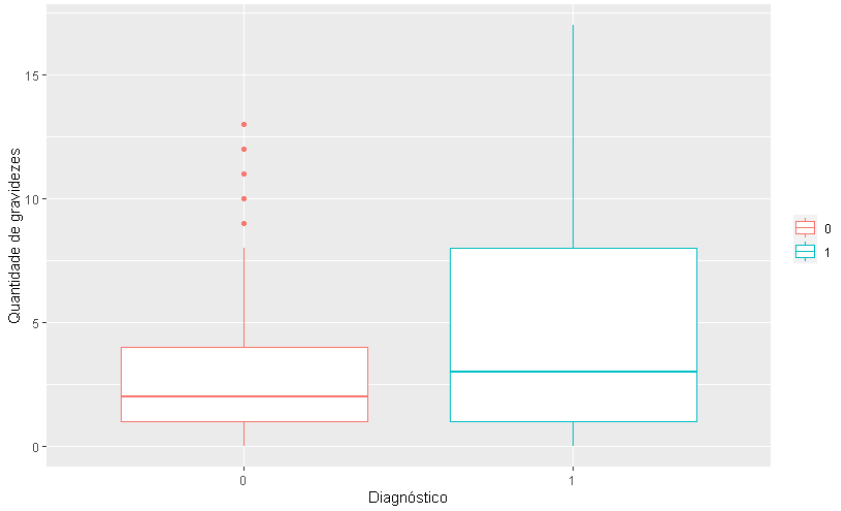


Figura 5.6: Boxplot para a variável gravidez

Pela Figura 5.6 é possível ver que o grupo de mulheres que possuem sinais de diabetes parece apresentar um número mediano de gravidezes semelhante ao outro grupo, mas com um terceiro quartil bem superior. Também vemos que a variabilidade do número de gravidezes das mulheres que apresentam sinais de diabetes parece ser maior que o das demais mulheres.

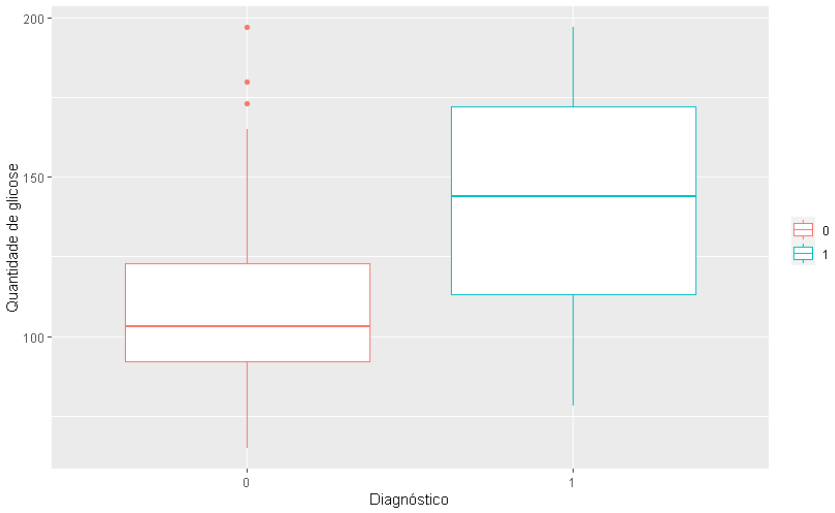


Figura 5.7: Boxplot para a variável glicose

Pela Figura 5.7 percebemos que a concentração de glicose do grupo de mulheres que apresentam sinais de diabetes parece ser, em geral, consideravelmente maior que a do grupo de mulheres que não apresentam, assim como a variabilidade dessa mesma concentração.

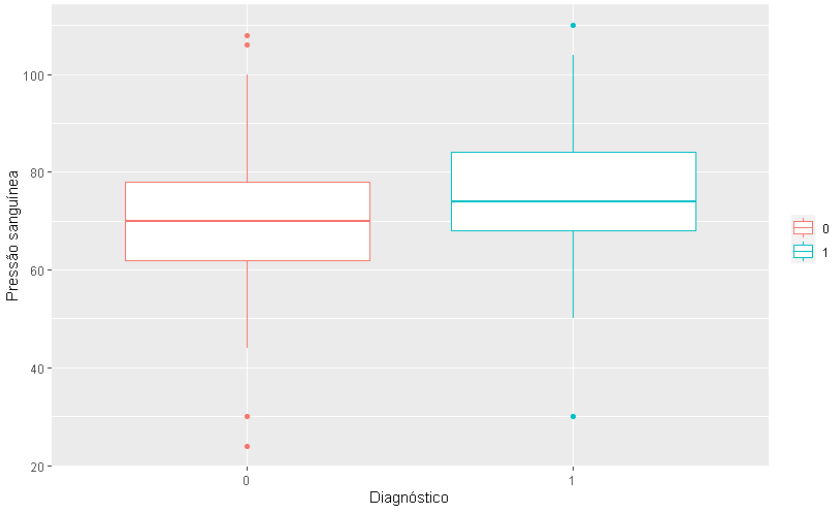


Figura 5.8: Boxplot para a variável diastólica

Analisando a Figura 5.8 é possível verificar que a mediana da pressão sanguínea das mulheres que possuem sinais de diabetes é ligeiramente maior do que a das que não possuem. Já a variabilidade da pressão sanguínea de ambos os grupos, parece ser bem similar.

