



UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE MATEMÁTICA
DEPARTAMENTO DE ESTATÍSTICA



Coeficiente de Correlação Intraclasse: Comparação entre métodos de estimação clássico e bayesianos.

Autora: Greice Helen da Costa Laureano

Orientador: Professora MSc. Vanessa Bielefeldt Leotti Torman

Co-orientador: Professora Dra. Stela Maris de Jesus Castro

Porto Alegre, 14 de Dezembro de 2011.

Universidade Federal do Rio Grande do Sul
Instituto de Matemática
Departamento de Estatística

Coeficiente de Correlação Intraclasse: Comparação entre métodos de estimação clássico e bayesianos.

Autora: Greice Helen da Costa Laureano

Monografia apresentada para obtenção
do grau de Bacharel em Estatística.

Banca Examinadora:

Professora MSc. Vanessa Bielefeldt Leotti Torman

Professor Dr. João Riboldi

Porto Alegre, 14 de Dezembro de 2011.

Dedico esta monografia aos meus pais.

Agradecimentos

Agradeço os meus mestres por toda a aprendizagem que me proporcionaram, principalmente às professoras Vanessa Torman e Stela Castro pela excelente orientação na disciplina de monografia, assim como por todas aulas e ensinamentos que me deram durante a graduação. Agradeço também aos professores João Riboldi e Cleber Bisognin pela oportunidade de trabalho na área de pesquisa. A Acxiom Brasil por possibilitar o contato direto com análise de dados e por proporcionar meu enriquecimento profissional. Aos meus amigos queridos que sempre estiveram ao meu lado nas horas boas e ruins, principalmente ao Ruben Ladwing, Piel Souza, Cecilia Biguelini e Marilyn Agranonik. A família Coster por todo carinho e acolhimento. Ao meu amor, Rodrigo Coster, que além de me dar todo o apoio e carinho também me faz querer ser uma pessoa melhor a cada dia que passa. Por último a minha família que sempre me apoiou durante essa jornada, principalmente aos meus pais que independentemente das minhas escolhas sempre me incentivaram a lutar pelos meus sonhos.

Resumo

O coeficiente de correlação intraclasse (Intraclass correlation coefficient - ICC) é uma das ferramentas estatísticas mais utilizadas para a mensuração da confiabilidade de medidas. Ele pode ser utilizado na mensuração da homogeneidade de duas ou mais medidas e é interpretado como a medida da proporção da variabilidade total atribuída ao objeto medido. O ICC é calculado com base em uma razão de variâncias e nem sempre os métodos clássicos são capazes de estimar estes componentes, uma vez que se podem obter estimativas negativas dos componentes de variância. Portanto se torna interessante compará-los com os métodos bayesianos, nos quais se a priori for corretamente estabelecida e as posteriores corretamente avaliadas, é impossível de se obter estimativas negativas de variância. Com o objetivo de identificar qual a melhor forma de estimarmos o ICC foram comparados os métodos inferenciais clássico e bayesianos em quatro cenários distintos através de simulação. Utilizaram-se dois métodos para aproximar as posteriores de interesse: aproximação de Laplace (INLA) e simulação Monte Carlo via cadeias de Markov (MCMC). Concluiu-se que os métodos clássico e bayesianos INLA e MCMC são precisos nas estimativas dos ICC's, destacando-se o método INLA.

Sumário

1. Introdução	8
2. Metodologia	10
2.1 Método Clássico	12
2.2 Métodos Bayesianos	13
2.2.1 INLA	13
2.2.2 MCMC.....	14
2.3 Simulação	14
3. Resultados e discussão.....	16
4. Conclusões.....	20
5. Referências.....	21
6. Quadros e Gráficos.....	24
6.1. Estimativas pontuais para o cenário 1.	24
6.2. Estimativas pontuais para o cenário 2.	25
6.3. Estimativas pontuais para o cenário 3.	26
6.4. Estimativas pontuais para o cenário 4.	27
6.5. Estimativas dos intervalos de confiança e credibilidade para o cenário 1.	28
6.6. Estimativas dos intervalos de confiança e credibilidade para o cenário 2.	29
6.7. Estimativas dos intervalos de confiança e credibilidade para o cenário 3.	30
6.8. Estimativas dos intervalos de confiança e credibilidade para o cenário 4.	31
6.9. <i>Boxplot's</i> para o cenário 1.....	32
6.10. <i>Boxplot's</i> para o cenário 2.....	35
6.11. <i>Boxplot's</i> para o cenário 3.....	39
6.12. <i>Boxplot's</i> para o cenário 4.....	42
6.13. Tempos de simulação em horas, por cenário	46
Anexos.....	46
Anexo I: Exemplo prático	46
Anexo II: Código	52

Este artigo será submetido ao “*Journal of Statistical Computation and Simulation*”.

1. Introdução

O coeficiente de correlação intraclasse (*Intraclass correlation coefficient* - ICC) é uma das ferramentas estatísticas mais utilizadas para a mensuração da confiabilidade de medidas. O ICC é adequado para mensurar a homogeneidade de duas ou mais medidas e é interpretado como a proporção da variabilidade total atribuída ao objeto medido [1].

O conceito de correlação intraclasse foi introduzido por Fisher [2], que também primeiramente interpretou o termo “classe” dos coeficientes de correlação intraclasse. Para ele, a classe é constituída pela correlação entre as medidas realizadas [2], enquanto outros autores [3 e 4] usam o termo “classe” para se referir aos sujeitos nos quais foram feitas as análises de correlação.

O ICC é utilizado em distintas áreas do conhecimento e teve suas primeiras implementações para medir a homogeneidade da herdabilidade [2]. Na área da saúde, por exemplo, existem estudos que utilizam o ICC para fazer a mensuração de homogeneidade genética [5], para medir a precisão de instrumentos de medida [6-12], para verificar a associação entre níveis plasmáticos de carotenóides e tipos diferentes de dieta [13], para medir a falta de concordância entre médicos na hora de avaliar os custos e a qualidade dos cuidados de pacientes com doenças crônicas [14], entre outros.

O ICC é calculado através de uma razão de variâncias, as quais podem ser obtidas a partir de diferentes métodos de estimação, que podem ser classificados como clássicos ou bayesianos. A estimação de componentes de variância via métodos clássicos é, muitas vezes, ineficiente, uma vez que podem ser obtidas estimativas negativas destes componentes. Uma alternativa para a solução deste problema pode ser a utilização de métodos de estimação bayesianos, pois, para estes, se a priori for corretamente estabelecida e as posteriores corretamente avaliadas, a ocorrência de estimativas deste tipo é impossível.

A estimação dos componentes de variância e dos ICC's utilizando métodos bayesianos é um tema amplamente discutido na área científica. Diversos autores [15-24] propuseram inúmeras maneiras de se achar formas fechadas para as posteriores dos componentes de variância e também para razões de variâncias para modelos unifatoriais. Esses autores utilizaram diversas distribuições a priori e vários resultados matemáticos para

obter ou aproximar as posteriores de interesse.

Palmer e Broemeling [22], por exemplo, utilizaram a priori gama inversa para os componentes de variância e algumas aproximações para obter um estimador bayesiano para o ICC em modelos unifatoriais aleatórios. Por conseguinte, compararam o método bayesiano com o de máxima verossimilhança (MV) na estimação da correlação intraclasse com dados completos e incompletos. Para comparar os métodos, eles utilizaram simulação de 126 cenários, onde variavam o valor do ICC (0,0000; 0,0909; 0,1667; 0,3333; 0,5000; 0,6667 ou 0,8333), a quantidade de observações por classe (5, 10 ou 15), a quantidade de classes (3, 6 ou 9) e se o conjunto de dados foi completo ou incompleto. Compararam para cada cenário o método MV com o estimador bayesiano da mediana utilizando o cálculo do vício e do erro quadrático médio. Eles concluíram que o estimador bayesiano é adequado para todos os cenários e que deve ser preferencialmente usado quando o ICC for pequeno ou quando houver pelo menos uma classe com no máximo uma observação.

Burch e Harris [24] primeiramente compararam quatro métodos de estimação (que eles denominam de “*objective Bayesian estimators*”) do ICC para modelos unifatoriais com dados completos e incompletos: o primeiro é um estimador bayesiano que utiliza a máxima verossimilhança da quantidade pivotal do ICC, o segundo é também um estimador bayesiano que utiliza prioris de referência [23], o terceiro utiliza o método bayesiano que utiliza a máxima verossimilhança pivotal da função de máxima verossimilhança restrita na obtenção do estimador para o ICC e o quarto é o estimador bayesiano do ICC proposto por Palmer e Broemeling [22]. Posteriormente eles propõem estimadores (que eles denominam de “*subjective estimators*”) para o ICC que utilizam prioris beta e a máxima verossimilhança da quantidade pivotal para a derivação da forma fechada da posteriori baseada em um tipo específico de aproximação.

Para fazer a comparação dos quatro primeiros estimadores os autores utilizam dois exemplos, o primeiro oriundo de Winsor e Clarke [25] com dados completos e o segundo de Charlner [26] com dados incompletos. Para cada exemplo, compararam o erro quadrático médio. Eles concluíram que os melhores estimadores, para estimação do ICC, foram os que utilizaram a máxima verossimilhança da quantidade pivotal, ao passo que eles recomendam fortemente o uso do mesmo para dados completos. Já para o quinto estimador, eles analisaram o seu desempenho em termos do risco de bayes e recomendaram o seu uso

quando a análise de variância não fornece valores muito próximos de zero para os componentes de variância.

Jelenkowska [27] utilizou os resultados aproximados obtidos por Cook *et al* [21] e propôs um estimador para o ICC em modelos mistos. Também procedeu a uma comparação empírica entre este estimador e o estimador clássico obtido pelo método dos momentos (ANOVA) unifatoriais e com dois fatores.

Uma vez que não foi encontrada publicação na área científica que compare o método clássico com os métodos bayesianos para modelos com dois fatores, por simulação, nos motivamos a identificar qual a melhor forma de estimarmos o ICC comparando os métodos inferenciais clássico e bayesianos em quatro cenários simulados distintos.

Este artigo está organizado como se segue: primeiramente, na metodologia, abordaremos detalhes dos métodos comparados e do estudo de simulação procedido; posteriormente os resultados são apresentados e discutidos. Nos anexos, quadros e gráficos, bem como códigos do aplicativo R [28] utilizados, são apresentados.

2. Metodologia

O estudo de simulação procedido considerou uma situação de um estudo de confiabilidade de medidas onde cada unidade, chamadas de sujeito, tinha a característica de interesse mensurada repetidas vezes através de diferentes condições de aferição, que aqui consideramos como sendo diferentes aferidores. O delineamento experimental segue o mesmo escopo utilizado por Eliasziw *et al* [29].

Os modelos experimentais foram simulados de acordo com o modelo:

$$y_{ijk} = \mu + \pi_i + \gamma_j + (\pi\gamma)_{ij} + \varepsilon_{ijk} \quad (1)$$

$$i = 1, 2, \dots, a \quad j = 1, 2, \dots, b \quad k = 1, 2, \dots, r$$

onde, μ é a média geral, π_i é o efeito aleatório do sujeito i , γ_j é o efeito aleatório do aferidor j , $(\pi\gamma)_{ij}$ é o efeito aleatório da interação sujeito i x aferidor j , ε_{ijk} é o erro experimental e o

y_{ijk} se refere à medida realizada em cada sujeito (i), para um determinado aferidor (j) em uma dada repetição (k).

As suposições do modelo definido em (1) são: $\pi_i \stackrel{iid}{\sim} N(0, \sigma_\pi^2)$, $\gamma_j \stackrel{iid}{\sim} N(0, \sigma_\gamma^2)$, $\pi\gamma_{ij} \stackrel{iid}{\sim} N(0, \sigma_{\pi\gamma}^2)$, $\varepsilon_{ijk} \stackrel{iid}{\sim} N(0, \sigma_\varepsilon^2)$ e independentes entre si.

O foco do estudo está na comparação dos métodos na estimação de dois tipos de ICC:

$$\rho^* = \frac{\sigma_\pi^2}{\sigma_\pi^2 + \sigma_\gamma^2 + \sigma_{\pi\gamma}^2 + \sigma_\varepsilon^2}$$

$$\rho^{**} = \frac{\sigma_\pi^2 + \sigma_\gamma^2 + \sigma_{\pi\gamma}^2}{\sigma_\pi^2 + \sigma_\gamma^2 + \sigma_{\pi\gamma}^2 + \sigma_\varepsilon^2}$$

Eliasziw *et al* [29] denomina ρ^* e ρ^{**} , respectivamente, como interaferidor e intraferidor. O coeficiente ρ^* se refere à correlação entre as medidas do mesmo sujeito independentemente do aferidor responsável pelas mesmas. Já o coeficiente ρ^{**} se refere à correlação entre as medidas do mesmo sujeito levando em consideração o aferidor utilizado. Por exemplo, se $\rho^* = 0,8$ significa que a correlação entre duas medidas do mesmo sujeito, tomadas por diferentes aferidores, é 0,8. Da mesma forma, se $\rho^{**} = 0,9$ significa que a correlação entre duas medidas do mesmo sujeito, tomadas pelo mesmo aferidor, é igual a 0,9. Note que ρ^{**} é sempre maior que ρ^* .

Comparamos três métodos de estimação: um clássico e dois bayesianos. A seguir, detalharemos estes métodos assim como o estudo de simulação conduzido. Posteriormente, no anexo, apresentamos um exemplo prático e os códigos utilizados no estudo de simulação.

2.1 Método Clássico

Dentre os diferentes métodos de estimação de componentes de variância clássicos, escolhemos o método de máxima verossimilhança restrita (REML), uma vez que é o mais indicado na literatura, principalmente quando se tem dados incompletos [30]. Para a obtenção dos intervalos de confiança para os ICC's, optou-se pela utilização da técnica de *Bootstrap*. Na literatura foram encontradas expressões para os limites superior e inferior dos intervalos de confiança dos ICC's baseados no método de estimação dos momentos [29 e 31], e, como este método necessita de propriedades assintóticas para sua implementação, não julgamos sua utilização adequada para os cenários trabalhados, em especial quando do uso de dados incompletos.

O método de estimação REML teve sua primeira abordagem aplicada na estimação de componentes de variância com dados incompletos feita por Patterson e Thompson [30]. Este método foi proposto como solução para um problema de estimativas viesadas que ocorre quando do uso do método de estimação ML. O REML, diferentemente do ML, maximiza separadamente as funções de verossimilhança dos efeitos fixos e aleatórios, ao passo que a maximização para os efeitos aleatórios, em relação aos componentes de variância, retira o viés introduzido pela perda de graus de liberdade na estimação dos efeitos fixos. Apesar de ser uma técnica amplamente utilizada, existem restrições na sua utilização, pois algumas vezes o algoritmo numérico, utilizado por ele, pode convergir para valores negativos para os componentes de variância.

Na estimativa dos intervalos de confiança foi utilizado o *Bootstrap* não-paramétrico. Esta é uma técnica não tem suposições a respeito da distribuição do conjunto de dados e vem a ser uma abordagem alternativa ao cálculo de intervalos de confiança. O *Bootstrap* utiliza amostras combinadas como uma pseudo-população (amostra representativa da população da qual os dados são oriundos) e gera várias amostras aleatórias, com reposição, da mesma.

No caso deste estudo, utilizou-se a abordagem de amostrar o sujeito, com reposição, para criar as amostras *Bootstrap*, e assim manter a correlação existente entre as medidas do mesmo sujeito. Para isto, os dados devem ser organizados de modo que cada linha contenha um sujeito com todas as suas respectivas medidas. Em seguida, sorteia-se com reposição uma nova amostra com o mesmo número de sujeitos. Repete-se este procedimento um

número razoável de vezes, gerando várias amostras *Bootstrap*. Para cada amostra, ajusta-se o modelo (1) e se estima os efeitos fixos e aleatórios através do método de estimação REML. Os intervalos de confiança são obtidos através dos percentis 2,5 e 97,5 das estimativas obtidas em cada amostra para cada parâmetro.

Para o ajuste do modelo (1), foi utilizado o pacote lme4 [32] do aplicativo R.

2.2 Métodos Bayesianos

O principal resultado de uma análise bayesiana é a distribuição a posteriori marginal de cada parâmetro desconhecido. No modelo que estamos considerando, não existe forma analítica para essas posterioris, então as mesmas foram aproximadas através de métodos computacionais. Neste trabalho, foram utilizados dois métodos para obtenção das distribuições a posteriori de interesse: INLA (*Integrated Nested Laplace Approximations*) e simulação MCMC (*Markov Chain Monte Carlo*) via algoritmo *Gibbs Sampling*.

Nas estimativas pontuais dos componentes de variância e ICC's foram utilizados os estimadores média, mediana e moda (esta última estimada a partir da densidade não paramétrica de cada posteriori), e para os intervalos de credibilidade foram utilizados o intervalo de credibilidade central e o intervalo de credibilidade HPD (*highest posteriori density* – intervalos que possuem a menor amplitude dos intervalos com $100(1-\alpha)\%$ de credibilidade). Tanto no INLA quanto na simulação MCMC, prioris vagas foram utilizadas para propiciar uma comparação justa com o método clássico.

2.2.1 INLA

O INLA [33] é um novo método bayesiano que utiliza a aproximação de Laplace para aproximar as posterioris via métodos computacionais determinísticos, ou seja, sem utilizar simulação. Assim, o INLA é mais rápido que o MCMC e não tem o inconveniente de ter que se fazer diagnóstico de convergência das cadeias simuladas. Este método está implementado em um pacote do R disponível na internet. Instruções para instalação deste pacote bem como o ajuste do modelo (1) através do mesmo podem ser encontradas no anexo I e II.

As prioris padrões do INLA foram utilizadas neste trabalho. O INLA define os

modelos em função do logaritmo natural das precisões dos efeitos aleatórios, sendo que a precisão de um efeito aleatório é o inverso do seu componente de variância. Assim, as distribuições a priori utilizadas foram as seguintes: $p(\mu) \propto 1$, $\log\left(\frac{1}{\sigma_\pi^2}\right) \sim \log\text{Gama}(1; 0,00005)$, $\log\left(\frac{1}{\sigma_\gamma^2}\right) \sim \log\text{Gama}(1; 0,00005)$, $\log\left(\frac{1}{\sigma_{\pi\gamma}^2}\right) \sim \log\text{Gama}(1; 0,00005)$ e $\log\left(\frac{1}{\sigma_\varepsilon^2}\right) \sim \log\text{Gama}(1; 0,00005)$.

2.2.2 MCMC

A utilização do MCMC já está bastante difundida, e seu uso elevou a aplicação de métodos bayesianos em problemas práticos. Apesar de ser eficaz, trata-se de um método por vezes demorado e que necessita que se faça o diagnóstico de convergência das cadeias simuladas.

Foram utilizadas as seguintes prioris para os parâmetros do modelo (1): $\mu \sim N(0, 1000)$, $s_\pi \sim U(0, 100)$, $s_\gamma \sim U(0, 100)$, $s_{\pi\gamma} \sim U(0, 100)$ e $s_\varepsilon \sim U(0, 100)$, onde $\sigma_\pi^2 = \left(\frac{1}{s_\pi}\right)^2$, $\sigma_\gamma^2 = \left(\frac{1}{s_\gamma}\right)^2$, $\sigma_{\pi\gamma}^2 = \left(\frac{1}{s_{\pi\gamma}}\right)^2$ e $\sigma_\varepsilon^2 = \left(\frac{1}{s_\varepsilon}\right)^2$. A utilização da família uniforme para as prioris dos desvios padrões dos efeitos aleatórios e erro experimental é devida a recomendação de Spiegelhalter *et al* [34] e Gelman [35].

Para o ajuste do modelo (1), foi utilizado o aplicativo OpenBUGS versão 3.2.1 [36], ao passo que se utilizou o pacote BRugs [37] para fazer a interface entre o OpenBUGS e o R. O aplicativo OpenBUGS é um programa para análise bayesiana de modelos estatísticos complexos que implementa algoritmos MCMC.

2.3 Simulação

De acordo com o modelo (1), foram gerados dados (1000 amostras) em quatro cenários distintos: dois aferidores com dados completos (cenário 1), quatro aferidores com dados completos (cenário 2), dois aferidores com dados incompletos (cenário 3) e quatro aferidores com dados incompletos (cenário 4). O resultado é um delineamento experimental tendo r medidas feitas por cada aferidor b em cada um dos sujeitos a, como disposto no Quadro 1.

Quadro 1: Delineamento experimental.

	Aferidor						
	1		...	b			
	Medidas			Medidas			
Sujeito	1	...	r	...	1	...	r
1	y_{111}	...	y_{11r}	...	y_{1b1}	...	y_{1br}
2	y_{211}	...	y_{21r}	...	y_{2b1}	...	y_{2br}
.		
.	...			y_{ijk}	...		
.		
a	y_{a11}	...	y_{a1r}	.	y_{ab1}	...	y_{abr}

$$i = 1, 2, \dots, a. j = 1, 2, \dots, b. k = 1, 2, \dots, r.$$

Assim, no cenário 1 temos $a = 29$, $b = 2$ e $r = 3$, no cenário 2 temos $a = 29$, $b = 4$ e $r = 3$, no cenário 3 temos $a = 29$, $b = 2$ e r varia para cada sujeito e aferidor, no cenário 4 temos $a = 29$, $b = 4$ e r é novamente variável.

Em todos os cenários, os verdadeiros valores dos parâmetros considerados foram os dados no Quadro 2, retirados de Eliasziw *et al* [29].

Quadro 2: Verdadeiros valores dos parâmetros.

Parâmetro	Valor
ρ^*	0,9457
ρ^{**}	0,9844
μ	-1,2480
σ_π^2	51,9877
σ_γ^2	0,9155
$\sigma_{\pi\gamma}^2$	1,2155
σ_ε^2	0,8563

O primeiro passo para a geração das amostras aleatórias foi a geração dos efeitos aleatórios, segundo as suposições do modelo (1) e os verdadeiros valores dos componentes de variância. Com esses valores gerados e o verdadeiro valor do efeito fixo, calculou-se o valor da esperança condicional da variável resposta: $(\theta_{ijk} = E(Y_{ijk} | \mu, \pi_i, \gamma_j, (\pi\gamma)_{ij})) = \mu + \pi_i + \gamma_j + (\pi\gamma)_{ij}$. Então os valores Y_{ijk} foram gerados a partir de uma distribuição normal com média θ_{ijk} e variância σ_ε^2 .

Para simular as situações com dados incompletos, primeiramente se simulou dados completos e então se retirou 17 observações no cenário 3 e 34 observações no cenário 4. Estes valores faltantes foram selecionados a partir de um sorteio, sem reposição, e representam aproximadamente 10% do tamanho total da amostra de cada um destes cenários ($a \times b \times r$).

Para o método clássico foi utilizado um número de amostras *Bootstrap* igual a 1000. Os intervalos de confiança para todos os parâmetros do modelo foram então estimados através dos percentis 2,5 e 97,5 das estimativas obtidas em cada amostra *Bootstrap*.

Em qualquer simulação MCMC, é necessário especificar o período de aquecimento do algoritmo (*burn-in*), número de cadeias, número de interações e o pulo (*thin*) entre as interações para reduzir a autocorrelação. Após diversos testes optamos por utilizar neste trabalho: *burn-in* igual a 2000, número de cadeias de Markov igual a 2, quantidade de interações igual a 10000 e *thin* igual a 20.

Após as 1000 replicações da simulação calculamos o EQM (erro quadrático médio), o vício (média das estimativas menos o verdadeiro valor do parâmetro), o vício relativo absoluto (VRA, módulo da razão entre o vício e o verdadeiro valor do parâmetro), a cobertura (proporção de vezes que o verdadeiro valor do parâmetro estava contido nos intervalos de confiança ou credibilidade) e a amplitude média dos intervalos de confiança ou de credibilidade (média da diferença entre limite superior e inferior dos intervalos).

3. Resultados e discussão

Os resultados de todas as simulações para cada cenário estão dispostos na sessão Quadros e Gráficos.

Para o método clássico houve algumas simulações em que as estimativas de alguns componentes de variância foram negativas. Nestes casos, o aplicativo R trunca o valor estimado do componente em zero. O Quadro 3 mostra quantas vezes ocorreu esse fenômeno.

Quadro 3: Ocorrência de truncamento em zero dos componentes de variância para cada cenário.

Componentes de variância	Cenário			
	1	2	3	4
σ^2_{π}	0	0	0	0
σ^2_{γ}	142	23	166	28
$\sigma^2_{\pi\gamma}$	5	0	1	0
σ^2_{ε}	0	0	0	0

No Quadro 3 se observa que houve uma frequência maior de problemas de convergência do REML para os cenários 1 e 3, e que apenas as variâncias do aferidor e da interação foram problemáticas. Esse comportamento indica que houve dificuldade para método clássico estimar os componentes de variância quando há apenas dois aferidores. A ocorrência destas estimativas negativas pode ter sido causada pelo valor muito baixo atribuído a σ^2_{γ} .

Surpreendentemente, para o estimador INLA-Mediana, o componente de variância do aferidor obteve algumas estimativas negativas - Cenário 1 (144), Cenário 2 (88), Cenário 3 (124) e Cenário 4 (103) - por conseguinte, algumas estimativas para o ρ^* foram maiores que um (Quadro 4). Conforme discussão com um dos autores do método, isto aconteceu em situações extremas onde existia um conflito entre a priori estabelecida e os dados simulados. A priori atribui probabilidades não desprezíveis para valores altos de precisão e a amostra simulada apontava para um valor baixo, entretanto, a amostra não possuía informação suficiente para sobrepor a informação a priori. Como resultado desse conflito, a posteriori da precisão era bimodal, e os algoritmos do pacote não conseguem transformar corretamente uma posteriori bimodal.

Quadro 4: Ocorrências de estimativas maiores que um, em ao menos um estimador, do método INLA, para o ρ^*

Cenário	Frequência
1	44
2	0
3	29
4	0

Quando ocorreu incoerência na estimação dos componentes de variância, tanto para o método clássico quanto para o método INLA, excluímos todas as estimativas das amostras que obtiveram esse problema.

Para o método clássico, após a exclusão das simulações com estimativas negativas dos componentes de variância, os cenários ficaram com o seguinte número de observações: cenário 1 (854), cenário 2 (977), cenário 3 (833) e cenário 4 (972), já para o método INLA os cenários ficaram desta forma: cenário 1 (856), cenário 2 (912), cenário 3 (876) e cenário 4 (897). É importante salientar que a quantidade de replicações do método bayesiano MCMC permaneceu inalterada, ou seja, cada cenário contém 1000 replicações.

Como houve estimativas negativas de variância para os dois métodos (clássico e INLA) decidimos verificar se essas estimativas ocorreram na mesma amostra. A quantidade de amostras que obtiveram estimativas negativas dos componentes de variância para os dois métodos, simultaneamente, esta disposta a seguir: cenário 1 (1), cenário 2 (4), cenário 3 (1) e cenário 4 (5). Como a frequência de estimativas negativas dos componentes de variância dos métodos clássico e INLA - para a mesma amostra - é pequena não podemos associar que o comportamento amostral que gera problemas no método clássico é semelhante ao do INLA.

Na análise gráfica (ver sessão Quadros e Gráficos) observamos a variabilidade das estimativas e a precisão do método (proximidade da média e mediana do verdadeiro valor do parâmetro). Portanto, observando os gráficos para os parâmetros σ_{π}^2 , σ_{γ}^2 , $\sigma_{\pi\gamma}^2$ e σ_{ε}^2 (Figuras: 4-7; 11-14; 18-21; 25-28) é possível identificar que os métodos que obtiveram melhores comportamentos foram os métodos Clássico e INLA.

