

FBST em problemas de likelihood-free

Marcela Musetti

DISSERTAÇÃO APRESENTADA AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA UNIVERSIDADE DE SÃO PAULO
PARA OBTENÇÃO DO TÍTULO DE
MESTRA EM CIÊNCIAS

Programa: Estatística

Orientador: Prof. Dr. Luís Gustavo Esteves

São Paulo
10 de Dezembro de 2022

FBST em problemas de likelihood-free

Marcela Musetti

Esta é a versão original da dissertação
elaborada pela candidata Marcela Musetti,
tal como submetida à Comissão Julgadora.

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

*Todos estes que aí estão atravancando o
meu caminho, eles passarão. Eu passarinho!*
- Mário Quintana

Agradecimentos

Agradeço a minha família, Ilda, José, Gustavo, Mariana, Maria, Bianca e Heitor, toda força que eu tenho vem de vocês. Em especial, aos meus pais, por sonharem meus sonhos junto comigo.

Agradeço a orientação de Luís Esteves e Rafael Izbicki, obrigada por aceitarem estar comigo nesse trabalho, pelo apoio, paciência, e por compartilhar o imenso conhecimento de vocês

Agradeço a todos meus amigos por serem minha válvula de escape no meio do caos. Vocês conseguem deixar momentos difíceis muito mais leves. Em especial, Gabi e Ju que acompanham de perto todos os meus anseios, inseguranças, medos e ansiedades. Vocês foram essenciais para eu estar aqui hoje.

Por fim, agradeço a todos que passaram pela minha vida nesse período, fazendo parte da minha formação direta ou indiretamente. Alguns não estão mais nela, mas eu sempre lembrarei com muito carinho.

Resumo

Marcela Musetti. **FBST em problemas de likelihood-free**. Dissertação (Mestrado).
Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2022.

Na inferência Bayesiana, problemas com a obtenção da distribuição a posteriori analiticamente, em problemas de *likelihood-free* são frequentes. E mesmo quando a solução desse problema é feito de forma computacional, ainda temos desafios com a realização de inferências, como o teste de hipóteses precisas. Uma forma Bayesiana de testar hipóteses precisas é via o Full Bayesian Significance Test (FBST) no qual calcula-se uma medida de evidência, denominada por e-valor. Nessa dissertação, queremos encontrar as densidades a posteriori em problemas de *likelihood-free* pelo o método FlexCode, ao invés do tradicional MCMC e com as densidades em mãos propor uma solução para o cálculo do e-valor: através de métodos de classificação ou da integração computacional da densidade obtida via FlexCode. Dessa forma, conseguimos resolver problemas frequentes na inferência bayesiana transformando-os em um problema possível de resolver no universo de aprendizado de máquina

Palavras-chave: FBST. *likelihood-free*. Aprendizado de máquina.

Abstract

Marcela Musetti. **EM ingles**. Thesis (Master's). Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2022.

In Bayesian inference, issues with obtaining the analytics posterior distribution in likelihood-free 's problems are frequently. And yet, when the solution for these is made in a computational way, the challenges with the realization of inferences remains, like the precise hypothesis testing. One path to assemble the precise hypothesis testing is by the Full Bayesian Significance Test (FBST), calculating an evidence measure, the e-value. In this research, we aiming to find the posterior densities in problems of likelihood-free by the FlexCode method, instead the traditional MCMC. Then, with the densities in hands, offer a solution to the e-value calculation: through the classification methods or the computational integration of the density obtained by FlexCode. Therefore, we can solve frequent problems in Bayesian inference by transforming then in a problem that could be solve in the machine learning universes.

Keywords: FBST. *likelihood-free*. Machine Learning.

Lista de Abreviaturas

FBST	Teste de Significância Genuinamente Bayesiano (<i>Full Bayesian Significance Test</i>)
ABNT	Associação Brasileira de Normas Técnicas
URL	Localizador Uniforme de Recursos (<i>Uniform Resource Locator</i>)
IME	Instituto de Matemática e Estatística
USP	Universidade de São Paulo

Lista de Símbolos

ω	Frequência angular
ψ	Função de análise <i>wavelet</i>
Ψ	Transformada de Fourier de ψ

Lista de Figuras

4.1	Distribuição a posteriori Real vs Estimada	14
4.2	E-valor estimado em função de x segundo o método de aproximação . . .	16
4.3	Dispersão entre os e-valores estimados e o e-valor real em função de x .	16
4.4	Distribuição a posteriori Real vs Estimada	18
4.5	E-valor estimado para vários x	19
4.6	Dispersão entre os e-valores estimados e o e-valor real em função de x .	20
4.7	Distribuição a posteriori Real vs Estimada	22
4.8	Dispersão entre os e-valores estimados reduzidos e o e-valor real reduzido	23
4.9	Dispersão entre os e-valores estimados originais e o e-valor real original	24
4.10	Distribuição a posteriori de μ Real vs Estimada	26
4.11	Dispersão entre os e-valores estimados reduzidos e o e-valor real reduzido	27
4.12	Dispersão entre os e-valores estimados originais e o e-valor real original	28
4.13	Distribuição a posteriori Real vs Estimada	30
4.14	Dispersão entre os e-valores obtidos pela proposta de classificação e o e-valor real	31
4.15	Dispersão entre os e-valores obtidos pela proposta de classificação e o e-valor real	32

Lista de Tabelas

4.1	Tabela com o valor estimado da LogLoss na base de teste	15
-----	---	----

4.2	Cálculo do erro absoluto das evidências estimadas em relação ao e-valor real e seu respectivo erro padrão	15
4.3	Tabela com o valor estimado da LogLoss na base de teste	18
4.4	Cálculo do erro absoluto das evidências estimadas em relação ao e-valor real e seu respectivo erro padrão	19
4.5	Tabela com o valor estimado da LogLoss na base de teste	22
4.6	Cálculo do erro absoluto das evidências reduzidas estimadas em relação ao e-valor real reduzido e seu respectivo erro padrão	23
4.7	Tabela com o valor estimado da LogLoss na base de teste	26
4.8	Cálculo do erro absoluto das evidências reduzidas estimadas em relação ao e-valor real reduzido e seu respectivo erro padrão	26
4.9	Tabela com o valor estimado da LogLoss na base de teste	30
4.10	Cálculo do erro absoluto e seu respectivo erro padrão em relação ao e-valor real	30

Sumário

1	Introdução	1
1.1	Problemas de <i>likelihood-free</i>	1
1.2	Problemas de hipóteses precisas em Inferência Bayesiana	2
1.3	Problemas de hipóteses precisas em <i>likelihood-free</i>	2
1.4	Estrutura do trabalho	2
2	Inferência Bayesiana	5
2.1	Modelo Bayesiano	5
2.2	Teste de Significância Totalmente Bayesiano	5
3	Inferência Bayesiana e FBST em problemas de <i>likelihood-free</i>	7
3.1	Aproximação da posteriori via FlexCode	7
3.2	Calculando e-valor em problemas de <i>likelihood-free</i>	8
3.2.1	Utilizando a distribuição via <i>likelihood-free</i> para a obtenção do e-valor	8
3.2.2	Transformando a estimativa do e-valor em um problema de classificação	9
3.2.3	Estimação do e-valor em casos multivariados	10
3.3	Medidas de avaliação	11
3.3.1	Logloss	11
3.3.2	Média e erro padrão do erro absoluto	12
4	Exemplos	13
4.1	Exemplos Univariados	13
4.1.1	Modelo Normal com variância conhecida	13
4.1.2	Modelo Bernoulli	17
4.2	Exemplos Multivariados	21
4.2.1	Comparação de duas populações Poisson independentes	21
4.2.2	Distribuição Normal com variância desconhecida	25

4.2.3	Regressão Linear	29
5	Conclusão	33
Apêndices		
A		35
A.1	Cálculo do e-valor para o exemplo de duas populações Poisson	35
A.2	Cálculo do e-valor para o exemplo da Normal com variância desconhecida	36
	Referências	39
	Índice Remissivo	43

Capítulo 1

Introdução

Em Estatística, o objetivo principal, em geral, é sempre extrair a maior quantidade de informação possível dos dados sobre um determinado parâmetro desconhecido, θ . Na Inferência Bayesiana, essas incertezas são representadas por um modelo de probabilidade para θ , mais conhecido como distribuição a *priori*. Durante a análise dos dados, essa informação é combinada com as informações de uma amostra, representada pela função de verossimilhança, a fim de refinar a medida de incerteza sobre o parâmetro. Essa nova medida é chamada de distribuição a *posteriori* e representa a incerteza atualizada sobre o parâmetro de interesse. A partir da *posteriori*, conseguimos obter estimativas, pontuais e intervalares, e testar hipóteses para realizar inferências sobre θ , assim como na abordagem frequentista.

1.1 Problemas de *likelihood-free*

Mesmo com a importância da função de verossimilhança para o desenvolvimento tanto dos métodos bayesianos como dos frequentistas, em alguns casos, como, por exemplo, em áreas da ciência, física e na engenharia (FELDMAN e COUSINS, 1998, CRANMER *et al.*, 2015, COUSINS, 2018, MARIN *et al.*, 2012, SISSON, FAN e M. BEAUMONT, 2018, IZBICKI, A. LEE *et al.*, 2014), a função de verossimilhança proposta pode ser muito complexa, o que é muito diferente de um problema não paramétrico, pois mesmo sendo possível supor uma fórmula paramétrica para o problema terão muitos desafios analíticos e/ou computacionais. Sendo possível gerar esses dados através de algum simulador, um jeito de obter a função de verossimilhança é utilizando métodos computacionais, essa técnica é denominada por *likelihood-free*.

Com isso, vêm sendo difundidas formas de obter, sob a abordagem bayesiana, a distribuição a *posteriori* utilizando métodos computacionais sem fazer o uso da função de verossimilhança exata. A mais conhecida é baseada no método denominado ABC (M. A. BEAUMONT *et al.*, 2002) e, mais recentemente, técnicas de aprendizado de máquina, tomando dados simulados como dados de treinamento, também tem sido empregadas nessa área. As densidades geradas computacionalmente através desses métodos são as densidades que chamamos de *likelihood-free* (SISSON e FAN, 2011). Nesse trabalho, para estimar a densidade a *posteriori* e testar hipóteses, usaremos a abordagem de aprendizado de máquina, mais

precisamente o FlexCode (IZBICKI, A. B. LEE e POSPISIL, 2019), por ter mais benefícios comparativamente às técnicas usuais.

1.2 Problemas de hipóteses precisas em Inferência Bayesiana

Com a distribuição a posteriori calculada, independentemente do método escolhido para sua obtenção, o próximo passo no estudo bayesiano é a realização de estimativas, como o intervalo de credibilidade e o teste de hipóteses. Um problema bastante importante nessa área é o problema de teste de hipóteses precisas: quando θ é absolutamente contínuo e a medida do conjunto que caracteriza a hipótese nula de interesse é zero, o uso dos testes mais comuns, como os baseados na probabilidade a posteriori da hipótese, é inviável (BERGER e DELAMPADY, 1987). Uma alternativa para os testes de hipóteses precisas é o Teste de Significância Totalmente Bayesiano - FBST (PEREIRA e STERN, 1999), baseado em uma medida de evidência da hipótese nula, denominada de e-valor, ou, equivalentemente, em regiões HPD.

1.3 Problemas de hipóteses precisas em *likelihood-free*

Caso a densidade a posteriori seja determinada através de métodos *likelihood-free*, a obtenção do e-valor analiticamente não é possível e, por isso, nesse trabalho, temos como objetivo propor uma nova forma de fazer esse cálculo computacionalmente de modo satisfatório. Propomos duas ideias para esse cálculo: 1) obter a medida de evidência via a densidade do FlexCode ou 2) transformar o cálculo do e-valor em um problema de classificação, a fim de resolvê-lo através de modelos como Regressão Logística (HOSMER JR *et al.*, 2013), Florestas Aleatórias (BREIMAN, 2001), XGBoost (CHEN e GUESTRIN, 2016).

Resumidamente, a ideia dessa pesquisa é responder dois problemas da inferência bayesiana: obtenção da posteriori e teste de hipótese precisa em problemas de densidade *likelihood-free* de forma conjunta em exemplos variados. Ilustramos a proposta através de exemplos uniparamétricos, sendo eles o caso Normal com variância conhecida e o caso Bernoulli, e exemplos com mais de um parâmetro, como o caso Normal com média e variância desconhecidas, a comparação de duas populações Poisson e o modelo de Regressão Linear Simples.

Observe que nesses exemplos, temos a distribuição a *posteriori* e o cálculo do e-valor de forma analítica e, por conta disso, conseguimos comparar a precisão dos métodos propostos. Utilizamos algumas medidas para fazer tais comparações.

1.4 Estrutura do trabalho

No Capítulo 2, introduziremos conceitos bayesianos básicos que foram utilizados ao longo dessa dissertação. No Capítulo 3, teremos o detalhamento do método proposto,

que incluirá a obtenção da distribuição a *posteriori* em problemas de *likelihood-free*, via FlexCode, e a implementação do cálculo do e-valor para esse tipo de situação. No Capítulo 4, teremos os exemplos de utilização dessas técnicas em diversas distribuições conhecidas. Encerramos discutindo as conclusões e possíveis extensões desse trabalho no Capítulo 5.

Capítulo 2

Inferência Bayesiana

2.1 Modelo Bayesiano

O modelo bayesiano é uma forma de traduzirmos matematicamente a nossa incerteza sobre os valores observados de variáveis aleatórias e os parâmetros de interesse desconhecidos. De uma maneira geral, temos um vetor aleatório, X , o qual podemos observar, e um parâmetro desconhecido, θ , com uma distribuição de probabilidade associada, $f(\theta)$, denominada distribuição *a priori* para θ . Temos ainda a densidade condicional de X dado θ , $f(x|\theta)$. Essa distribuição é denominada distribuição amostral. Além disso, fixando x , $f(x|\cdot)$ como função de θ é a função de verossimilhança para θ . Após a observação dos dados, idealmente a atualização da incerteza sobre θ deve combinar a *priori* com a informação obtida na distribuição dos dados, obtendo a distribuição *a posteriori*, $f(\theta|x)$. Utilizando o Teorema de Bayes, conseguimos descrever essa atualização na distribuição de θ da seguinte maneira:

$$f(\theta|x) = \frac{f(x|\theta)f(\theta)}{f(x)} \propto f(x|\theta)f(\theta). \quad (2.1)$$

Note que para essa metodologia, é imprescindível o conhecimento da função de verossimilhança. Porém, como já mencionado no Capítulo 1, em alguns casos a função de verossimilhança pode ser difícil de ser trabalhada por conta da sua forma analítica ser muito complexa. Por isso, a ideia dessa pesquisa é trazer uma proposta de obtenção da distribuição *a posteriori* sem a utilização da função de verossimilhança, através de técnicas computacionais descritas no próximo capítulo.