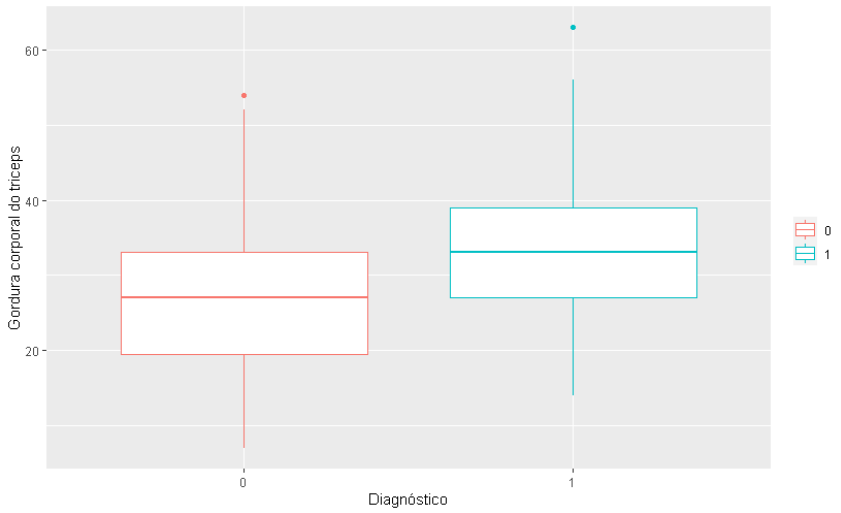


Figura 5.9: Boxplot para a variável tríceps

Na Figura 5.9 é possível ver que a espessura mediana da prega cutânea do tríceps das mulheres que apresentam sinais de diabetes parece ser superior a das mulheres que não apresentam. A variabilidade dessa espessura parece ser bem parecida para ambos os grupos.

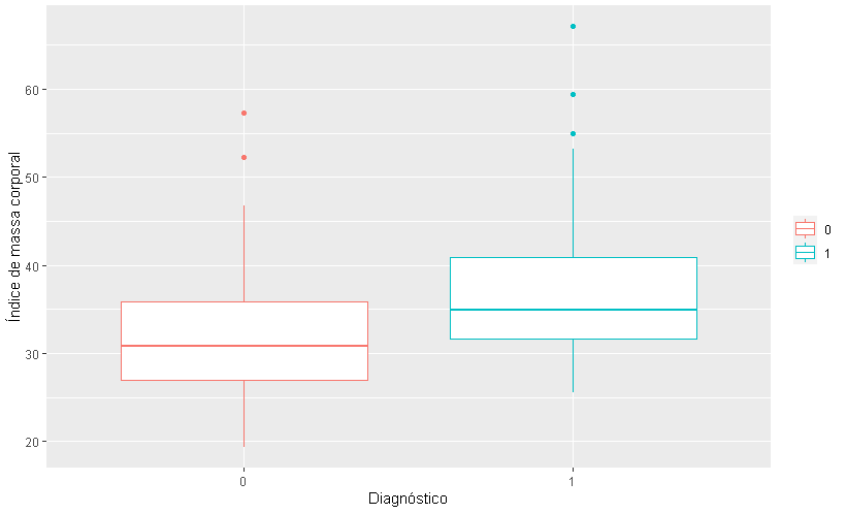


Figura 5.10: Boxplot para a variável IMC

Pela Figura 5.10 vemos que a variabilidade do IMC das mulheres de ambos os grupos seria bem parecida, se não houvesse um outlier superior no grupo das mulheres que não possuem sinais de diabetes. A mediana do grupo das mulheres que apresentam sinais de diabetes também parece ser maior que a do outro grupo.

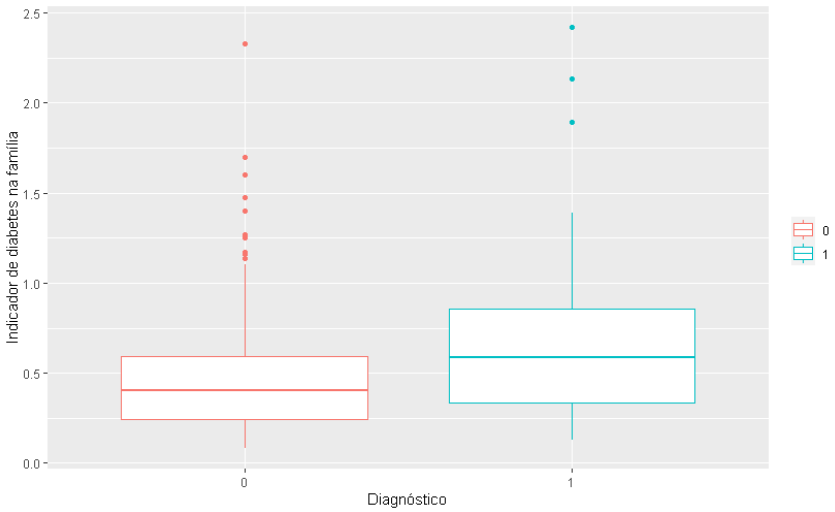


Figura 5.11: Boxplot para a variável diabetes

Na Figura 5.11 percebe-se que tanto a mediana quanto a variabilidade do indicador quantitativo do histórico de diabetes na família parecem ser maiores no grupo das mulheres que apresentam sinais de diabetes.