Para o parâmetro μ (Figuras: 3, 10, 17 e 24) todos os estimadores indicam adequabilidade. Ao passo que para o parâmetro ρ^* (Figuras: 1, 8, 15, 22) os estimadores que se mostraram mais adequados foram: para os cenários 1 e 3, ou seja, os cenários com apenas dois aferidores, os métodos Clássico e INLA, já para os cenários 2 e 4, onde haviam quatro aferidores, os métodos Clássico, INLA e MCMC-Moda.

Para o parâmetro ρ^{**} (Figuras: 2, 9, 16, 23) os estimadores que se mostraram mais adequados foram: para os cenários 2 e 4, todos os métodos, já para os cenários 1 e 3, os métodos Clássico e INLA.

O método bayesiano MCMC teve dificuldade de estimar o componente de variância σ_{γ}^2 , principalmente para cenários que tinham dois aferidores. Notou-se uma melhora nas estimativas quando o cenário tinha 4 aferidores. Este comportamento é explicado por

Gelman [35], que informa que quanto maior a quantidade de níveis do efeito aleatório melhores serão as estimativas do MCMC.

Ainda para os componentes de variância, observou-se que os estimadores INLA-Média e MCMC-Média foram os que tiveram estimações mais imprecisas em relação aos que utilizam a mediana ou a moda. Portanto recomendamos a utilização dos estimadores que utilizam mediana ou moda na estimação dos componentes de variância.

Os métodos que obtiveram melhores desempenhos para os ICCs em relação às medidas resumo EQM e VRA estão dispostos no Quadro 5.

Quadro 5: Métodos que obtiveram melhor desempenho para cada cenário

Cenário	Parâmetro	Menor EQM	Menor Vício Relativo	Maior Amplitude do EQM	Maior Amplitude do Vício Relativo
1	ρ^*	INLA-Moda	INLA-Mediana	0,1281	0,3445
	ρ^{**}	INLA-Moda	INLA-Moda	0,0001	0,0056
2	ρ^*	INLA-Moda	INLA-Mediana	0,0078	0,0636
	ρ^{**}	MCMC-Mediana	MCMC-Média	0,0000	0,0056
3	ρ^*	INLA-Moda	INLA-Mediana	0,1317	0,3515
	ρ^{**}	INLA-Moda	INLA-Moda	0,0001	0,0059
4	ρ^*	INLA-Moda	INLA-Mediana	0,0066	0,0600
	ρ^{**}	MCMC-Mediana	MCMC-Média	0,0000	0,0020

O estimador INLA-Moda, na maioria dos cenários, foi o que obteve menores EQM's. Por outro lado, os estimadores INLA-Mediana, MCMC-Média e INLA-Moda foram os que obtiveram menores valores de VRA. Observa-se, entretanto, que a amplitude entre os maiores e menores VRA's e EQM's muitas vezes é tão pequena que, em alguns cenários, não foi possível eleger apenas 1 como, de fato, o melhor estimador para os ICC's. As maiores diferenças foram encontradas nos cenários 1 e 3 para o parâmetro ρ^* , onde os resultados apontam que os métodos MCMC devem ser descartados. Isto é coerente com o fato de o método MCMC ter tido um desempenho pobre para estimar os componentes de variância nos cenários com apenas dois aferidores.

Também foi possível observar um pequeno aumento de dificuldade na estimação dos componentes de variância para os cenários 3 e 4, principalmente para os métodos clássico e MCMC. Como esses cenários são referentes a amostras com dados incompletos, esse tipo de comportamento já era esperado.

Os métodos intervalares que obtiveram menor amplitude e maior cobertura estão dispostos, a seguir, no Quadro 7.

Quadro 6: Métodos intervalares que obtiveram melhor desempenho para cada cenário.

Cenário	Parâmetro	Menor Amplitude	Maior Cobertura	Maior Amplitude da Amplitude	Maior Amplitude da cobertura
1	ρ^*	Clássico-Bootstrap	MCMC-Central	0,8551	12,5000
	ρ^{**}	INLA-HPD	Clássico-Bootstrap	0,0042	11,8000
2	ρ^*	Clássico-Bootstrap	MCMC-HPD	0,3153	15,5000
	ρ^{**}	INLA-HPD	MCMC-Central	0,9543	10,6000
3	ρ^*	Clássico-Bootstrap	MCMC-Central	0,8593	11,3000
	ρ^{**}	INLA-HPD	MCMC-Central	0,0041	8,0000
4	ρ^*	INLA-HPD	MCMC-HPD	0,2998	16,4000
	ρ^{**}	MCMC-HPD	MCMC-Central	0,0054	11,3000

Observa-se que existem grandes diferenças entre as coberturas e amplitudes médias dos métodos, podendo-se afirmar que, para o parâmetro ρ^* , o método que tem menor amplitude é geralmente o método Clássico-Bootstrap e o que tem maior cobertura é geralmente o MCMC-Central. Já para o parâmetro ρ^{**} , os métodos que tem, geralmente, menor amplitude e maior cobertura foram, respectivamente: INLA-HPD e MCMC-Central. É importante ressaltar que todos os métodos raramente atingiram o nível nominal de 95% de cobertura. Apenas os intervalos obtidos por MCMC atingiram esse nível em algumas situações.

Os tempos de simulação, por cenário, estão dispostos na sessão Quadros e Gráficos. Para todos os cenários o método INLA foi o mais rápido, seguido pelo método MCMC e posteriormente pelo método clássico. Para os cenários com 4 aferidores os tempos de simulação aumentaram consideravelmente para quase totalidade dos métodos. É importante destacar que foram usados 4 computadores diferentes para realizar as simulações dos cenários e seus respectivos métodos. A utilização de diferentes computadores, com características de *hardware* distintas, pode ter causado flutuações nos tempos de simulação dos cenários.

4. Conclusões

Após a análise dos resultados do estudo de simulação, concluímos que os métodos clássico e os bayesianos INLA e MCMC têm estimativas precisas para os ICC's. Destacamos o método INLA, que obteve bom desempenho em quase todas as comparações

realizadas (análise gráfica, VRA's, EQM's, amplitudes e coberturas dos intervalos de credibilidade).

Entretanto, como ocorreu em nossas simulações, o método INLA pode obter estimativas negativas de variância ou maiores que 1 para o ICC. Esse fenômeno ainda deve ser melhor investigado. No momento, se isso acontecer num problema prático, recomenda-se especificar outros valores para os hiperparâmetros das prioris, ou utilizar outro método de estimação (MCMC ou clássico).

Apesar de haver pouca publicação na área científica para a estimação dos ICC's, por métodos bayesianos, quando se fala de modelos unifatoriais e com dois fatores, os autores geralmente concordam que os métodos bayesianos são melhores, principalmente quando se tem dados incompletos, para estimação do ICC [22]. Neste estudo não foi possível fazer esse tipo de afirmação, pois todos os métodos foram bons na estimação pontual dos ICC's.

Para estudos posteriores seria interessante comparar os métodos com dados possuindo maior proporção de observações faltantes (mais que 10%), assim como também a verificar a influência do aumento de aferidores nas estimativas. Outra comparação relevante poderia ser testar valores menores de ICC e verificar o comportamento dos estimadores [22].

Pretende-se posteriormente aumentar o número de replicações da simulação, para verificar se a cobertura dos intervalos melhorará.

5. Referências

- [1] Shrout, P. E., Fleiss, J. L. (1979). Intraclass correlations: Uses in assessing reliability. *Psychological Bulletin*, Vol. 86, N° 2, Pag. 420-428.
- [2] Fisher, R. A. (1925). *Statistical methods for research workers*. New York: Hafner Press.
- [3] Haggard, E. A. (1958). *Intraclass correlation and the analysis of variance*. New York: Dryden Press.
- [4] Harris, J. A. (1913). On the calculation of intraclass and interclass coefficients of correlation from class moments when the number of possible combinations is large. *Biometrika*, Vol. 9, Pag. 446-472.
- [5] Robert, C., Foulley, J. L., Ducroq V. (1995). Genetic variation of traits measured in several environments. I. Estimation and testing of homogeneous genetic and intra-class correlations between environments. *Genet. Sel. Evol.*, Vol. 27, Pag. 111-123.
- [6] Luft, V. C., Beghetto, M. G., Castro, S. M. J., Mello, E. D. (2008). Validation of a new method developed to measure the height of adult patients in bed. *Nutrition in Clinical Practice*, Vol. 23, N° 4, Pag. 424-428.

- [7] Hade, E. M., Strickland, P. A. O., et all. (2010). Intraclass correlation estimates for cancer screening outcomes: Estimates and applications in the design of group-randomized cancer screening studies. *Journal of the National Cancer Institute Monographs*, N° 40, pag 97-103.
- [8] Venturini, C., Ituassú, N. T., Teixeira, L. M., Deus, C. V. O. (2006). Confiabilidade intra e interexaminadores de dois métodos de medida da amplitude ativa de dorsiflexão do tornozelo em indivíduos saudáveis. *Revista Brasileira de Fisioterapia*, São Carlos, Vol. 10, N°4, Pag. 407-411.
- [9] Lutosa, L. P., Silva, C. W. A., Brito, J. P., Valadão, R. (2008). Goniometria e Fleximetria: um estudo de confiabilidade e comparação das medidas nas articulações do cotovelo e joelho. *e-scentia*, Vol. 1, N° 1, Novembro.
- [10] Youdas, J. W., Carey, J. R., Garrett, T. R. (1991). Reliability of measurements of cervical spine range of motion – Comparison of three methods. *Physical Therapy*, Vol. 71, N° 2, February.
- [11] Watkins, M. A., Riddle, D. L., Lamb, R. L., Personius, W. J. (1991). Reliability of goniometric measurements and visual estimates of knee range of motion obtained in a clinical setting. *Physical Therapy*, Vol. 71, N° 2, February.
- [12] Clapper, M. P., Wolf, S. L. (1988). Comparison of the reliability of the orthoranger and the standart goniometer for assessing active lower extremity range of motion. *Physical Therapy*, Vol. 68, N° 2, February.
- [13] Ferrari, P., Riboli, E., et all. (2005). An Approach to estimate between- and within-group correlation coefficients in multicenter studies: Plasma carotenoids as biomarkers of intake of fruits and vegetables. *American Journal of Epidemiology*, Vol. 162, N° 6, Pag. 591-598.
- [14] Hofer, T. P., Hayward, R. A., Greenfield, S., Wagner, E. H., Kaplan, S. H., Manning, W. G. (1999). The unreliability of individual physician “report cards” for assessing the costs and quality of care of a chronic disease. *American Medical Association*, Vol. 281, N° 22, Pag. 2098-2105.
- [15] Tiao, G.C., Tan, W.Y. (1965). Bayesian analysis of random-effect models in the analysis of variance - I: Posterior distribution of variance components. *Biometrika* Vol. 52, Pag. 37-53.
- [16] Hill, B.M. (1965). Inference about Variance Components in the One-Way Model. *Journal of the American Statistical Association*, Vol. 60, Número 311, Pag. 806-825.
- [17] Stone, M., Springer, B.G.F. (1965). A paradox involving quasi-prior distributions. *Biometrika*, Vol. 52, Pag. 623-627.
- [18] Rajagopal, M., Broemeling, L.D. (1983) Bayesian inference for the variance components of the general mixed model. *Communications in Statistics* 12, No 6 , Pag. 701-724.
- [19] Gharraf, M. (1979). General Solution to Making Inference about the Parameters of Mixed Linear Models. Ph.D. dissertation, Oklahoma State University, Stillwater, Oklahoma.
- [20] Jelenkowska, T.H. (1988). Bayesian estimation of the variance components in a general random linear model, *Bayesian Statistics 3*. Oxford Sci. Publ. Oxford Univ. Press, New York, Pag. 653-656.
- [21] Cook, P., Broemeling, L.D., Gharraf, M. (1990). A Bayesian analysis of the mixed linear model. *Communications in Statistics A19*, Vol. 3, Pag. 987-1002.
- [22] Palmer, J., Broemelling, D. L. (1990). A comparison of bayes and maximum likelihood estimation of the intraclass correlation coefficient. *Communications in Statistics – Theory and Methods*, Vol. 19(3), Pag. 953-975.
- [23] Keying Ye (1994). Bayesian reference prior analysis on the ratio of variances for the balanced one-way random effect model. *Journal of Statistical Planning and Inference*, Vol. 41, Pag. 267-280.

- [24] Burch, D. B., Harris, R. I. (1999). Bayesian estimators of the intraclass correlation coefficient in the one-way random effects model. *Communications in Statistics – Theory and Methods*, Vol. 28(6), Pag. 1247-1272.
- [25] Winsor, C. P., Clarke, J. L. A statistical study of variation in the catch of plankton nets. *Journal of Marine Research*, Vol. 3, Pag. 1-34.
- [26] Charloner, K. (1987). A bayesian approach to the estimation of variance components for the unbalanced one-way random effects model. *Tecnometrics*, Vol. 29, Pag 323-337.
- [27] Jelenkowska, T. H. (1998). Bayesian estimation of the intraclass correlation coefficients in the mixed linear model. *Applications of Mathematics*, Vol. 2, Pag. 103-110.
- [28] R Development Core Team. (2008). *R: A Language and Environment for Statistical Computing*. Vienna Austria R Foundation for Statistical Computing, Vol. 1 10^o Edição. <http://www.r-project.org/>.
- [29] Eliasziw, M., Young, S. et all. (1994). Statistical Methodology for the concurrent assessment of interrater and intrarrater reliability using goniometric measurements as an example. *Physical Therapy*, Vol. 74 , N^o8, Agosto.
- [30] Patterson, H.D.; Thompson, R (1971). Recovery of interblock information when block sizes are unequal. *Biometrika*, Vol.58, Pag.545-554.
- [31] McGraw, K., Wong, S. (1996). Forming inferences about some intraclass correlation coefficients. *Psychological Methods* , Vol. 1 , N^o1, Pag. 30-46.
- [32] Bates, D., Martin, M., Bin, D. (2011). lme4: Linear mixed-effects models using S4 classes. R package version 0.999375-42. <http://cran.r-project.org/web/packages/lme4/index.html>.
- [33] Rue, H., Martino, S. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal Royal Statistical Society B*, Vol. 71, Part. 2, Pag 319-392.
- [34] Spiegelhalter, DJ., Abrams, KR., Myles, J. (2004). *Bayesian Approaches to Clinical Trials and Health-care Evaluation*. Wiley: London.
- [35] Gelman, P. (2006). Distributions for variance parameters in hierarchical models. *Bayesian Analysis*, Vol.1, N^o 3, Pag. 515-533.
- [36] Thomaz, A., Spiegelhalter, D., Best, N., Lunn, D., Rice, K. (2011). *Bayesian inference Using Gibbs Sampling*. Version 3.2.1. <http://www.openbugs.info/w/FrontPage>.
- [37] Thomaz, A., Spiegelhalter, D., Best, N., Lunn, D., Rice, K. (2011). *Bayesian inference Using Gibbs Sampling*. Version 0.7-1. <http://www.openbugs.info/w/UserContributedCode>.

6. Quadros e Gráficos

6.1. Estimativas pontuais para o cenário 1.

Cenário	Parâmetro	Método	Média	Vício	Variância	EQM	Vício Relativo
Cenário 1	μ	Clássico	-1,26923	-0,02128	2,25545	2,25590	1,7%
		MCMC-Média	-0,82369	0,42426	8,11173	8,29173	34,0%
		MCMC-Moda	-0,97265	0,27530	9,69710	9,77288	22,1%
		MCMC-Mediana	-0,83014	0,41781	6,52044	6,69500	33,5%
		INLA-Média	-1,23174	0,01621	2,23879	2,23906	1,3%
		INLA-Moda	-1,23174	0,01621	2,23879	2,23906	1,3%
		INLA-Mediana	-1,23175	0,01620	2,23879	2,23906	1,3%
	σ^2_{π}	Clássico	51,95925	-0,02849	186,37492	186,37573	2,3%
		MCMC-Média	58,10865	6,12091	231,13913	268,60465	11,8%
		MCMC-Moda	50,50263	-1,48511	178,09968	180,30524	2,9%
		MCMC-Mediana	55,13604	3,14830	208,34060	218,25238	6,1%
		INLA-Média	51,81034	-0,17740	179,65924	179,69071	0,3%
		INLA-Moda	45,53557	-6,45217	139,66038	181,29086	12,4%
		INLA-Mediana	49,53001	-2,45773	164,49862	170,53904	4,7%
	σ^2_{γ}	Clássico	0,99315	0,07761	2,20925	2,21528	8,5%
		MCMC-Média	328,84211	327,92657	49948,69097	157484,52365	35817,7%
		MCMC-Moda	13,54620	12,63066	716,90386	876,43739	1379,6%
		MCMC-Mediana	36,38547	35,46993	4507,74399	5765,85992	3874,2%
		INLA-Média	0,63054	-0,28500	1,63230	1,71353	31,1%
		INLA-Moda	0,13214	-0,78340	0,07956	0,69328	85,6%
		INLA-Mediana	0,30192	-0,61362	0,38799	0,76452	67,0%
	$\sigma^2_{\pi\gamma}$	Clássico	1,22435	0,00887	0,21209	0,21217	0,7%
		MCMC-Média	1,37783	0,16235	0,21019	0,23655	13,4%
		MCMC-Moda	1,16570	-0,04978	0,16193	0,16441	4,1%
		MCMC-Mediana	1,29294	0,07746	0,18848	0,19448	6,4%
		INLA-Média	1,36203	0,14655	0,37794	0,39942	12,1%
		INLA-Moda	1,16002	-0,05547	0,29564	0,29871	4,6%
		INLA-Mediana	1,28943	0,07395	0,34650	0,35197	6,1%
	σ^2_{ε}	Clássico	0,86265	0,00635	0,02588	0,02592	0,7%
		MCMC-Média	0,88831	0,03201	0,01372	0,01474	3,7%
		MCMC-Moda	0,86080	0,00450	0,01298	0,01300	0,5%
		MCMC-Mediana	0,87813	0,02183	0,01342	0,01390	2,5%
		INLA-Média	0,87230	0,01600	0,01398	0,01424	1,9%
		INLA-Moda	0,84349	-0,01281	0,01298	0,01314	1,5%
		INLA-Mediana	0,86227	0,00597	0,01364	0,01368	0,7%
	ρ^*	Clássico	0,94080	-0,00486	0,00112	0,00115	0,5%
		MCMC-Média	0,61732	-0,32834	0,01940	0,12720	34,7%
		MCMC-Moda	0,77726	-0,16840	0,10014	0,12850	17,8%
		MCMC-Mediana	0,70044	-0,24522	0,04665	0,10679	25,9%
		INLA-Média	0,94287	-0,00279	0,00077	0,00078	0,3%
		INLA-Moda	0,95413	0,00847	0,00036	0,00043	0,9%
		INLA-Mediana	0,94818	0,00252	0,00053	0,00053	0,3%
ρ^{**}	Clássico	0,98331	-0,00111	0,00003	0,00003	0,1%	
	MCMC-Média	0,98942	0,00500	0,00001	0,00004	0,5%	
	MCMC-Moda	0,99092	0,00650	0,00004	0,00008	0,7%	
	MCMC-Mediana	0,98969	0,00526	0,00002	0,00004	0,5%	
	INLA-Média	0,98191	-0,00251	0,00003	0,00004	0,3%	
	INLA-Moda	0,98346	-0,00097	0,00003	0,00003	0,1%	
	INLA-Mediana	0,98244	-0,00198	0,00003	0,00003	0,2%	

6.2. Estimativas pontuais para o cenário 2.

Cenário	Parâmetro	Método	Média	Vício	Variância	EQM	Vício Relativo
Cenário 2	μ	Clássico	-1,29065	-0,04270	1,89251	1,89433	3,4%
		MCMC-Média	-1,03074	0,21721	2,14094	2,18812	17,4%
		MCMC-Moda	-1,14965	0,09830	2,45979	2,46945	7,9%
		MCMC-Mediana	-1,06492	0,18303	2,14136	2,17487	14,7%
		INLA-Média	-1,29090	-0,04295	1,89123	1,89307	3,4%
		INLA-Moda	-1,29090	-0,04295	1,89123	1,89307	3,4%
		INLA-Mediana	-1,29091	-0,04296	1,89123	1,89307	3,4%
	σ^2_{π}	Clássico	52,15329	0,16555	185,64551	185,67291	13,3%
		MCMC-Média	58,56055	6,57281	235,43152	278,63334	12,6%
		MCMC-Moda	51,02594	-0,96180	180,45252	181,37757	1,9%
		MCMC-Mediana	55,56286	3,57512	211,97486	224,75633	6,9%
		INLA-Média	51,76949	-0,21825	185,62351	185,67115	0,4%
		INLA-Moda	45,55971	-6,42803	144,15840	185,47793	12,4%
		INLA-Mediana	49,50656	-2,48118	169,85460	176,01088	4,8%
	σ^2_{γ}	Clássico	0,91540	-0,00014	0,65227	0,65227	0,0%
		MCMC-Média	10,64065	9,72511	421,04136	515,61905	1062,2%
		MCMC-Moda	1,30385	0,38831	3,05407	3,20485	42,4%
		MCMC-Mediana	2,42581	1,51027	13,14591	15,42682	165,0%
		INLA-Média	0,85982	-0,05572	0,63065	0,63375	6,1%
		INLA-Moda	0,34189	-0,57366	0,12117	0,45026	62,7%
		INLA-Mediana	0,59612	-0,31942	0,32098	0,42301	34,9%
	$\sigma^2_{\pi\gamma}$	Clássico	1,21896	0,00347	0,05707	0,05709	0,3%
		MCMC-Média	1,26199	0,04650	0,05797	0,06013	3,8%
		MCMC-Moda	1,19836	-0,01712	0,05583	0,05612	1,4%
		MCMC-Mediana	1,23809	0,02260	0,05607	0,05658	1,9%
		INLA-Média	1,22813	0,01265	0,06128	0,06144	1,0%
		INLA-Moda	1,15872	-0,05676	0,05569	0,05891	4,7%
		INLA-Mediana	1,20427	-0,01121	0,05929	0,05942	0,9%
	σ^2_{ε}	Clássico	0,85953	0,00323	0,00630	0,00631	0,4%
		MCMC-Média	0,87164	0,01534	0,00650	0,00673	1,8%
		MCMC-Moda	0,85810	0,00180	0,00657	0,00658	0,2%
		MCMC-Mediana	0,86675	0,01045	0,00644	0,00655	1,2%
		INLA-Média	0,86475	0,00845	0,00645	0,00652	1,0%
		INLA-Moda	0,85046	-0,00584	0,00622	0,00625	0,7%
		INLA-Mediana	0,85974	0,00344	0,00637	0,00638	0,4%
	ρ^*	Clássico	0,94230	-0,00336	0,00049	0,00050	0,4%
		MCMC-Média	0,88374	-0,06192	0,00423	0,00807	6,5%
		MCMC-Moda	0,94099	-0,00467	0,00055	0,00057	0,5%
		MCMC-Mediana	0,91944	-0,02622	0,00229	0,00297	2,8%
		INLA-Média	0,93983	-0,00583	0,00049	0,00052	0,6%
		INLA-Moda	0,94903	0,00337	0,00029	0,00030	0,4%
		INLA-Mediana	0,94386	-0,00180	0,00038	0,00039	0,2%
ρ^{**}	Clássico	0,98342	-0,00100	0,00002	0,00002	0,1%	
	MCMC-Média	0,98498	0,00055	0,00002	0,00002	0,1%	
	MCMC-Moda	0,98601	0,00159	0,00002	0,00002	0,2%	
	MCMC-Mediana	0,98532	0,00090	0,00002	0,00002	0,1%	
	INLA-Média	0,98214	-0,00228	0,00002	0,00003	0,2%	
	INLA-Moda	0,98337	-0,00105	0,00002	0,00002	0,1%	
	INLA-Mediana	0,98255	-0,00187	0,00002	0,00003	0,2%	

6.3. Estimativas pontuais para o cenário 3.

Cenário	Parâmetro	Método	Média	Vício	Variância	EQM	Vício Relativo
Cenário 3	μ	Clássico	-1,27091	-0,02296	2,23325	2,23378	1,8%
		MCMC-Média	-0,89580	0,35215	7,90446	8,02847	28,2%
		MCMC-Moda	-1,03141	0,21654	7,58648	7,63337	17,4%
		MCMC-Mediana	-0,88232	0,36563	6,15692	6,29060	29,3%
		INLA-Média	-1,26597	-0,01802	2,26992	2,27024	1,4%
		INLA-Moda	-1,26597	-0,01802	2,26992	2,27024	1,4%
		INLA-Mediana	-1,26598	-0,01803	2,26991	2,27024	1,4%
	σ^2_{π}	Clássico	51,97360	-0,01414	185,81666	185,81686	1,1%
		MCMC-Média	58,16548	6,17774	233,44038	271,60484	11,9%
		MCMC-Moda	50,62116	-1,36658	179,62506	181,49260	2,6%
		MCMC-Mediana	55,20192	3,21418	210,47903	220,80998	6,2%
		INLA-Média	51,76895	-0,21879	181,88090	181,92877	0,4%
		INLA-Moda	45,50312	-6,48462	141,40604	183,45630	12,5%
		INLA-Mediana	49,49187	-2,49587	166,54413	172,77351	4,8%
	σ^2_{γ}	Clássico	0,98835	0,07280	2,12961	2,13491	8,0%
		MCMC-Média	340,18999	339,27445	47587,19096	162694,34067	37057,1%
		MCMC-Moda	13,61436	12,69882	579,46538	740,72539	1387,0%
		MCMC-Mediana	35,96473	35,04919	3660,75648	4889,20214	3828,2%
		INLA-Média	0,61641	-0,29913	1,61188	1,70136	32,7%
		INLA-Moda	0,12884	-0,78670	0,07765	0,69655	85,9%
		INLA-Mediana	0,29464	-0,62090	0,38037	0,76589	67,8%
	$\sigma^2_{\pi\gamma}$	Clássico	1,21350	-0,00198	0,19020	0,19020	0,2%
		MCMC-Média	1,36937	0,15389	0,22458	0,24826	12,7%
		MCMC-Moda	1,15430	-0,06118	0,17357	0,17731	5,0%
		MCMC-Mediana	1,28489	0,06941	0,20231	0,20713	5,7%
		INLA-Média	1,36908	0,15360	0,41853	0,44212	12,6%
		INLA-Moda	1,15974	-0,05575	0,32952	0,33263	4,6%
		INLA-Mediana	1,29386	0,07837	0,38427	0,39041	6,4%
	σ^2_{ϵ}	Clássico	0,86346	0,00716	0,01618	0,01623	0,8%
		MCMC-Média	0,89270	0,03640	0,01644	0,01777	4,3%
		MCMC-Moda	0,85899	0,00269	0,01539	0,01540	0,3%
		MCMC-Mediana	0,88041	0,02411	0,01601	0,01659	2,8%
		INLA-Média	0,87813	0,02183	0,01701	0,01748	2,5%
		INLA-Moda	0,84381	-0,01249	0,01560	0,01576	1,5%
		INLA-Mediana	0,86625	0,00995	0,01654	0,01664	1,2%
	ρ^*	Clássico	0,94102	-0,00464	0,00111	0,00113	0,5%
		MCMC-Média	0,61081	-0,33485	0,01782	0,12994	35,4%
		MCMC-Moda	0,77211	-0,17355	0,10206	0,13218	18,4%
		MCMC-Mediana	0,69304	-0,25262	0,04484	0,10865	26,7%
		INLA-Média	0,94284	-0,00282	0,00078	0,00078	0,3%
		INLA-Moda	0,95410	0,00844	0,00037	0,00044	0,9%
		INLA-Mediana	0,94811	0,00245	0,00054	0,00054	0,3%
ρ^{**}	Clássico	0,98327	-0,00116	0,00003	0,00003	0,1%	
	MCMC-Média	0,98944	0,00501	0,00001	0,00004	0,5%	
	MCMC-Moda	0,99118	0,00676	0,00004	0,00008	0,7%	
	MCMC-Mediana	0,98974	0,00532	0,00002	0,00004	0,5%	
	INLA-Média	0,98177	-0,00265	0,00003	0,00004	0,3%	
	INLA-Moda	0,98344	-0,00098	0,00003	0,00003	0,1%	
	INLA-Mediana	0,98234	-0,00209	0,00003	0,00003	0,2%	