2.2 Teste de Significância Totalmente Bayesiano

O objeto do Teste de Significância Totalmente Bayesiano (FBST) (PEREIRA e STERN, 1999) é testar hipóteses precisas (de medida zero) a partir do cálculo de uma medida de evidência para a hipótese nula. Descreveremos, a seguir, a formulação de tal medida para uma hipótese simples no caso unidimensional. Para testar $H_0 : \theta = \theta_0$ vs $H_1 : \theta \neq \theta_0$,

$\theta_0 \in \mathbb{R}$, após a observação da amostra \mathbf{x} , ontem-se a densidade a posteriori de θ e considera-se a região *Highest Posterior Density*, HPD (YAO e LINDSAY (2009)) tangente à hipótese nula

$$S_x = \{\theta \in \Theta : f(\theta|\mathbf{X} = \mathbf{x}) > f(\theta_0|\mathbf{X} = \mathbf{x})\}.$$

Define-se então a medida de evidência de Pereira - Stern por

$$\hat{e}(\Theta_0, \mathbf{x}) = 1 - \mathbb{P}(\theta \in S_x | \mathbf{X} = \mathbf{x}).$$

Em alguns casos, a obtenção de uma expressão para o e-valor é difícil ou impossível de ser determinada analiticamente. Nesses casos, uma solução computacional é usar o método de Monte Carlo para obter uma estimativa para o e-valor. O método de Monte Carlo consiste em aproximar integrais computacionalmente (BIERNACKI *et al.* (2010)). Especificamente, para o e-valor, podemos proceder do seguinte modo:

Defina

$$g(\theta, \mathbf{x}) = \begin{cases} 1, & \text{se } f(\theta|\mathbf{X} = \mathbf{x}) > f(\theta_0|\mathbf{x}) \\ 0, & \text{c.c.} \end{cases}$$

Observe que

$$\mathbb{P}(\theta \in S_x | \mathbf{X} = \mathbf{x}) = E_x(g(\theta, \mathbf{x})) = \int_{\theta \in \Theta} g(\theta, \mathbf{x}) f(\theta|\mathbf{x}) dx.$$

Amostrando $\theta_1, \dots, \theta_B$ da distribuição a *posteriori* de θ dado \mathbf{x} , temos que

$$\mathbb{P}(\theta \in S_x | \mathbf{X} = \mathbf{x}) \approx \frac{\sum_{i=1}^B g(\theta_i, \mathbf{x})}{B}, \text{ para } B \text{ grande.}$$

Logo, o e-valor estimado via Monte Carlo, utilizando a distribuição a posteriori analítica, é dado por:

$$e_valor \approx 1 - \frac{\sum_{i=1}^B g(\theta_i, \mathbf{x})}{B}.$$

No caso de hipóteses precisas compostas em espaços paramétricos de dimensão maior que um, o raciocínio é análogo (MADRUGA *et al.*, 2003): é apenas necessário a substituição de $f(\theta_0|\mathbf{x})$ por $\sup_{\theta \in \Theta_0} f(\theta|\mathbf{x})$ na definição de S_x . No Capítulo 3, abordaremos o cenário em que a função de verossimilhança $f(\mathbf{x}|\theta)$ não é analítica e a Equação 2.1 não pode ser efetuada.

Capítulo 3

Inferência Bayesiana e FBST em problemas de *likelihood-free*

3.1 Aproximação da posteriori via FlexCode

Para a obtenção da posteriori aproximada, suponha que seja possível amostrar da distribuição *a priori* θ e do modelo estatístico $f(\mathbf{x}|\theta)$ (ou da distribuição conjunta de (θ, \mathbf{x})), capturando os pares $(\theta_1, \mathbf{X}_1), \dots, (\theta_B, \mathbf{X}_B)$. Com isso, conseguimos um conjunto de dados para estimar a distribuição *a posteriori* de θ .

O algoritmo para a geração desses dados é apresentado a seguir:

Algorithm 1: Gerando a base de dados $\{(\theta_1, \mathbf{x}_1), \dots, (\theta_B, \mathbf{x}_B)\}$

Entrada: Distribuições Priori, Verossimilhança

Saída: Base de dados para a obtenção da *posteriori*

Para $i = 1, \dots, B$, faça:

- 1) Gerar i -ésima observação, θ_i , da distribuição suposta *a priori*, $\theta \sim f(\theta)$
 - 2) Gerar \mathbf{x}_i da distribuição de $\mathbf{X}|\theta = \theta_i \sim f(\mathbf{x}|\theta = \theta_i)$, utilizando o θ_i obtido no passo 1
-

Com os pares gerados em mãos, a ideia agora é estimar $f(\theta|\mathbf{x})$, que nada mais é que uma densidade condicional. É possível realizar essa aproximação de duas maneiras: estimando diretamente a densidade *a posteriori* (IZBICKI, A. B. LEE e POSPISIL (2019)) ou usando um classificador probabilístico com foco na resolução desse problema (Niccolò DALMASSO *et al.*, 2020, Niccolò DALMASSO *et al.*, 2021). Nesta dissertação, utilizaremos a primeira maneira com o enfoque no FlexCode (IZBICKI e A. B. LEE, 2017), descrita a seguir.

Sejam $(\mathbf{X}_1, \theta_1), \dots, (\mathbf{X}_n, \theta_n)$ i.i.d., em que $\mathbf{x} \in \mathbb{R}^d$. O objetivo desse método não é prever a função de regressão $E[\theta|\mathbf{x}]$, como em métodos de predições usuais, mas sim estimar a função densidade $f(\theta|\mathbf{x})$. A ideia chave do FlexCode é expandir $f(\theta|\mathbf{x})$ em uma base ortonormal $\phi_i(\theta)$ da seguinte forma

$$f(\theta|\mathbf{x}) = \sum_{i \in \mathbb{N}} \beta_i(\mathbf{x}) \phi_i(\theta).$$

Por causa da ortogonalidade das funções ϕ_i , temos que $\beta_i(\mathbf{x}) = E[\phi_i(\theta)|\mathbf{x}]$. Dessa forma, os β_i 's são estimados via regressão

$$\hat{\beta}_i(\mathbf{x}) = \hat{E}[\phi_i(\theta)|\mathbf{x}],$$

isto é, regredindo cada $\phi_i(\theta)$ em \mathbf{x} . Note que diferentes métodos de regressão levam a resultados diferentes na estimativa dos β 's via FlexCode. Os métodos considerados neste trabalho serão KNN (GUO *et al.*, 2003), para casos unidimensionais de \mathbf{X} , e Florestas Aleatórias (CUTLER *et al.*, 2012) nos demais casos.

Por fim, o estimador FlexCode para $f(\theta|\mathbf{x})$ é definido por

$$\hat{f}(\theta|\mathbf{x}) = \sum_{i=1}^I \hat{\beta}_i(\mathbf{x})\phi_i(\theta),$$

onde o número de termos I na série de expansão está associado com o equilíbrio entre viés e variância e é escolhido através de *Data Splitting* (JAMES *et al.* (2013)).

Note que não é feita nenhuma restrição para a distribuição dos dados para chegar na estimativa da densidade condicional. Por isso, o método pode ser utilizado para qualquer distribuição. Além disso, observe que a proposta descrita é desenvolvida para parâmetros unidimensionais. Com isso, para casos multidimensionais, uma alternativa de solução é realizar alguma transformação do parâmetro e/ou utilizar a distribuição marginal dos parâmetros para a solução do problema em cada caso.

3.2 Calculando e-valor em problemas de *likelihood-free*

Da mesma forma que em alguns casos é difícil o cálculo analítico da distribuição a posteriori, a partir do momento que temos sua estimativa via *likelihood-free* também se torna complicado a obtenção de todas as inferências para θ sob essa distribuição. Nessa dissertação, estudaremos como obter a medida de evidência de Pereira-Stern a partir da distribuição a posteriori obtida via FlexCode de duas maneiras diferentes, sendo possível generalizar tais procedimentos para qualquer método de *likelihood-free*. E, após isso, avaliaremos, através de métricas descritas nessa seção, a eficiência de cada método proposto.

3.2.1 Utilizando a distribuição via *likelihood-free* para a obtenção do e-valor

O primeiro método proposto para estimar a medida de evidência tem uma ideia bastante intuitiva: substituir $f(\theta|\mathbf{x})$ por $\hat{f}(\theta|\mathbf{x})$ no cálculo da estimativa do e-valor. Na seção 2.2, descrevemos como seria o cálculo do e-valor via Monte Carlo (METROPOLIS e ULAM, 1949) com a posteriori encontrada analiticamente $f(\theta|\mathbf{X} = \mathbf{x})$. Note que para calcular o e-valor via Monte Carlo, nós não precisamos fazer nenhum cálculo analítico com a posteriori exata por ser uma aproximação feita através de recursos computacionais. Por conta disso,

nós conseguimos substituir, nesse procedimento de aproximação, a posteriori exata pela posteriori obtida através de *likelihood-free*, ou seja, substituir $f(\theta|X = \mathbf{x})$ por $\hat{f}(\theta|X = \mathbf{x})$. Nesse caso, o e-valor estimado é definido por:

$$\hat{e}(\Theta_0, \mathbf{x}) = 1 - \hat{\mathbb{P}}(\theta \in \hat{S}_x | X = \mathbf{x}),$$

onde

$$\hat{S}_x = \{\theta \in \Theta : \hat{f}(\theta|X = \mathbf{x}) > \hat{f}(\theta^*|X = \mathbf{x})\},$$

com $\theta^* = \operatorname{argmax}_{\theta \in \Theta} \hat{f}(\theta|X = \mathbf{x})$ e $\hat{\mathbb{P}}$ a medida de probabilidade a posteriori estimada.

Uma vez que tenhamos uma aproximação boa da posteriori analítica, é razoável supor, sob certas condições, que teremos também uma boa aproximação da medida de evidência através do procedimento acima.

Nesse trabalho, avaliaremos se utilizar a posteriori estimada pelo FlexCode e então estimar a medida de evidência via Monte Carlo terá um valor próximo daquele calculado pela posteriori analítica. Além disso, outra forma de calcular o e-valor é sob a perspectiva de um problema de classificação.

3.2.2 Transformando a estimativa do e-valor em um problema de classificação

Para determinar a medida de evidência contra a hipótese nula $H_0 : \theta = \theta_0$ versus $H_1 : \theta \neq \theta_0$, $\theta_0 \in \Theta$ denominada de e-valor em situações em que a posteriori exata não é tratável analiticamente, podemos usar a aproximação de Monte Carlo baseada em $\hat{f}(\theta|\mathbf{x})$, a estimativa da distribuição à posteriori. Porém, além da obtenção desse cálculo via Monte Carlo, este trabalho tem como objetivo propor uma nova forma de calcular a medida de evidência desejada sob o enfoque de um problema de classificação, eliminando a complexidade de calcular essa integral com a densidade estimada via FlexCode. Para isso, vamos definir, para $i = 1, \dots, B$, uma nova variável dada por

$$Z_i = \begin{cases} 1, & \text{se } \hat{f}(\theta_i|X = \mathbf{x}_i) > \hat{f}(\theta_0|X = \mathbf{x}_i), i = 1, \dots, B. \\ 0, & \text{caso contrário} \end{cases} \quad (3.1)$$

Considerando agora os pontos $(X_1, Z_1), \dots, (X_B, Z_B)$, a estimação de $\hat{\mathbb{P}}(\theta \in \hat{S}_x | X = \mathbf{x}) = E(Z|\mathbf{x})$ é abordado como um problema de classificação de Z em X .

Logo, para cada i , θ_i é obtido da *priori*, $\mathbf{x}_i|\theta_i$ é obtido de $f(\mathbf{x}|\theta_i)$, estimada via *likelihood-free*, e valores de $\hat{f}(\theta_i|X = \mathbf{x}_i)$ e $\hat{f}(\theta_0|X = \mathbf{x}_i)$ são calculados de modo a determinar z_i . Replicando esse procedimento M vezes temos o conjunto de dados $\{(\mathbf{x}_1, z_1), \dots, (\mathbf{x}_M, z_M)\}$

para o problema de classificação. Temos então o seguinte algoritmo:

Algorithm 2: Gerando a base de dados $\{(\mathbf{x}_1, z_1), \dots, (\mathbf{x}_M, z_M)\}$

Entrada: Priori, Verossimilhança, Posteriori estimada, θ_0

Saída: Base de dados para a obtenção da medida de evidência

Para $i = 1, \dots, B$, faça:

- 1) Gerar i -ésima observação, θ_i , da distribuição suposta para a priori, $\theta \sim f(\theta)$
 - 2) Gerar \mathbf{x}_i utilizando o θ_i obtido no passo 1, ou seja, obter \mathbf{x}_i da distribuição $X|\theta = \theta_i \sim f(\mathbf{x}|\theta_i)$
 - 3) Construir $z_i = \begin{cases} 1, & \text{se } \hat{f}(\theta_i|X = \mathbf{x}_i) > \hat{f}(\theta_0|X = \mathbf{x}_i) \\ 0, & \text{c.c.} \end{cases}$
 - 4) Manter (z_i, \mathbf{x}_i)
-

Com os dados $\{(\mathbf{x}_1, z_1), \dots, (\mathbf{x}_M, z_M)\}$ em mãos, temos o problema de determinação do e-valor simplificado para um problema de classificação, cujo objetivo é estimar $E(Z|X)$, sendo possível utilizar qualquer modelo de Aprendizado de Máquina para a sua solução. Nesse trabalho, utilizamos Regressão Logística, XGBoost (CHEN e GUESTRIN, 2016) e Florestas Aleatórias.

3.2.3 Estimação do e-valor em casos multivariados

Em casos de espaços paramétricos de dimensão maior que 1, a ideia explorada nesse trabalho consiste em obter a medida de evidência para alguma transformação do parâmetro original (por exemplo, uma marginal unidimensional do parâmetro) uma vez que o FlexCode é um estimador que funciona muito bem para densidades com apenas uma dimensão.

Dessa forma, obtemos um e-valor "marginal" ou reduzido para hipótese estatística, gerado por *likelihood-free* que, por sua vez, não é exatamente igual ao e-valor obtido a partir da densidade conjunta. Por isso, precisamos aplicar uma transformação ao e-valor reduzido, $e(\Theta_0, \mathbf{x})_{\text{reduzido}}$, para obtermos uma aproximação do e-valor original do teste, $e(\Theta_0, \mathbf{x})_{\text{original}}$, que nada mais é que o e-valor $\hat{e}(\Theta_0, \mathbf{x})$ definido na seção 3.2.