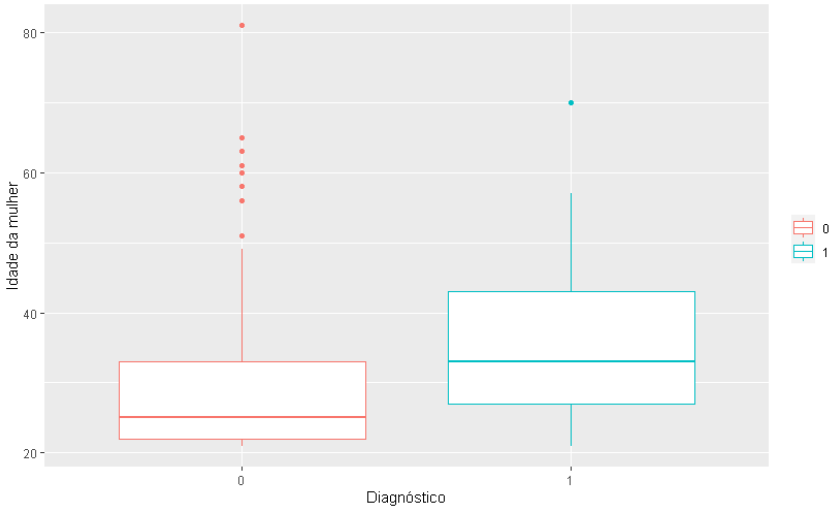


Figura 5.12: Boxplot para a variável idade

Pela Figura 5.12 vemos que as mulheres que possuem sinais de diabetes parecem ter idade mediana maior que as demais. Contudo, o *boxplot* que representa a idade das mulheres que não possuem sinais de diabetes contém um grande número de pontos discrepantes.

Resumindo, os *boxplots* parecem sugerir que as mulheres que mais tendem a ter diabetes são as que possuem um número maior de gravidezes durante a vida, maior concentração de glicose, maior pressão sanguínea, assim como a espessura da prega cutânea do tríceps, índice de massa corporal, indicador de diabetes na família e idade.

5.2.2 Proporções

Na Tabela 5.2 apresentamos as proporções encontradas a partir do banco de dados Pima.te. A divisão das bases foi feita de forma análoga à apresentada na Subseção 5.1.2.

Tabela 5.2: Proporções encontradas a partir do banco de dados Pima.te

Medida	**Base**	**Proporções**					
		Backward			**Lasso**		
		$p1$	$p2$	$p3$	$p1$	$p2$	$p3$
Gini	**Treino**	0,818	0,182	0,319	0,380	0,620	0,121
	Teste	0,287	0,713		0,143	0,857	
KS	**Treino**	0,433	0,567	0,139	0,137	0,863	0,229
	Teste	0,200	0,800		0,139	0,861	
H	Treino	0,665	0,335	0,228	0,182	0,818	0,202
	Teste	0,217	0,783		0,114	0,886	

Analisando a Tabela 5.2, temos que para a base de treinamento, utilizando o método de seleção de variáveis Backward, o coeficiente de Gini apresenta a proporção $p1$ bem maior do que as outras medidas estudadas neste trabalho. O mesmo acontece, quando utilizamos o Lasso. Já quando estamos trabalhando com a base de validação, é perceptível que as medidas, apresentam proporções $p1$ muito parecidas, utilizando o método Lasso. Porém, mesmo com pouca diferença, o coeficiente de Gini ainda apresenta as maiores proporções. Já utilizando o Backward, a proporção $p1$ apresentada pelo coefiente de Gini, também é maior que as demais, mas a diferença é maior do que a observada com o método Lasso.

Em relação a proporção $p3$, para o caso em que a seleção de variáveis foi feita pelo Backward, o coeficiente de Gini é o que apresenta a maior proporção. Já para o Lasso, quem se desempenha melhor é a estatística KS.

Abaixo apresentamos os *boxplots* que representam as diferenças entre as natureza para as três medidas, utilizando os métodos de seleção de variáveis Backward e Lasso.