6.4. Estimativas pontuais para o cenário 4.

Cenário	Parâmetro	Método	Média	Vício	Variância	EQM	Vício Relativo
Cenário 4	μ	Clássico	-1,29161	-0,04366	1,89005	1,89195	3,5%
		MCMC-Média	-1,11719	0,13076	2,16654	2,18364	10,5%
		MCMC-Moda	-1,20219	0,04576	2,29362	2,29571	3,7%
		MCMC-Mediana	-1,14702	0,10093	2,07979	2,08998	8,1%
		INLA-Média	-1,28232	-0,03437	1,87505	1,87623	2,8%
		INLA-Moda	-1,28232	-0,03437	1,87505	1,87623	2,8%
		INLA-Mediana	-1,28233	-0,03438	1,87505	1,87623	2,8%
	σ^2_{π}	Clássico	52,14077	0,15303	185,69073	185,71415	12,3%
		MCMC-Média	58,45136	6,46362	234,47693	276,25530	12,4%
		MCMC-Moda	50,92641	-1,06134	184,81698	185,94342	2,0%
		MCMC-Mediana	55,49284	3,50510	212,03332	224,31904	6,7%
		INLA-Média	51,82319	-0,16455	183,95774	183,98482	0,3%
		INLA-Moda	45,61183	-6,37591	142,88966	183,54195	12,3%
		INLA-Mediana	49,55892	-2,42882	168,34701	174,24616	4,7%
	σ^2_{γ}	Clássico	0,91622	0,00068	0,65127	0,65127	0,1%
		MCMC-Média	9,29902	8,38348	342,02678	412,30951	915,7%
		MCMC-Moda	1,29253	0,37698	3,24120	3,38332	41,2%
		MCMC-Mediana	2,21220	1,29666	8,06320	9,74453	141,6%
		INLA-Média	0,86872	-0,04683	0,63290	0,63509	5,1%
		INLA-Moda	0,34439	-0,57115	0,12201	0,44822	62,4%
		INLA-Mediana	0,60203	-0,31351	0,32210	0,42038	34,2%
	$\sigma^2_{\pi\gamma}$	Clássico	1,22033	0,00485	0,05966	0,05968	0,4%
		MCMC-Média	1,23941	0,02393	0,05937	0,05994	2,0%
		MCMC-Moda	1,19489	-0,02059	0,05825	0,05868	1,7%
		MCMC-Mediana	1,23941	0,02393	0,05937	0,05994	2,0%
		INLA-Média	1,22843	0,01294	0,06677	0,06694	1,1%
		INLA-Moda	1,15709	-0,05839	0,06066	0,06407	4,8%
		INLA-Mediana	1,20395	-0,01153	0,06458	0,06471	0,9%
	σ^2_{ϵ}	Clássico	0,86054	0,00424	0,00756	0,00758	0,5%
		MCMC-Média	0,87533	0,01903	0,00791	0,00828	2,2%
		MCMC-Moda	0,85862	0,00232	0,00769	0,00770	0,3%
		MCMC-Mediana	0,86945	0,01315	0,00779	0,00796	1,5%
		INLA-Média	0,86745	0,01115	0,00758	0,00771	1,3%
		INLA-Moda	0,85057	-0,00573	0,00726	0,00730	0,7%
		INLA-Mediana	0,86158	0,00528	0,00748	0,00750	0,6%
	ρ^*	Clássico	0,94221	-0,00345	0,00050	0,00051	0,4%
		MCMC-Média	0,88703	-0,05863	0,00344	0,00688	6,2%
		MCMC-Moda	0,94098	-0,00468	0,00056	0,00058	0,5%
		MCMC-Mediana	0,92158	-0,02408	0,00172	0,00230	2,5%
		INLA-Média	0,93969	-0,00597	0,00049	0,00053	0,6%
		INLA-Moda	0,94905	0,00339	0,00029	0,00031	0,4%
		INLA-Mediana	0,94377	-0,00189	0,00039	0,00039	0,2%
ρ^{**}	Clássico	0,98339	-0,00103	0,00002	0,00002	0,1%	
	MCMC-Média	0,98481	0,00039	0,00002	0,00002	0,0%	
	MCMC-Moda	0,98581	0,00139	0,00002	0,00002	0,1%	
	MCMC-Mediana	0,98517	0,00075	0,00002	0,00002	0,1%	
	INLA-Média	0,98211	-0,00231	0,00003	0,00003	0,2%	
	INLA-Moda	0,98338	-0,00104	0,00002	0,00002	0,1%	
	INLA-Mediana	0,98254	-0,00188	0,00002	0,00003	0,2%	

6.5. Estimativas dos intervalos de confiança e credibilidade para o cenário 1.

Cenário	Parâmetro	Método	Amplitude	Cobertura
Cenário 1	μ	Clássico-Bootstrap	5,129449	90,5
		MCMC-Central	13,23142	97,9
		MCMC-HPD	12,90344	98,1
		INLA-Central	5,548911	79,4
		INLA-HPD	5,546538	79,3
	σ^2_{π}	Clássico-Bootstrap	49,07694	89,4
		MCMC-Central	67,15846	94,9
		MCMC-HPD	61,82171	95,2
		INLA-Central	54,98633	80,5
		INLA-HPD	51,98632	79,4
	σ^2_{γ}	Clássico-Bootstrap	1,500316	46,4
		MCMC-Central	3188,255	86,1
		MCMC-HPD	1662,667	99
		INLA-Central	3,142427	25,6
		INLA-HPD	2,025207	25,4
	$\sigma^2_{\pi\gamma}$	Clássico-Bootstrap	1,625322	89,5
		MCMC-Central	1,929614	94,5
		MCMC-HPD	1,779147	95
		INLA-Central	1,746456	73,7
		INLA-HPD	1,650897	75
	σ^2_{ε}	Clássico-Bootstrap	0,60777	92,4
		MCMC-Central	0,468427	94,9
		MCMC-HPD	0,453128	95,1
		INLA-Central	0,446962	80,8
		INLA-HPD	0,440258	80,6
	ρ^*	Clássico-Bootstrap	0,079246	82,3
		MCMC-Central	0,934377	85,4
		MCMC-HPD	0,908454	79,2
		INLA-Central	0,096006	74,4
		INLA-HPD	0,082946	72,9
ρ^{**}	Clássico-Bootstrap	0,022995	92,2	
	MCMC-Central	0,023239	92	
	MCMC-HPD	0,021267	83,7	
	INLA-Central	0,01953	80,4	
	INLA-HPD	0,019027	82,2	

6.6. Estimativas dos intervalos de confiança e credibilidade para o cenário 2.

Cenário	Parâmetro	Método	Amplitude	Cobertura
Cenário 2	μ	Clássico-Bootstrap	5,118575	92,1
		MCMC-Central	6,887281	97,2
		MCMC-HPD	6,676705	96,6
		INLA-Central	5,477754	85,7
		INLA-HPD	5,475523	85,7
	σ^2_{π}	Clássico-Bootstrap	48,74553	89,1
		MCMC-Central	67,39163	95,3
		MCMC-HPD	62,01059	95,5
		INLA-Central	54,50322	85
		INLA-HPD	51,52864	83,9
	σ^2_{γ}	Clássico-Bootstrap	0,941689	46,8
		MCMC-Central	67,16253	93,5
		MCMC-HPD	37,47882	98,4
		INLA-Central	2,997024	71,8
		INLA-HPD	0,85689	90,1
	$\sigma^2_{\pi\gamma}$	Clássico-Bootstrap	0,85689	90,1
		MCMC-Central	0,971774	94,8
		MCMC-HPD	0,935103	94,5
		INLA-Central	0,920857	85,1
		INLA-HPD	0,902117	85,2
	σ^2_{ε}	Clássico-Bootstrap	0,301198	93,1
		MCMC-Central	0,320793	95,2
		MCMC-HPD	0,312692	95
		INLA-Central	0,309551	86,4
		INLA-HPD	0,307124	86,6
	ρ^*	Clássico-Bootstrap	0,067823	82,9
		MCMC-Central	0,383103	95
		MCMC-HPD	0,299941	98,4
INLA-Central		0,084781	85	
INLA-HPD		0,076543	83,8	
ρ^{**}	Clássico-Bootstrap	0,019261	89,5	
	MCMC-Central	0,017902	95,7	
	MCMC-HPD	0,971774	94,8	
	INLA-Central	0,017828	85,1	
	INLA-HPD	0,017475	86,7	

6.7. Estimativas dos intervalos de confiança e credibilidade para o cenário 3.

Cenário	Parâmetro	Método	Amplitude	Cobertura
Cenário 3	μ	Clássico-Bootstrap	5,118289	90,1
		MCMC-Central	13,63767	98,1
		MCMC-HPD	13,26028	98,3
		INLA-Central	5,541584	80,8
		INLA-HPD	5,53922	80,9
	σ^2_{π}	Clássico-Bootstrap	49,04101	89,2
		MCMC-Central	67,42458	94,6
		MCMC-HPD	62,04345	94,9
		INLA-Central	54,9485	82,2
		INLA-HPD	51,95298	81,7
	σ^2_{γ}	Clássico-Bootstrap	1,471085	46,4
		MCMC-Central	3316,222	85,5
		MCMC-HPD	1738,988	98,9
		INLA-Central	3,06815	25,5
		INLA-HPD	1,977313	25,5
	$\sigma^2_{\pi\gamma}$	Clássico-Bootstrap	1,492364	89,2
		MCMC-Central	1,97206	94,9
		MCMC-HPD	1,819384	95,3
		INLA-Central	1,813856	75,6
		INLA-HPD	1,713721	74,4
	σ^2_{ε}	Clássico-Bootstrap	0,463269	92,9
		MCMC-Central	0,512699	94,3
		MCMC-HPD	0,494708	94,7
		INLA-Central	0,490061	82,3
		INLA-HPD	0,481507	82,4
ρ^*	Clássico-Bootstrap	0,07765	80,4	
	MCMC-Central	0,936918	85,3	
	MCMC-HPD	0,914184	79,1	
	INLA-Central	0,095982	76,2	
	INLA-HPD	0,083219	74	
ρ^{**}	Clássico-Bootstrap	0,021318	89,8	
	MCMC-Central	0,023571	90,2	
	MCMC-HPD	0,021536	83,3	
	INLA-Central	0,020076	82,2	
	INLA-HPD	0,01952	83,3	

6.8. Estimativas dos intervalos de confiança e credibilidade para o cenário 4.

Cenário	Parâmetro	Método	Amplitude	Cobertura
Cenário 4	μ	Clássico-Bootstrap	5,129449	90,5
		MCMC-Central	6,741301	96,1
		MCMC-HPD	6,544924	95,5
		INLA-Central	5,485123	84,5
		INLA-HPD	5,482889	84,5
	σ^2_{π}	Clássico-Bootstrap	49,07694	89,4
		MCMC-Central	67,31924	95,1
		MCMC-HPD	61,86068	95,8
		INLA-Central	54,56286	83,9
		INLA-HPD	51,5863	82,8
	σ^2_{γ}	Clássico-Bootstrap	1,500316	46,4
		MCMC-Central	57,70771	93,5
		MCMC-HPD	32,56328	98,4
		INLA-Central	3,029418	70,8
		INLA-HPD	2,2675	67,8
	$\sigma^2_{\pi\gamma}$	Clássico-Bootstrap	1,625322	89,5
		MCMC-Central	1,00605	95,2
		MCMC-HPD	0,968774	95,7
		INLA-Central	0,951796	84,6
		INLA-HPD	0,932785	83,7
	σ^2_{ε}	Clássico-Bootstrap	0,60777	92,4
		MCMC-Central	0,349395	94,8
		MCMC-HPD	0,340225	94,4
		INLA-Central	0,336598	84
		INLA-HPD	0,333552	84,8
	ρ^*	Clássico-Bootstrap	0,079246	82,3
		MCMC-Central	0,376847	94,2
		MCMC-HPD	0,291261	98,7
INLA-Central		0,085505	84,1	
INLA-HPD		0,077092	83,2	
ρ^{**}	Clássico-Bootstrap	0,022995	92,2	
	MCMC-Central	0,018054	95,4	
	MCMC-HPD	0,017571	94,4	
	INLA-Central	0,018051	84,1	
	INLA-HPD	0,017672	84,7	

6.9. *Boxplot's* para o cenário 1

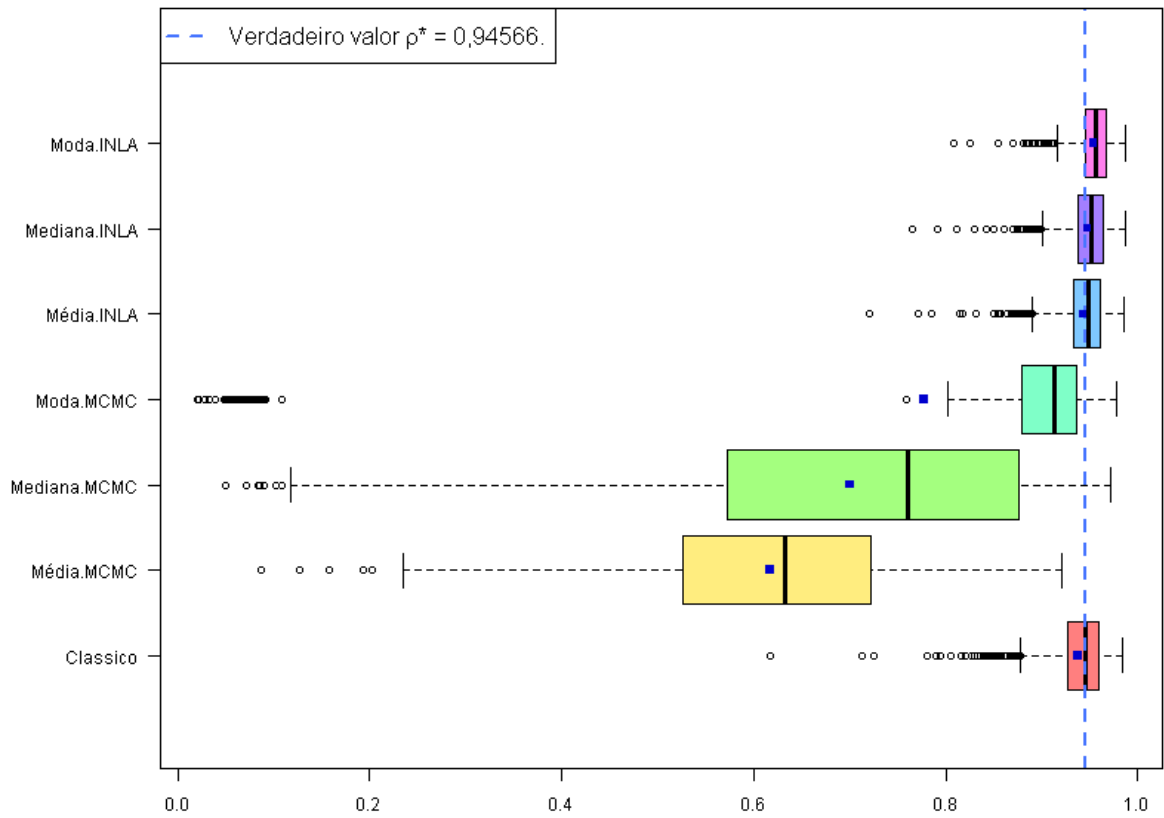


Figura 1: Estimativa de ρ^* para 2 aferidores e dados completos, por método.

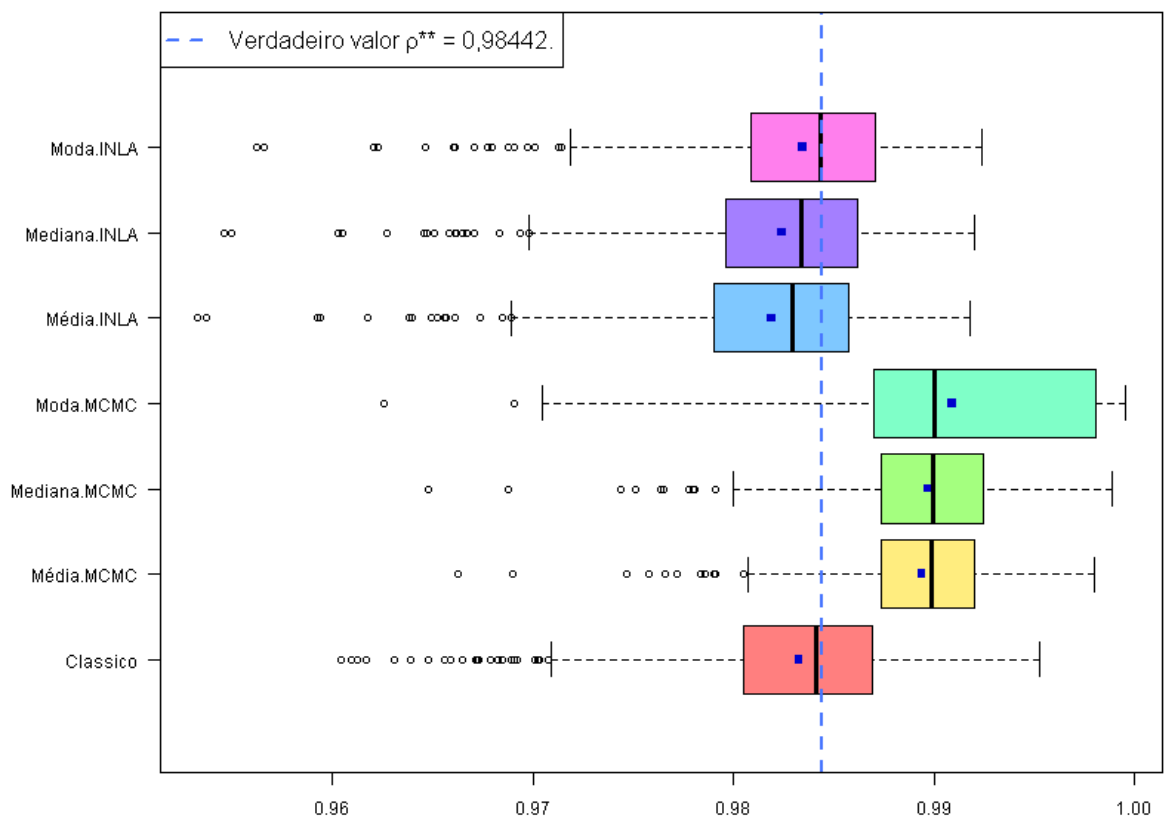


Figura 2: Estimativa de ρ^{**} para 2 aferidores e dados completos, por método.

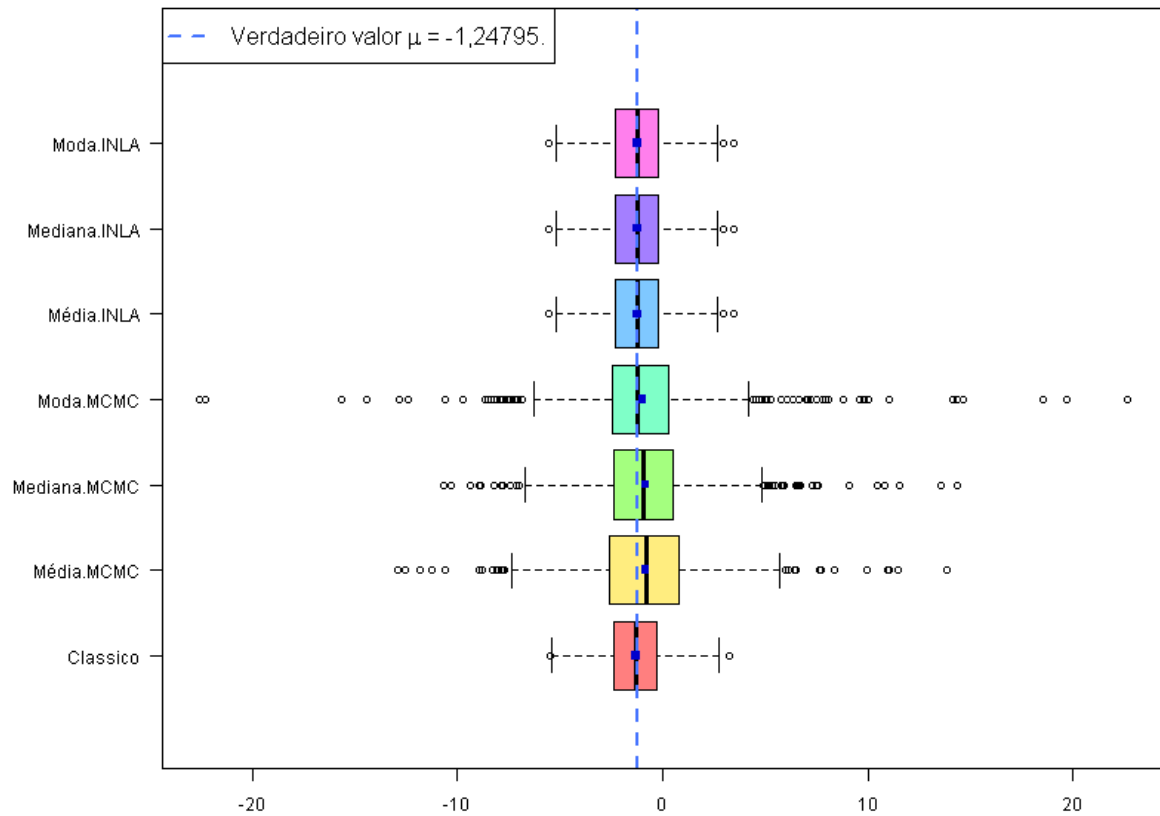


Figura 3: Estimativa de μ para 2 aferidores e dados completos, por método.

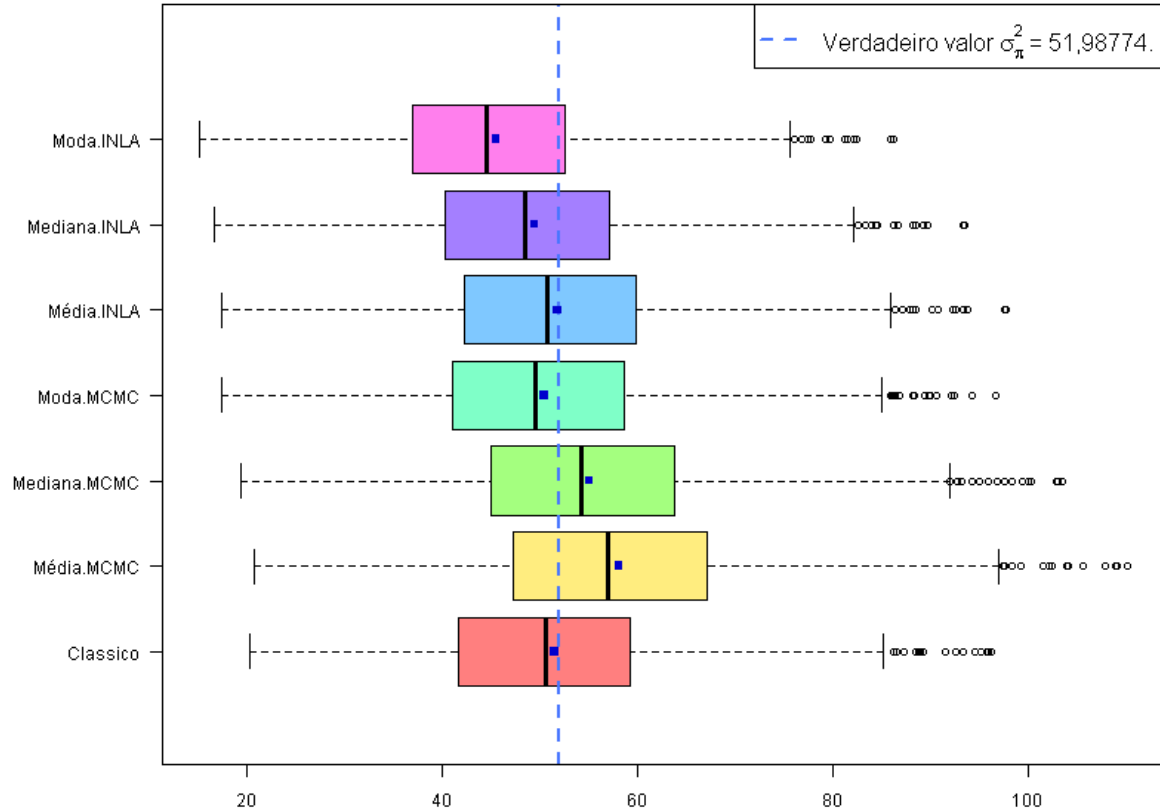


Figura 4: Estimativa de σ_{π}^2 para 2 aferidores e dados completos, por método.

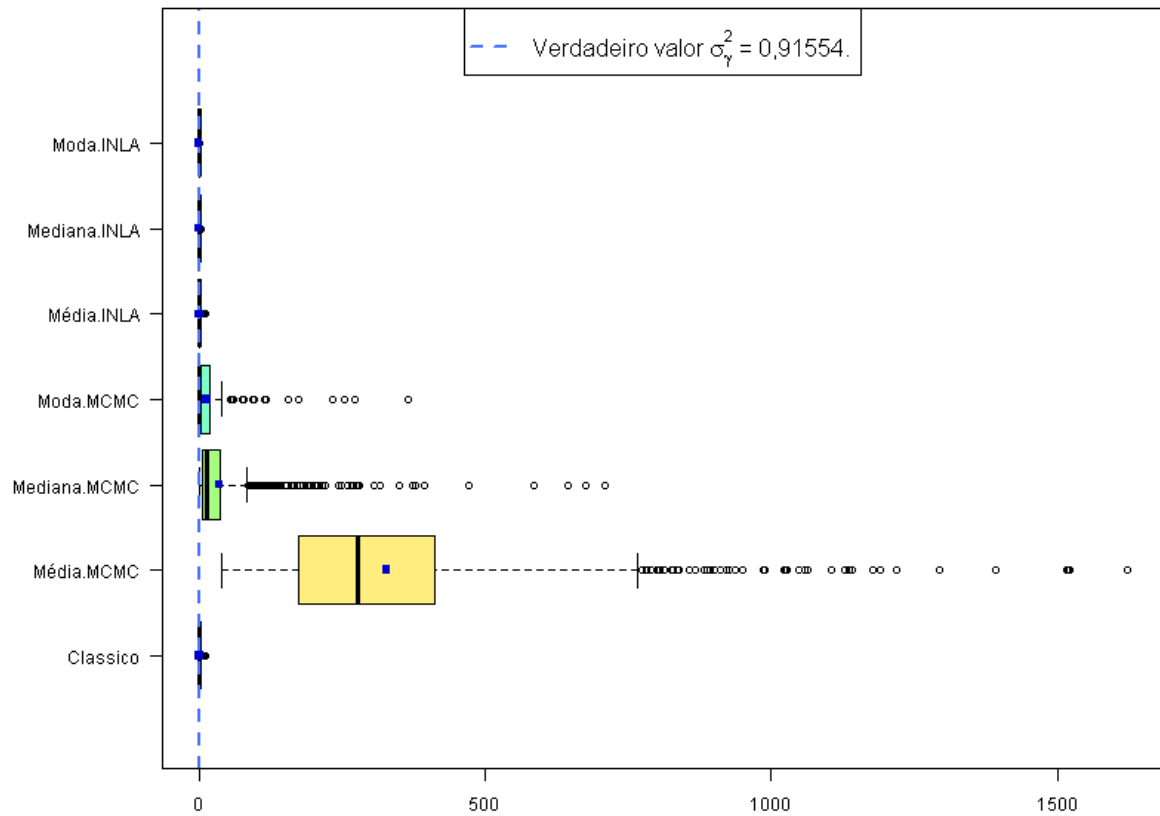


Figura 5: Estimativa de σ_{γ}^2 para 2 aferidores e dados completos, por método.