Para isso, utilizaremos uma relação assintótica entre o e-valor e o p-valor descrito em (DINIZ *et al.*, 2012). Essa relação estabelece que quando $\dim(\Theta) = d$, $\dim(\Theta_0) = d_0$, $d > d_0$, e F_k é a distribuição acumulada de modelo χ^2 com k graus de liberdade, vale que

$$e(\Theta_0, \mathbf{x})_{\text{original}} \approx 1 - F_d(F_{d-d_0}^{-1}(1 - p_{\text{valor}})). \quad (3.2)$$

Sendo $e(\Theta_0, \mathbf{x})_{\text{reduzido}}$ o e-valor que calculamos a partir da *posteriori* da transformação (marginal) unidimensional de θ , temos que $d = 1$ e $d_0 = 0$. Logo, usando novamente a relação em (DINIZ *et al.*, 2012)

$$e(\Theta_0, \mathbf{x})_{\text{reduzido}} \approx 1 - F_1(F_{1-0}^{-1}(1 - p_{\text{valor}})) \approx 1 - F_1(F_1^{-1}(1 - p_{\text{valor}})) \approx 1 - 1 + p_{\text{valor}} \approx p_{\text{valor}}. \quad (3.3)$$

Dessa forma, utilizando as duas equações (3.2) e (3.3), temos:

$$e(\Theta_0, \mathbf{x})_{original} \approx 1 - F_d(F_{d-d_0}^{-1}(1 - e(\Theta_0, \mathbf{x})_{reduzido})).$$

Assim, conseguimos uma transformação que relacione, aproximadamente, o $e(\Theta_0, \mathbf{x})_{reduzido}$ com o $e(\Theta_0, \mathbf{x})_{original}$.

3.3 Medidas de avaliação

Podemos analisar a eficácia do ajuste da distribuição a posteriori via FlexCode graficamente e como utilizaremos exemplos nos quais conseguimos obter a *posteriori* computacionalmente e analiticamente, vamos comparar a posteriori gerada via FlexCode com a posteriori exata.

Para analisar o desempenho dos cálculos dos e-valores estimados via Aprendizado de Máquina no problema de classificação, utilizaremos a métrica de LogLoss, que será explicada ao longo dessa seção.

Entretanto, também teremos e-valores obtidos a partir da integração da densidade gerada via FlexCode e, por conta disso, utilizaremos mais uma métrica para a comparação, o erro absoluto, com o objetivo de avaliar todas as estimativas propostas nesse trabalho com o e-valor exato.

3.3.1 Logloss

A Logloss é uma métrica cujo objetivo é analisar a qualidade de uma predição de classificação binária (BISHOP, 2006). Seja $z \in (0,1)$ retirado da base de teste do conjunto de dados definido pelo Algoritmo 2 e $\hat{e}(\Theta_0, \mathbf{x})$ a estimativa do e-valor obtida pelas técnicas de classificação. A métrica Logloss(LL) é definida por:

$$LL = \frac{1}{M} \sum_{i=1}^M ll_i$$

onde,

$$ll_i = - \{ z_i \log(\hat{e}_i(\Theta_0, \mathbf{x})) + (1 - z_i) \log(1 - \hat{e}_i(\Theta_0, \mathbf{x})) \},$$

para $i = 1, \dots, M$.

Pela formula de LL, nós conseguimos avaliar que se $z_i = 1$, o valor do custo dessa função será $-\log(\hat{e}_i(\Theta_0, \mathbf{x}))$, que penaliza mais intensamente estimativas menores de e-valor. Caso contrário, se $z_i = 0$, o valor do custo dessa função será $-\log(1 - \hat{e}_i(\Theta_0, \mathbf{x}))$, que penaliza estimativas maiores do e-valor. Com isso, conseguimos penalizar casos em que a probabilidade de classificação não condiz com o valor observado, e não penalizar quando condiz.

Efetivamente, dado que temos os valores que queremos prever z_i , o que a função de

custo vai avaliar é se a probabilidade obtida na classificação é de boa qualidade segundo certo critério, retornando valores altos para previsões ruins e valores baixos para previsões boas.

3.3.2 Média e erro padrão do erro absoluto

Seja, $e(\Theta_0, \mathbf{x})$ o e-valor que queremos estimar via as técnicas apresentadas no trabalho e $\hat{e}(\Theta_0, \mathbf{x})$ a estimativa dessa medida. O erro absoluto é definido, para cada observação gerada, pela diferença entre o e-valor analítico e o e-valor estimado através das metodologias propostas, isto é,

$$ea_i = |e_i(\Theta_0, \mathbf{x}) - \hat{e}_i(\Theta_0, \mathbf{x})|,$$

para cada $i = 1, \dots, M$.

Com isso, conseguimos avaliar o quanto a estimativa obtida está próxima do e-valor original em média:

$$\text{média}(ea) = \frac{\sum_{i=1}^M ea_i}{M} = \bar{ea}.$$

A dispersão dos erros absolutos é dada pelo erro padrão

$$\text{erro_padrão}(ea) = \frac{\sqrt{\frac{\sum_i^M (ea_i - \bar{ea})^2}{M}}}{\sqrt{M}},$$

que servirão de medidas para comparação dos métodos no próximo capítulo.

Capítulo 4

Exemplos

Para avaliarmos as metodologias propostas no Capítulo 3, utilizaremos variados exemplos conhecidos, univariados e multivariados, com suas mais diferentes particularidades.

4.1 Exemplos Univariados

4.1.1 Modelo Normal com variância conhecida

O modelo Normal é um dos mais conhecidos em Estatística. Sob a perspectiva Bayesiana, destaca-se a propriedade denominada conjugação para diversos cenários envolvendo o modelo Normal. A conjugação acontece quando a priori e a posteriori pertencem a uma mesma específica família de distribuição (PRATT *et al.*, 1995). Uma situação bem simples de conjugação é a seguinte:

Suponha que

$$\mu \sim N(\mu_0, \sigma_0^2), \mu_0 \in \mathbb{R}, \sigma_0^2 > 0, \quad (4.1)$$

e que

$$X|\mu \sim N(\mu, \sigma^2), \text{ com } \sigma^2 > 0 \text{ conhecido.} \quad (4.2)$$

Nesse caso,

$$\mu|X = x \sim \text{Normal}(\mu_x, \sigma_x^2),$$

em que

- $\mu_x = \frac{\sigma_0^2 x + \sigma^2 \mu_0}{\sigma_0^2 + \sigma^2}$
- $\sigma_x^2 = \left(\frac{1}{\sigma_0^2} + \frac{1}{\sigma^2} \right)^{-1}$.

Com a metodologia apresentada em 3.1, obtemos o conjunto de dados, $(\mu_1, X_1), \dots, (\mu_B, X_B)$, através do Algoritmo 1, considerando a priori Normal como em (4.1) e a distribuição

amostral como em (4.2). Com os dados, realizamos um data splitting (PICARD e BERK (1990)) para a modelagem de 70% de dados de treinamento, 20% de dados de validação e 10% de dados de teste fixando uma amostra de tamanho $B = 10.000$. Uma observação importante é que o algoritmo retorna um conjunto de regras, no qual, para a obtenção da distribuição a posteriori, utiliza-se a função *predict* do software R, declarando um valor amostral, x , para a obtenção de $\hat{f}(\mu|X = x)$, ou seja, o resultado da posteriori estimada está condicionado a um valor específico de x .

Como o estudo trata de um caso Normal, pela propriedade de conjugação conhecemos a distribuição a posteriori de μ exata. Dessa forma, os resultados obtidos pelo método proposto podem ser comparados com a distribuição a posteriori real. Considerando $\mu_0 = 0$, $\sigma^2 = \sigma_0^2 = 1$, temos que $\mu|X = x \sim N(\frac{x}{2}, \frac{1}{2})$, em que x é o mesmo valor utilizado para se obter $\hat{f}(\mu|X = x)$ através do FlexCode.

Na Figura 4.1, temos as curvas real e estimada para a posteriori de μ considerando alguns valores de x . Note que elas estão muito próximas, indicando que a posteriori estimada via FlexCode é muito próxima da distribuição a posteriori exata. Além disso, observe que quanto mais próximos os valores de x são de μ_0 , a aproximação parece ser melhor.

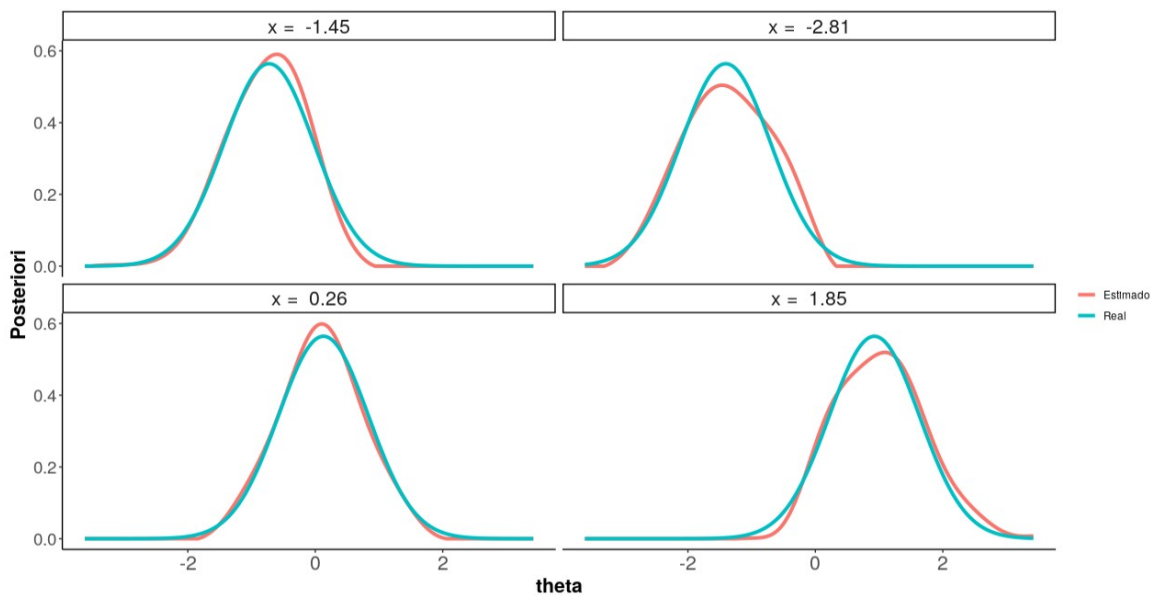


Figura 4.1: Distribuição a posteriori Real vs Estimada

Agora, para testar $H_0 : \mu = 0,5$ vs $H_1 : \mu \neq 0,5$ utilizando o FBST, temos duas abordagens possíveis. A primeira é a alternativa de estimar a evidência de Pereira-Stern utilizando a integração computacional com a distribuição estimada via FlexCode, conforme detalhada na Seção 3.2.1. A segunda alternativa é transformar o cálculo do e-valor em um problema de classificação, como na Seção 3.2.2. Compararemos os resultados dessas duas alternativas com o valor exato da evidência que, no caso Normal, considerado é dado por $Ev(\Theta_0; \mathbf{x}) = 2[1 - \Phi(|x - 1|)]$, com $\mu_0 = 0,5$.

Obtido o e-valor através do modelo de classificação, utilizando uma amostra de tamanho

100.000, temos a LogLoss na base de teste (10%) de todos os modelos testados na Tabela 4.1. Nota-se que o XGBoost e a Regressão Logística apresentaram um desempenho muito melhor do que a Floresta Aleatória quanto à logloss.

Método	LogLoss
Logística	1
Floresta	7,6
XGBoost	1,26

Tabela 4.1: Tabela com o valor estimado da LogLoss na base de teste

Já segundo o erro absoluto apresentado na Tabela 4.2, o modelo de classificação com resultado mais próximo do e-valor real é o de XGBoost. Outro ponto importante a se destacar é o desempenho do cálculo do e-valor via FlexCode (descrito na seção 3.2.1): essa técnica de estimativa apresentou desempenho superior a todos os modelos de classificação propostos.

Método	Erro absoluto médio	Erro padrão
XGBoost	0,0723	0,0021
Logística	0,1734	0,0046
Floresta	0,2329	0,0059
Flexcode	0,0437	0,0015

Tabela 4.2: Cálculo do erro absoluto das evidências estimadas em relação ao e-valor real e seu respectivo erro padrão

Na Figura 4.2, comparamos as estimativas do e-valor para vários valores de x gerados da priori com o e-valor analítico. Podemos notar que, mesmo não tendo um bom desempenho na medida LogLoss, a Floresta parece estar mais próximo do valor real da medida de evidência do que a Logística por ter o comportamento simétrico em torno de 0,5 similar ao do e-valor analítico. Além disso, observe que o e-valor Real-Analítico e o e-valor Real-Computacional, obtido pela integração de Monte Carlo da função a posteriori real, são muito próximos. Por isso, para casos em que o e-valor é muito difícil de se calcular analiticamente, olharemos apenas o e-valor estimado pelo método de Monte Carlo (Real - Computacional).

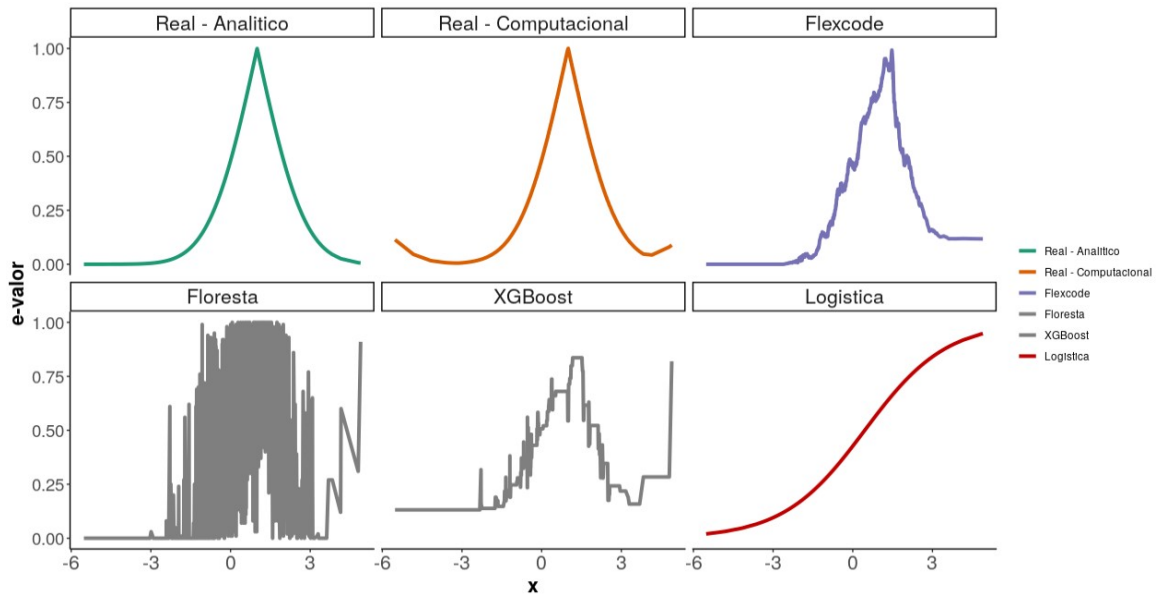


Figura 4.2: E-valor estimado em função de x segundo o método de aproximação

Podemos ver na Figura 4.3 que os e-valores obtidos via XGBoost estão fortemente correlacionados com os e-valores reais (correlação = 0,949), indicando que dentre as três propostas de classificação é a que apresenta melhor desempenho. Porém, dentre todos os métodos estudados, a aproximação do e-valor obtido pela integração da densidade obtida via FlexCode é o melhor (correlação = 0,976). Na diagonal principal, conseguimos ver a distribuição dos e-valores estimados em função de x .