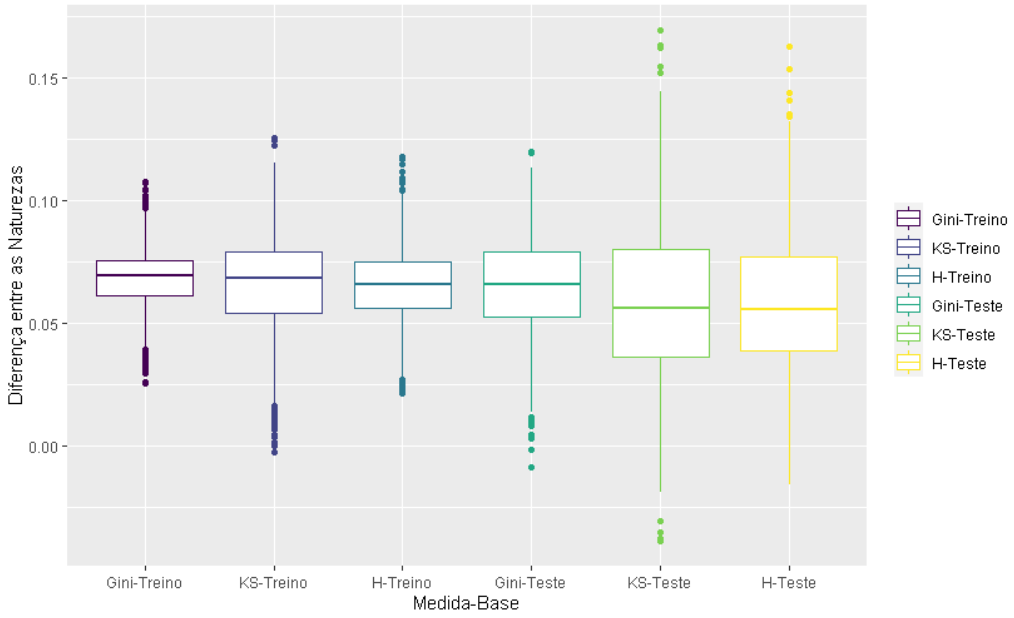


Figura 5.13: Boxplot utilizando o Backward.

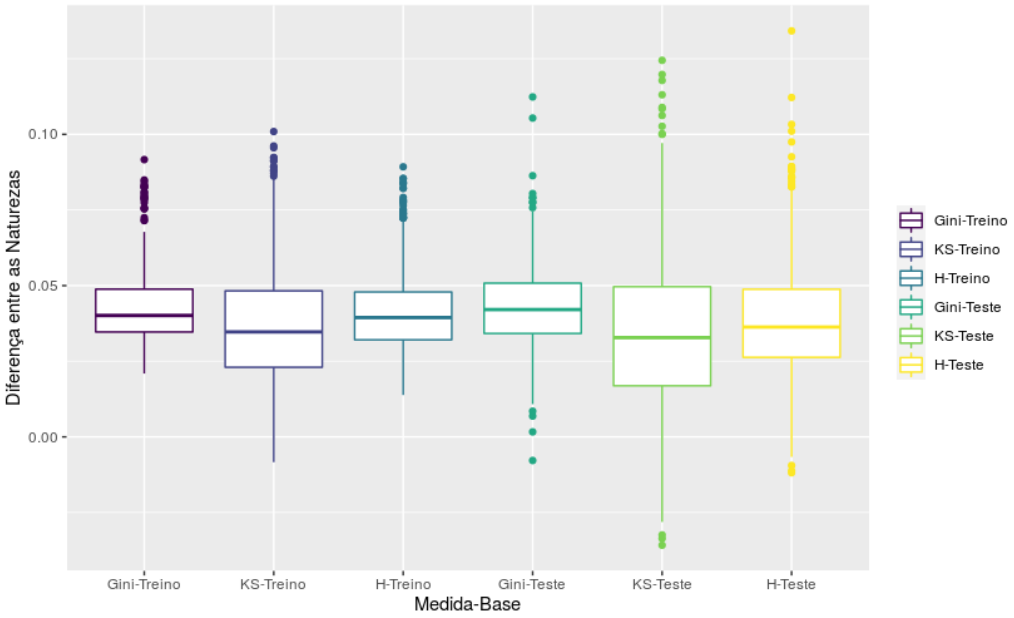


Figura 5.14: Boxplot utilizando o Lasso.

Analisando as Figuras 5.13 e 5.14, observamos que a diferença entre as naturezas na maioria dos casos, é positiva. Portanto, os modelos selecionados pelo Backward e Lasso possuem medidas com valores mais altos do que as médias dos modelos da substituição.

A mediana de todas as medidas, tanto para as bases de treinamento quanto para as de teste são próximas. Porém é possível ver que, as medianas relacionadas ao coeficiente de Gini são suavemente maiores, em ambas as bases. Os *boxplots* relacionados

a estatística KS são os que possuem maior variabilidade e menor mediana da diferença entre as naturezas. Portanto, há evidências de que as medidas possuem desempenho semelhantes, mas o coeficiente de Gini ainda parece ser a melhor opção.

5.3 Resumo dos resultados

Pelas tabelas contendo as proporções encontradas a partir das 1000 réplicas, considerando as bases de validação, podemos verificar que as medidas possuem performances muito parecidas na avaliação do poder preditivo de modelos com resposta binária. Já pelos *boxplots* , também considerando as bases de validação, o coeficiente de Gini é a medida que apresenta o melhor desempenho para avaliar o poder preditivo de modelos para dados binários, pois ela possui as maiores diferenças positivas entre as naturezas.

No decorrer deste capítulo, notamos que quando estamos trabalhando com dados reais, as proporções $p1$ e $p3$ encontradas não são tão altas. Isso acontece, pois a análise realizada depende diretamente do resultado dos métodos de seleção de variáveis, que muitas vezes pode retirar alguma variável importante do modelo, aumentando assim as chances do modelo selecionado como melhor, através das medidas, estar contido na natureza 2.

Capítulo 6

Conclusões

Neste trabalho estudamos três medidas de avaliação do poder preditivo de modelos de regressão para variáveis respostas binárias. São elas o coeficiente de Gini, o KS e a medida H. Descrevemos essa medidas e as comparamos em estudos de simulação e também em bancos de dados reais, através do cálculo de proporções e da análise da distribuição da diferença do valor da medida entre o modelo correto ou selecionado por algum método e modelos alternativos.