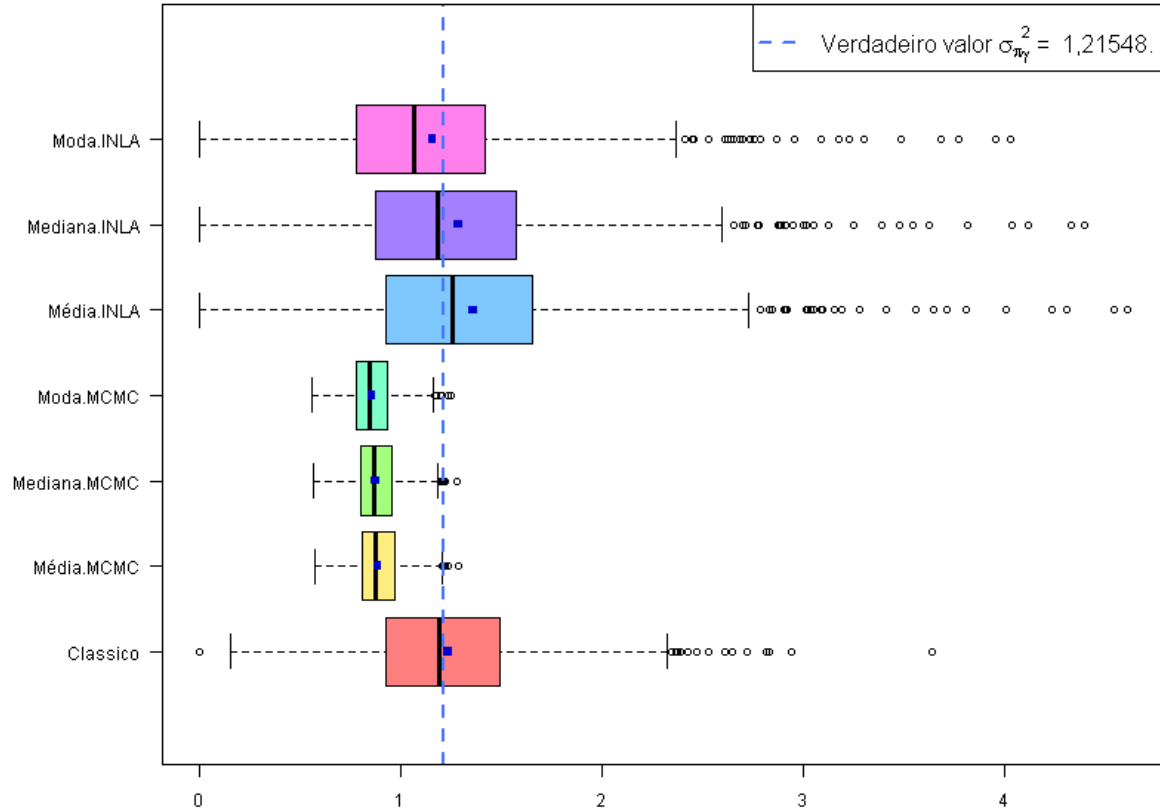


Figura 6: Estimativa de $\sigma_{\pi\gamma}^2$ para 2 aferidores e dados completos, por método.

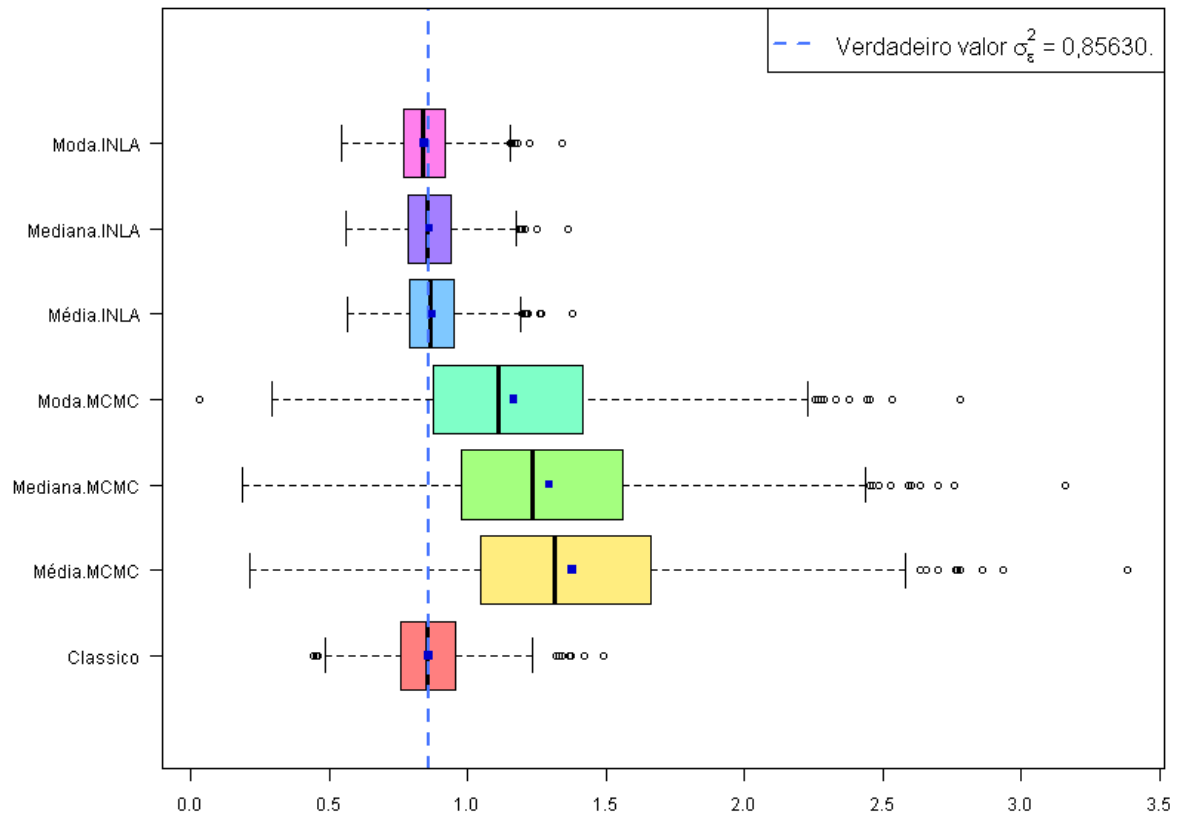


Figura 7: Estimativa de σ_{ϵ}^2 para 2 aferidores e dados completos, por método.

6.10. Boxplot's para o cenário 2

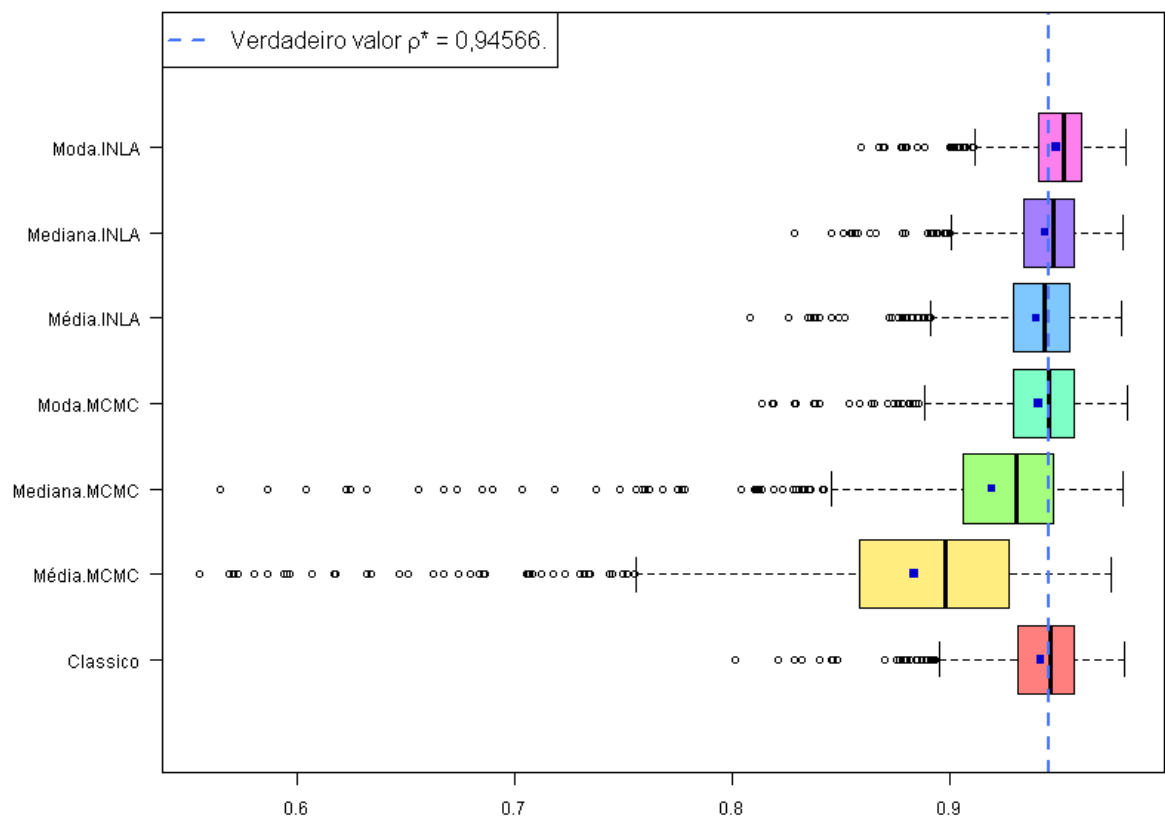


Figura 8: Estimativa de ρ^* para 4 aferidores e dados completos, por método.

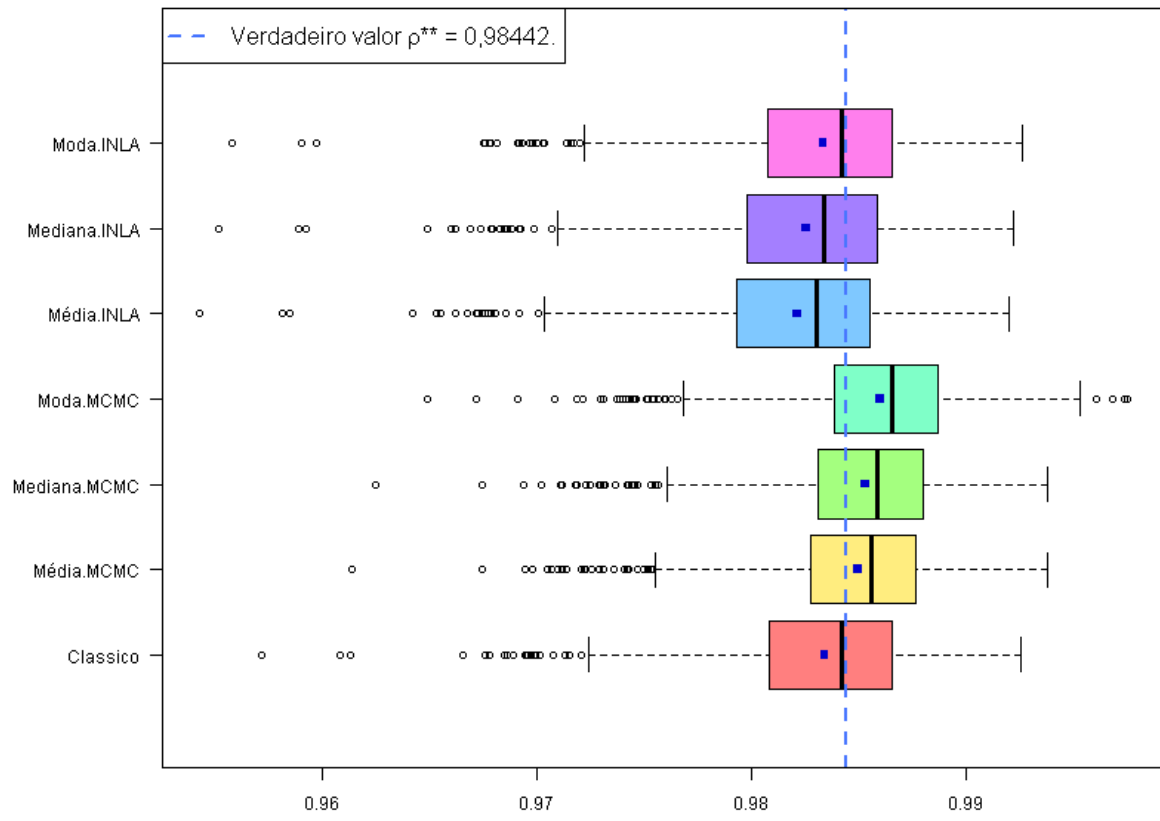


Figura 9: Estimativa de ρ^{**} para 4 aferidores e dados completos, por método.

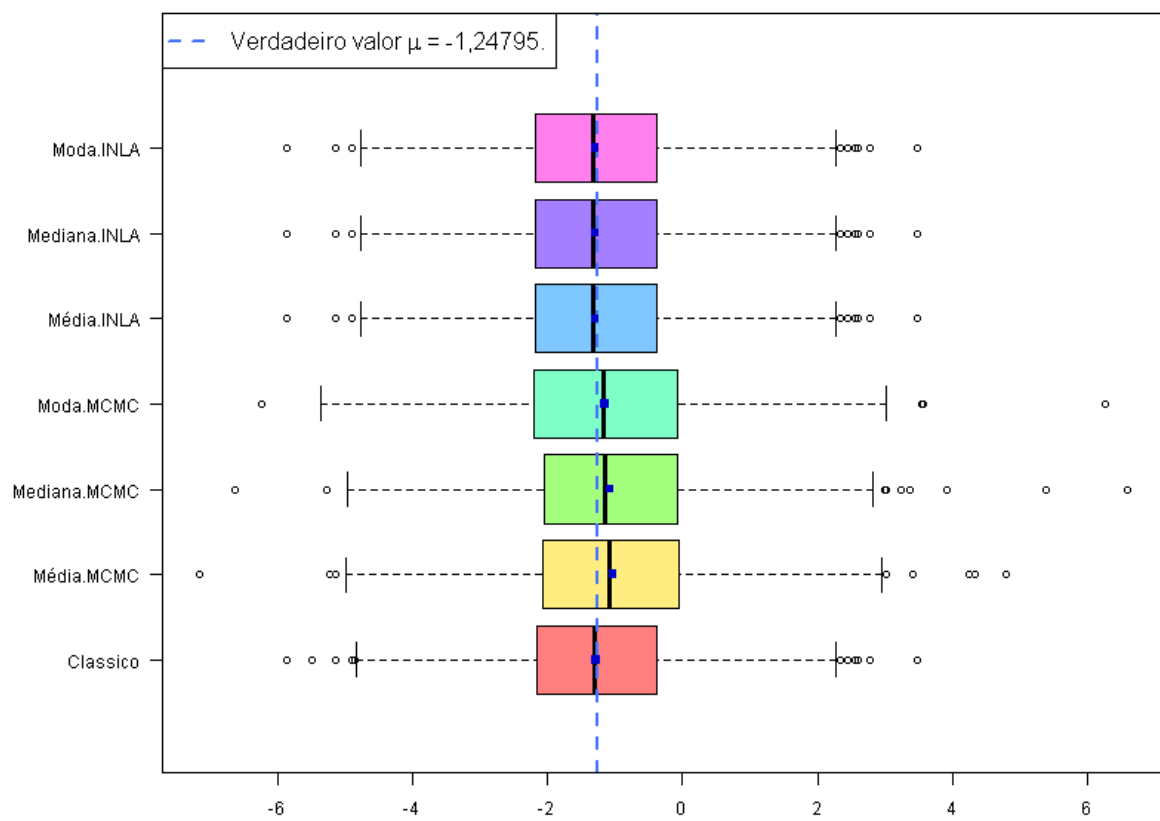


Figura 10: Estimativa de μ para 4 aferidores e dados completos, por método.

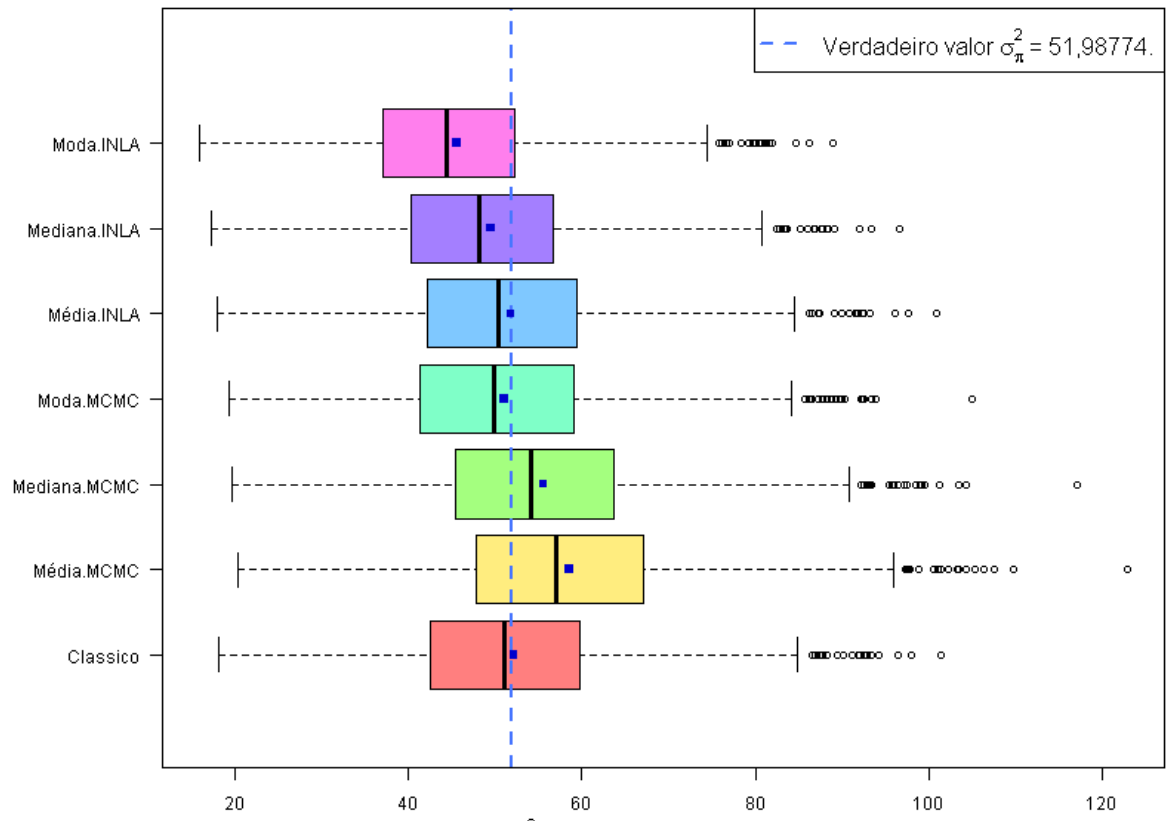


Figura 11: Estimativa de σ_{π}^2 para 4 aferidores e dados completos, por método.

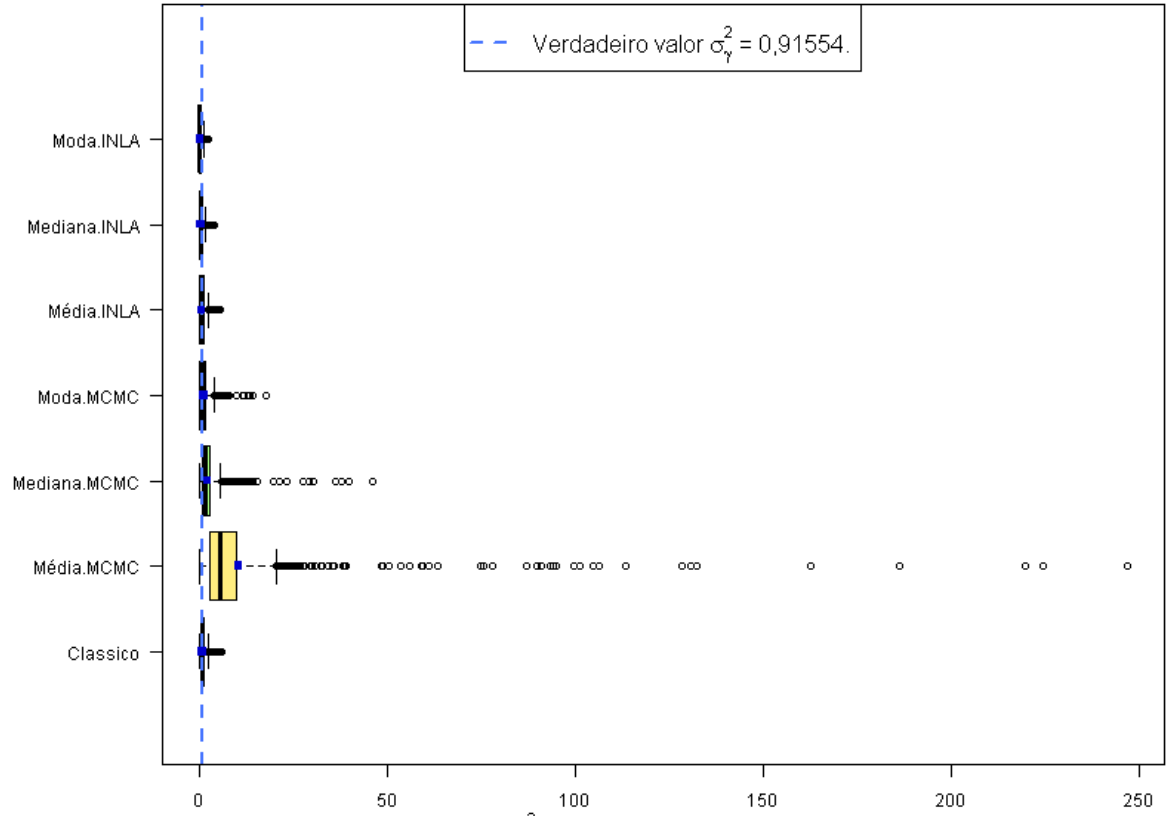


Figura 12: Estimativa de σ_{γ}^2 para 4 aferidores e dados completos, por método.

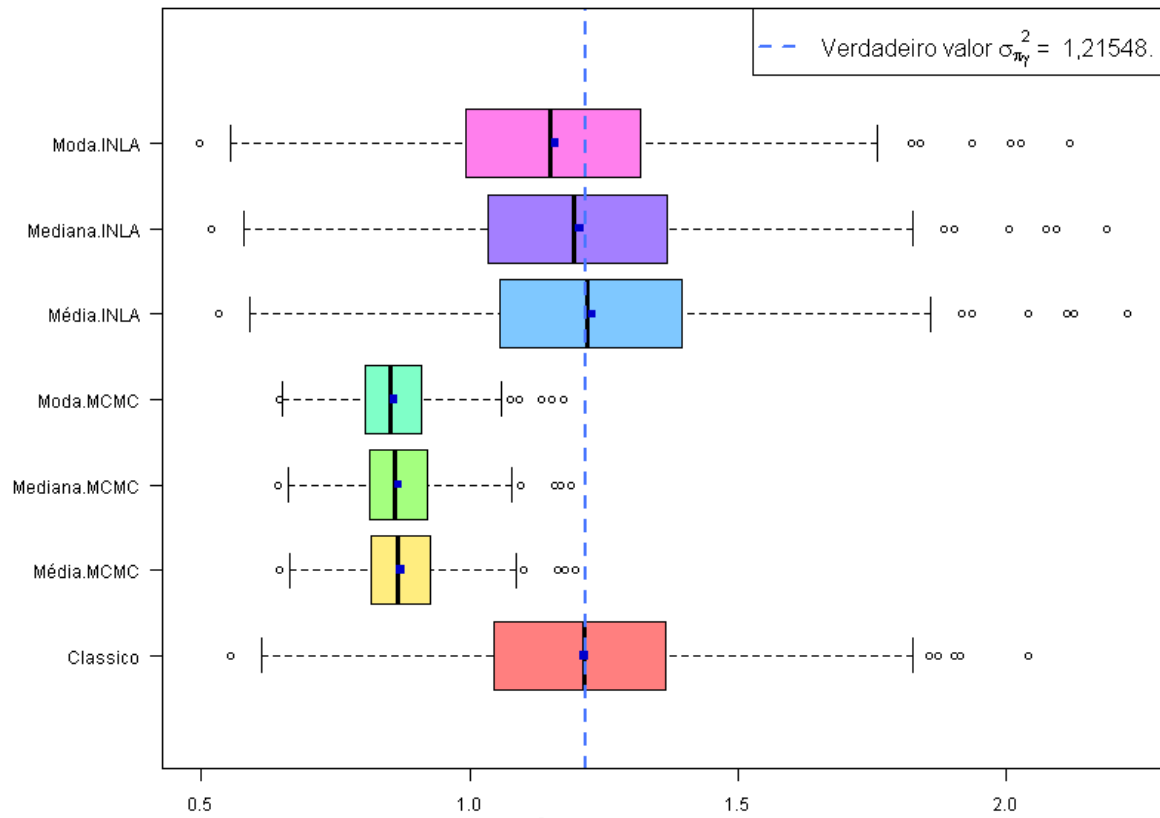


Figura 13: Estimativa de $\sigma_{\pi\gamma}^2$ para 4 aferidores e dados completos, por método.

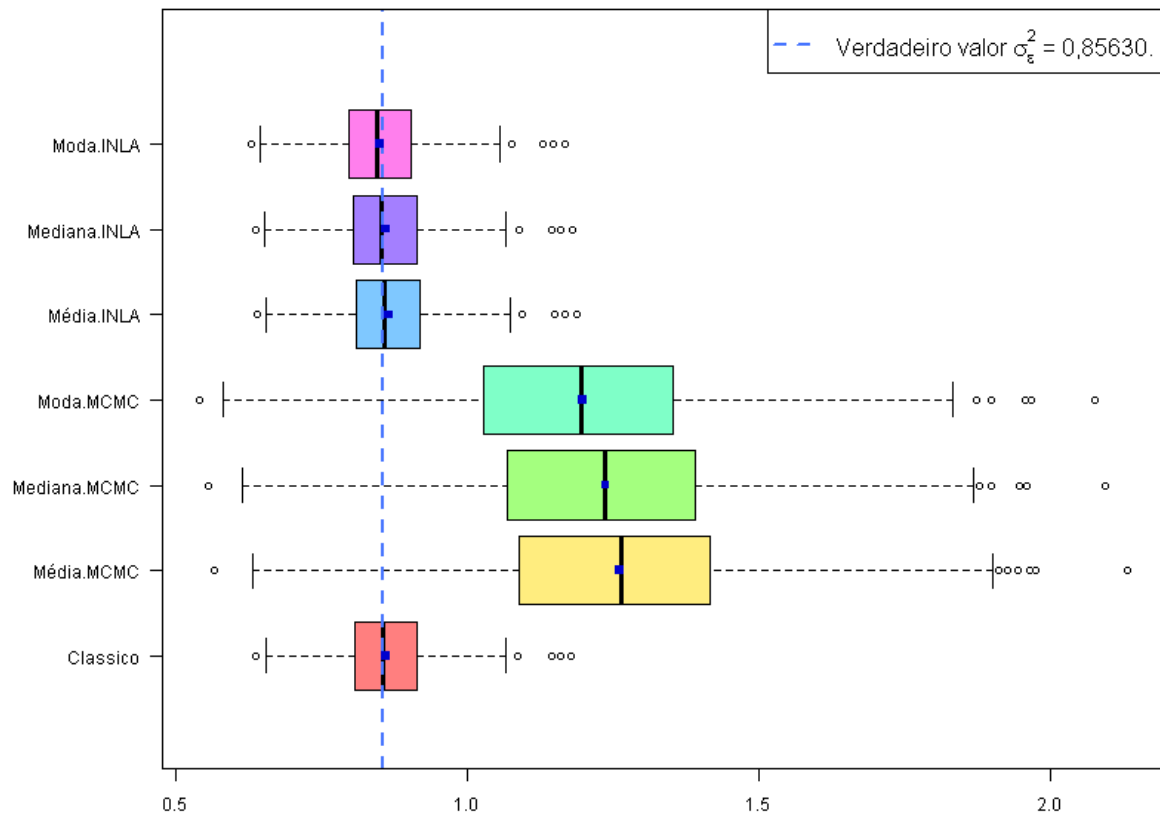


Figura 14: Estimativa de σ_{ϵ}^2 para 4 aferidores e dados completos, por método.