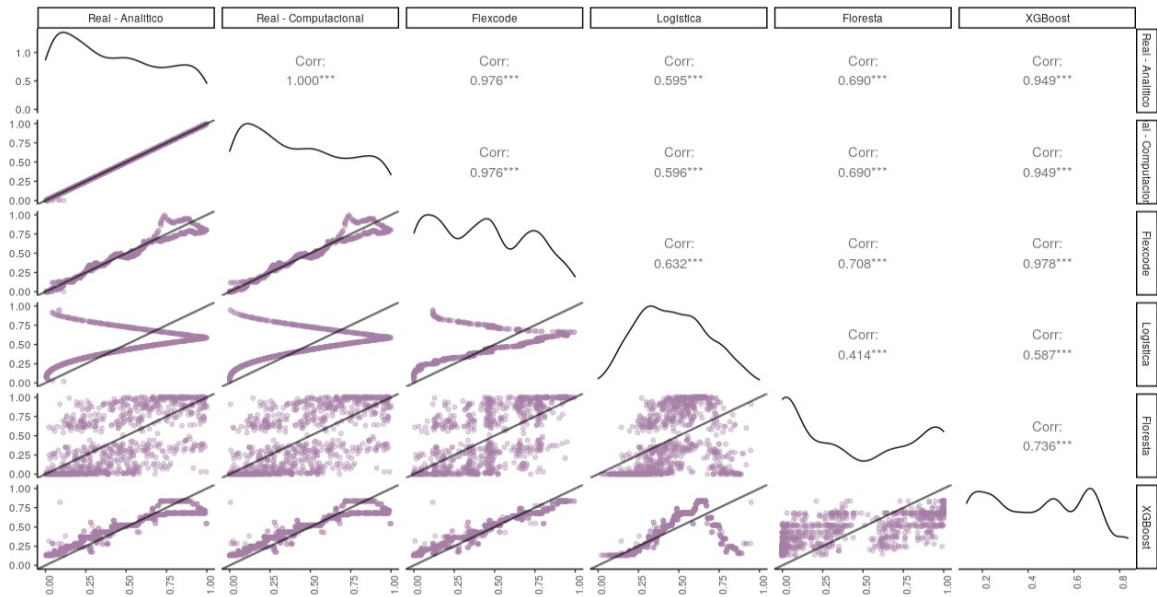


Figura 4.3: Dispersão entre os e-valores estimados e o e-valor real em função de x

4.1.2 Modelo Bernoulli

Vimos com os resultados obtidos através de simulação para a distribuição Normal que os métodos propostos na obtenção da *posteriori* e do e-valor apresentaram um bom desempenho para esse caso. E agora, será que conseguimos expandir satisfatoriamente os métodos para outras situações, muitas vezes mais complexas do que o cenário proposto na seção anterior? Para responder essa pergunta, examinamos os métodos em novos exemplos avaliando os resultados e destacando os pontos positivos e negativos.

Vamos começar por um caso ainda uniparamétricos e com propriedade de conjugação, porém com *posteriors* assimétricas e com dados discretos. Suponha que,

$$\theta \sim \text{Beta}(a, b), a, b > 0,$$

e que

$$X|\theta \sim \text{Binomial}(n, \theta), n \in \mathbb{N}^*$$

Temos, pela propriedade de conjugação (DEGROOT, 2005), que

$$\theta|X = x \sim \text{Beta}(a + x, b + n - x).$$

Utilizando o algoritmo FlexCode implementado no software R, fixando uma amostra de tamanho $n = 200$, com $a = 1$ e $b = 2$ na distribuição a priori, temos as seguintes curvas estimadas da *posteriori* para alguns valores de x na Figura 4.4. Note que as distribuições exatas e estimadas estão bem próximas. Um ponto a se destacar é que para $x = 6$ a distribuição real parece destoar um pouco mais da estimada do que para outros valores de x .

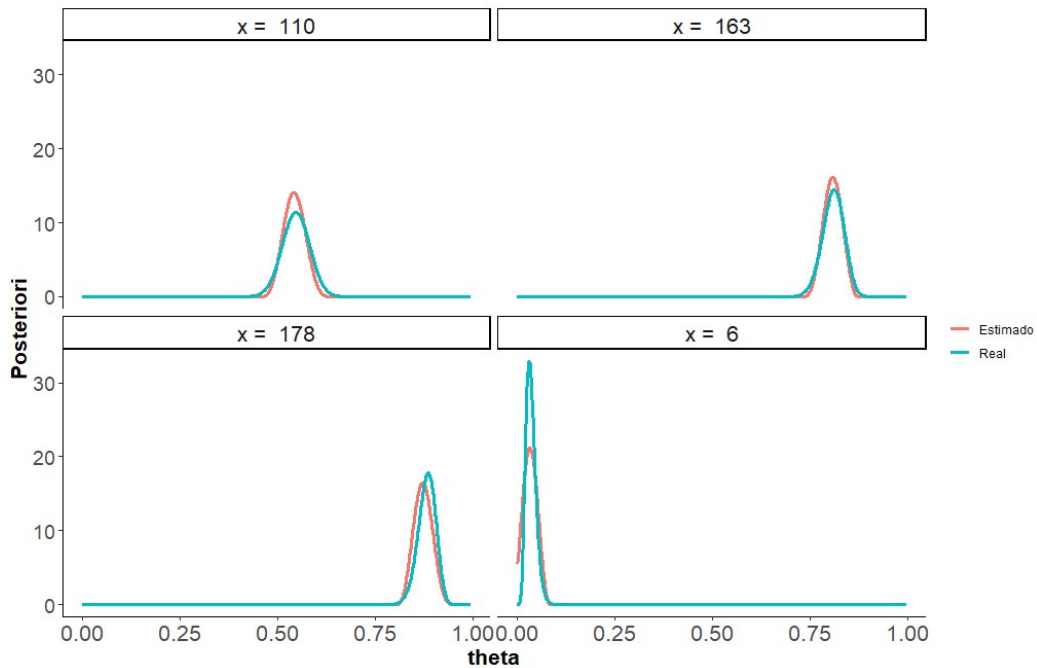


Figura 4.4: Distribuição a posteriori Real vs Estimada

Nesse exemplo, vamos testar $H_0 : \theta = 0,85$ vs $H_1 : \theta \neq 0,85$ através do FBST usando o método de cálculo da medida de evidência proposto neste trabalho. Vamos comparar o resultado com o valor obtido via integração computacional da densidade real da posteriori. Sob a abordagem de problema de classificação, conseguimos comparar a qualidade das estimativas da evidência através da LogLoss na base de teste. Na Tabela 4.3, segundo essa métrica, a Logística foi o que apresentou o melhor desempenho. Porém, a avaliação segundo essa métrica no caso da distribuição Normal não coincide com os resultados apresentados graficamente, pois o que apresentou melhor desempenho naquele exemplo via LogLoss foi a Regressão Logística e, no entanto, na análise gráfica, conseguimos identificar que essa foi a técnica de pior desempenho.

Método	LogLoss
Logística	0,92
Floresta	24,81
XGBoost	1,11

Tabela 4.3: Tabela com o valor estimado da LogLoss na base de teste

Na Tabela 4.4, apresentamos o erro absoluto. Diferente da LogLoss, o modelo de classificação que obteve o melhor desempenho segundo o erro absoluto foi o XGBoost. Entretanto, a estimativa via FlexCode foi a técnica com menor erro absoluto médio segundo essa métrica.

Método	Erro absoluto médio	Erro padrão
XGBoost	0,1413	0,0027
Logística	0,2599	0,0060
Floresta	0,1752	0,0051
Flexcode	0,0890	0,0022

Tabela 4.4: Cálculo do erro absoluto das evidências estimadas em relação ao e-valor real e seu respectivo erro padrão

Na Figura 4.5, note que a aproximação por FlexCode foi a que mais se assemelhou ao e-valor real calculado computacionalmente como função de x , confirmando o desempenho medido pelo erro absoluto na Tabela 4.4. Já o XGBoost, que também apresentou resultados bons tanto na LogLoss quanto no erro absoluto, apresenta estimativas para o e-valor que preservam o formato da curva do e-valor real em função de x . Porém, para valores de x menores que 165 e maiores que 180, a medida de evidência estimada via XGBoost apresenta um valor muito acima daquele calculado a partir da medida de evidência real, onde a primeira possui valores em torno de 0,25, no intervalo de x destacado, e a segunda apresenta valores próximos de 0.

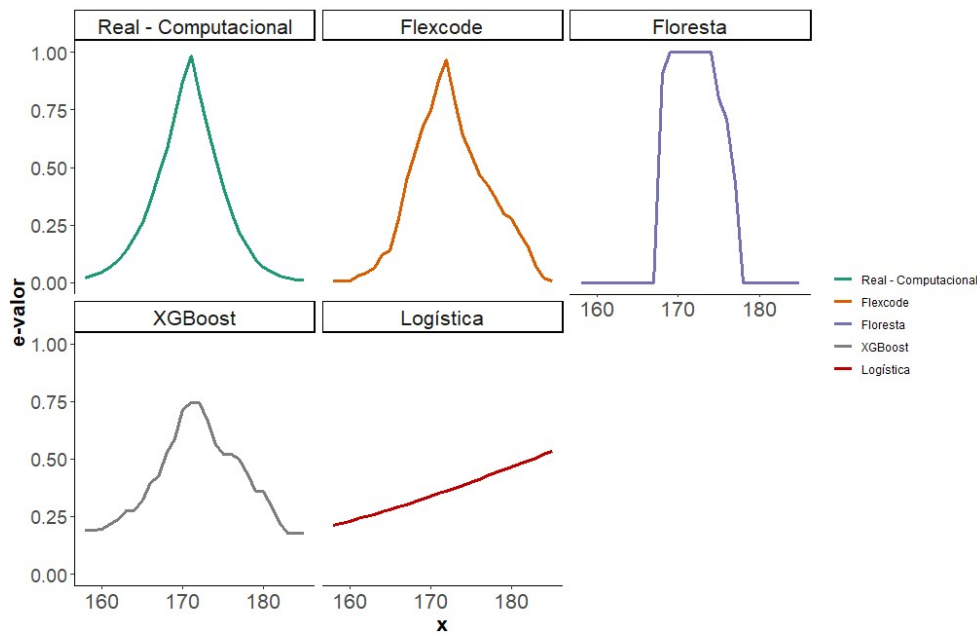


Figura 4.5: E-valor estimado para vários x

Analisando os e-valores obtidos aos pares, podemos entender melhor os desempenhos dos métodos. Na Figura 4.6, vemos que entre os métodos de classificação a estimativa do XGBoost é o mais correlacionado ($R = 0,947$) com o e-valor Real-Computacional, e graficamente, a dispersão dos pontos aproxima a reta identidade, exceto pelas caudas. Isso é um indicativo do que vimos nas caudas do e-valor estimado via XGboost no gráfico 4.5. Já o método via FlexCode, embora apresente uma correlação um pouco menor do que o XGBoost ($R = 0,937$), graficamente, o comportamento da dispersão é muito mais próximo da reta identidade do que o XGBoost para todos os pontos.

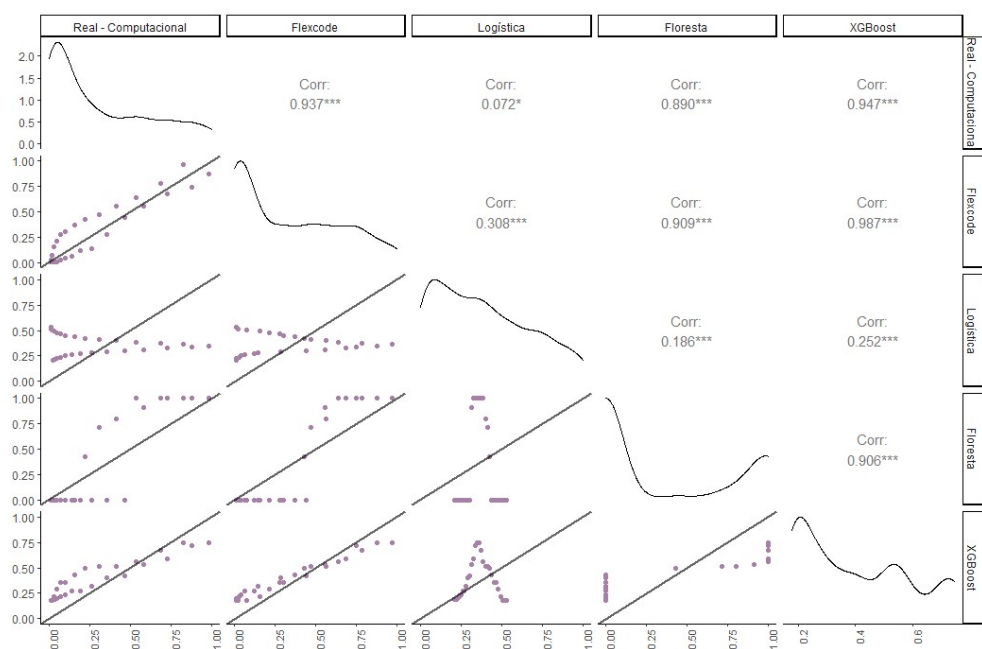


Figura 4.6: Dispersão entre os e-valores estimados e o e-valor real em função de x

4.2 Exemplos Multivariados

Agora testaremos as metodologias para exemplos multivariados. Para utilizar o FBST, a ideia é "reduzir" o problema original multivariado utilizando alguma transformação do parâmetro a valores reais (por exemplo, uma de suas componentes) e determinar o que chamamos de evidências restritas no Capítulo 3, para depois obtermos o e-valor original utilizando a aproximação descrita na seção 3.2.3. Outro ponto interessante é que como agora, em alguns casos, a entrada para estimação do Flexcode não será apenas de uma variável, podendo ser duas ou até mais, estimaremos o FlexCode via Florestas Aleatórias. Além disso, nesses casos, como as medidas resumo que entram no FlexCode podem ser bidimensionais ou tridimensionais, analisaremos o desempenho dos métodos apenas com as métricas descritas na seção 3.3. No primeiro exemplo, faremos ainda uma reparametrização do modelo de modo a aplicar a nova metodologia.