A principal conclusão que obtemos neste trabalho é que, para quando os coeficientes de regressão são pequenos (em torno do valor 1), o coeficiente de Gini apresenta melhor performance que as demais medidas na avaliação do poder preditivo de modelos de regressão para resposta binária. Já quando o ajuste de regressão apresenta coeficientes com altos valores (em torno do valor 3), a medida H também torna-se um medida com uma performance compatível ao coeficiente de Gini.

É importante destacar que, quando estamos trabalhando com regressão logística, utilizamos razão de chances para interpretar os parâmetros, como apresentado na Seção 2.4. Sendo assim, cada parâmetro é interpretado a partir da resultante $\exp(\beta_j)$. É por esse motivo que estamos considerando 1 como um coeficiente pequeno e 3 como alto, pois $\exp(1) \approx 2,71$ e $\exp(3) \approx 20$, ou seja, o aumento de duas unidades no parâmetro representa uma diferença relativamente grande na resultante $\exp(\beta_j)$.

Nas aplicações, a superioridade do coeficiente de Gini não ficou tão evidente quanto nos estudos de simulação. Isso pode ser explicado porque nesses casos não temos um modelo correto e nem sempre o modelo escolhido por um método de seleção de variáveis é o melhor modelo.

Além de apresentar a melhor performance, o coeficiente de Gini é uma medida

com embasamento teórico muito mais simples que a medida H. A estatística KS também é simples, mas em geral, apresentou resultados bem inferiores ao coeficiente de Gini.

Neste trabalho, tanto nos estudos de simulação quanto nas aplicações, está sendo considerado um número pequeno de variáveis. Portanto, ainda não existe evidências de que essas conclusões obtidas são válidas para situações onde existe alta dimensionalidade. Um outro estudo pode ser desenvolvido para fazer tal verificação.

Além do estudo mencionado no parágrafo anterior, outros trabalhos futuros podem ser desenvolvidos. Pode-se considerar por exemplo outras medidas para avaliar o poder preditivo de modelos com respostas binárias. Também poder ser comparada as medidas estudadas neste trabalho em outras técnicas adequadas para variáveis respostas binárias, como, por exemplo, *gradient boosting* (Bello, 2018) e florestas aleatórias (Friedman *et al.*, 2001).

Referências Bibliográficas

Abreu, H. J. (2005). Aplicação da análise de sobrevivência em um problema de credit scoring e comparação com a regressão logística. Dissertação de mestrado, Universidade Federal de São Carlos.

Agresti, A. (1990). *Categorical data analysis*. John Wiley & Sons.

Akaike, H. (1974). A new look at the statistical model identification. In *Selected Papers of Hirotugu Akaike*, pages 215–222. Springer.

Akaike, H. (1978). A bayesian analysis of the minimum aic procedure. *Annals of the Institute of Statistical Mathematics A*, pages 9–14.

Alves, M. C. (2008). Estratégias para o desenvolvimento de modelos de credit score com inferência de rejeitados. Dissertação de mestrado, Universidade de São Paulo.

Anagnostopoulos, C. e Hand, D. J. (2019). *hmeasure: The H-Measure and Other Scalar Classification Performance Metrics*. R package version 1.0-2.

Antonio, F. d. C. V. (2009). Análise da média e dispersão em experimentos fatoriais não replicados para otimização de processos industriais. Tese de doutorado, Departamento de Engenharia Industrial, Pontifícia Universidade Católica do Rio de Janeiro.

Barakat, H., Nigm, E. S. e Khaled, O. (2019). *Statistical Techniques for Modelling Extreme Value Data and Related Applications*. Cambridge Scholars Publishing, first edition.

Bello, S. N. (2018). Gradient tree boosting teoria e uma aplicação a dados reais. Trabalho de conclusão de curso, Universidade Federal de São Carlos.

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, **30**, 1145–1559.

Brolo, C. L. (2019). Comparação da performance do lasso e do método da máxima verossimilhança com seleção de variáveis em modelos de regressão para dados binários. Trabalho de conclusão de curso, Universidade Federal de São Carlos.

Conover, W. (1999). *Practical nonparametric statistics*. John Wiley & Sons, third edition.

Delacour, H., Servonnet, A., Perrot, A., Vigezzi, J. F. e Ramirez, J. M. (2013). Roc (receiver operating characteristics) curve: Principles and application in biology. *Ann Biol Clin (Paris)*, pages 145–154.

Demirtas, H., Allozi, R. e Gao, R. (2020). *MultiRNG: Multivariate Pseudo-Random Number Generation*. R package version 1.2.3.

Demétrio, C. G. B. (2002). *Modelos Lineares Generalizados em Experimentação Agronômica*. ESALQ/USP.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, **27**(8), 861–874.

Forti, M. (2019). Técnicas de machine learning aplicadas na recuperação de crédito do mercado brasileiro. Trabalho de conclusão de curso, Fundação Getulio Vargas.

Friedman, J., Hastie, T. e Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in Statistics New York.

Friedman, J., Hastie, T. e Tibshirani, R. (2009). *glmnet: Lasso and elastic-net regularized generalized linear models*. R package version 1.1-4.