6.11. *Boxplot's* para o cenário 3

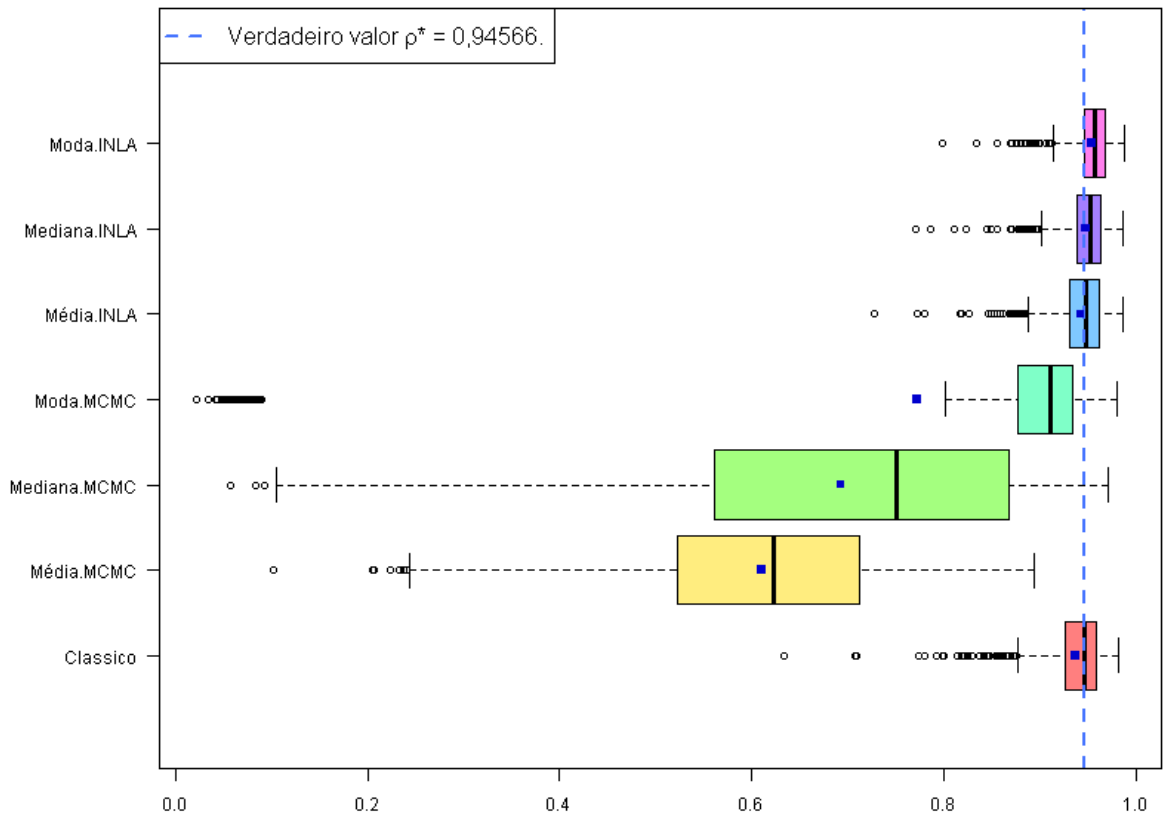


Figura 15: Estimativa de ρ^* para 2 aferidores e dados incompletos, por método.

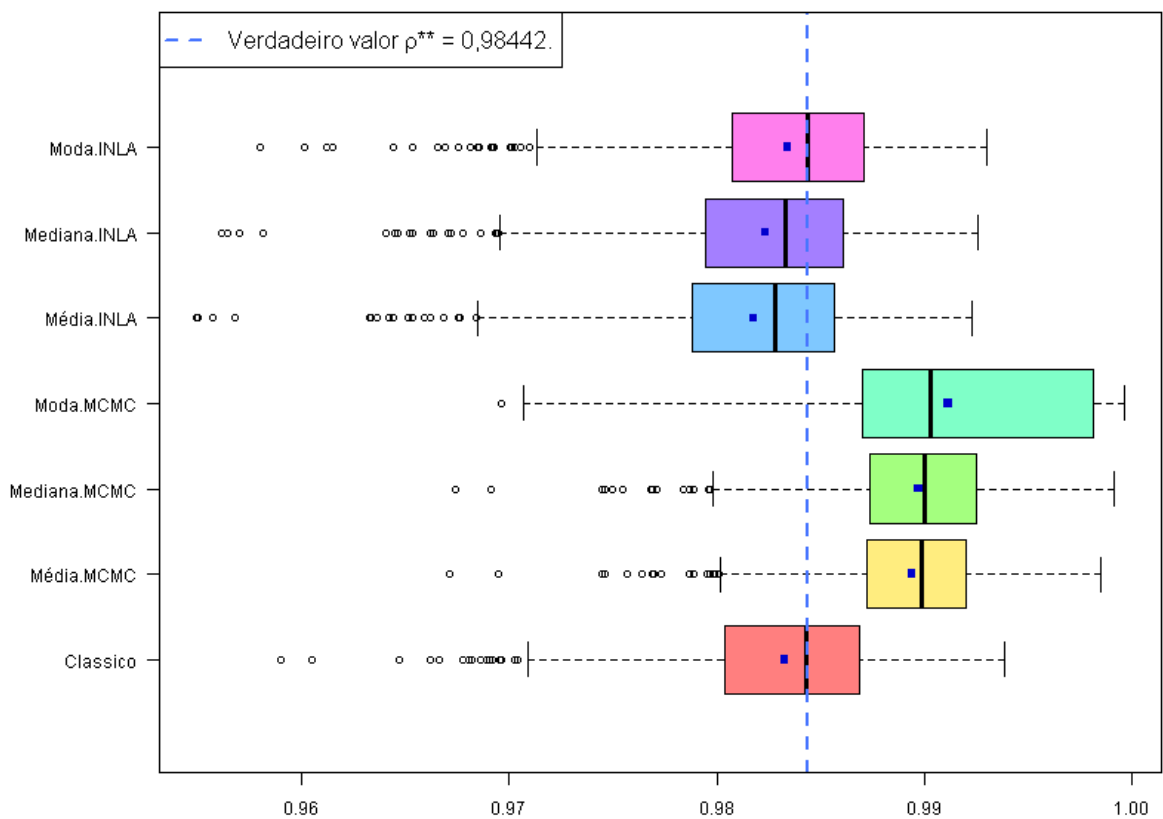


Figura 16: Estimativa de ρ^{**} para 2 aferidores e dados incompletos, por método.

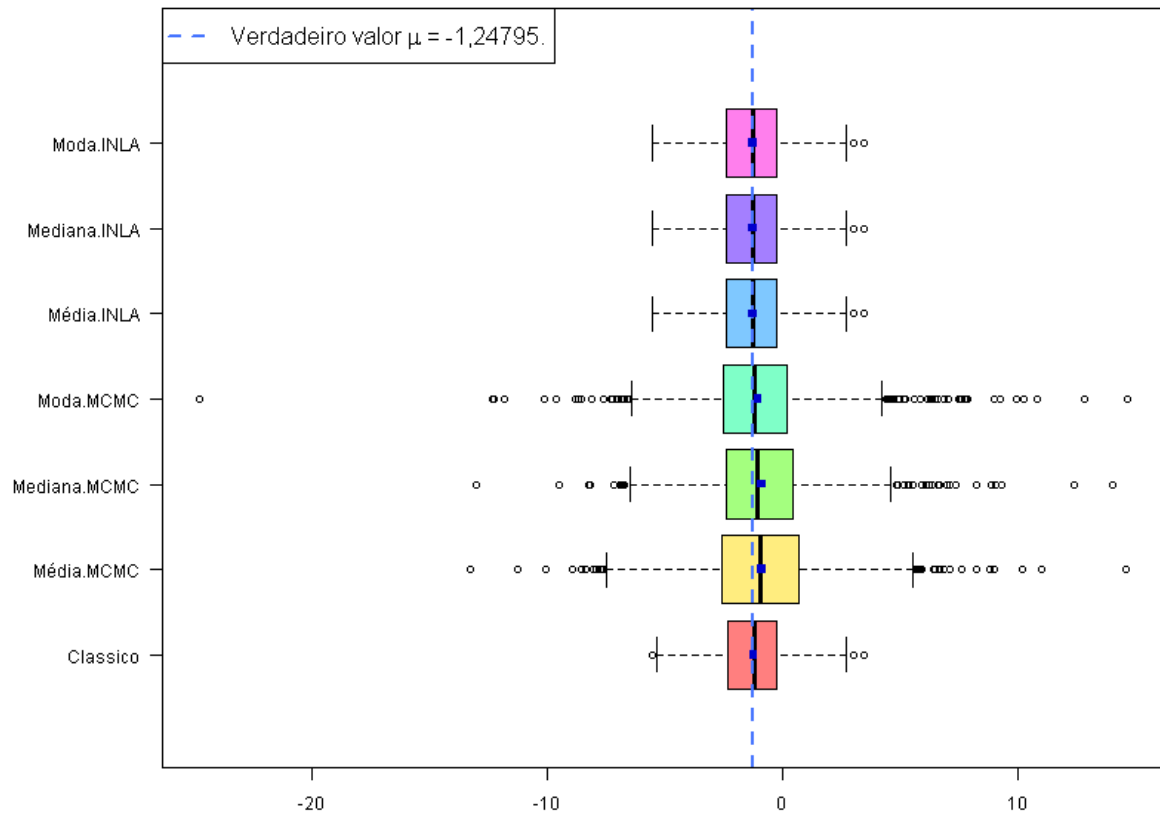


Figura 17: Estimativa de μ para 2 aferidores e dados incompletos, por método.

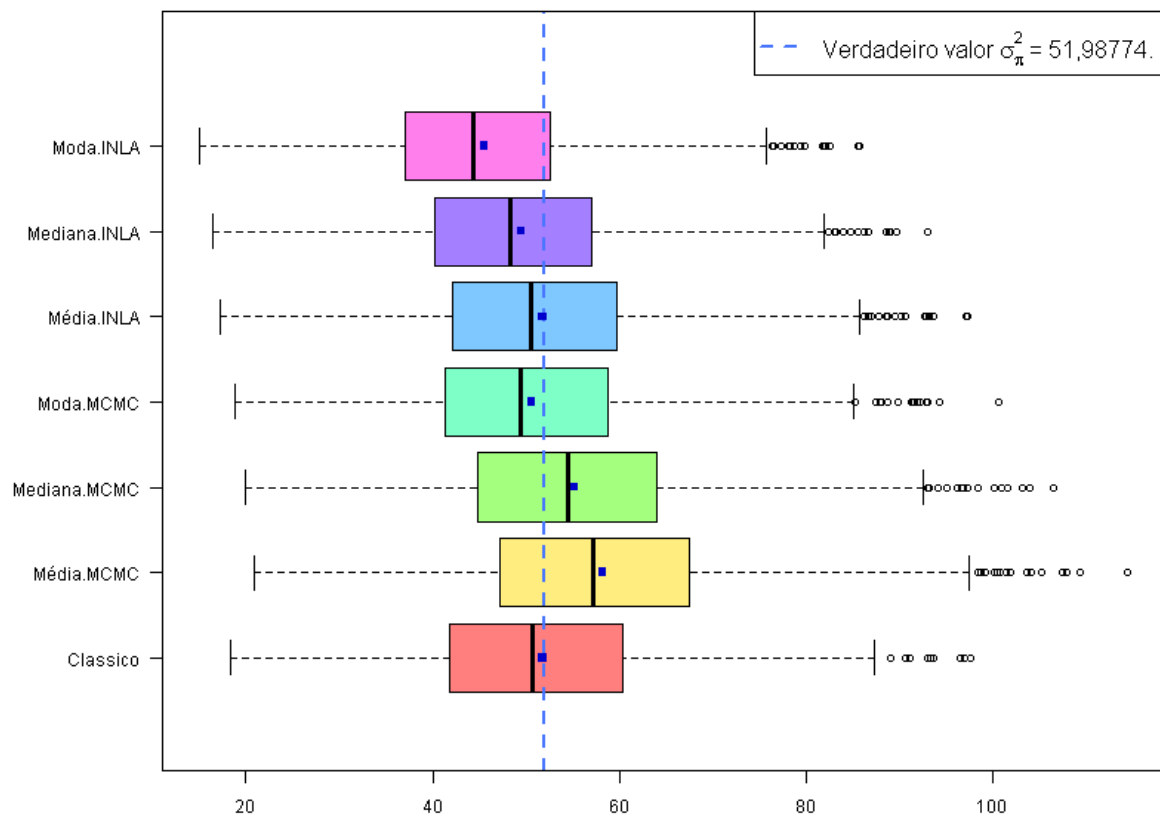


Figura 18: Estimativa de σ_{π}^2 para 2 aferidores e dados incompletos, por método.

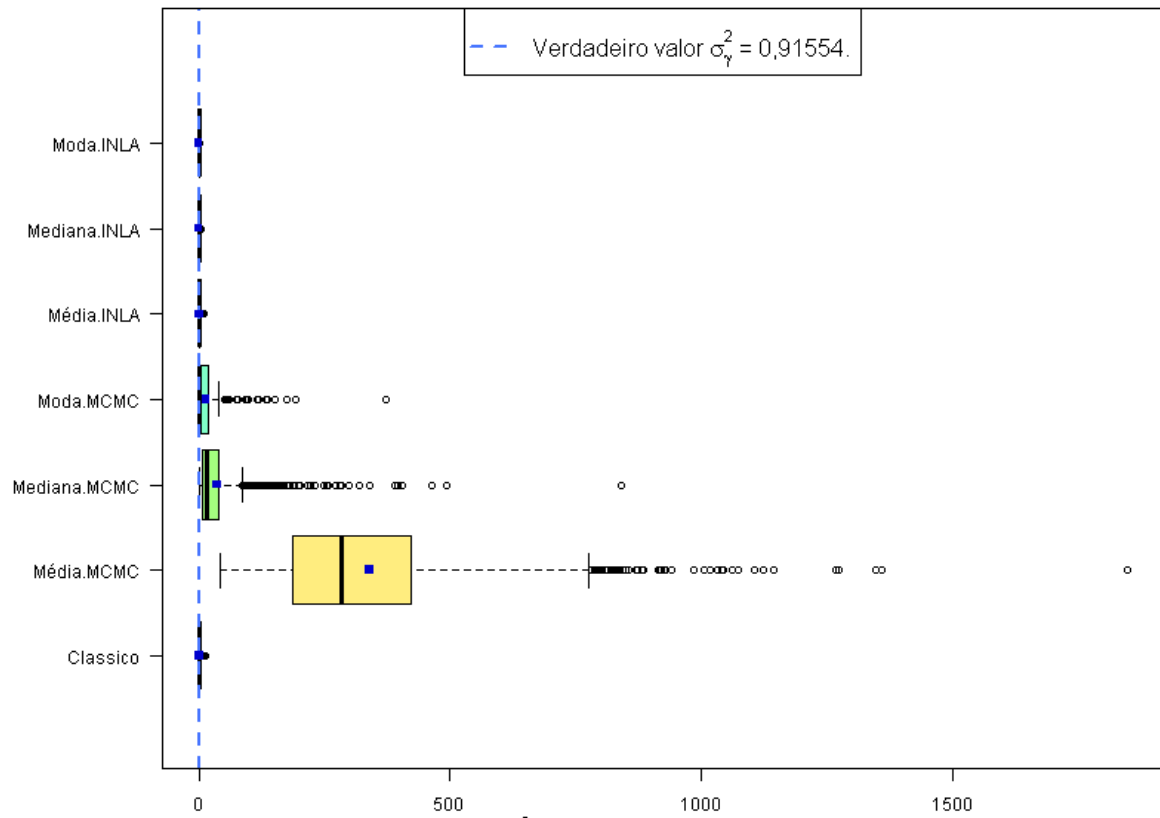


Figura 19: Estimativa de σ_γ^2 para 2 aferidores e dados incompletos, por método.

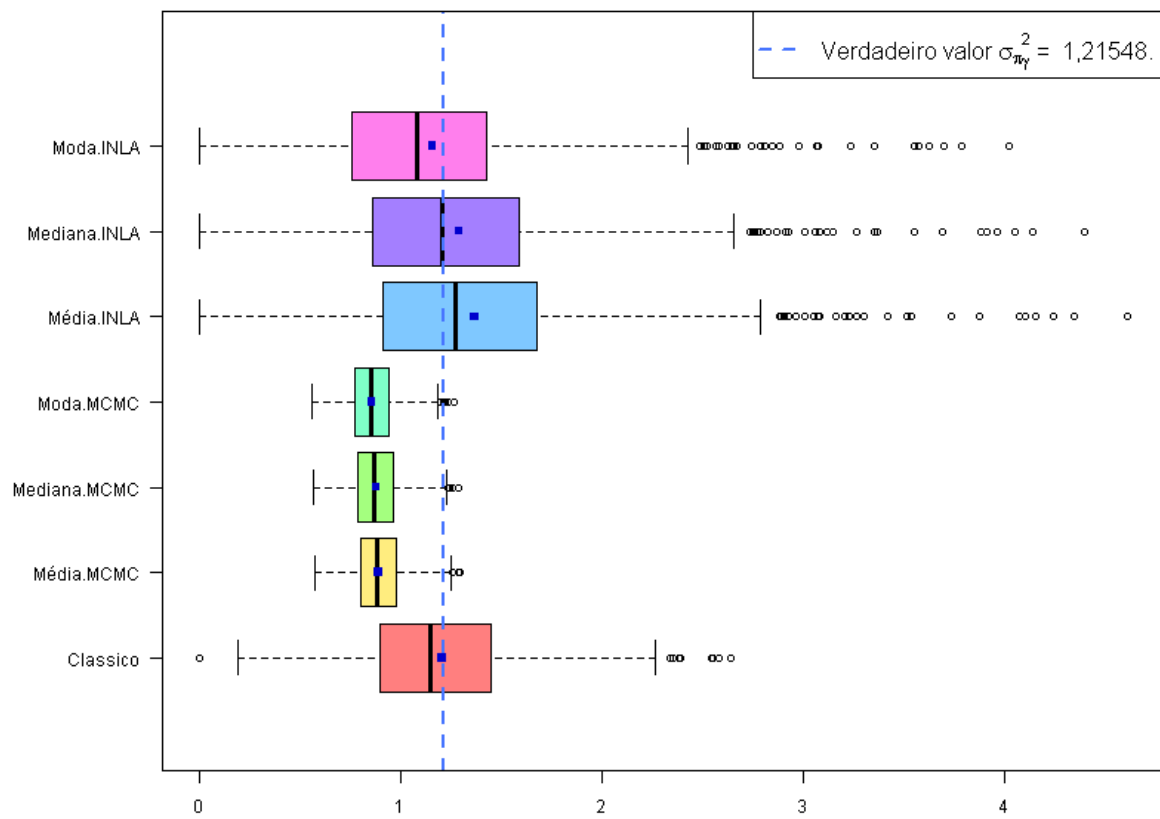


Figura 20: Estimativa de $\sigma_{\pi\gamma}^2$ para 2 aferidores e dados incompletos, por método.

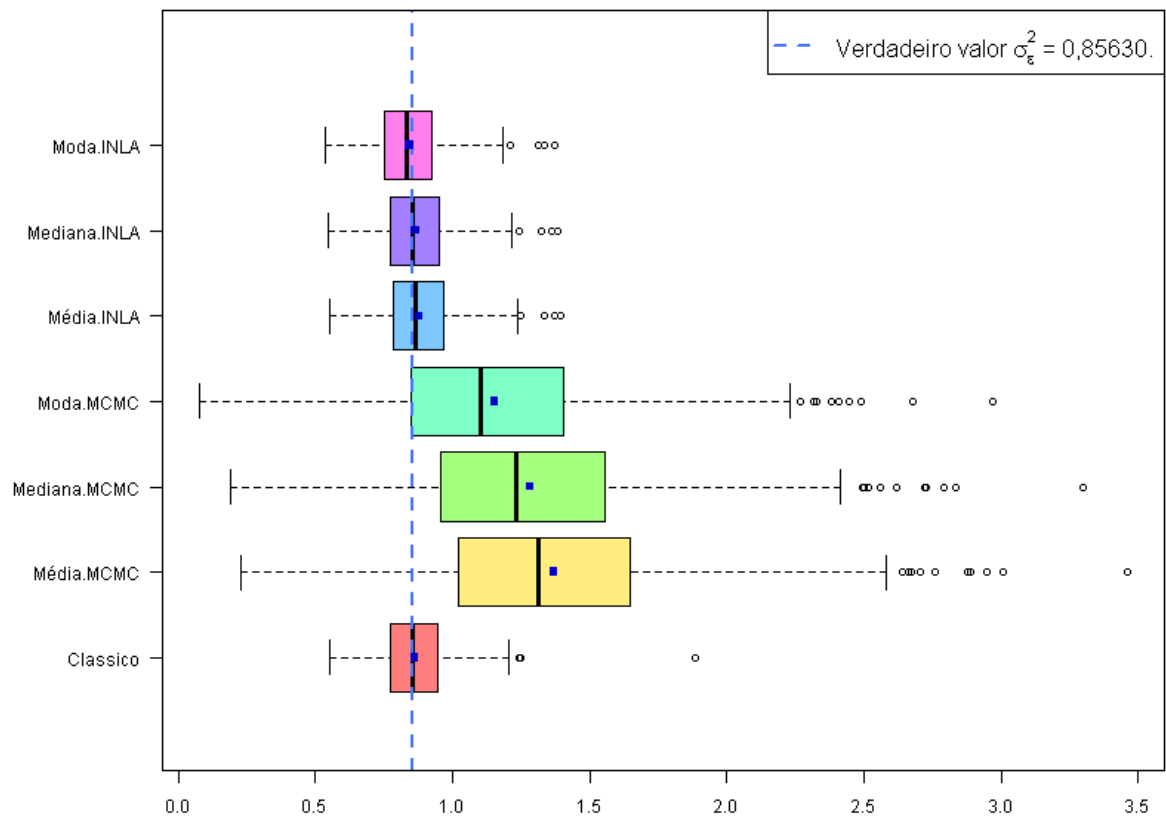


Figura 21: Estimativa de σ_{ϵ}^2 para 2 aferidores e dados incompletos, por método.

6.12. Boxplot's para o cenário 4

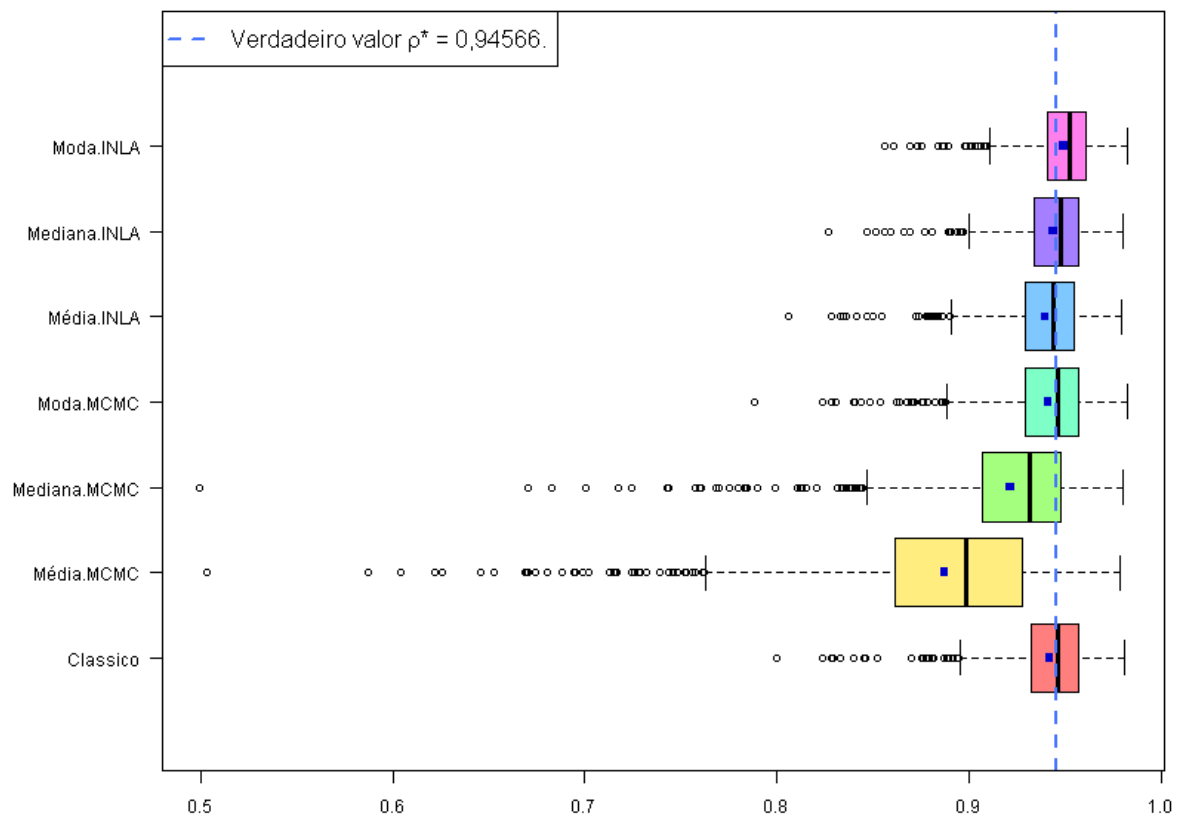


Figura 22: Estimativa de ρ^* para 4 aferidores e dados incompletos, por método.