4.2.1 Comparação de duas populações Poisson independentes

Seja $\theta = (\theta_1, \theta_2)$ vetor de variáveis aleatórias independentes e tais que $\theta_i \sim \text{Gamma}(a_i, b_i)$ com $a_i, b_i > 0$, $i = 1, 2$. Seja $D = (X_1, \dots, X_n, Y_1, \dots, Y_m)$ vetor de variáveis que, dado θ , são independentes e tais que $X_i | \theta \sim \text{Poisson}(\theta_1)$, $i = 1, \dots, n$ e $Y_j | \theta \sim \text{Poisson}(\theta_2)$, $j = 1, \dots, m$. Logo, dado $D = d = (x_1, \dots, x_n, y_1, \dots, y_m) \in \mathbb{N}^{n+m}$, θ_1 e θ_2 são independentes e tais que

$$\theta_1 | D = d \sim \text{Gamma} \left(a_1 + \sum_{i=1}^n x_i, n + b_1 \right),$$

e

$$\theta_2 | D = d \sim \text{Gamma} \left(a_2 + \sum_{j=1}^m y_j, m + b_2 \right).$$

Consideremos a seguinte transformação de \mathbb{R}_+^2 a valores em $\mathbb{R}_+ \times [0, 1]$:

- $\lambda_1 = \theta_1 + \theta_2$
- $\lambda_2 = \frac{\theta_1}{\theta_1 + \theta_2}$

Tal reparametrização $\lambda = (\lambda_1, \lambda_2)$ torna mais simples o uso do método FlexCode para a estimação de densidades e também a obtenção da estimativa da medida de evidência para a comparação das duas populações Poisson. Note que na nova parametrização, a hipótese de igualdade das taxas, $\theta_1 = \theta_2$, pode ser reescrita como $\lambda_2 = 1/2$, que depende só de um componente do vetor λ .

Tomando $n = m$, pois nesse exemplo queremos garantir independência na posteriori conjunta, e $a_1 = a_2 = b_1 = b_2 = 1$, a fim de simplificar cálculos, temos que, dado $D = d$, λ_1 e λ_2 são independentes, com $\lambda_1 | D = d \sim \text{Gamma} \left(2 + \sum_{i=1}^n x_i + \sum_{j=1}^n y_j, n + 1 \right)$ e $\lambda_2 | D = d \sim \text{Beta} \left(1 + \sum_{i=1}^n x_i, 1 + \sum_{j=1}^n y_j \right)$. Para simular via Flexcode a posteriori de λ_2 , vamos gerar amostras de λ_2 e amostras de D , com $n = 200$. Na Figura 4.7, podemos comparar as

densidades de λ_2 a *posteriori* para alguns valores da estatística $\left(\sum_{i=1}^n x_i, \sum_{i=1}^n y_j \right)$. Observe que as curvas, real e estimada, estão próximas entre si. Um ponto interessante é que a densidade estimada parece ter mais oscilação, possuindo em alguns casos como em $\sum_{i=1}^n x_i = 66$ e $\sum_{i=1}^n y_i = 149$ três picos de maior densidade.

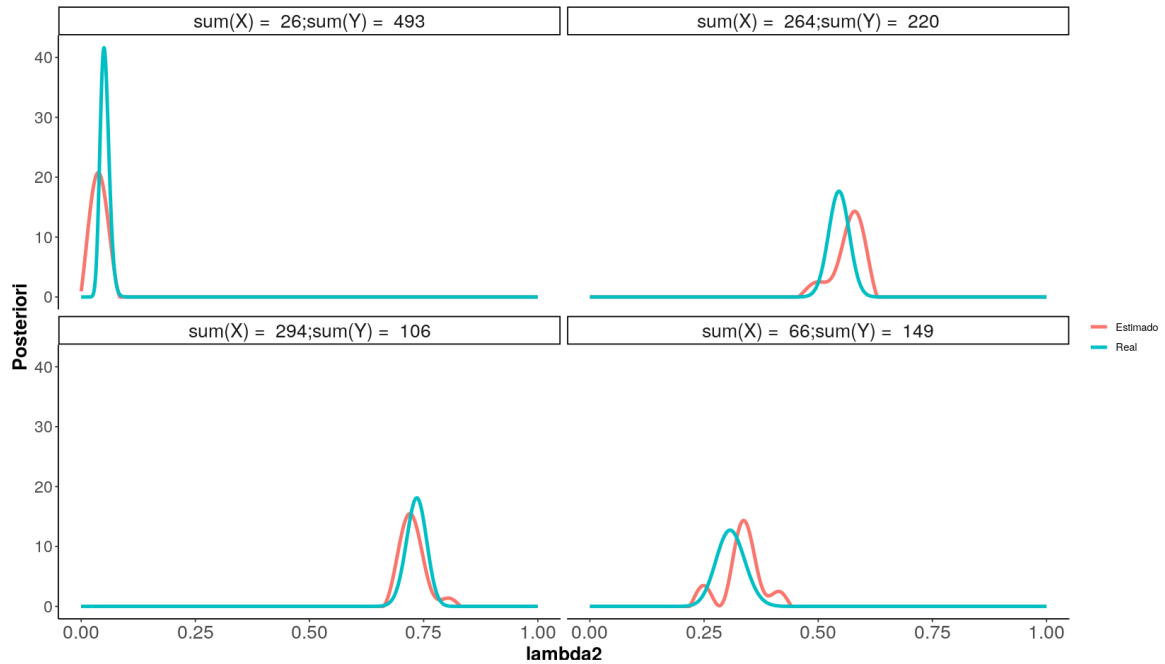


Figura 4.7: Distribuição a *posteriori* Real vs Estimada

Calculamos o e-valor para a igualdade das taxas das Poisson na nova parametrização, isto é, calculamos o e-valor para a hipótese $H_0 : \lambda_2 = 1/2$. Para a solução do problema de classificação, temos, na Tabela 4.5, a LogLoss dos métodos usados na estimação do e-valor. Note que a Floresta Aleatória é o que apresenta pior desempenho, pois exibe o maior valor de LogLoss (4,11), e a Regressão Logística (0,83) é o melhor.

Métodos	LogLoss
Logística	0,83
Floresta	4,11
XGBoost	1,87

Tabela 4.5: Tabela com o valor estimado da LogLoss na base de teste

Entretanto, ao contrário da LogLoss, segundo o erro absoluto (na Tabela 4.6) o modelo de classificação que melhor estima e-valor real é o XGBoost, indicando, novamente, que a avaliação dos métodos a partir da métrica LogLoss não chega aos mesmos resultados obtidos pelos demais critérios. Outro ponto importante a destacar é o desempenho da abordagem via FlexCode no cálculo do e-valor (descrito na seção 3.2.1): este método tem desempenho superior a todos os modelos de classificação propostos segundo o erro absoluto médio.

Método	Erro absoluto médio	Erro padrão
XGBoost	0.1938	0.0062
Logística	0.2696	0.0048
Floresta	0.2051	0.0077
Flexcode	0.1885	0.0076

Tabela 4.6: Cálculo do erro absoluto das evidências reduzidas estimadas em relação ao e-valor real reduzido e seu respectivo erro padrão

Na Figura 4.8, podemos ver que por mais que a regressão Logística apresente um bom desempenho medida pela LogLoss, a correlação de suas estimativas com os e-valores reduzidos reais é muito baixa ($R = 0,279$), se comparada ao XGBoost ($R = 0,615$), Florestas Aleatórias ($R = 0,590$) e o FlexCode ($R = 0,567$). Vale ressaltar que o FlexCode apresentou um desempenho superior ao olharmos o erro absoluto, porém quando observamos a correlação linear, XGBoost e Florestas apresentam melhores resultados. Pelo gráfico, podemos justificar esse resultado devido à maior variabilidade que existe na dispersão entre o e-valor real (Real- Computacional) e o e-valor gerado via FlexCode.

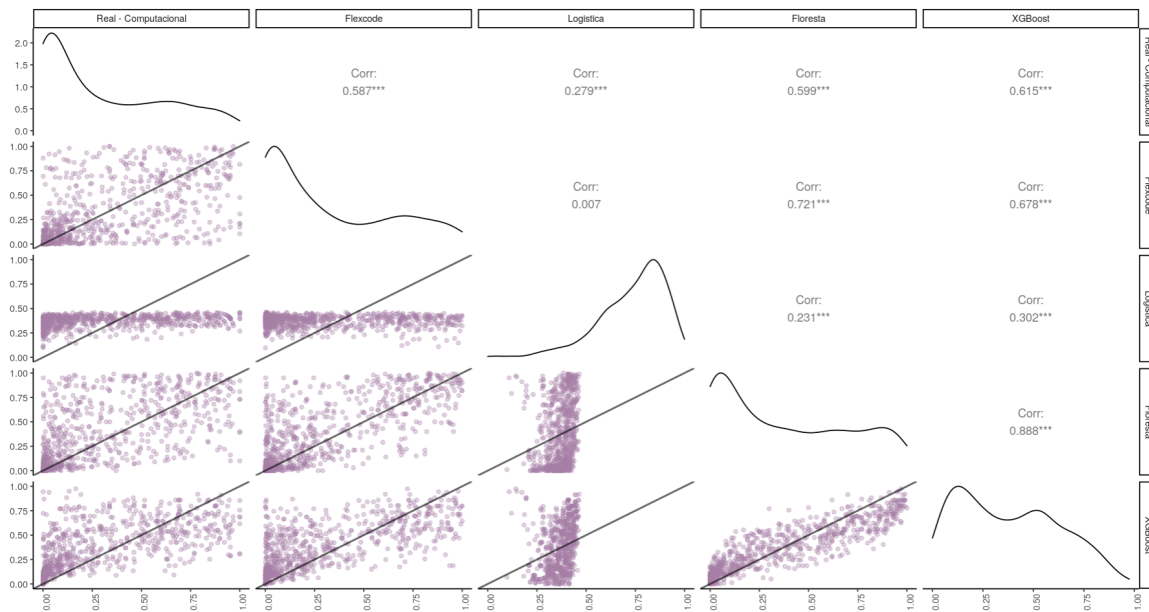


Figura 4.8: Dispersão entre os e-valores estimados reduzidos e o e-valor real reduzido

Agora que temos o e-valor restrito, para conseguirmos uma aproximação para o e-valor original realizaremos a transformação descrita na seção 3.2.3. Além disso, compararemos tal aproximação com o e-valor original calculado analiticamente para avaliar também a eficácia da transformação.

Obtido o e-valor no espaço original de forma analítica (Original - Analítico), no Apêndice A.1, conseguimos calcular o e-valor original analítico com o intuito de verificar se a transformação assintótica proposto em 3.2.3 para e-valores reduzidos realmente funciona. Note que o e-valor calculado a partir da integração da densidade real, utilizado anteriormente para comparação da eficácia dos e-valores reduzidos calculados via Flexcode

ou classificação, também deve ser transformado para o espaço original, da mesma forma que os demais. Continuaremos denominando tal e-valor por "Real - Computacional".

Da Figura 4.9, ao comparar o e-valor Original- Analítico com o e-valor Real - Computacional, cuja definição foi dada acima, vemos que eles são virtualmente idênticos. Isso traz um forte indício que a transformação via qui-quadrado proposta na seção 3.2.3 no e-valor reduzido se aproxima, bem, do e-valor obtido no espaço original. Comparando os demais e-valores estimados via FlexCode e classificação, agora transformados para o espaço original, vemos que, assim como no caso marginal, o melhor desempenho é o obtido pela técnica de classificação via XGBoost ($R = 0.708$).

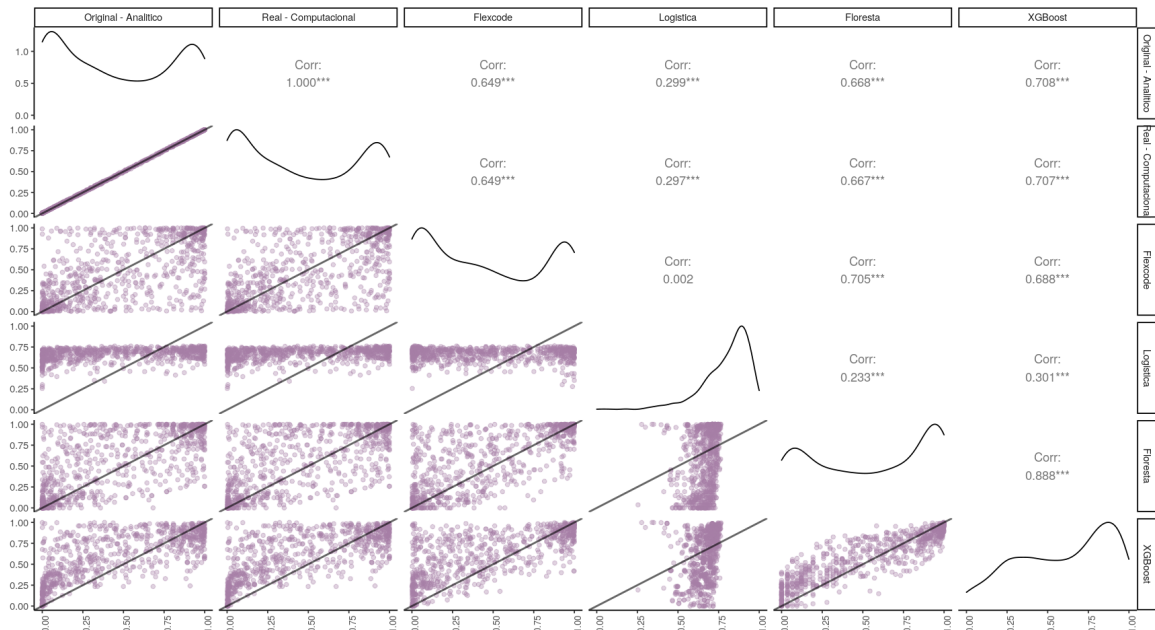


Figura 4.9: Dispersão entre os e-valores estimados originais e o e-valor real original

4.2.2 Distribuição Normal com variância desconhecida

Sejam X_1, \dots, X_n variáveis aleatórias que, dado $\theta = (\mu, \phi)$, são independentes e identicamente distribuídas $N(\mu, \phi^{-1})$ onde ϕ é a precisão de X . Nosso interesse é testar hipóteses precisas acerca de μ da forma $\mu = \mu_0$. Para continuar esse problema, devemos determinar a marginal a posteriori de $\mu|X = \mathbf{x} = (x_1, \dots, x_n)$ com o objetivo de calcular primeiramente o e-valor marginal da hipótese de interesse.