Gouvêa, M. A., Gonçalves, E. B. e Mantovani, D. M. N. (2013). Análise de risco de crédito com aplicação de regressão logística e redes neurais. *Revista Contabilidade Vista Revista*, pages 96–123.

Hajian-Tilaki, K. (2013). Receiver operating characteristic (roc) curve analysis for medical diagnostic test evaluation. *Caspian Journal of Internal Medicine*, **3**, 627—-635.

Hand, D. (2009). Measuring classifier performance: A coherent alternative to the area under the roc curve. *Machine Learning*, **77**, 103–123.

Hand, D. e Anagnostopoulos, C. (2014). A better beta for the h measure of classification performance. *Pattern Recognition Letters*, **40**, 41–46.

Hastie, T., Tibshirani, R. e Wainwright, M. (2015). *Statistical learning with sparsity: The lasso and generalizations*. Chapman Hall/CRC.

Hosmer Jr, D. W. e Lemeshow, S. (2004). *Applied logistic regression*. John Wiley & Sons.

Izbicki, R. e dos Santos, T. M. (2020). *Aprendizado de máquina: uma abordagem estatística*.

Khan, M. R. A. e Brandenburger, T. (2020). *ROCit: Performance Assessment of Binary Classifier with Visualization*. R package version 2.1.1.

Magalhães, M. N. (2011). *Probabilidade e Variáveis Aleatórias*. EDUSP.

Manfio, F. (2007). *O Risco Nosso de Cada Dia*. Estação das Letras.

Martinez, E., Louzada, F. e Pereira, B. (2003). A curva roc para testes diagnósticos. *Cad Saúde Coletiva*, **11**, 7–31.

McCullagh, P. e Nelder, J. (1989). *Generalized linear models*. Chapman Hall.

Nelder, J. A. e Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, **135**(3), 370–384.

Neter, J., Kutner, M. H., Nachtsheim, C. J. e Wasserman, W. (1996). *Applied linear statistical models*. Irwin Chicago.

Nunes, L. L. (2011). Aplicação do modelo de regressão logística para apoio à decisão de crédito. Trabalho de conclusão de curso, Universidade Federal de Juiz de Fora.

Paula, G. A. (2004). *Modelos de regressão: com apoio computacional*. IME-USP São Paulo.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, **6**(2), 461–464.

Souza, C. (2019). Análise de poder discriminativo através de curva roc. Disponível em: < http://crsouza.com/2009/07/13/analise-de-poder-discriminativo-atraves-de-curvas-roc >. Acessado em: 29 outubro. 2020.

Thomas, L. C., Edelman, D. B. e Crook, J. N. (2002). *Credit scoring and its applications*. SIAM.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*.

UCI (1999). Haberman. Disponível em: < https://archive.ics.uci.edu/ml/datasets/Haberman's%2BSurvival >. Acessado em: 10 outubro. 2020.

Vaz, J. C. L. (2009). Regiões de incerteza para a curva roc em testes diagnósticos. Dissertação de mestrado, Universidade Federal de São Carlos.

Venables, W. N. e Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition.