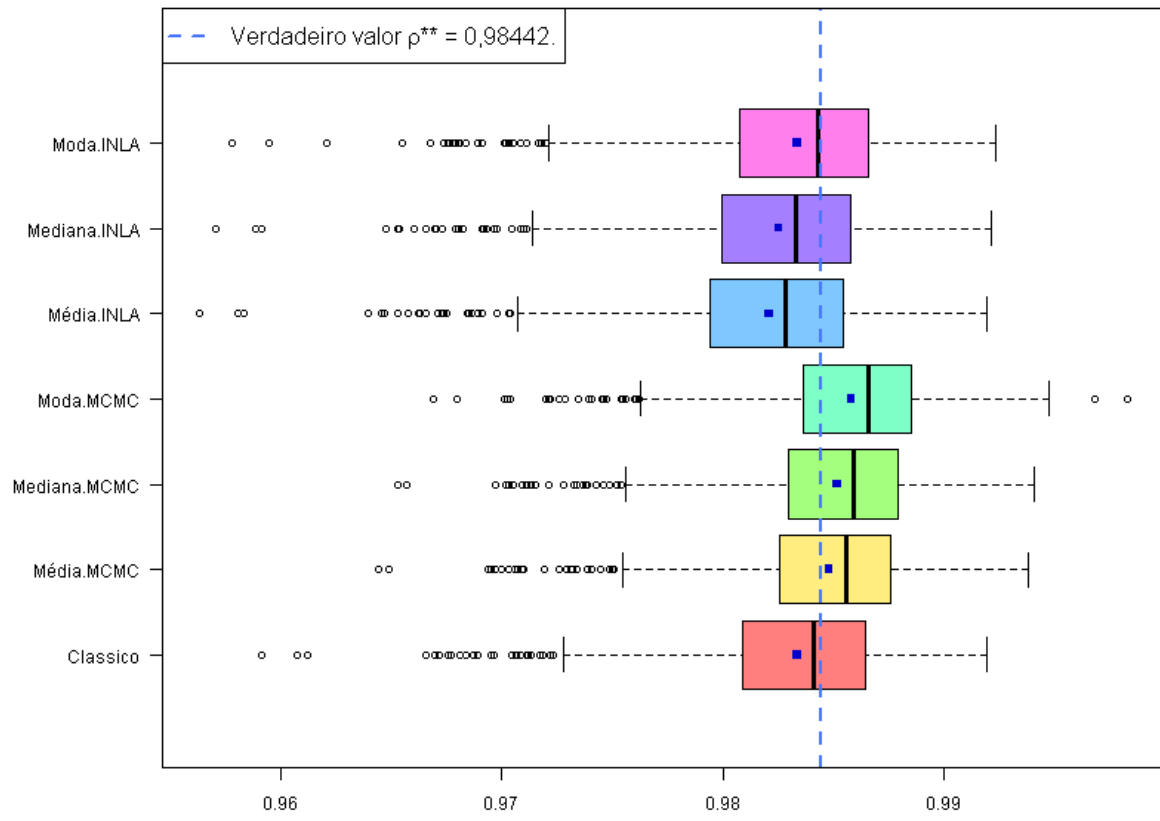


Figura 23: Estimativa de ρ^{**} para 4 aferidores e dados incompletos, por método.

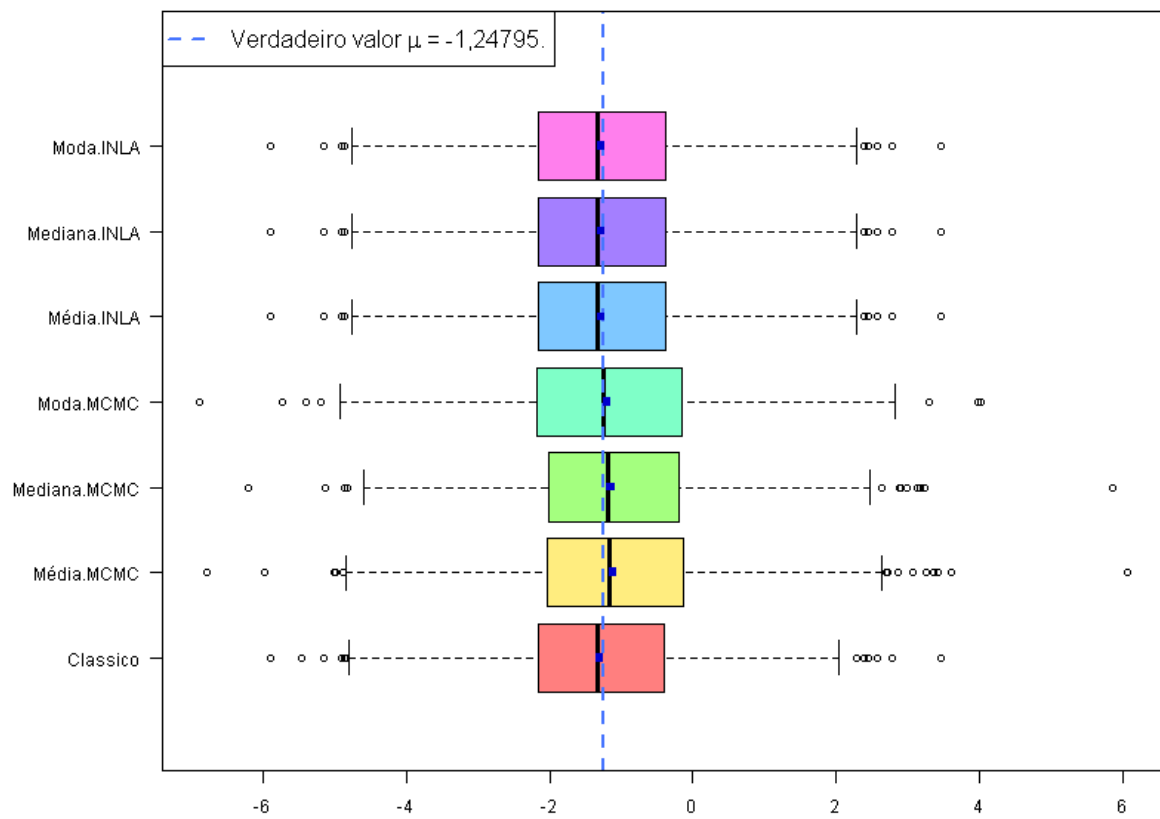
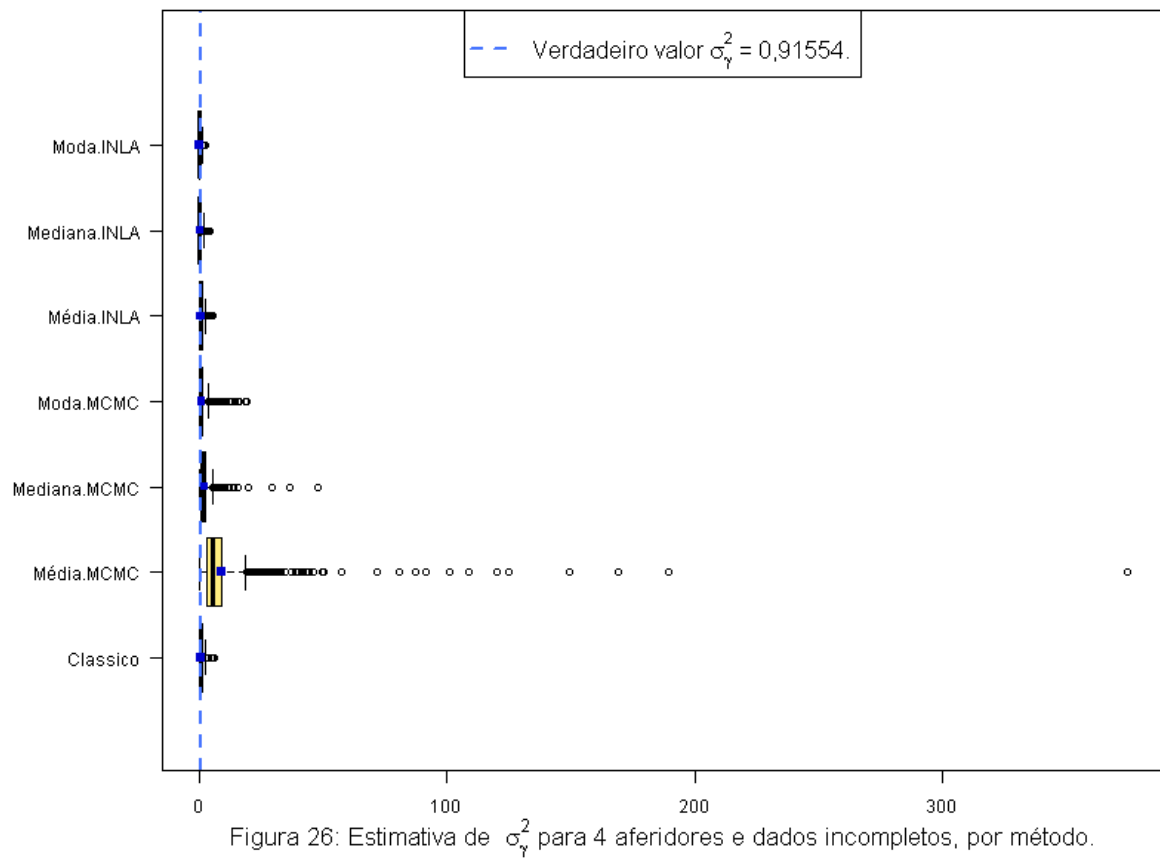
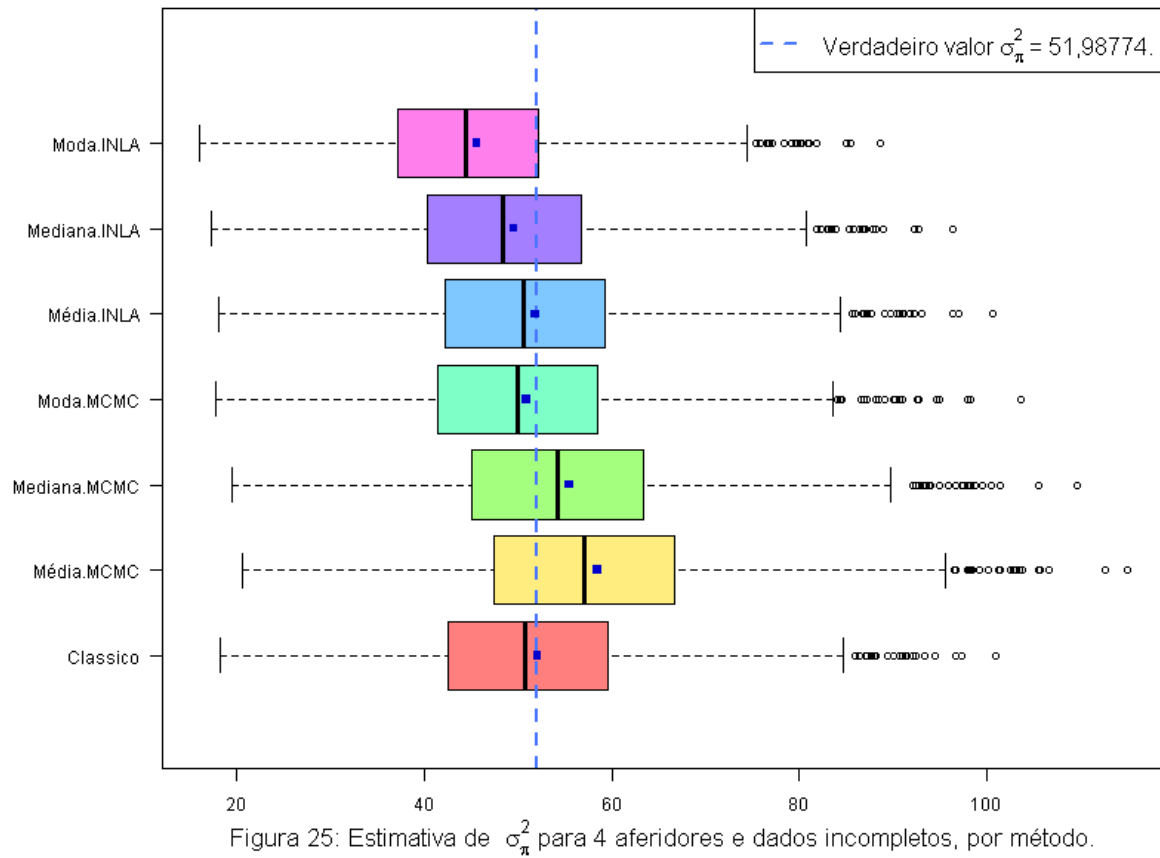


Figura 24: Estimativa de μ para 4 aferidores e dados incompletos, por método.



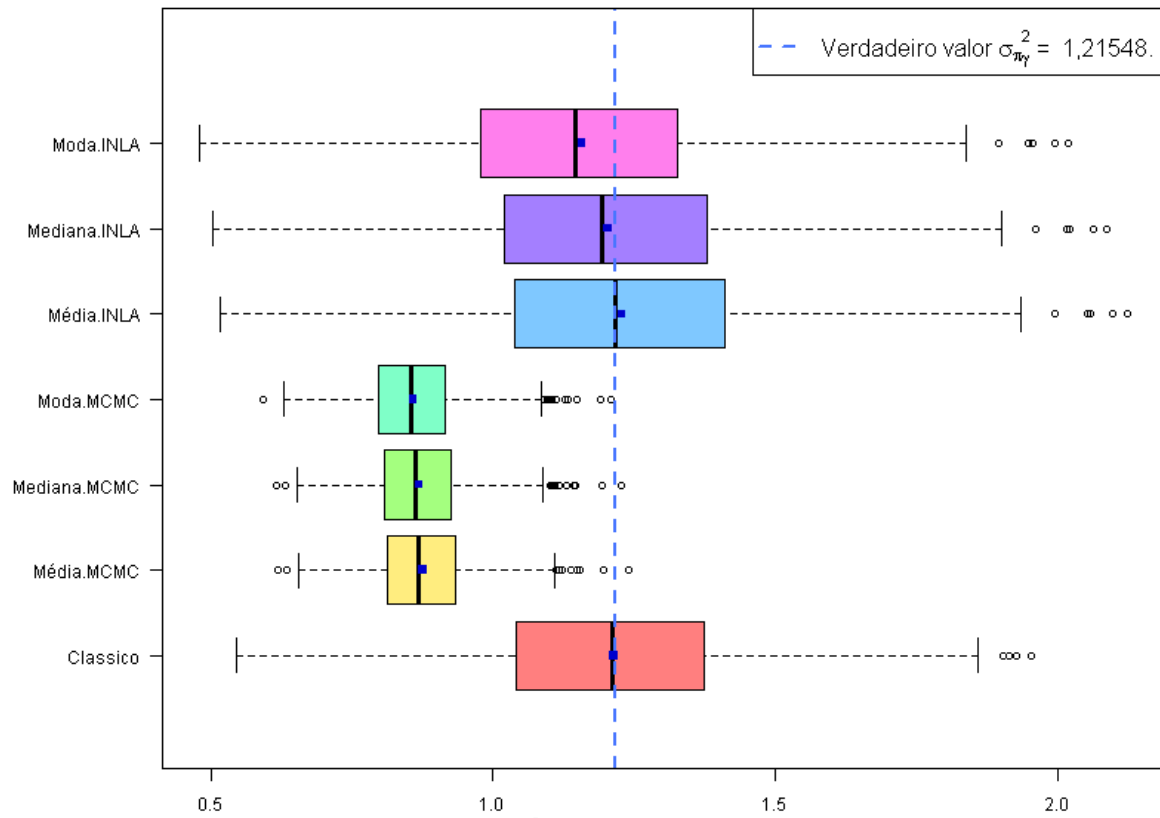


Figura 27: Estimativa de $\sigma_{\pi\gamma}^2$ para 4 aferidores e dados incompletos, por método.

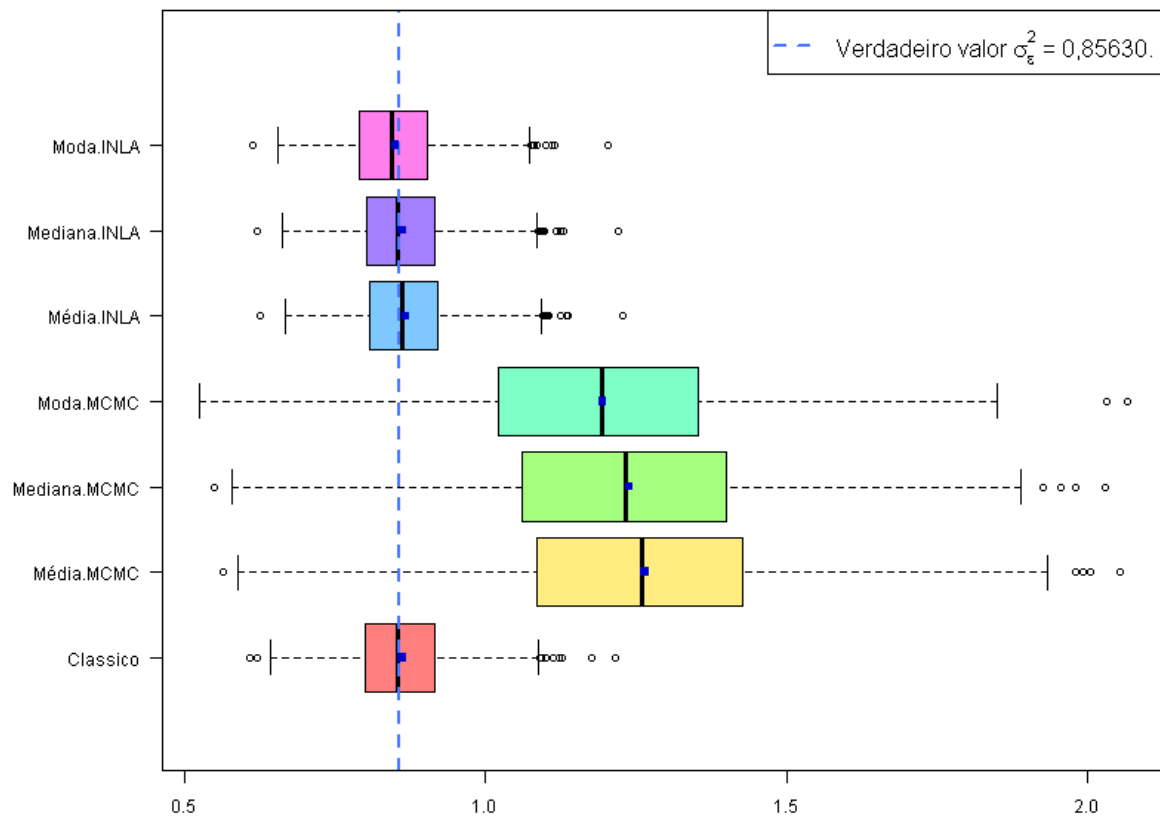


Figura 28: Estimativa de σ_{ϵ}^2 para 4 aferidores e dados incompletos, por método.

6.13. Tempos de simulação em horas, por cenário

Cenário	Método	Tempo
1	Clássico	26,5307
	MCMC	8,8277
	INLA	5,6270
2	Clássico	72,4777
	MCMC	20,7101
	INLA	7,8790
3	Clássico	28,5138
	MCMC	9,2982
	INLA	1,9997
4	Clássico	61,1929
	MCMC	17,6582
	INLA	8,1363

Anexos

Anexo I: Exemplo prático

Para a utilização dos métodos é necessário fazer o download dos pacotes: lme4, coda e BRugs do aplicativo R.

Para utilização do método MCMC é necessário fazer o download do aplicativo OpenBUGS que se encontra disponível no site: <http://openbugs.info/w/Downloads>

Para a utilização do método INLA é necessário a instalação do pacote INLA utilizando as seguintes funções dentro do aplicativo R:

```
source("http://www.math.ntnu.no/inla/givemeINLA.R")
library(INLA)
inla.upgrade(testing=T)
```

O Banco de dados, Modelo para o MCMC e valores iniciais disponível no link abaixo:

<https://docs.google.com/open?id=0B2phBRcWin3lNmE5NWU4YzgtOWEwMy00OTk4LTk0ZGEtM2I0Yjk3ZjczMTUz>

```
# É necessário criar uma pasta no computador para que sejam colocados
# os arquivos baixados do link acima. Nesta pasta será salva as
# estimativas para os ICC's também.
```

```
# Lendo os dados
```

```
setwd("Pasta do computador que contém os dados, inits e modelo")
```

```
dados <- read.table("Pasta que contém os dados",header=T)
```

```

n.af = 3 # número de aferidores
n.suj = 20 # número de sujeitos
n.rep = 2 # número de repetições
ntot = n.rep*n.af*n.suj # número total de observações

# Clássico
nboots = 1000 # número de amostras bootstrap

# Bayesiano

nchains = 2 # número de cadeias
nreplicacao = 10000 # número de replicações
burnin = 2000 # Aquecimento do algoritmo
thin = 20 # Pulo

##### Método MCMC #####

library(BRugs)

### Função HPD
emp.hpd.m <- function(x, conf=0.95){
  conf <- min(conf, 1-conf)
  n <- length(x)
  nn <- floor( n*conf )
  x <- sort(x)
  xx <- x[ (n-nn):(n) ] - x[1:(nn+1)]
  nnn <- which(xx==min(xx))[1]
  return( c( x[ nnn ], x[ n-nn+nnn-1 ] ) )
}

dados_bayes <- list(n.rep=n.rep, M=n.af, O=n.suj,
y=structure(.Data=dados$valor,.Dim=c(n.rep,n.af,n.suj)))

fit.brugs <- BRugsFit(modelFile="model.mcmc.txt", data=dados_bayes, inits
= c("inits1.txt", "inits2.txt") ,
numChains = nchains,
parametersToSave=c("vari","vara","varinter","vare","theta","icc","icc2"),
nBurnin = burnin, nIter = nreplicacao, nThin = thin, coda = FALSE,
DIC = TRUE, working.directory = NULL, digits = 5, seed=NULL,
BRugsVerbose = getOption("BRugsVerbose"))

# Estatísticas e intervalo de credibilidade central

x <- samplesStats("*")

amostra.icc<-samplesSample('icc')
amostra.icc2<-samplesSample('icc2')

# média

icc.mean<-x['icc','mean']
icc2.mean<-x['icc2','mean']

# Quantil 2,5%

icc.val2.5pc<-x['icc','val2.5pc']
icc2.val2.5pc<-x['icc2','val2.5pc']

# Quantil 97,5%

```

```

icc.val97.5pc<-x['icc','val97.5pc']
icc2.val97.5pc<-x['icc2','val97.5pc']

# mediana

icc.median<-x['icc','median']
icc2.median<-x['icc2','median']

# HPDs

icc.hpd.mcmc <- emp.hpd.m(amostra.icc)
icc2.hpd.mcmc <- emp.hpd.m(amostra.icc2)

icc.hpd.mcmc.inf <- icc.hpd.mcmc[1]
icc.hpd.mcmc.sup <- icc.hpd.mcmc[2]
icc2.hpd.mcmc.inf <- icc2.hpd.mcmc[1]
icc2.hpd.mcmc.sup <- icc2.hpd.mcmc[2]

# Moda

kernel1 = density(amostra.icc, bw="nrd0", n = 512,
from=min(amostra.icc), to=max(amostra.icc))
posmo1 = which.max(kernel1$y)
icc.moda.mcmc = kernel1$x[posmo1]

kernel2 = density(amostra.icc2, bw="nrd0", n = 512,
from=min(amostra.icc2), to=max(amostra.icc2))
posmo2 = which.max(kernel2$y)
icc2.moda.mcmc = kernel2$x[posmo2]

##### Método INLA #####

library(INLA)

dados<-as.data.frame(dados)

formula = valor ~ f(sujeito,model="iid") + f(aferidor,model="iid") +
f(interacao,model="iid")

mod.inla = inla(formula,data=dados,family="gaussian")

h.inla = inla.hyperpar(mod.inla)

# Erro
marg.prec.erro.inla = h.inla$marginals.hyperpar$`Precision for the
Gaussian observations`
marg.var.erro.inla = inla.tmarginal(function(x) 1/x, marg.prec.erro.inla)

# Sujeito

marg.prec.suj.inla = h.inla$marginals.hyperpar$`Precision for sujeito`
marg.var.suj.inla = inla.tmarginal(function(x) 1/x, marg.prec.suj.inla)

# Aferidor

marg.prec.af.inla = h.inla$marginals.hyperpar$`Precision for aferidor`
marg.var.af.inla = inla.tmarginal(function(x) 1/x, marg.prec.af.inla)

# Interação

marg.prec.inter.inla = h.inla$marginals.hyperpar$`Precision for interacao`

```



```

marg.var.inter.inla = inla.tmarginal(function(x) 1/x,
marg.prec.inter.inla)

# ICC's e intervalos de credibilidade

var.erro.ran.inla = inla.rmarginal(10000, marg.var.erro.inla)
var.af.ran.inla = inla.rmarginal(10000, marg.var.af.inla)
var.suj.ran.inla = inla.rmarginal(10000, marg.var.suj.inla)
var.inter.ran.inla = inla.rmarginal(10000, marg.var.inter.inla)

icc.inla = var.suj.ran.inla/(var.suj.ran.inla+ var.af.ran.inla+
var.inter.ran.inla+ var.erro.ran.inla)

icc.mean.inla = mean(icc.inla)
icc.median.inla = median(icc.inla)
icc.2.5.inla = quantile(icc.inla,0.025)
icc.97.5.inla = quantile(icc.inla, 0.975)

icc.hpd.inla = emp.hpd.m(icc.inla)
icc.hpd.inla.inf = icc.hpd.inla[1]
icc.hpd.inla.sup = icc.hpd.inla[2]

kernel6 = density(icc.inla, bw="nrd0", n = 512, from=min(icc.inla),
to=max(icc.inla))
posmo6 = which.max(kernel6$y)
icc.moda.inla = kernel6$x[posmo6]

icc.inla2 = (var.suj.ran.inla+var.af.ran.inla+
var.inter.ran.inla)/(var.suj.ran.inla+ var.af.ran.inla+
var.inter.ran.inla+ var.erro.ran.inla)

icc.mean.inla2 = mean(icc.inla2)
icc.median.inla2 = median(icc.inla2)
icc.2.5.inla2 = quantile(icc.inla2,0.025)
icc.97.5.inla2 = quantile(icc.inla2, 0.975)

icc.hpd.inla2 = emp.hpd.m(icc.inla2)
icc.hpd.inla2.inf = icc.hpd.inla2[1]
icc.hpd.inla2.sup = icc.hpd.inla2[2]

kernel7 = density(icc.inla2, bw="nrd0", n = 512, from=min(icc.inla2),
to=max(icc.inla2))
posmo7 = which.max(kernel7$y)
icc.moda.inla2 = kernel7$x[posmo7]

#### Método Clássico ####

library(lme4)

sujeito = rep(1:n.suj,each=(n.af*n.rep))
aferidor = rep(rep(1:n.af,each=n.rep),n.suj)
interacao = rep(1:(n.af*n.suj),each=n.rep)

### Função para gerar amostras bootstrap

dadosboot = function(dados,n.suj,n.rep,n.af,sujeito,aferidor,interacao){
  dadostransp = cbind(unique(dados$sujeito),matrix(dados$valor,ncol =
(n.rep*n.af),byrow=T))

```

```

    dadosaleat <- as.data.frame(dadostransp[sample(1:n.suj,
n.suj,replace=T), 1])
    yboot = as.real(t(as.matrix(dadosaleat[,-1])))
    bancoboot = cbind(sujeito,aferidor,interacao,yboot)
    saidaboot = yboot
    return(saidaboot)
}

### Função para obter estimativas dentro de cada amostra bootstrap

analiseboot = function(valoratual){
  bancoatual = data.frame(sujeito,aferidor,interacao,valoratual)

  af <- as.factor(bancoatual$aferidor)
  id <- as.factor(bancoatual$sujeito)

  fit.boot <- lmer(valoratual ~ (1|af) + (1|id) + (1|af:id),bancoatual)
  theta.boot = fixef(fit.boot)[[1]]
  var.inter.boot = VarCorr(fit.boot)$'af:id'[1]
  var.suj.boot = VarCorr(fit.boot)$'id'[1]
  var.af.boot = VarCorr(fit.boot)$'af'[1]
  var.erro.boot = (attr(VarCorr(fit.boot),"sc" ))^2
  icc1.boot <-
(var.suj.boot/(var.suj.boot+var.af.boot+var.inter.boot+var.erro.boot))
  icc2.boot <-
((var.suj.boot+var.af.boot+var.inter.boot)/(var.suj.boot+var.af.boot+var.i
nter.boot+var.erro.boot))
  saidaanboot = list(theta.boot, var.inter.boot, var.suj.boot,
var.af.boot, var.erro.boot, icc1.boot, icc2.boot)
  names(saidaanboot) = c("theta.boot", "var.inter.boot", "var.suj.boot",
"var.af.boot", "var.erro.boot", "icc1.boot", "icc2.boot")
  return(saidaanboot)
}

### Função para análise clássica

dados<-as.data.frame(dados)

# Estimativas pontuais clássicas

af <- as.factor(dados$aferidor)
id <- as.factor(dados$sujeito)

fit <- lmer(valor ~ (1|af) + (1|id) + (1|af:id),dados)

theta.class = fixef(fit)[[1]]
var.inter.class = VarCorr(fit)$'af:id'[1]
var.suj.class = VarCorr(fit)$'id'[1]
var.af.class = VarCorr(fit)$'af'[1]
var.erro.class = (attr(VarCorr(fit),"sc" ))^2
icc1.class <-
(var.suj.class/(var.suj.class+var.af.class+var.inter.class+var.erro.class)
)
icc2.class <-
((var.suj.class+var.af.class+var.inter.class)/(var.suj.class+var.af.class+
var.inter.class+var.erro.class))

# Implementando bootstrap

amsboot = replicate(nboots,
dadosboot(dados,n.suj,n.rep,n.af,sujeito,aferidor,interacao))