Consideramos que, a priori,

$$\phi \sim \text{Gama}\left(\frac{n_0}{2}, \frac{n_0 \sigma_0^2}{2}\right),$$

e

$$\mu|\phi \sim N(\mu_0, (c_0 \phi)^{-1}),$$

isto é, temos uma priori *Normal - Gama*($\mu_0, c_0, n_0, \sigma_0^2$) (LENK, 1988) para θ e, com isso, a marginal de μ é tem uma distribuição $t(n_0, \mu_0, \sigma_0^2/c_0)$. Como a família Normal-Gama é conjugada para amostragem Normal, temos que:

$$(\mu, \phi)|X = \mathbf{x} = (x_1, \dots, x_n) \sim \text{Normal - Gama}(\mu_1, c_1, n_1, \sigma_1^2),$$

isto é,

$$\phi|X = \mathbf{x} \sim \text{Gama}\left(\frac{n_1}{2}, \frac{n_1 \sigma_1^2}{2}\right)$$

$$\mu|\phi, X = \mathbf{x} \sim N(\mu_1, (c_1 \phi)^{-1})$$

$$\mu|X = \mathbf{x} \sim t(n_1, \mu_1, \sigma_1^2/c_1),$$

em que

- $\mu_1 = \frac{c_0 \mu_0 + n \bar{x}}{c_0 + n}$
- $c_1 = c_0 + n$
- $n_1 = n_0 + n$
- $\sigma_1^2 = \frac{1}{n_1} \left[n_0 \sigma_0^2 + \sum_{i=1}^n x_i^2 - n \bar{x}^2 + \frac{c_0 n (\mu_0 - \bar{x})^2}{c_0 + n} \right]$.

Para gerar as densidades via FlexCode, vamos gerar amostra da distribuição de (ϕ, μ) e valores de $\sum_{i=1}^n x_i$ e $\sum_{i=1}^n x_i^2$. Com $\mu_0 = 0$, $c_0 = 10$, $n_0 = 20$ e $\sigma_0^2 = 20$, podemos observar, na Figura 4.10, que para $\sum_{i=1}^n x_i = -3$, $\sum_{i=1}^n x_i^2 = 333$ e para $\sum_{i=1}^n x_i = 20$ e $\sum_{i=1}^n x_i^2 = 267$, o FlexCode produz estimativas boas para a média da distribuição marginal a *posteriori* de μ . Já para os demais casos, tais estimativas não são tão boas.

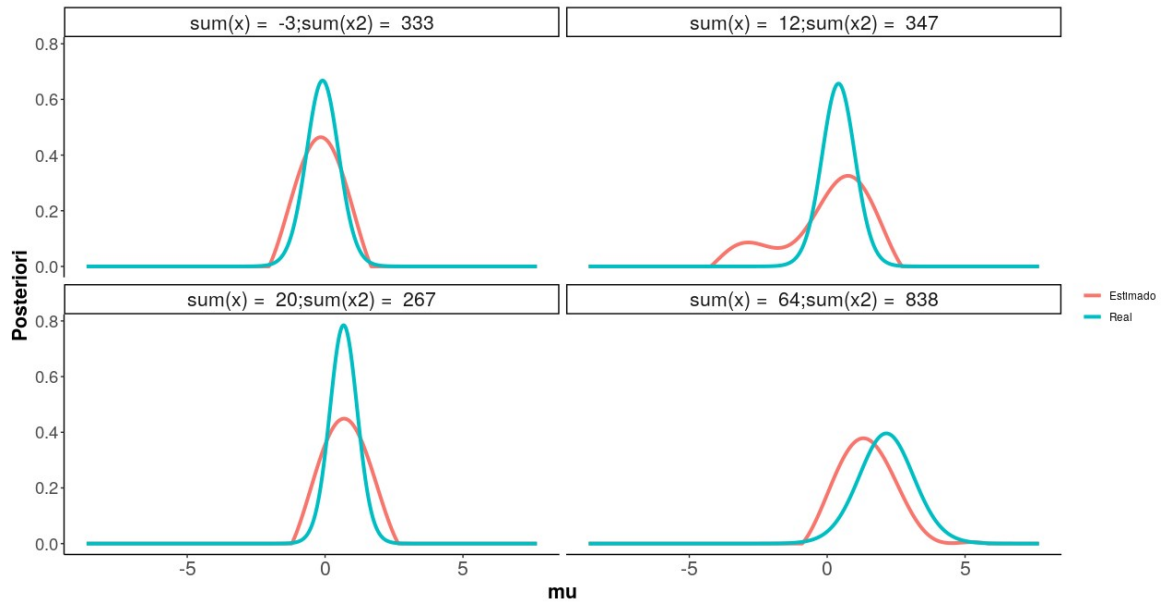


Figura 4.10: Distribuição a posteriori de μ Real vs Estimada

Agora, vamos testar $H_0 : \mu = 0$ vs $H_1 : \mu \neq 0$. Para a solução do problema de classificação, temos, na Tabela 4.7, a LogLoss dos modelos para estimar o e-valor restrito. Note que a Floresta Aleatória é o que apresenta pior desempenho, segundo essa métrica, e a Regressão Logística é o que apresenta o melhor ajuste.

Método	LogLoss
Logística	0,75
Floresta	2,68
XGBoost	1,29

Tabela 4.7: Tabela com o valor estimado da LogLoss na base de teste

Porém, já vimos em exemplos anteriores que nem sempre o melhor algoritmo segundo a LogLoss produz o e-valor mais próximo do e-valor real. Como nesse caso o parâmetro é bidimensional, conseguimos avaliar essa proximidade apenas através do erro absoluto médio e seu respectivo erro padrão. Da Tabela 4.8, observamos que, segundo essa métrica, o cálculo do e-valor restrito mais próximo do e-valor reduzido real é obtida pela classificação via XGBoost.

Método	Erro absoluto médio	Erro padrão
XGBoost	0,0948	0,0024
Logística	0,2660	0,0047
Floresta	0,1502	0,0043
Flexcode	0,1439	0,0045

Tabela 4.8: Cálculo do erro absoluto das evidências reduzidas estimadas em relação ao e-valor real reduzido e seu respectivo erro padrão

Na Figura 4.11, conseguimos identificar que a dispersão entre o e-valor real e o e-valor calculado via XGBoost se aproxima muito da reta identidade ($R=0,928$), indicando que realmente ambos possuem valores bem próximos. Outro ponto interessante é que os e-valores obtidos pela densidade do FlexCode e pela classificação via Florestas também apresentam alta correlação, $R= 0,794$ e $R=0,784$ respectivamente, porém com maior variabilidade.

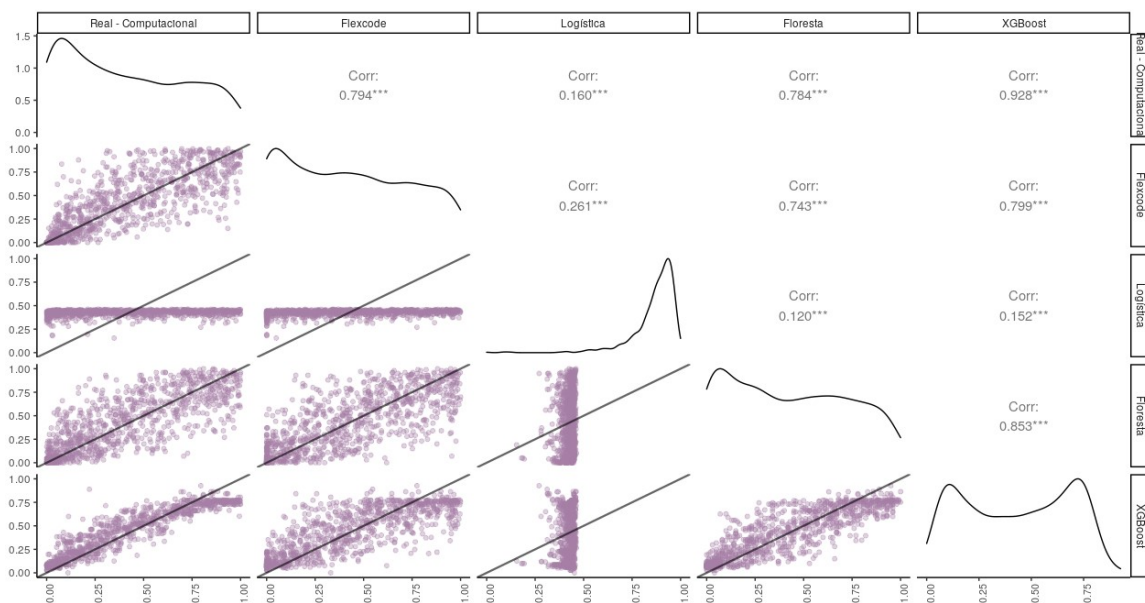


Figura 4.11: Dispersão entre os e-valores estimados reduzidos e o e-valor real reduzido

Como já vimos anteriormente (Seção 3.2.1), há uma certa diferença entre o e-valor obtido através da densidade marginal e o e-valor obtido da densidade conjunta. Por conta disso, é necessário fazer a transformação proposta nessa pesquisa. Para compararmos o e-valor transformado com o e-valor original, o cálculo do e-valor original real é obtido a partir da distribuição conjunta de θ no Apêndice A.2.

Na Figura 4.12, obtemos os e-valores corrigidos para o espaço original através da transformação para os diversos métodos considerados. Como no exemplo anterior, temos o "Original-Analítico", que é o e-valor calculado analiticamente no espaço original, para comparar se a transformação aproxima bem o e-valor reduzido. A resposta dessa comparação vem quando olhamos a correlação entre o e-valor "Original-Analítico" com e-valor "Real-Computacional", que é o valor obtido calculando-se o e-valor real reduzido e aplicando-se a transformação da seção 3.2.3. Note que a correlação entre tais valores é alta de 0,99, com os pontos muito próximos da reta identidade.

Além disso, olhando a transformação nas outras formas de estimar o e-valor, vemos que, assim como no caso reduzido, a técnica que produziu resultados mais próximo do e-valor original foi a classificação via XGBoost ($R=0,937$).

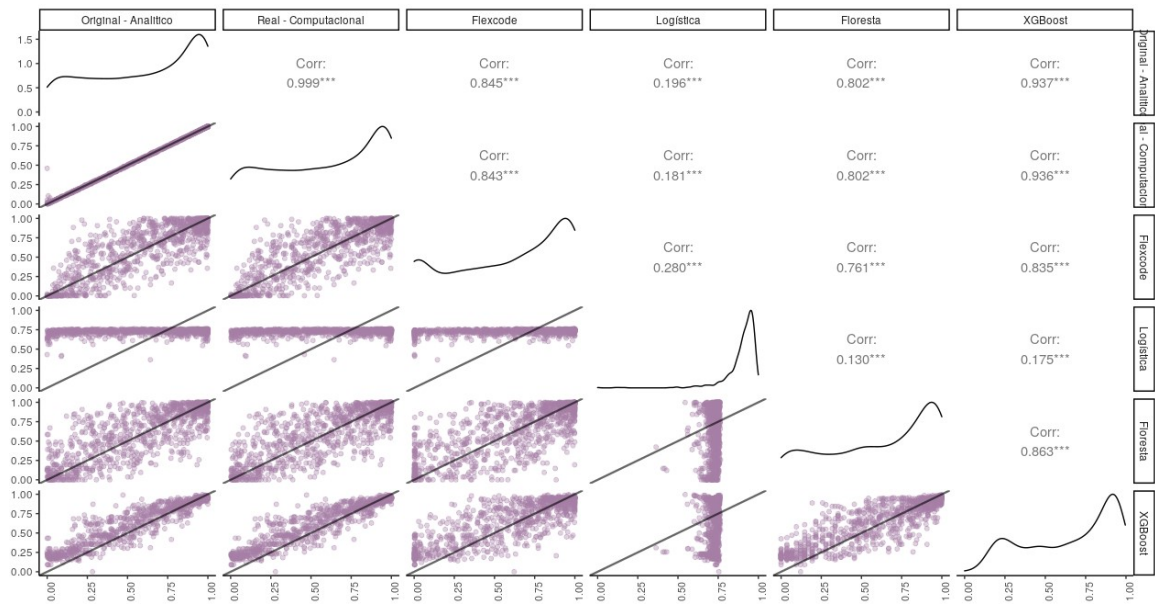


Figura 4.12: Dispersão entre os e-valores estimados originais e o e-valor real original

4.2.3 Regressão Linear

Consideramos agora o modelo de Regressão Linear com uma variável preditora. Seja $\theta = (\beta_0, \beta_1, \phi)$ e $\epsilon_1, \dots, \epsilon_n$, dado θ , c.i.i.d $N(0, \phi^{-1})$. Além disso, seja, $D = ((X_1, Y_1), \dots, (X_n, Y_n))$ com $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$ para $i = 1, \dots, n$. Consideremos a seguinte priori para θ :

$$\phi \sim \text{Gama}\left(\frac{c_0}{2}, \frac{c_0 \sigma_0^2}{2}\right)$$

e

$$(\beta_0, \beta_1) | \phi \sim N_2(\mu_0, V_0 \phi^{-1}).$$

Assim, θ tem distribuição conjunta $Normal_2 - \text{Gama}(\mu_0, V_0, c_0, \sigma_0^2)$ (HOYOS, s.d.). Com isso, a distribuição marginal de β_1 é dado por uma distribuição $t(c_0, \mu_0^{(2)}, V_0^{(2,2)} \sigma_0^2)$, onde $\mu_0^{(2)}$ é o segundo elemento do vetor μ_0 e $v_0^{(2,2)}$ é o elemento da posição (2,2) na matriz de variância e covariância V_0 . Pela propriedade de conjugação (WALTER *et al.*, 2007), temos, para $d = ((x_1, y_1), \dots, (x_n, y_n))$, a seguinte posteriori para θ :

$$(\beta_0, \beta_1, \phi) | D = d \sim Normal_2 - \text{Gama}(\mu_1, V_1, c_1, \sigma_1^2)$$

e

$$\beta_1 | D = d \sim t(c_1, \mu_1^{(2)}, V_1^{(2,2)} \sigma_1^2),$$

onde

- $V_1 = (V_0^{-1} + x^t x)^{-1}$
- $\mu_1 = V_1 (V_0^{-1} \mu_0 + x^t y)$
- $c_1 = c_0 + n$
- $c_1 \sigma_1^2 = c_0 \sigma_0^2 + \mu_0^t V_0^{-1} \mu_0 + y^t y - \mu_1^t V_1^{-1} \mu_1$,

com $x = (x_1, \dots, x_n)^t$ e $y = (y_1, \dots, y_n)^t$. Nosso interesse é testar $H_0 : \beta_1 = 0$ versus $H_1 : \beta_1 \neq 0$ através do FBST. Assim, inicialmente, vamos avaliar a aproximação à posteriori de β_1 pelos métodos propostos. No FlexCode, geramos amostras de β_1 através da distribuição t apresentada acima e amostras da estatística suficiente 5-dimensional para o modelo de regressão $(\sum_{i=1}^n y_i, \sum_{i=1}^n y_i^2, \sum_{i=1}^n y_i x_i, \sum_{i=1}^n x_i^2, \sum_{i=1}^n x_i)$. Consideramos, a priori, $c_0 = 20$, $\sigma_0^2 = 20$, $\mu_0 = (2 \quad 5)$ e $V_0 = \begin{bmatrix} 10 & 0.8 \\ 0.8 & 20 \end{bmatrix}$ e $n = 20$.