Capítulo 7

Anexo

```
 #### BIBLIOTECA ############
library(MASS)
library(pROC)
library(tidyverse)
library(hmeasure)
library(glmnet)
library(MASS)



###### CÓDIGO USADO PARA CALCULAR AS PROPORÇÕES UTILIZANDO MODELO CORRETO #####
### Gerando os dados simulados
set.seed(3657)
xxx = c(sample(1:10000,1000),replace=F)
n_var = 5
dados = matrix(c(rep(1,n_var*500)),500,n_var)
dados
for( iii in 2:n_var){
  set.seed(xxx[iii])
  dados[,iii] = (runif(500,0,1))
}
betas = matrix(c(1,2,-2,0,0),n_var,1)
e = dados%*%betas
```

```
dados1 = as.data.frame(dados)


u = exp(e)/(1+exp(e))
u


#### Calculando medidas para as naturezas 1 e 2
max_step_treino_G = c()
max_step_teste_G = c()
treino_teste_igual_G = c()
max_sub_treino_G = c()
max_sub_teste_G =c()
m_step_treino_G = c()
m_step_teste_G = c()
m_sub_treino_G = c()
m_sub_teste_G = c()

max_step_treino_H = c()
max_step_teste_H = c()
treino_teste_igual_H = c()
max_sub_treino_H = c()
max_sub_teste_H =c()
m_step_treino_H = c()
m_step_teste_H = c()
m_sub_treino_H = c()
m_sub_teste_H = c()

max_step_treino_KS = c()
max_step_teste_KS = c()
treino_teste_igual_KS = c()
max_sub_treino_KS = c()
```

```
max_sub_teste_KS =c()
m_step_treino_KS = c()
m_step_teste_KS = c()
m_sub_treino_KS = c()
m_sub_teste_KS = c()

set.seed(494939)
x = c(sample(1:10000,10000),replace=F)
cont=0
v=0
j=0
n=0
for(k in 1:10000){
  set.seed(x[k])
  for (tt in 1:500){
    dados1$TESTE[tt]=rbernoulli(1,u[tt])
  }
  dados1$TESTE
  dados1$V1 = NULL
#### Separação do banco em teste e treino
  set.seed(x[k])
  div = c(treino = .7, teste = .3)
  g = sample(cut(
    seq(nrow(dados1)),
    nrow(dados1)*cumsum(c(0,div)),
    labels = names(div)
  ))

  res = split(dados1, g)
  nrow(res$treino) #
  nrow(res$teste) #
  teste <- res$teste
  treino <- res$treino
```

```
#### ajustando o modelo com todas as variáveis
  modelo1<-  glm(TESTE ~ .,data=treino,family = binomial(link=logit))
  aa = data.matrix(modelo1$coefficients)
  aa = c(rownames(aa))
  aa=aa[-1]

#### ajustando o modelo correto
  modelstep = glm(TESTE ~ V2+V3,data=treino,family = binomial(link=logit))
  pred = predict(modelstep)



  results <- HMeasure(treino$TESTE,pred)

  a = results$metrics$H

  bb = data.matrix(modelstep$coefficients)
  bb = c(rownames(bb))
  bb=bb[-1]
  bb

#### calculando medidas para o modelo correto
  if(identical(aa,bb)==T){v=v+1}
  if(length(bb)==1){j=j+1}
  if(length(bb)==0){n=n+1}
  if(identical(aa,bb)==F & length(bb)!=0){
    cont=cont+1
    G1 = c()
    H1 = c()
    KS1 = c()
    G1[1] = results$metrics$Gini
    H1[1] = results$metrics$H
    KS1[1] = results$metrics$KS
```

```
    entram = setdiff(aa,bb)
    formula <- rep(NA,length(entram)*length(bb) )

#### ajudando demais modelos (os contidos na natureza 2)
    conts = 1
    for(d in 1:length(bb)){
      covs <- paste(bb, collapse = "+")
      covs1 = paste0(c(covs, bb[d]), collapse = "-")
      covs1
      for(g in 1:length(entram)){
        covs2 = paste0(c(covs1, entram[g]), collapse = "+")
        formula[conts] <- paste0("TESTE ~ ", covs2)
        conts = conts+1
      }

    }

#### calculando medidas nos modelos da natureza 2
    for(i in 1:(length(entram)*length(bb))){
      modi<- glm(formula[i], data = treino,family = binomial(link=logit))
      pred = predict(modi)
      results <- HMeasure(treino$TESTE,pred)

      G1[i+1] = results$metrics$Gini
      H1[i+1] = results$metrics$H
      KS1[i+1] = results$metrics$KS
    }


#####Predição #######

    xnovo = teste
```

```
xnovo$TESTE=NULL
pred = predict(modelstep,newdata=xnovo)
results <- HMeasure(teste$TESTE,pred)

G2 = c()
H2 = c()
KS2 = c()
G2[1] = results$metrics$Gini
H2[1] = results$metrics$H
KS2[1] = results$metrics$KS



for(ff in 1:(length(entram)*length(bb))){
  modi<- glm(formula[ff], data = treino,family = binomial(link=logit))
  pred = predict(modi,newdata=xnovo)
  results = HMeasure(teste$TESTE,pred)
  G2[ff+1] = results$metrics$Gini
  H2[ff+1] = results$metrics$H
  KS2[ff+1] = results$metrics$KS

}

max_step_treino_G[k] = ifelse(which.max(G1)==1,1,0)
max_step_teste_G[k] = ifelse(which.max(G2)==1,1,0)
max_sub_treino_G[k] = ifelse(which.max(G1)>1,1,0)
max_sub_teste_G[k] = ifelse(which.max(G2)>1,1,0)
treino_teste_igual_G[k] = ifelse(which.max(G1)==which.max(G2),1,0)

m_step_treino_G[k] = G1[1]
m_step_teste_G[k] = G2[1]
m_sub_treino_G[k]=mean(G1[-1])
m_sub_teste_G[k]=mean(G2[-1])
```

```
    max_step_treino_H[k] = ifelse(which.max(H1)==1,1,0)
    max_step_teste_H[k] = ifelse(which.max(H2)==1,1,0)
    max_sub_treino_H[k] = ifelse(which.max(H1)>1,1,0)
    max_sub_teste_H[k] = ifelse(which.max(H2)>1,1,0)
    treino_teste_igual_H[k] = ifelse(which.max(H1)==which.max(H2),1,0)

    m_step_treino_H[k] = H1[1]