```

```

resboot = apply(amsboot, 2, analiseboot)

# Matriz de estimativas bootstrap: nas linhas são estimativas pontuais e
nas colunas são as repetições do bootstrap

matestboot = sapply(resboot,function(x) { c(x$theta.boot, x$var.suj.boot,
x$var.af.boot, x$var.inter.boot, x$var.erro.boot, x$icc1.boot,
x$icc2.boot) })
dimnames(matestboot) = list(c("theta.boot", "var.suj.boot", "var.af.boot",
"var.inter.boot", "var.erro.boot", "icc1.boot", "icc2.boot"))

icc1.class.inf = quantile(matestboot['icc1.boot', ], 0.025)
icc1.class.sup = quantile(matestboot['icc1.boot', ], 0.975)

icc2.class.inf = quantile(matestboot['icc2.boot', ], 0.025)
icc2.class.sup = quantile(matestboot['icc2.boot', ], 0.975)

# Estimativas gerais

estimativas <-
c(icc.mean,icc2.mean,icc.val2.5pc,icc2.val2.5pc,icc.val97.5pc,icc2.val97.5
pc,
icc.median,icc2.median,icc.hpd.mcmc.inf,icc.hpd.mcmc.sup,icc2.hpd.mcmc.inf
,
icc2.hpd.mcmc.sup,icc.moda.mcmc,icc2.moda.mcmc,icc.mean.inla,icc.median.in
la,
icc.moda.inla,icc.2.5.inla,icc.97.5.inla,icc.hpd.inla.inf,icc.hpd.inla.sup
,icc.mean.inla2,icc.median.inla2,icc.moda.inla2,icc.2.5.inla2,icc.97.5.inl
a2,icc.hpd.inla2.inf,
icc.hpd.inla2.sup,icc1.class,icc2.class,icc1.class.inf,icc1.class.sup,icc2
.class.inf,icc2.class.sup)

names(estimativas) <-
c("icc.mcmc.mean","icc2.mcmc.mean","icc.2.5.percentil","icc2.mcmc.2.5.perc
entil","icc.mcmc.97.5.percentil",
"icc2.mcmc.97.5.percentil","icc.mcmc.median","icc2.mcmc.median","icc.hpd.m
cmmc.inf","icc.hpd.mcmc.sup","icc2.hpd.mcmc.inf",
"icc2.hpd.mcmc.sup","icc.moda.mcmc","icc2.moda.mcmc","icc.mean.inla","icc.
median.inla",
"icc.moda.inla","icc.inla.2.5.percentil","icc.inla.97.5.percentil","icc.hp
d.inla.inf","icc.hpd.inla.sup",
"icc2.mean.inla","icc2.median.inla","icc2.moda.inla","icc2.2.5.inla","icc2
.97.5.inla","icc2.hpd.inla.inf",
"icc2.hpd.inla.sup","icc1.class","icc2.class","icc1.class.inf","icc1.class
.sup","icc2.class.inf","icc2.class.sup")

### Escrevendo os resultados

write.csv2(round(estimativas,6),"Resultados.csv")

# salva todas as estimativas em um arquivo csv que pode ser aberto no
#excel

```

Anexo II: Código

O Modelo para o MCMC e valores iniciais disponível no link abaixo:

<https://docs.google.com/open?id=0B2phBRcWin3lNWFizjcwZjMtNDMwNC00MzNjLTlmZGQtN2M3MjBLY2JkOTE1>

Para a utilização dos métodos é necessário fazer o download dos pacotes: lme4, coda e BRugs do aplicativo R.

Para utilização do método MCMC é necessário fazer o download do aplicativo OpenBUGS que se encontra disponível no site: <http://openbugs.info/w/Downloads>

Para a utilização do método INLA é necessário a instalação do pacote INLA utilizando as seguinte funções dentro do aplicativo R:

```
source("http://www.math.ntnu.no/inla/givemeINLA.R")
library(INLA)
inla.upgrade(testing=T)

### Geração dos dados completos e parâmetros necessários para simulação
### Equivalente para n aferidores ###

n.af=n
n.suj=29
n.rep=1000
repl=5
var.af=0.915544
var.sj=51.987741
var.sjaf=1.215483
var.erro=0.8563
theta = -1.24795
icc1 = var.sj/(var.sj+var.af+var.sjaf+var.erro)
icc2 = (var.sj+var.af+var.sjaf)/(var.sj+var.af+var.sjaf+var.erro)

ntot = n.rep*n.af*n.suj

params = c(theta, var.sj, var.af, var.sjaf, var.erro, icc1, icc2)

# Clássico #
nboots = 1000

# Bayesiano #

nchains=2
nreplicacao=10000
burnin=2000
thin=20

### Dados

sujeito = rep(1:n.suj,each=(n.af*n.rep))
aferidor = rep(rep(1:n.af,each=n.rep),n.suj)
interacao = rep(1:(n.af*n.suj),each=n.rep)

### Efeitos aleatórios

set.seed(123)

ef.aferidor = replicate(repl,rnorm(n.af,0,sqrt(var.af)))
```

```

ef.sujeito = replicate(repl,rnorm(n.suj,0,sqrt(var.sj)))
ef.inter = replicate(repl,rnorm((n.suj*n.af),0,sqrt(var.sjaf)))

ef.suj.rep = apply(ef.sujeito, 2, function(x) rep(x,each=(n.af*n.rep)))
ef.af.rep = apply(ef.aferidor, 2, function(x)
rep(rep(x,each=n.rep),n.suj))
ef.inter.rep = apply(ef.inter, 2, function(x) rep(x,each=n.rep))

### Gerando resposta

media = theta + ef.suj.rep + ef.af.rep + ef.inter.rep

y = apply(media, 2, function(x) rnorm(ntot,x,sqrt(var.erro)))

save.image("cenarioN.Rdata")

### Geração dos dados incompletos e parâmetros necessários para simulação
### Equivalente para n aferidores ###

n.af=n
n.suj=29
n.rep=3
repl=2
var.af=0.915544
var.sj=51.987741
var.sjaf=1.215483
var.erro=0.8563
theta = -1.24795
icc1 = var.sj/(var.sj+var.af+var.sjaf+var.erro)
icc2 = (var.sj+var.af+var.sjaf)/(var.sj+var.af+var.sjaf+var.erro)

ntot = n.rep*n.af*n.suj

params = c(theta, var.sj, var.af, var.sjaf, var.erro, icc1, icc2)

# Clássico #
nboots = 1000

# Bayesiano #

nchains=2
nreplicacao=10000
burnin=2000
thin=20

### Dados

sujeito = rep(1:n.suj,each=(n.af*n.rep))
aferidor = rep(rep(1:n.af,each=n.rep),n.suj)
interacao = rep(1:(n.af*n.suj),each=n.rep)

### Efeitos aleatórios

set.seed(123)

ef.aferidor = replicate(repl,rnorm(n.af,0,sqrt(var.af)))
ef.sujeito = replicate(repl,rnorm(n.suj,0,sqrt(var.sj)))
ef.inter = replicate(repl,rnorm((n.suj*n.af),0,sqrt(var.sjaf)))

ef.suj.rep = apply(ef.sujeito, 2, function(x) rep(x,each=(n.af*n.rep)))

```

```

ef.af.rep = apply(ef.aferidor, 2, function(x)
rep(rep(x,each=n.rep),n.suj))
ef.inter.rep = apply(ef.inter, 2, function(x) rep(x,each=n.rep))

### Gerando resposta

media = theta + ef.suj.rep + ef.af.rep + ef.inter.rep

yc = apply(media, 2, function(x) rnorm(ntot,x,sqrt(var.erro)))

### Colocando não-respostas

nmis = 17

imputNA = function(y,ntot,nmis){
posmis = sample(1:ntot, nmis, replace = FALSE)
y[posmis] = NA
return(y)
}

y = apply(yc,2,imputNA,ntot,nmis)

save.image("cenarioN.Rdata")

##### Função de Simulação Geral #####

load("cenarioN.RData")

source("funcao_classico.r")

save.image("simulacoes.classica.RData")
write.csv2(Quadro1.class, "simulacoes1.Class1.csv")
write.csv2(Quadro2.class, "simulacoes1.Class2.csv")

rm(list=ls(all=TRUE))

load("cenarioN.RData")

source("funcao_inla.r")
save.image("simulacoes.INLA.RData")
write.csv2(Quadro.INLA.1, "simulacoes1.INLA1.csv")
write.csv2(Quadro.INLA.2, "simulacoes1.INLA2.csv")

rm(list=ls(all=TRUE))

load("cenarioN.RData")

source("funcao_MCMC.r")

save.image("simulacoes.MCMC.RData")
write.csv2(Quadro.MCMC.1, "simulacoes1.MCMC1.csv")
write.csv2(Quadro.MCMC.2, "simulacoes1.MCMC2.csv")

##### Função Clássico #####

library(lme4)

### Função para gerar amostras bootstrap

dadosboot = function(dados,n.suj,n.rep,n.af,sujeito,aferidor,interacao){

```

```

    dadostransp = cbind(unique(dados$sujeito),matrix(dados$valor,ncol =
(n.rep*n.af),byrow=T))
    dadosaleat <- as.data.frame(dadostransp[sample(1:n.suj,
n.suj,replace=T), ])
    yboot = as.real(t(as.matrix(dadosaleat[,-1])))
    bancoboot = cbind(sujeito,aferidor,interacao,yboot)
    saidaboot = yboot
    return(saidaboot)
}

### Função para obter estimativas dentro de cada amostra bootstrap
analiseboot = function(valoratual){
  bancoatual = data.frame(sujeito,aferidor,interacao,valoratual)

  af <- as.factor(bancoatual$aferidor)
  id <- as.factor(bancoatual$sujeito)

  fit.boot <- lmer(valoratual ~ (1|af) + (1|id) + (1|af:id),bancoatual)
  theta.boot = fixef(fit.boot)[[1]]
  var.inter.boot = VarCorr(fit.boot)$'af:id'[1]
  var.suj.boot = VarCorr(fit.boot)$'id'[1]
  var.af.boot = VarCorr(fit.boot)$'af'[1]
  var.erro.boot = (attr(VarCorr(fit.boot),"sc" ))^2
  iccl.boot <-
(var.suj.boot/(var.suj.boot+var.af.boot+var.inter.boot+var.erro.boot))
  icc2.boot <-
((var.suj.boot+var.af.boot+var.inter.boot)/(var.suj.boot+var.af.boot+var.i
nter.boot+var.erro.boot))
  saidaanboot = list(theta.boot, var.inter.boot, var.suj.boot,
var.af.boot, var.erro.boot, iccl.boot, icc2.boot)
  names(saidaanboot) = c("theta.boot", "var.inter.boot", "var.suj.boot",
"var.af.boot", "var.erro.boot", "iccl.boot", "icc2.boot")
  return(saidaanboot)
}

### Função para análise clássica

classica =
function(y,sujeito,aferidor,interacao,dadosboot,analiseboot,nboots,n.suj){

dados <- cbind(sujeito,aferidor,interacao,y)
colnames(dados) <- c("sujeito", "aferidor", "interacao", "valor")
dados<-as.data.frame(dados)

# Estimativas pontuais clássicas

af <- as.factor(dados$aferidor)
id <- as.factor(dados$sujeito)

fit <- lmer(valor ~ (1|af) + (1|id) + (1|af:id),dados)
theta.class = fixef(fit)[[1]]
var.inter.class = VarCorr(fit)$'af:id'[1]
var.suj.class = VarCorr(fit)$'id'[1]
var.af.class = VarCorr(fit)$'af'[1]
var.erro.class = (attr(VarCorr(fit),"sc" ))^2
iccl.class <-
(var.suj.class/(var.suj.class+var.af.class+var.inter.class+var.erro.class)
)
icc2.class <-
((var.suj.class+var.af.class+var.inter.class)/(var.suj.class+var.af.class+
var.inter.class+var.erro.class))

```

```

# Implementando bootstrap

amsboot = replicate(nboots,
dadosboot(dados,n.suj,n.rep,n.af,sujeito,aferidor,interacao))

resboot = apply(amsboot, 2, analiseboot)

# Matriz de estimativas bootstrap: nas linhas são estimativas pontuais e
nas colunas são as repetições do bootstrap
matestboot = sapply(resboot,function(x) { c(x$theta.boot, x$var.suj.boot,
x$var.af.boot, x$var.inter.boot, x$var.erro.boot, x$iccl.boot,
x$icc2.boot) })
dimnames(matestboot) = list(c("theta.boot", "var.suj.boot", "var.af.boot",
"var.inter.boot", "var.erro.boot", "iccl.boot", "icc2.boot"))

ic1.theta.class = quantile(matestboot['theta.boot', ], 0.025)
ic2.theta.class = quantile(matestboot['theta.boot', ], 0.975)

ic1.var.suj.class = quantile(matestboot['var.suj.boot', ], 0.025)
ic2.var.suj.class = quantile(matestboot['var.suj.boot', ], 0.975)

ic1.var.af.class = quantile(matestboot['var.af.boot', ], 0.025)
ic2.var.af.class = quantile(matestboot['var.af.boot', ], 0.975)

ic1.var.inter.class = quantile(matestboot['var.inter.boot', ], 0.025)
ic2.var.inter.class = quantile(matestboot['var.inter.boot', ], 0.975)

ic1.var.erro.class = quantile(matestboot['var.erro.boot', ], 0.025)
ic2.var.erro.class = quantile(matestboot['var.erro.boot', ], 0.975)

ic1.iccl.class = quantile(matestboot['iccl.boot', ], 0.025)
ic2.iccl.class = quantile(matestboot['iccl.boot', ], 0.975)

ic1.icc2.class = quantile(matestboot['icc2.boot', ], 0.025)
ic2.icc2.class = quantile(matestboot['icc2.boot', ], 0.975)

# Saída da simulação

saidaclassica = list(theta.class, var.suj.class , var.af.class ,
var.inter.class , var.erro.class , iccl.class , icc2.class ,
ic1.theta.class, ic2.theta.class, ic1.var.suj.class,
ic2.var.suj.class, ic1.var.af.class, ic2.var.af.class ,
ic1.var.inter.class , ic2.var.inter.class,
ic1.var.erro.class, ic2.var.erro.class,
ic1.iccl.class, ic2.iccl.class, ic1.icc2.class, ic2.icc2.class, amsboot,
matestboot
)
names(saidaclassica) = c("theta.class", "var.suj.class" , "var.af.class" ,
"var.inter.class" , "var.erro.class" , "iccl.class" , "icc2.class",
"ic1.theta.class", "ic2.theta.class",
"ic1.var.suj.class", "ic2.var.suj.class", "ic1.var.af.class",
"ic2.var.af.class" , "ic1.var.inter.class" , "ic2.var.inter.class",
"ic1.var.erro.class", "ic2.var.erro.class",
"ic1.iccl.class", "ic2.iccl.class", "ic1.icc2.class", "ic2.icc2.class",
"amsboot", "matestboot")

return(saidaclassica)
}

### Implementando para todos os bancos gerados

inicio.class = Sys.time()

```



```

simuls.classica =
apply(y,2,classica,sujeito=sujeito,aferidor=aferidor,interacao=interacao,d
adosboot=dadosboot,analiseboot=analiseboot,nboots=nboots,n.suj=n.suj)

fim.class = Sys.time()
tempo.class = fim.class - inicio.class

# Matriz de estimativas: nas linhas são as estimativas pontuais ou limites
#de ics, nas colunas são as repetições da simulação

matest.class = sapply(simuls.classica, function(x) { c(x$theta.class,
x$var.suj.class , x$var.af.class , x$var.inter.class , x$var.erro.class ,
x$iccl.class , x$icc2.class ,
x$ic1.theta.class, x$ic2.theta.class,
x$ic1.var.suj.class, x$ic2.var.suj.class, x$ic1.var.af.class,
x$ic2.var.af.class , x$ic1.var.inter.class , x$ic2.var.inter.class,
x$ic1.var.erro.class, x$ic2.var.erro.class,
x$ic1.iccl.class, x$ic2.iccl.class, x$ic1.icc2.class, x$ic2.icc2.class) })
dimnames(matest.class) = list(c("theta.class", "var.suj.class" ,
"var.af.class" , "var.inter.class" , "var.erro.class" , "iccl.class" ,
"icc2.class",
"ic1.theta.class", "ic2.theta.class",
"ic1.var.suj.class", "ic2.var.suj.class", "ic1.var.af.class",
"ic2.var.af.class", "ic1.var.inter.class", "ic2.var.inter.class",
"ic1.var.erro.class",
"ic2.var.erro.class","ic1.iccl.class", "ic2.iccl.class",
"ic1.icc2.class", "ic2.icc2.class"))

# Separando estimativas pontuais, de limite inferior de ic e limite
superior de ic
matestpont.class = matest.class[c("theta.class", "var.suj.class" ,
"var.af.class" , "var.inter.class" , "var.erro.class" , "iccl.class" ,
"icc2.class"),]
matestic1.class = matest.class[c("ic1.theta.class", "ic1.var.suj.class",
"ic1.var.af.class", "ic1.var.inter.class", "ic1.var.erro.class",
"ic1.iccl.class", "ic1.icc2.class"), ]
matestic2.class = matest.class[c("ic2.theta.class", "ic2.var.suj.class",
"ic2.var.af.class", "ic2.var.inter.class", "ic2.var.erro.class",
"ic2.iccl.class", "ic2.icc2.class"), ]

# Calculando Medidas resumo
media.class = apply(matestpont.class,1,mean)
vicio.class = media.class - params
varian.class = apply(matestpont.class,1,var)
eqm.class = varian.class + vicio.class^2

cober.class = rep(0,7)
ampl.class = rep(0,7)

for(j in 1:7){
cober.class[j] =
100*sum(matestic1.class[j,]<=params[j]&params[j]<=matestic2.class[j,])/rep
l
ampl.class[j] = mean(matestic2.class[j,]-matestic1.class[j,])
}
names(cober.class) = c("ic.theta.class", "ic.var.suj.class",
"ic.var.af.class", "ic.var.inter.class" , "ic.var.erro.class",
"ic.iccl.class", "ic.icc2.class")
names(ampl.class) = c("ic.theta.class", "ic.var.suj.class",
"ic.var.af.class", "ic.var.inter.class" , "ic.var.erro.class",
"ic.iccl.class", "ic.icc2.class")

```

```

# Salvando Quadro de resultados
Quadro1.class =
data.frame(media=media.class,vicio=vicio.class,variancia=varian.class,eqm=
eqm.class)
Quadro2.class = data.frame(amplitude=ampl.class, cobertura=cober.class)

##### Função INLA #####

library(coda)
library(INLA)

### Função HPD emp.hpd.m <- function(x, conf=0.95){
  conf <- min(conf, 1-conf)
  n <- length(x)
  nn <- floor( n*conf )
  x <- sort(x)
  xx <- x[ (n-nn):(n) ] - x[1:(nn+1)]
  nnn <- which(xx==min(xx))[1]
  return( c( x[ nnn ], x[ n-nn+nnn-1 ] ) )
}

simula.INLA <- function(y,sujeito,aferidor,interacao){

dados <- cbind(sujeito,aferidor,interacao,y)
colnames(dados) <- c("sujeito", "aferidor", "interacao", "valor")
dados<-as.data.frame(dados)
rm(y)

formula = valor ~ f(sujeito,model="iid") + f(aferidor,model="iid") +
f(interacao,model="iid")

mod.inla = inla(formula,data=dados,family="gaussian")

h.inla = inla.hyperpar(mod.inla)

theta.mean.inla = h.inla$summary.fixed[, 'mean']
theta.sd.inla = h.inla$summary.fixed[, 'sd']
theta.median.inla = h.inla$summary.fixed[, '0.5quant']
theta.2.5.inla = h.inla$summary.fixed[, '0.025quant']
theta.97.5.inla = h.inla$summary.fixed[, '0.975quant']
marg.theta = h.inla$marginals.fixed$(Intercept)
theta.hpd.inla = inla.hpdmarginal(0.95, marg.theta)
theta.hpd.inla.inf<-theta.hpd.inla[, 'low']
theta.hpd.inla.sup<-theta.hpd.inla[, 'high']

posmol = which.max(marg.theta[,2])
theta.moda.inla = marg.theta[posmol,1]

##### Não tem as estimativas sd para os componentes de variância #####

##### Erro #####

marg.prec.erro.inla = h.inla$marginals.hyperpar$`Precision for the
Gaussian observations`
marg.var.erro.inla = inla.tmarginal(function(x) 1/x, marg.prec.erro.inla)
var.erro.mean.inla = inla.emarginal(function(x) x^1, marg.var.erro.inla)
var.erro.median.inla = inla.qmarginal(c(0.5), marg.var.erro.inla)
var.erro.2.5.inla = inla.qmarginal(c(0.025), marg.var.erro.inla)
var.erro.97.5.inla = inla.qmarginal(c(0.975), marg.var.erro.inla)
var.erro.hpd.inla = inla.hpdmarginal(0.95, marg.var.erro.inla)

```

```

var.erro.hpd.inla.inf<-var.erro.hpd.inla[,'low']
var.erro.hpd.inla.sup<-var.erro.hpd.inla[,'high']

posmo2 = which.max(marg.var.erro.inla[,2])
var.erro.moda.inla = marg.var.erro.inla[posmo2,1]

#### Sujeito #####

marg.prec.suj.inla = h.inla$marginals.hyperpar$`Precision for sujeito`
marg.var.suj.inla = inla.tmarginal(function(x) 1/x, marg.prec.suj.inla)
var.suj.mean.inla = inla.emarginal(function(x) x^1, marg.var.suj.inla)
var.suj.median.inla = inla.qmarginal(c(0.5), marg.var.suj.inla)
var.suj.2.5.inla = inla.qmarginal(c(0.025), marg.var.suj.inla)
var.suj.97.5.inla = inla.qmarginal(c(0.975), marg.var.suj.inla)
var.suj.hpd.inla = inla.hpdmarginal(0.95, marg.var.suj.inla)
var.suj.hpd.inla.inf<-var.suj.hpd.inla[,'low']
var.suj.hpd.inla.sup<-var.suj.hpd.inla[,'high']

posmo3 = which.max(marg.var.suj.inla[,2])
var.suj.moda.inla = marg.var.suj.inla[posmo3,1]

#### Aferidor #####

marg.prec.af.inla = h.inla$marginals.hyperpar$`Precision for aferidor`
marg.var.af.inla = inla.tmarginal(function(x) 1/x, marg.prec.af.inla)
var.af.mean.inla = inla.emarginal(function(x) x^1, marg.var.af.inla)
var.af.median.inla = inla.qmarginal(c(0.5), marg.var.af.inla)
var.af.2.5.inla = inla.qmarginal(c(0.025), marg.var.af.inla)
var.af.97.5.inla = inla.qmarginal(c(0.975), marg.var.af.inla)
var.af.hpd.inla = inla.hpdmarginal(0.95, marg.var.af.inla)
var.af.hpd.inla.inf<-var.af.hpd.inla[,'low']
var.af.hpd.inla.sup<-var.af.hpd.inla[,'high']

posmo4 = which.max(marg.var.af.inla[,2])
var.af.moda.inla = marg.var.af.inla[posmo4,1]

### Interação #####

marg.prec.inter.inla = h.inla$marginals.hyperpar$`Precision for interacao`
marg.var.inter.inla = inla.tmarginal(function(x) 1/x,
marg.prec.inter.inla)
var.inter.mean.inla = inla.emarginal(function(x) x^1, marg.var.inter.inla)
var.inter.median.inla = inla.qmarginal(c(0.5), marg.var.inter.inla)
var.inter.2.5.inla = inla.qmarginal(c(0.025), marg.var.inter.inla)
var.inter.97.5.inla = inla.qmarginal(c(0.975), marg.var.inter.inla)
var.inter.hpd.inla = inla.hpdmarginal(0.95, marg.var.inter.inla)
var.inter.hpd.inla.inf<-var.inter.hpd.inla[,'low']
var.inter.hpd.inla.sup<-var.inter.hpd.inla[,'high']

posmo5 = which.max(marg.var.inter.inla[,2])
var.inter.moda.inla = marg.var.inter.inla[posmo5,1]

#### ICC's #####

var.erro.ran.inla = inla.rmarginal(10000, marg.var.erro.inla)
var.af.ran.inla = inla.rmarginal(10000, marg.var.af.inla)
var.suj.ran.inla = inla.rmarginal(10000, marg.var.suj.inla)
var.inter.ran.inla = inla.rmarginal(10000, marg.var.inter.inla)

```

```

icc.inla = var.suj.ran.inla/(var.suj.ran.inla+ var.af.ran.inla+
var.inter.ran.inla+ var.erro.ran.inla)

icc.mean.inla = mean(icc.inla)
icc.median.inla = median(icc.inla)
icc.2.5.inla = quantile(icc.inla,0.025)
icc.97.5.inla = quantile(icc.inla, 0.975)

icc.hpd.inla = emp.hpd.m(icc.inla)
icc.hpd.inla.inf = icc.hpd.inla[1]
icc.hpd.inla.sup = icc.hpd.inla[2]

kernel6 = density(icc.inla, bw="nrd0", n = 512, from=min(icc.inla),
to=max(icc.inla))
posmo6 = which.max(kernel6$y)
icc.moda.inla = kernel6$x[posmo6]

icc.inla2 = (var.suj.ran.inla+var.af.ran.inla+
var.inter.ran.inla)/(var.suj.ran.inla+ var.af.ran.inla+
var.inter.ran.inla+ var.erro.ran.inla)

icc.mean.inla2 = mean(icc.inla2)
icc.median.inla2 = median(icc.inla2)
icc.2.5.inla2 = quantile(icc.inla2,0.025)
icc.97.5.inla2 = quantile(icc.inla2, 0.975)

icc.hpd.inla2 = emp.hpd.m(icc.inla2)
icc.hpd.inla2.inf = icc.hpd.inla2[1]
icc.hpd.inla2.sup = icc.hpd.inla2[2]

kernel7 = density(icc.inla2, bw="nrd0", n = 512, from=min(icc.inla2),
to=max(icc.inla2))
posmo7 = which.max(kernel7$y)
icc.moda.inla2 = kernel7$x[posmo7]

# Saida Bayesiano INLA #

saidaINLA =
list(theta.mean.inla,theta.sd.inla,theta.median.inla,theta.2.5.inla,theta.
97.5.inla,theta.moda.inla,theta.hpd.inla.inf,theta.hpd.inla.sup,
var.erro.mean.inla,var.erro.median.inla,var.erro.2.5.inla,var.erro.97.5.in
la,var.erro.hpd.inla.inf,var.erro.hpd.inla.sup,var.suj.mean.inla,
var.suj.median.inla,var.suj.2.5.inla,var.suj.97.5.inla,var.suj.hpd.inla.in
f,var.suj.hpd.inla.sup,var.af.mean.inla,var.af.median.inla,var.af.2.5.inla
,
var.af.97.5.inla,var.af.hpd.inla.inf,var.af.hpd.inla.sup,var.inter.mean.in
la,var.inter.median.inla,var.inter.2.5.inla,var.inter.97.5.inla,
var.inter.hpd.inla.inf,var.inter.hpd.inla.sup,icc.mean.inla,icc.median.inl
a,icc.2.5.inla,icc.97.5.inla,icc.mean.inla2,
icc.median.inla2,icc.2.5.inla2,icc.97.5.inla2,var.erro.moda.inla,var.suj.m
oda.inla, var.af.moda.inla, var.inter.moda.inla,
icc.hpd.inla.inf,icc.hpd.inla.sup,icc.hpd.inla2.inf,icc.hpd.inla2.sup,icc.
moda.inla , icc.moda.inla2 )

names(saidaINLA) =
c("theta.mean.inla","theta.sd.inla","theta.median.inla","theta.2.5.inla","
theta.97.5.inla","theta.moda.inla", "theta.hpd.inla.inf",
"theta.hpd.inla.sup","var.erro.mean.inla","var.erro.median.inla","var.erro
.2.5.inla","var.erro.97.5.inla","var.erro.hpd.inla.inf",
"var.erro.hpd.inla.sup","var.suj.mean.inla","var.suj.median.inla","var.suj
.2.5.inla","var.suj.97.5.inla","var.suj.hpd.inla.inf","var.suj.hpd.inla.su
p",