Na Figura 4.13, podemos ver que a densidade estimada não aproxima muito bem a densidade real, variando muito a qualidade da aproximação para cada amostra gerada do vetor $(\sum_{i=1}^n y_i, \sum_{i=1}^n y_i^2, \sum_{i=1}^n y_i x_i, \sum_{i=1}^n x_i^2, \sum_{i=1}^n x_i)$. Note que no cenário representado pelo gráfico superior à direita desta figura temos a melhor aproximação, pois as curvas destoam bem pouco entre si. Já na situação representada pelo gráfico superior esquerdo da figura, temos a pior aproximação, pois além das curvas destoarem em relação à média elas também destoam bastante em suas variâncias.

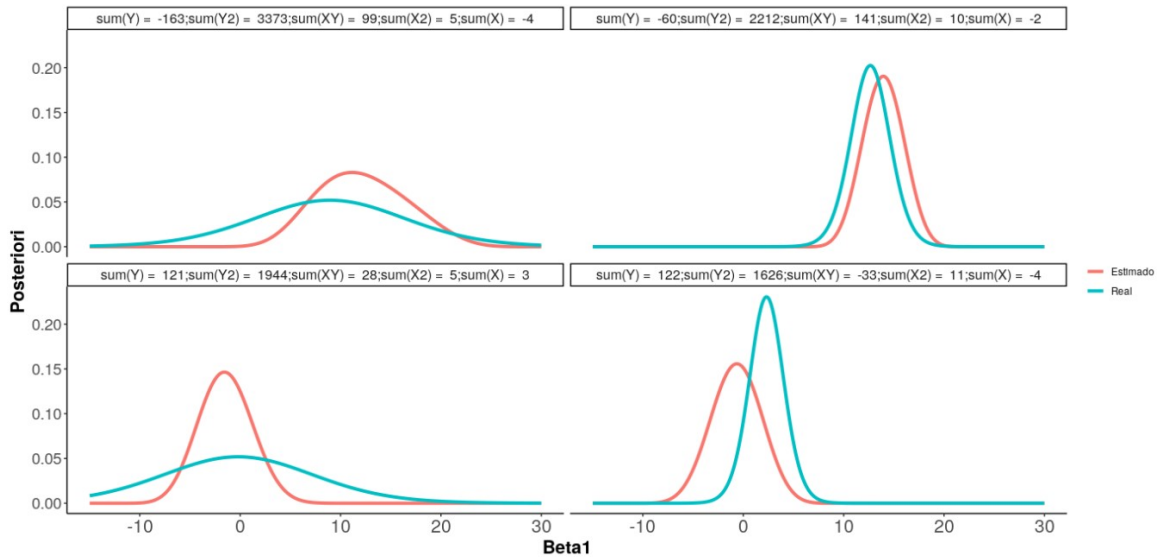


Figura 4.13: Distribuição a posteriori Real vs Estimada

Finalmente, iremos testar $H_0 : \beta_1 = 0$ vs $H_1 : \beta_1 \neq 0$. Para a solução do problema de classificação, temos, na Tabela 4.9, a LogLoss dos métodos usados para estimar o e-valor. Note que a Floresta é o que apresenta pior desempenho, segundo essa métrica, e a Regressão Logística o melhor.

Métodos	LogLoss
Logística	1,86
Floresta	13
XGBoost	3,10

Tabela 4.9: Tabela com o valor estimado da LogLoss na base de teste

Entretanto, ao contrário da LogLoss, vemos, na Tabela 4.10, que Regressão Logística é o modelo que apresenta maior erro absoluto médio, implicando que as estimativas do e-valor geradas por esse método não são os mais próximas do e-valor calculado computacionalmente. E entre os algoritmos de classificação, o melhor é o XGBoost. Note que o Flexcode, mesmo gerando densidades não tão próxima das densidades reais, teve um desempenho superior a qualquer outra técnica em termos do erro absoluto médio.

Método	Erro absoluto médio	Erro padrão
XGBoost	0,1139	0,0039
Logística	0,1739	0,0041
Floresta	0,1146	0,0041
Flexcode	0,0871	0,0037

Tabela 4.10: Cálculo do erro absoluto e seu respectivo erro padrão em relação ao e-valor real

O que vimos na Tabela dos erros absolutos também pode ser confirmado nos diagramas de dispersão da Figura 4.14: note que o e-valor obtido por Floresta e XGBoost apresentam

maiores correlações com o e-valor real, do que a Regressão Logística. Além disso, o FlexCode apresenta a maior correlação com o e-valor real ($R = 0,711$), indicando ser a técnica com o melhor desempenho conforme já apontado acima.

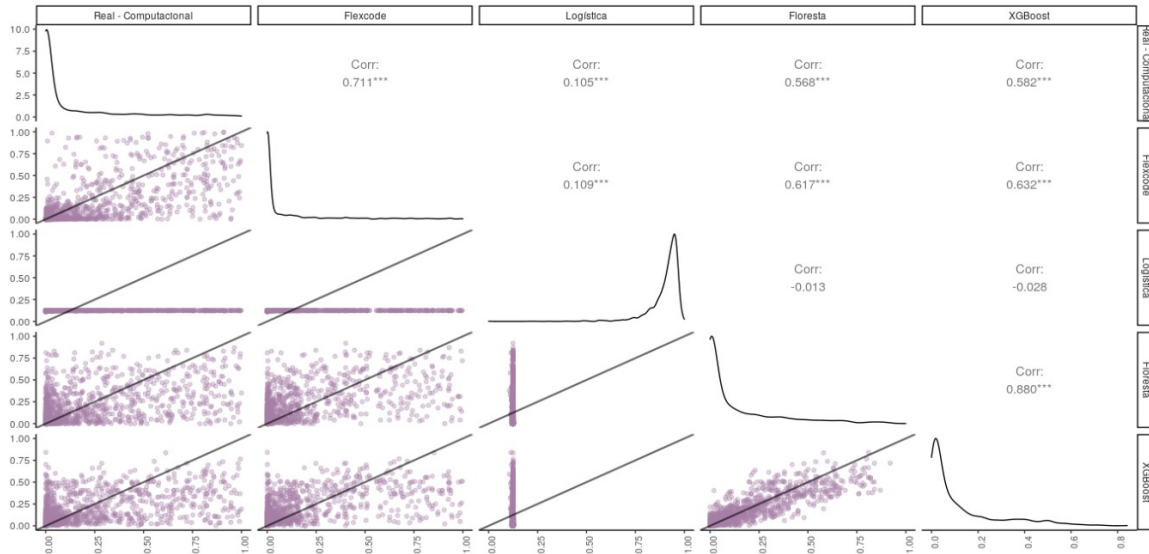


Figura 4.14: Dispersão entre os e-valores obtidos pela proposta de classificação e o e-valor real

Realizamos, por fim, a transformação descrita na seção 3.2.3 para o cálculo do e-valor original. Isso pode ser feito utilizando os resultados apresentados em (Hoyos, s.d.). Vale ressaltar que, nesse exemplo, devido à alta complexidade de calcular a medida de evidência no espaço original analiticamente, assumiremos que a transformação do e-valor reduzido para o e-valor original, utilizando a aproximação qui-quadrado, é plausível. Os exemplos da Poisson e Normal com variância desconhecida nesta seção endossam essa suposição.

Na Figura 4.15, podemos ver que, assim como aconteceu nos exemplos anteriores, a transformação qui-quadrado não alterou significativamente a correlação dos e-valores em relação ao e-valor real: ao comparar as duas figuras, 4.14 e 4.15, podemos dizer que elas estão bem parecidas entre si.

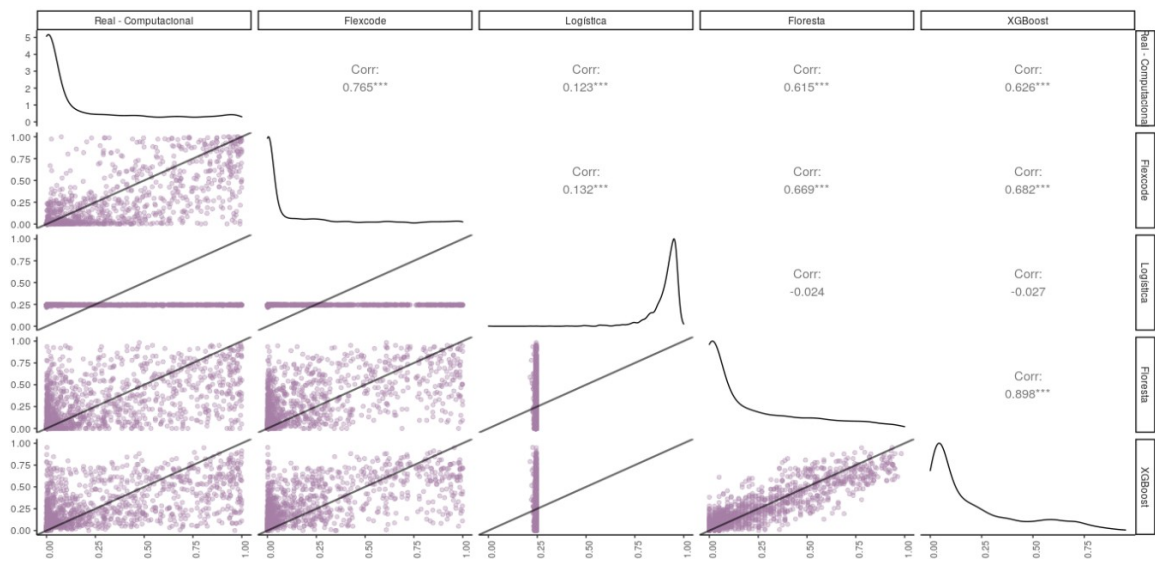


Figura 4.15: Dispersão entre os e-valores obtidos pela proposta de classificação e o e-valor real

Capítulo 5

Conclusão

Neste trabalho foram apresentadas técnicas de como realizar o teste FBST em problemas de *likelihood-free*. Mais precisamente, estimamos a distribuição a posteriori via FlexCode e calculamos a medida de evidência de Pereira-Stern para hipóteses precisas através de dois métodos: a integração da densidade estimada e a transformação do cálculo do e-valor em um problema de classificação. Um ponto importante a ser ressaltado é que o cálculo da posteriori via FlexCode em nenhum momento traz limitação para as técnicas de obtenção do FBST apresentadas. Ou seja, pode-se utilizar esses métodos propostos para outras maneiras de se obter a distribuição a posteriori via *likelihood-free*, como Redes Neurais por exemplo. Além disso, avaliamos as técnicas apresentadas em exemplos uni e multivariados.

Vimos que a obtenção da distribuição a posteriori via FlexCode teve bom desempenho nos exemplos citados, funcionando muito bem, em geral, e para casos uni-paramétricos em particular. Sobre as propostas para o cálculo de evidência do teste de hipótese FBST com essa densidade gerada, apresentamos quatro técnicas: FlexCode, XGBoost, Florestas Aleatórias e Regressão Logística. Pelos resultados obtidos para o teste de hipótese nos exemplos considerados, a Regressão Logística foi uma técnica que não estimou bem, segundo gráficos apresentados e o erro absoluto. Já medida de evidência, obtida por meio dos métodos XGBoost e Florestas Aleatórias capturaram melhor o comportamento do e-valor e seus resultados acabavam sendo bem próximos entre si, e o FlexCode teve melhor desempenho, segundo gráficos apresentados e o erro absoluto, em quase todos os exemplos, de todas as técnicas de classificação propostas, mesmo quando a densidade não estava tão bem estimada, como nos exemplos multivariados.

Um ponto interessante dessa pesquisa é que já conseguimos destacar técnicas de classificação para o cálculo do e-valor que não funcionaram bem nos exemplos citados, como, por exemplo, a estimação realizada via Regressão Logística.

Também vale destacar que, na mensuração dos resultados de classificação via LogLoss, essa métrica não detectou os métodos que produzem estimativas de e-valor mais próximas quando comparadas ao e-valor real. Em muitos casos, essa métrica apresentou resultados muito diferentes das análises gráficas e do erro absoluto apresentado. Isso pode ser um

problema na prática, pois acaba tornando difícil a avaliação da qualidade do ajuste sem ter um e-valor real para comparação.

Nos casos multidimensionais, realizamos a transformação do e-valor marginal para o e-valor original, e o que podemos observar é que se o e-valor marginal está bem ajustado, ele continua sendo bem ajustado após a transformação e caso ele não esteja bem ajustado a transformação não produziu boas estimativas para a medida de evidência, como no exemplo da Regressão Linear Simples.

Por fim, chegamos em resultados bem interessantes para estudos futuros nessa área: seria interessante encontrar uma métrica para avaliação de desempenho dos métodos de classificação, exceto a logloss, que mensure melhor as estimativas obtidas por esses modelos; propor outros modelos para realizar a classificação e tentar encontrar novas propostas de modelos que melhorem a estimação da densidade a posteriori ou até mesmo explorar a parametrização do FlexCode.

Apêndice A

A.1 Cálculo do e-valor para o exemplo de duas populações Poisson

Sejam λ_1 e λ_2 , que dado $D = d$, são independentes, com $\lambda_1|D = d \sim \text{Gama}(2 + \sum_{i=1}^n x_i + \sum_{j=1}^n y_j, n + 1)$ e $\lambda_2|D = d \sim \text{Beta}(1 + \sum_{i=1}^n x_i, 1 + \sum_{j=1}^n y_j)$.