    m_step_teste_H[k] = H2[1]
    m_sub_treino_H[k]=mean(H1[-1])
    m_sub_teste_H[k]=mean(H2[-1])



    max_step_treino_KS[k] = ifelse(which.max(KS1)==1,1,0)
    max_step_teste_KS[k] = ifelse(which.max(KS2)==1,1,0)
    max_sub_treino_KS[k] = ifelse(which.max(KS1)>1,1,0)
    max_sub_teste_KS[k] = ifelse(which.max(KS2)>1,1,0)
    treino_teste_igual_KS[k] = ifelse(which.max(KS1)==which.max(KS2),1,0)

    m_step_treino_KS[k] = KS1[1]
    m_step_teste_KS[k] =  KS2[1]
    m_sub_treino_KS[k]=mean(KS1[-1])
    m_sub_teste_KS[k]=mean(KS2[-1])
  }
  if(cont==1000){
    stop()
  }

}

### calculano proporções
prop_max_step_treino_G = mean(max_step_treino_G, na.rm = T)
prop_max_step_teste_G = mean(max_step_teste_G, na.rm = T)
```

```
prop_max_sub_treino_G = mean(max_sub_treino_G, na.rm = T)
prop_max_sub_teste_G = mean(max_sub_teste_G, na.rm = T)
prop_igual_teste_treino_G = mean(treino_teste_igual_G,na.rm = T)

prop_max_step_treino_G
prop_max_step_teste_G
prop_max_sub_treino_G
prop_max_sub_teste_G
prop_igual_teste_treino_G

prop_max_step_treino_KS = mean(max_step_treino_KS, na.rm = T)
prop_max_step_teste_KS = mean(max_step_teste_KS, na.rm = T)
prop_max_sub_treino_KS = mean(max_sub_treino_KS, na.rm = T)
prop_max_sub_teste_KS = mean(max_sub_teste_KS, na.rm = T)
prop_igual_teste_treino_KS = mean(treino_teste_igual_KS,na.rm = T)

prop_max_step_treino_KS
prop_max_step_teste_KS
prop_max_sub_treino_KS
prop_max_sub_teste_KS
prop_igual_teste_treino_KS

prop_max_step_treino_H = mean(max_step_treino_H, na.rm = T)
prop_max_step_teste_H = mean(max_step_teste_H, na.rm = T)
prop_max_sub_treino_H = mean(max_sub_treino_H, na.rm = T)
prop_max_sub_teste_H = mean(max_sub_teste_H, na.rm = T)
prop_igual_teste_treino_H = mean(treino_teste_igual_H,na.rm = T)

prop_max_step_treino_H
prop_max_step_teste_H
prop_max_sub_treino_H
prop_max_sub_teste_H
prop_igual_teste_treino_H
```

```
diferenca = c(m_step_treino_G-m_sub_treino_G,m_step_treino_KS-m_sub_treino_KS
              ,m_step_treino_H-m_sub_treino_H,m_step_teste_G-m_sub_teste_G,
              m_step_teste_KS-m_sub_teste_KS,m_step_teste_H-m_sub_teste_H
                )

medida = c(rep('Gini-Treino',length(diferenca)/6),rep('KS-Treino',
length(diferenca)/6), rep('H-Treino',length(diferenca)/6),
rep('Gini-Teste',length(diferenca)/6),rep('KS-Teste',length(diferenca)/6),
           rep('H-Teste',length(diferenca)/6))

medida = ordered(medida,levels= c('Gini-Treino','KS-Treino','H-Treino',
                                  'Gini-Teste','KS-Teste','H-Teste'))


banco = cbind.data.frame(medida,diferenca)


##### Criando boxplots #####
ggplot(banco, aes(x=medida, y=diferenca, color=factor(medida))) +
  geom_boxplot()+ xlab('Medida-Base')+ ylab('Diferença entre as Naturezas') +
  theme(legend.title=element_blank())

 ##### AS PROPORÇÕES DOS DEMAIS CENÁRIOS SÃO FEITAS DE MANEIRA ANÁLOGA,
 SÓ MUDA A FORMA EM COMO OS DADOS SÃO SIMULADOS E EM ALGUMAS SITUAÇÕES É FEITA
 IMPLEMENTADA A SELEÇÃO DE VARIÁVEIS. O CÓDIGO UTILIZADO PARA OS DADOS REAIS
 TAMBÉM É O MESMO, MAS NO LUGAR DOS DADOS SIMULADOS ESTÁ A ENTRADA DOS
 DADOS REAIS


 ##############################################
```

```
 ################## ANÁLISE DESCRITIVA #################
####### Pima.te
data(Pima.te)
per = Pima.te
per$type =  as.numeric(per$type)
names(per)[8]=c('TESTE')
per$TESTE
per$TESTE = ifelse(per$TESTE==2,1,0)
sum(per$TESTE==0)
sum(per$TESTE==1)

var(Pima.te$age[Pima.te$type=='Yes'])
var(Pima.te$age[Pima.te$type=='No'])

## Gravidez
plot(Pima.te$type,Pima.te$npreg)
ggplot(per, aes(x=factor(TESTE), y=npreg, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Quantidade de gravidezes') +
  theme(legend.title=element_blank())

## Glicose

ggplot(per, aes(x=factor(TESTE), y=glu, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Quantidade de glicose') +
  theme(legend.title=element_blank())

## Diastolica

ggplot(per, aes(x=factor(TESTE), y=bp, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Pressão sanguínea') +
  theme(legend.title=element_blank())
```

```
## Triceps

ggplot(per, aes(x=factor(TESTE), y=skin, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Gordura corporal do triceps') +
  theme(legend.title=element_blank())


##IMC

ggplot(per, aes(x=factor(TESTE), y=bmi, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Índice de massa corporal') +
  theme(legend.title=element_blank())

## Diabetes

ggplot(per, aes(x=factor(TESTE), y=ped, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Indicador de diabetes na família') +
  theme(legend.title=element_blank())


## Idade

ggplot(per, aes(x=factor(TESTE), y=age, color=factor(TESTE))) +
  geom_boxplot()+ xlab('Diagnóstico')+ ylab('Idade da mulher') +
  theme(legend.title=element_blank())
```