Para o cálculo do e-valor original, considere a posteriori conjunta

$$\begin{aligned} \pi(\lambda_1, \lambda_2|D = d) &= \pi(\lambda_1|D = d)\pi(\lambda_2|D = d) = \\ &\propto \lambda_1^{(2 + \sum_{i=1}^n x_i + \sum_{j=1}^n y_j - 1)} e^{-(n+1)\lambda_1} \mathbb{I}_{\mathbb{R}_+}(\lambda_1) \lambda_2^{\sum_{i=1}^n x_i + 1 - 1} (1 - \lambda_2)^{\sum_{j=1}^n y_j + 1 - 1} \mathbb{I}_{[0,1]}(\lambda_2) \end{aligned}$$

Para avaliar o máximo da posteriori sob $H_0 : \lambda_2 = 1/2$, basta maximizar $\pi(\lambda_1|D = d)$ em λ_1 . Como $\pi(\lambda_1|D = d)$ é diferenciável em \mathbb{R}_+ , temos

$$\frac{d\pi(\lambda_1|d)}{d\lambda_1} = \lambda_1^{(\sum_{i=1}^n x_i + \sum_{j=1}^n y_j)} e^{-(n+1)\lambda_1} \left[\left(1 + \sum_{i=1}^n x_i + \sum_{j=1}^n y_j \right) - (n+1)\lambda_1 \right]$$

Igualando a última expressão a zero e resolvendo a equação obtida, temos que o valor λ_1^* que maximiza $\pi(\lambda_1|d)$ é $\lambda_1^* = \frac{(1 + \sum_{i=1}^n x_i + \sum_{j=1}^n y_j)}{(n+1)}$, pois é fácil verificar que $\frac{d^2\pi(\lambda_1|d)}{d\lambda_1^2} < 0$ no ponto $\lambda_1 = \lambda_1^*$.

Assim, o cálculo do e-valor original analítico é dado por:

$$e(\lambda_2 = 1/2, d)_{\text{original}} = 1 - \mathbb{P}(\lambda \in T_d|D = d),$$

onde

$$T_d = \{(\lambda_1, \lambda_2) \in \mathbb{R}_+ \times [0,1] : \pi(\lambda_1, \lambda_2 | D = d) > \pi(\lambda_1^*, 1/2 | D = d)\}.$$

A.2 Cálculo do e-valor para o exemplo da Normal com variância desconhecida

Sejam X_1, \dots, X_n variáveis aleatórias que, dado $\theta = (\mu, \phi)$, são independentes e identicamente distribuídos $N(\mu, \phi^{-1})$ onde ϕ é a precisão de X . Nosso interesse é testar as hipóteses $H_0 : \mu = 0$ vs $H_1 : \mu \neq 0$, acerca de μ . Para calcular o e-valor da hipótese de interesse, consideramos que

$$\phi \sim \text{Gama}\left(\frac{n_0}{2}, \frac{n_0 \sigma_0^2}{2}\right),$$

e

$$\mu | \phi \sim N(\mu_0, (c_0 \phi)^{-1}),$$

donde resulta, pelo resultado de conjugação, que

$$(\mu, \phi) | \mathbf{X} = \mathbf{x} = (x_1, \dots, x_n) \sim \text{Normal} - \text{Gama}(\mu_1, c_1, n_1, \sigma_1^2),$$

isto é,

$$\phi | \mathbf{X} = \mathbf{x} \sim \text{Gama}\left(\frac{n_1}{2} e^{-\frac{n_1 \sigma_1^2}{2}}\right)$$

$$\mu | \phi, \mathbf{X} = \mathbf{x} \sim N(\mu_1, (c_1 \phi)^{-1})$$

em que

- $\mu_1 = \frac{c_0 \mu_0 + n \bar{x}}{c_0 + n}$
- $c_1 = c_0 + n$
- $n_1 = n_0 + n$
- $\sigma_1^2 = \frac{1}{n_1} \left[n_0 \sigma_0^2 + \sum_{i=1}^n x_i^2 - n \bar{x}^2 + \frac{c_0 n (\mu_0 - \bar{x})^2}{c_0 + n} \right]$

Assim,

$$\pi(\mu, \phi | x) \propto \phi^{(n_1+1)/2-1} \exp \left\{ -\frac{\phi}{2} (n_1 \sigma_1^2 + c_1 (\mu - \mu_1)^2) \right\}.$$

Temos, sob H_0 , que

$$\pi(\mu = 0, \phi | x) \propto \phi^{(n_1+1)/2-1} \exp \left\{ -\frac{\phi}{2} (n_1 \sigma_1^2 + c_1 \mu_1^2) \right\} = g(\phi).$$

Para especificar o conjunto tangente a H_0 , $T(x)$, devemos maximizar $g(\phi)$ em ϕ . Assim,

$$\frac{dg(\phi)}{d\phi} = \phi^{(n_1+1)/2-2} \exp \left\{ -\frac{\phi}{2} (n_1 \sigma_1^2 + c_1 \mu_1^2) \right\} \left(\frac{(n_1+1)}{2} - \frac{2}{2} - \frac{\phi}{2} (n_1 \sigma_1^2 + c_1 \mu_1^2) \right) = 0 \Leftrightarrow$$

$$\Leftrightarrow \phi = \frac{n_1 - 1}{n_1 \sigma_1^2 + c_1 \mu_1^2}.$$

Encontramos assim que $\phi^* = \frac{n_1 - 1}{n_1 \sigma_1^2 + \mu_1^2 c_1}$ é ponto de máximo de $\pi(0, \phi|x)$, pois é fácil verificar que $\frac{d^2 \pi(\mu = 0, \phi|x)}{d\phi^2} < 0$ no ponto $\phi = \phi^*$. Sendo assim, $\theta^* = (0, \phi^*) = \operatorname{argmáx}_{\theta \in \Theta_0} \pi(\theta|X = x)$. O cálculo do e-valor então é dado por:

$$e(\Theta_0, d)_{original} = 1 - \mathbb{P}(\theta \in T(x)|X = x),$$

onde

$$T(x) = \{(\mu, \phi) \in \mathbb{R}x\mathbb{R}^+ : \pi(\mu, \phi|x) > \pi(0, \phi^*|x)\}.$$

Referências

- [M. A. BEAUMONT *et al.* 2002] Mark A BEAUMONT, Wenyang ZHANG e David J BALDING. “Approximate bayesian computation in population genetics”. Em: *Genetics* 162.4 (2002), pgs. 2025–2035 (citado na pg. 1).
- [BERGER e DELAMPADY 1987] James O BERGER e Mohan DELAMPADY. “Testing precise hypotheses”. Em: *Statistical Science* (1987), pgs. 317–335 (citado na pg. 2).
- [BIERNACKI *et al.* 2010] Christophe BIERNACKI, Gilles CELEUX e Gérard GOVAERT. “Exact and monte carlo calculations of integrated likelihoods for the latent class model”. Em: *Journal of Statistical Planning and Inference* 140.11 (2010), pgs. 2991–3002 (citado na pg. 6).
- [BISHOP 2006] Christopher M BISHOP. “Pattern recognition”. Em: *Machine learning* 128.9 (2006) (citado na pg. 11).
- [BREIMAN 2001] Leo BREIMAN. “Random forests”. Em: *Machine learning* 45.1 (2001), pgs. 5–32 (citado na pg. 2).
- [CHEN e GUESTRIN 2016] Tianqi CHEN e Carlos GUESTRIN. “Xgboost: a scalable tree boosting system”. Em: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM. 2016, pgs. 785–794 (citado nas pgs. 2, 10).
- [COUSINS 2018] Robert D COUSINS. “Lectures on statistics in theory: prelude to statistics in practice”. Em: *arXiv preprint* (2018) (citado na pg. 1).
- [CRANMER *et al.* 2015] Kyle CRANMER, Juan PAVEZ e Gilles LOUPPE. “Approximating likelihood ratios with calibrated discriminative classifiers”. Em: *arXiv preprint arXiv:1506.02169* (2015) (citado na pg. 1).
- [CUTLER *et al.* 2012] Adele CUTLER, D Richard CUTLER e John R STEVENS. “Random forests”. Em: *Ensemble machine learning*. Springer, 2012, pgs. 157–175 (citado na pg. 8).

- [Niccolo DALMASSO *et al.* 2021] Niccolo DALMASSO, David ZHAO, Rafael IZBICKI e Ann B LEE. “Likelihood-free frequentist inference: bridging classical statistics and machine learning in simulation and uncertainty quantification”. Em: *arXiv preprint arXiv:2107.03920* (2021) (citado na pg. 7).
- [Niccolò DALMASSO *et al.* 2020] Niccolò DALMASSO *et al.* “Validation of approximate likelihood and emulator models for computationally intensive simulations”. Em: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pgs. 3349–3361 (citado na pg. 7).
- [DEGROOT 2005] Morris H DEGROOT. *Optimal statistical decisions*. John Wiley & Sons, 2005 (citado na pg. 17).
- [DINIZ *et al.* 2012] Marcio DINIZ, Carlos AB PEREIRA, Adriano POLPO, Julio M STERN e Sergio WECHSLER. “Relationship between bayesian and frequentist significance indices”. Em: *International Journal for Uncertainty Quantification* 2.2 (2012) (citado na pg. 10).
- [FELDMAN e COUSINS 1998] Gary J FELDMAN e Robert D COUSINS. “Unified approach to the classical statistical analysis of small signals”. Em: *Physical review D* 57.7 (1998), pg. 3873 (citado na pg. 1).
- [GUO *et al.* 2003] Gongde GUO, Hui WANG, David BELL, Yaxin BI e Kieran GREER. “Knn model-based approach in classification”. Em: *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*. Springer. 2003, pgs. 986–996 (citado na pg. 8).
- [HOSMER JR *et al.* 2013] David W HOSMER JR, Stanley LEMESHOW e Rodney X STURDIVANT. *Applied logistic regression*. Vol. 398. John Wiley & Sons, 2013 (citado na pg. 2).
- [HOYOS s.d.] Alejandra Estefanía Patiño Hoyos. “Adaptive significance levels in linear regression models”. Tese de dout. Universidade de São Paulo (citado nas pgs. 29, 31).
- [IZBICKI, A. LEE *et al.* 2014] Rafael IZBICKI, Ann LEE e Chad SCHAFFER. “High-dimensional density ratio estimation with extensions to approximate likelihood computation”. Em: *Artificial intelligence and statistics*. PMLR. 2014, pgs. 420–429 (citado na pg. 1).
- [IZBICKI e A. B. LEE 2017] Rafael IZBICKI e Ann B LEE. “Converting high-dimensional regression to high-dimensional conditional density estimation”. Em: *Electronic Journal of Statistics* 11.2 (2017), pgs. 2800–2831 (citado na pg. 7).
- [IZBICKI, A. B. LEE e POSPISIL 2019] Rafael IZBICKI, Ann B LEE e Taylor POSPISIL. “Abc-cde: toward approximate bayesian computation with complex high-dimensional data and limited simulations”. Em: *Journal of Computational and Graphical Statistics* 28.3 (2019), pgs. 481–492 (citado nas pgs. 2, 7).

- [JAMES *et al.* 2013] Gareth JAMES, Daniela WITTEN, Trevor HASTIE e Robert TIBSHIRANI. *An introduction to statistical learning*. Vol. 112. Springer, 2013 (citado na pg. 8).
- [LENK 1988] Peter J LENK. “The logistic normal distribution for bayesian, nonparametric, predictive densities”. Em: *Journal of the American Statistical Association* 83.402 (1988), pgs. 509–516 (citado na pg. 25).
- [MADRUGA *et al.* 2003] Maria Regina MADRUGA, CA de B PEREIRA e Julio Michael STERN. “Bayesian evidence test for precise hypotheses”. Em: *Journal of Statistical Planning and Inference* 117.2 (2003), pgs. 185–198 (citado na pg. 6).
- [MARIN *et al.* 2012] Jean-Michel MARIN, Pierre PUDLO, Christian P ROBERT e Robin J RYDER. “Approximate bayesian computational methods”. Em: *Statistics and Computing* 22.6 (2012), pgs. 1167–1180 (citado na pg. 1).
- [METROPOLIS e ULAM 1949] Nicholas METROPOLIS e Stanislaw ULAM. “The monte carlo method”. Em: *Journal of the American statistical association* 44.247 (1949), pgs. 335–341 (citado na pg. 8).
- [PEREIRA e STERN 1999] Carlos Alberto PEREIRA e Julio Michael STERN. “Evidence and credibility: full bayesian significance test for precise hypotheses”. Em: *Entropy* 1.4 (1999), pgs. 99–110 (citado nas pgs. 2, 5).
- [PICARD e BERK 1990] Richard R PICARD e Kenneth N BERK. “Data splitting”. Em: *The American Statistician* 44.2 (1990), pgs. 140–147 (citado na pg. 14).
- [PRATT *et al.* 1995] John Winsor PRATT, Howard RAIFFA, Robert SCHLAIFER *et al.* *Introduction to statistical decision theory*. MIT press, 1995 (citado na pg. 13).
- [SISSON e FAN 2011] Scott A SISSON e Yanan FAN. “Likelihood-free mcmc”. Em: *Handbook of Markov Chain Monte Carlo* (2011), pgs. 313–335 (citado na pg. 1).
- [SISSON, FAN e M. BEAUMONT 2018] Scott A SISSON, Yanan FAN e Mark BEAUMONT. *Handbook of approximate Bayesian computation*. CRC Press, 2018 (citado na pg. 1).
- [WALTER *et al.* 2007] Gero WALTER, Thomas AUGUSTIN e Annette PETERS. “Linear regression analysis under sets of conjugate priors”. Em: *ISIPTA*. Vol. 7. 2007, pgs. 445–455 (citado na pg. 29).
- [YAO e LINDSAY 2009] Weixin YAO e Bruce G LINDSAY. “Bayesian mixture labeling by highest posterior density”. Em: *Journal of the American Statistical Association* 104.486 (2009), pgs. 758–767 (citado na pg. 6).

Índice Remissivo

C

Captions, *veja* Legendas

Código-fonte, *veja* Floats

E

Equações, *veja* Modo Matemático

F

Figuras, *veja* Floats

Floats

 Algoritmo, *veja* Floats, Ordem

Fórmulas, *veja* Modo Matemático

I

Inglês, *veja* Língua estrangeira

P

Palavras estrangeiras, *veja* Língua es-

trangeira

R

Rodapé, notas, *veja* Notas de rodapé

S

Subcaptions, *veja* Subfiguras

Sublegendas, *veja* Subfiguras

T

Tabelas, *veja* Floats

V

Versão corrigida, *veja* Tese/Dissertação,
versões

Versão original, *veja* Tese/Dissertação,
versões