```

```

"var.af.mean.inla", "var.af.median.inla", "var.af.2.5.inla", "var.af.97.5.inla",
"var.af.hpd.inla.inf", "var.af.hpd.inla.sup", "var.inter.mean.inla",
"var.inter.median.inla", "var.inter.2.5.inla", "var.inter.97.5.inla", "var.inter.hpd.inla.inf",
"var.inter.hpd.inla.sup", "icc.mean.inla", "icc.median.inla", "icc.2.5.inla", "icc.97.5.inla",
"icc.mean.inla2", "icc.median.inla2", "icc.2.5.inla2", "icc.97.5.inla2",
"var.erro.moda.inla", "var.suj.moda.inla", "var.af.moda.inla",
"var.inter.moda.inla",
"icc.hpd.inla.inf", "icc.hpd.inla.sup", "icc.hpd.inla2.inf", "icc.hpd.inla2.sup",
"icc.moda.inla", "icc.moda.inla2")

return(saidaINLA)

}

inicio.INLA = Sys.time()

simuls.INLA =
apply(y, 2, simula.INLA, sujeito=sujeito, aferidor=aferidor, interacao=interacao)

fim.INLA = Sys.time()

tempo.INLA = fim.INLA - inicio.INLA

# Matriz de estimativas: nas linhas são as estimativas pontuais ou limites de ics, nas colunas são as repetições da simulação

matest.INLA = sapply(simuls.INLA, function(x) {
c(x$theta.mean.inla, x$theta.sd.inla, x$theta.median.inla, x$theta.2.5.inla, x$theta.97.5.inla,
x$theta.moda.inla, x$theta.hpd.inla.inf, x$theta.hpd.inla.sup, x$var.erro.mean.inla,
x$var.erro.median.inla, x$var.erro.2.5.inla, x$var.erro.97.5.inla, x$var.erro.hpd.inla.inf,
x$var.erro.hpd.inla.sup, x$var.suj.mean.inla, x$var.suj.median.inla, x$var.suj.2.5.inla,
x$var.suj.97.5.inla, x$var.suj.hpd.inla.inf, x$var.suj.hpd.inla.sup,
x$var.af.mean.inla, x$var.af.median.inla, x$var.af.2.5.inla, x$var.af.97.5.inla,
x$var.af.hpd.inla.inf, x$var.af.hpd.inla.sup, x$var.inter.mean.inla, x$var.inter.median.inla,
x$var.inter.2.5.inla, x$var.inter.97.5.inla, x$var.inter.hpd.inla.inf, x$var.inter.hpd.inla.sup,
x$icc.mean.inla, x$icc.median.inla, x$icc.2.5.inla, x$icc.97.5.inla, x$icc.mean.inla2, x$icc.median.inla2,
x$icc.2.5.inla2, x$icc.97.5.inla2, x$var.erro.moda.inla, x$var.suj.moda.inla, x$var.af.moda.inla,
x$var.inter.moda.inla,
x$icc.hpd.inla.inf,
x$icc.hpd.inla.sup, x$icc.hpd.inla2.inf, x$icc.hpd.inla2.sup, x$icc.moda.inla,
x$icc.moda.inla2) })

dimnames(matest.INLA) =
list(c("theta.mean.inla", "theta.sd.inla", "theta.median.inla", "theta.2.5.inla",
"theta.97.5.inla", "theta.moda.inla", "theta.hpd.inla.inf", "theta.hpd.inla.sup",
"var.erro.mean.inla", "var.erro.median.inla", "var.erro.2.5.inla", "var.erro.97.5.inla",
"var.erro.hpd.inla.inf", "var.erro.hpd.inla.sup", "var.suj.mean.inla", "var.suj.median.inla",
"var.suj.2.5.inla", "var.suj.97.5.inla", "var.suj.hpd.inla.inf", "var.suj.hpd.inla.sup",
"var.af.mean.inla", "var.af.median.inla", "var.af.2.5.inla", "var.af.97.5.inla",
"var.af.hpd.inla.inf", "var.af.hpd.inla.sup", "var.inter.mean.inla", "var.inter.median.inla",
"var.inter.2.5.inla", "var.inter.97.5.inla", "var.inter.hpd.inla.inf", "var.inter.hpd.inla.sup",
"icc.mean.inla", "icc.median.inla", "icc.2.5.inla", "icc.97.5.inla", "icc.mean.inla2", "icc.median.inla2",
"icc.2.5.inla2", "icc.97.5.inla2", "var.erro.moda.inla", "var.suj.moda.inla", "var.af.moda.inla",
"var.inter.moda.inla", "icc.hpd.inla.inf", "icc.hpd.inla.sup", "icc.hpd.inla2.inf", "icc.hpd.inla2.sup",
"icc.moda.inla", "icc.moda.inla2")

```

```

"icc.median.inla","icc.2.5.inla","icc.97.5.inla","icc.mean.inla2","icc.med
ian.inla2","icc.2.5.inla2","icc.97.5.inla2",
"var.erro.moda.inla","var.suj.moda.inla", "var.af.moda.inla",
"var.inter.moda.inla",
"icc.hpd.inla.inf","icc.hpd.inla.sup","icc.hpd.inla2.inf","icc.hpd.inla2.s
up","icc.moda.inla" , "icc.moda.inla2"))

# Separando estimativas pontuais, de limite inferior de ic e limite
superior de ic

matestmean.INLA =
matest.INLA[c("theta.mean.inla","var.suj.mean.inla","var.af.mean.inla","va
r.inter.mean.inla","var.erro.mean.inla","icc.mean.inla","icc.mean.inla2"),
]
matestmedian.INLA =
matest.INLA[c("theta.median.inla","var.suj.median.inla","var.af.median.inl
a","var.inter.median.inla","var.erro.median.inla","icc.median.inla","icc.m
edian.inla2"),]
matestmoda.INLA=
matest.INLA[c("theta.moda.inla","var.suj.moda.inla","var.af.moda.inla","va
r.inter.moda.inla","var.erro.moda.inla","icc.moda.inla","icc.moda.inla2"),
]

matestpont.INLA = rbind(matestmean.INLA,matestmedian.INLA,matestmoda.INLA)
matestsd.INLA = matest.INLA[c("theta.sd.inla"),]
matestic1.INLA =
matest.INLA[c("theta.2.5.inla","var.suj.2.5.inla","var.af.2.5.inla","var.i
nter.2.5.inla","var.erro.2.5.inla","icc.2.5.inla","icc.2.5.inla2"), ]
matestic2.INLA =
matest.INLA[c("theta.97.5.inla","var.suj.97.5.inla","var.af.97.5.inla","va
r.inter.97.5.inla","var.erro.97.5.inla","icc.97.5.inla","icc.97.5.inla2"),
]
matestiHPDc1.INLA =
matest.INLA[c("theta.hpd.inla.inf","var.suj.hpd.inla.inf","var.af.hpd.inla
.inf","var.inter.hpd.inla.inf","var.erro.hpd.inla.inf","icc.hpd.inla.inf",
"icc.hpd.inla2.inf"), ]
matestiHPDc2.INLA =
matest.INLA[c("theta.hpd.inla.sup","var.suj.hpd.inla.sup","var.af.hpd.inla
.sup","var.inter.hpd.inla.sup","var.erro.hpd.inla.sup","icc.hpd.inla.sup",
"icc.hpd.inla2.sup"), ]
matestlic.INLA = rbind(matestic1.INLA,matestiHPDc1.INLA)
matestlsc.INLA = rbind(matestic2.INLA,matestiHPDc2.INLA)

# Calculando Medidas resumo
media.INLA = apply(matestpont.INLA,1,mean)
vicio.INLA = media.INLA - rep(params,3)
varian.INLA = apply(matestpont.INLA,1,var)
eqm.INLA = varian.INLA + vicio.INLA^2

cober.INLA = rep(0,14)
ampl.INLA = rep(0,14)
params2 = rep(params,2)

for(j in 1:14){
  cober.INLA[j] =
100*sum(matestlic.INLA[j,]<=params2[j]&params2[j]<=matestlsc.INLA[j,])/rep
1
  ampl.INLA[j] = mean(matestlsc.INLA[j,]-matestlic.INLA[j,])
}

```

```

names(cober.INLA) =
c("ictheta.inla", "ic.var.suj.inla", "ic.var.af.inla", "ic.var.inter.inla", "i
c.var.erro.inla", "ic.icc.inla", "ic.icc.inla2",
"hpdtheta.inla", "hpd.var.suj.inla", "hpd.var.af.inla", "hpd.var.inter.inla",
"hpd.var.erro.inla", "hpd.icc.inla", "hpd.icc.inla2")
names(ampl.INLA) =
c("ictheta.inla", "ic.var.suj.inla", "ic.var.af.inla", "ic.var.inter.inla", "i
c.var.erro.inla", "ic.icc.inla", "ic.icc.inla2",
"hpdtheta.inla", "hpd.var.suj.inla", "hpd.var.af.inla", "hpd.var.inter.inla",
"hpd.var.erro.inla", "hpd.icc.inla", "hpd.icc.inla2")

# Salvando Quadro de resultados
Quadro.INLA.1 =
data.frame(media=media.INLA, vicio=vicio.INLA, variancia=varian.INLA, eqm=eqm
.INLA)
Quadro.INLA.2 = data.frame(amplitude=ampl.INLA, cobertura=cober.INLA)

##### Função MCMC #####

library(BRugs)

### Função HPD
emp.hpd.m <- function(x, conf=0.95){
  conf <- min(conf, 1-conf)
  n <- length(x)
  nn <- floor( n*conf )
  x <- sort(x)
  xx <- x[ (n-nn):(n) ] - x[1:(nn+1)]
  nnn <- which(xx==min(xx))[1]
  return( c( x[ nnn ], x[ n-nn+nnn-1 ] ) )
}

simula.MCMC <-
function(y, n.rep, n.af, n.suj, sujeito, aferidor, interacao, nchains, nreplicacao
, burnin, thin){

dados <- cbind(sujeito, aferidor, interacao, y)
colnames(dados) <- c("sujeito", "aferidor", "interacao", "valor")
dados <- as.data.frame(dados)
dados_bayes <- list(n.rep=n.rep, M=n.af, O=n.suj,
y=structure(.Data=dados$valor, .Dim=c(n.rep, n.af, n.suj)))

fit.brugs <- BRugsFit(modelFile="model.MCMC.txt", data=dados_bayes, inits
= c("inits1.txt", "inits2.txt"),
numChains = nchains,
parametersToSave=c("vari", "vara", "varinter", "vare", "theta", "icc", "icc2"),
nBurnin = burnin, nIter = nreplicacao, nThin = thin, coda = FALSE,
DIC = TRUE, working.directory = NULL, digits = 5, seed=NULL,
BRugsVerbose = getOption("BRugsVerbose"))

x <- samplesStats("*")

##### Salvando as estatísticas #####

## média ##

icc.mean <- x['icc', 'mean']
icc2.mean <- x['icc2', 'mean']
theta.mean <- x['theta', 'mean']
vara.mean <- x['vara', 'mean']
vare.mean <- x['vare', 'mean']
vari.mean <- x['vari', 'mean']

```

```

varinter.mean<-x['varinter','mean']

## desvio padrão ##

icc.sd<-x['icc','sd']
icc2.sd<-x['icc2','sd']
theta.sd<-x['theta','sd']
vara.sd<-x['vara','sd']
vare.sd<-x['vare','sd']
vari.sd<-x['vari','sd']
varinter.sd<-x['varinter','sd']

## MC error ##

icc.MC_error<-x['icc','MC_error']
icc2.MC_error<-x['icc2','MC_error']
theta.MC_error<-x['theta','MC_error']
vara.MC_error<-x['vara','MC_error']
vare.MC_error<-x['vare','MC_error']
vari.MC_error<-x['vari','MC_error']
varinter.MC_error<-x['varinter','MC_error']

## Quantil 2,5% ##

icc.val2.5pc<-x['icc','val2.5pc']
icc2.val2.5pc<-x['icc2','val2.5pc']
theta.val2.5pc<-x['theta','val2.5pc']
vara.val2.5pc<-x['vara','val2.5pc']
vare.val2.5pc<-x['vare','val2.5pc']
vari.val2.5pc<-x['vari','val2.5pc']
varinter.val2.5pc<-x['varinter','val2.5pc']

## Quantil 97,5% ##

icc.val97.5pc<-x['icc','val97.5pc']
icc2.val97.5pc<-x['icc2','val97.5pc']
theta.val97.5pc<-x['theta','val97.5pc']
vara.val97.5pc<-x['vara','val97.5pc']
vare.val97.5pc<-x['vare','val97.5pc']
vari.val97.5pc<-x['vari','val97.5pc']
varinter.val97.5pc<-x['varinter','val97.5pc']

## mediana ##

icc.median<-x['icc','median']
icc2.median<-x['icc2','median']
theta.median<-x['theta','median']
vara.median<-x['vara','median']
vare.median<-x['vare','median']
vari.median<-x['vari','median']
varinter.median<-x['varinter','median']

#### Amostras para o PHD #####

amostra.icc<-samplesSample('icc')
amostra.icc2<-samplesSample('icc2')
amostra.theta<-samplesSample('theta')
amostra.vari<-samplesSample('vari')
amostra.vara<-samplesSample('vara')
amostra.varinter<-samplesSample('varinter')
amostra.vare<-samplesSample('vare')

#### HPDs

```



```

icc.hpd.mcmc <- emp.hpd.m(amostra.icc)
icc2.hpd.mcmc <- emp.hpd.m(amostra.icc2)
theta.hpd.mcmc <- emp.hpd.m(amostra.theta)
vari.hpd.mcmc <- emp.hpd.m(amostra.vari)
vara.hpd.mcmc <- emp.hpd.m(amostra.vara)
varinter.hpd.mcmc <- emp.hpd.m(amostra.varinter)
vare.hpd.mcmc <- emp.hpd.m(amostra.vare)

theta.hpd.mcmc.inf <- theta.hpd.mcmc[1]
theta.hpd.mcmc.sup <- theta.hpd.mcmc[2]
icc.hpd.mcmc.inf <- icc.hpd.mcmc[1]
icc.hpd.mcmc.sup <- icc.hpd.mcmc[2]
icc2.hpd.mcmc.inf <- icc2.hpd.mcmc[1]
icc2.hpd.mcmc.sup <- icc2.hpd.mcmc[2]
vara.hpd.mcmc.inf <- vara.hpd.mcmc[1]
vara.hpd.mcmc.sup <- vara.hpd.mcmc[2]
vari.hpd.mcmc.inf <- vari.hpd.mcmc[1]
vari.hpd.mcmc.sup <- vari.hpd.mcmc[2]
vare.hpd.mcmc.inf <- vare.hpd.mcmc[1]
vare.hpd.mcmc.sup <- vare.hpd.mcmc[2]
varinter.hpd.mcmc.inf <- varinter.hpd.mcmc[1]
varinter.hpd.mcmc.sup <- varinter.hpd.mcmc[2]

#### Moda

kernel1 = density(amostra.icc, bw="nrd0", n = 512,
from=min(amostra.icc), to=max(amostra.icc))
posmo1 = which.max(kernel1$y)
icc.moda.mcmc = kernel1$x[posmo1]

kernel2 = density(amostra.icc2, bw="nrd0", n = 512,
from=min(amostra.icc2), to=max(amostra.icc2))
posmo2 = which.max(kernel2$y)
icc2.moda.mcmc = kernel2$x[posmo2]

kernel3 = density(amostra.theta, bw="nrd0", n = 512,
from=min(amostra.theta), to=max(amostra.theta))
posmo3 = which.max(kernel3$y)
theta.moda.mcmc = kernel3$x[posmo3]

kernel4 = density(amostra.vari, bw="nrd0", n = 512,
from=min(amostra.vari), to=max(amostra.vari))
posmo4 = which.max(kernel4$y)
vari.moda.mcmc = kernel4$x[posmo4]

kernel5 = density(amostra.vara, bw="nrd0", n = 512,
from=min(amostra.vara), to=max(amostra.vara))
posmo5 = which.max(kernel5$y)
vara.moda.mcmc = kernel5$x[posmo5]

kernel6 = density(amostra.varinter, bw="nrd0", n = 512,
from=min(amostra.varinter), to=max(amostra.varinter))
posmo6 = which.max(kernel6$y)
varinter.moda.mcmc = kernel6$x[posmo6]

kernel7 = density(amostra.vare, bw="nrd0", n = 512,
from=min(amostra.vare), to=max(amostra.vare))
posmo7 = which.max(kernel7$y)
vare.moda.mcmc = kernel7$x[posmo7]

```

```
#### Posteriori para as variáveis #####
```

```
post.icc<-buildMCMC('icc')
post.icc2<-buildMCMC('icc2')
post.theta<-buildMCMC('theta')
post.vari<-buildMCMC('vari')
post.vara<-buildMCMC('vara')
post.varinter<-buildMCMC('varinter')
post.vare<-buildMCMC('vare')
```

```
saida_MCMC =
list(icc.mean,icc2.mean,theta.mean,vara.mean,vare.mean,vari.mean,varinter.
mean,icc.sd,icc2.sd,theta.sd,vara.sd,vare.sd,vari.sd,
varinter.sd,icc.MC_error,icc2.MC_error,theta.MC_error,vara.MC_error,vare.M
C_error,vari.MC_error,varinter.MC_error,icc.val2.5pc,
icc2.val2.5pc,theta.val2.5pc,vara.val2.5pc,vare.val2.5pc,vari.val2.5pc,var
inter.val2.5pc,icc.val97.5pc,icc2.val97.5pc,theta.val97.5pc,
vara.val97.5pc,vare.val97.5pc,vari.val97.5pc,varinter.val97.5pc,icc.median
,icc2.median,theta.median,vara.median,vare.median,vari.median,varinter.med
ian,
amostra.icc, amostra.icc2, amostra.theta, amostra.vari, amostra.vara,
amostra.varinter, amostra.vare,
theta.hpd.mcmc,icc.hpd.mcmc, icc2.hpd.mcmc, theta.hpd.mcmc, vari.hpd.mcmc,
vara.hpd.mcmc, varinter.hpd.mcmc, vare.hpd.mcmc,
theta.hpd.mcmc.inf, theta.hpd.mcmc.sup,icc.hpd.mcmc.inf, icc.hpd.mcmc.sup,
icc2.hpd.mcmc.inf, icc2.hpd.mcmc.sup, vara.hpd.mcmc.inf,
vara.hpd.mcmc.sup,
vari.hpd.mcmc.inf, vari.hpd.mcmc.sup, vare.hpd.mcmc.inf,
vare.hpd.mcmc.sup, varinter.hpd.mcmc.inf, varinter.hpd.mcmc.sup,
icc.moda.mcmc, icc2.moda.mcmc, theta.moda.mcmc, vari.moda.mcmc,
vara.moda.mcmc, varinter.moda.mcmc, vare.moda.mcmc,
post.icc, post.icc2, post.theta, post.vari, post.vara, post.varinter,
post.vare)
```

```
names(saida_MCMC) =
c('icc.mean.mcmc','icc2.mean.mcmc','theta.mean.mcmc','vara.mean.mcmc','var
e.mean.mcmc','vari.mean.mcmc','varinter.mean.mcmc',
'icc.sd.mcmc','icc2.sd.mcmc','theta.sd.mcmc','vara.sd.mcmc',
'vare.sd.mcmc','vari.sd.mcmc','varinter.sd.mcmc','icc.MC_error.mcmc','icc2
.MC_error.mcmc','theta.MC_error.mcmc','vara.MC_error.mcmc','vare.MC_error.
mcmc',
'vari.MC_error.mcmc','varinter.MC_error.mcmc','icc.quantil2.5.mcmc','icc2.
quantil2.5.mcmc','theta.quantil2.5.mcmc','vara.quantil2.5.mcmc',
'vare.quantil2.5.mcmc','vari.quantil2.5.mcmc','varinter.quantil2.5.mcmc',
'icc.quantil97.5.mcmc','icc2.quantil97.5.mcmc','theta.quantil97.5.mcmc',
'vara.quantil97.5.mcmc','vare.quantil97.5.mcmc','vari.quantil97.5.mcmc','v
arinter.quantil97.5.mcmc','icc.median.mcmc',
'icc2.median.mcmc','theta.median.mcmc','vara.median.mcmc','vare.median.mcm
c','vari.median.mcmc','varinter.median.mcmc',
'amostra.icc','amostra.icc2','amostra.theta','amostra.vari','amostra.vara'
,'amostra.varinter','amostra.vare',
'theta.hpd.mcmc','icc.hpd.mcmc','icc2.hpd.mcmc','theta.hpd.mcmc','vari.hpd
.mcmc','vara.hpd.mcmc','varinter.hpd.mcmc','vare.hpd.mcmc',
'theta.hpd.mcmc.inf',
'theta.hpd.mcmc.sup','icc.hpd.mcmc.inf','icc.hpd.mcmc.sup','icc2.hpd.mcmc.
inf','icc2.hpd.mcmc.sup','vara.hpd.mcmc.inf','vara.hpd.mcmc.sup',
'vari.hpd.mcmc.inf','vari.hpd.mcmc.sup','vare.hpd.mcmc.inf','vare.hpd.mcmc
.sup','varinter.hpd.mcmc.inf','varinter.hpd.mcmc.sup',
'icc.moda.mcmc','icc2.moda.mcmc','theta.moda.mcmc','vari.moda.mcmc','vara.
moda.mcmc','varinter.moda.mcmc','vare.moda.mcmc',
'post.icc','post.icc2','post.theta','post.vari','post.vara','post.varinter
','post.vare')
```

```

return(saida_MCMC)

}

inicio.MCMC = Sys.time()

simuls.MCMC =
apply(y, 2, simula.MCMC, n.rep=n.rep, n.af=n.af, n.suj=n.suj, sujeito=sujeito, af
eridor=aferidor, interacao=interacao, nchains=nchains, nreplicacao=nreplicaca
o, burnin=burnin, thin=thin)

fim.MCMC = Sys.time()

tempo.MCMC = fim.MCMC - inicio.MCMC

matest.MCMC = sapply(simuls.MCMC, function(x) {
c(x$icc.mean.mcmc, x$icc2.mean.mcmc, x$theta.mean.mcmc, x$vara.mean.mcmc, x$va
re.mean.mcmc, x$vari.mean.mcmc, x$varinter.mean.mcmc,
x$icc.sd.mcmc, x$icc2.sd.mcmc, x$theta.sd.mcmc, x$vara.sd.mcmc,
x$vare.sd.mcmc, x$vari.sd.mcmc, x$varinter.sd.mcmc, x$icc.MC_error.mcmc, x$icc
2.MC_error.mcmc, x$theta.MC_error.mcmc, x$vara.MC_error.mcmc, x$vare.MC_error
.mcmc,
x$vari.MC_error.mcmc, x$varinter.MC_error.mcmc, x$icc.quantil2.5.mcmc, x$icc2
.quantil2.5.mcmc, x$theta.quantil2.5.mcmc, x$vara.quantil2.5.mcmc,
x$vare.quantil2.5.mcmc, x$vari.quantil2.5.mcmc, x$varinter.quantil2.5.mcmc, x
$icc.quantil97.5.mcmc, x$icc2.quantil97.5.mcmc, x$theta.quantil97.5.mcmc,
x$vara.quantil97.5.mcmc, x$vare.quantil97.5.mcmc, x$vari.quantil97.5.mcmc, x$
varinter.quantil97.5.mcmc, x$icc.median.mcmc,
x$icc2.median.mcmc, x$theta.median.mcmc, x$vara.median.mcmc, x$vare.median.mc
mc, x$vari.median.mcmc, x$varinter.median.mcmc,
x$theta.hpd.mcmc.inf, x$theta.hpd.mcmc.sup, x$icc.hpd.mcmc.inf, x$icc.hpd.mcm
c.sup, x$icc2.hpd.mcmc.inf, x$icc2.hpd.mcmc.sup, x$vara.hpd.mcmc.inf, x$vara.h
pd.mcmc.sup,
x$vari.hpd.mcmc.inf, x$vari.hpd.mcmc.sup, x$vare.hpd.mcmc.inf, x$vare.hpd.mcm
c.sup, x$varinter.hpd.mcmc.inf, x$varinter.hpd.mcmc.sup,
x$icc.moda.mcmc, x$icc2.moda.mcmc, x$theta.moda.mcmc, x$vari.moda.mcmc, x$vara
.moda.mcmc, x$varinter.moda.mcmc, x$vare.moda.mcmc)})

#
x$amostra.icc, x$amostra.icc2, x$amostra.theta, x$amostra.vari, x$amostra.vara
, x$amostra.varinter, x$amostra.vare,
#
x$post.icc, x$post.icc2, x$post.theta, x$post.vari, x$post.vara, x$post.varinte
r, x$post.vare,
#
x$gr.icc, x$gr.icc2, x$gr.theta, x$gr.vari, x$gr.vara, x$gr.varinter, x$gr.vare

dimnames(matest.MCMC) =
list(c('icc.mean.mcmc', 'icc2.mean.mcmc', 'theta.mean.mcmc', 'vara.mean.mcmc'
, 'vare.mean.mcmc', 'vari.mean.mcmc', 'varinter.mean.mcmc',
'icc.sd.mcmc', 'icc2.sd.mcmc', 'theta.sd.mcmc', 'vara.sd.mcmc',
'vare.sd.mcmc', 'vari.sd.mcmc', 'varinter.sd.mcmc', 'icc.MC_error.mcmc', 'icc2
.MC_error.mcmc', 'theta.MC_error.mcmc', 'vara.MC_error.mcmc', 'vare.MC_error.
mcmc',
'vari.MC_error.mcmc', 'varinter.MC_error.mcmc', 'icc.quantil2.5.mcmc', 'icc2.
quantil2.5.mcmc', 'theta.quantil2.5.mcmc', 'vara.quantil2.5.mcmc',
'vare.quantil2.5.mcmc', 'vari.quantil2.5.mcmc', 'varinter.quantil2.5.mcmc', '
icc.quantil97.5.mcmc', 'icc2.quantil97.5.mcmc', 'theta.quantil97.5.mcmc',
'vara.quantil97.5.mcmc', 'vare.quantil97.5.mcmc', 'vari.quantil97.5.mcmc', 'v
arinter.quantil97.5.mcmc', 'icc.median.mcmc',
'icc2.median.mcmc', 'theta.median.mcmc', 'vara.median.mcmc', 'vare.median.mcm
c', 'vari.median.mcmc', 'varinter.median.mcmc',

```

```

'theta.hpd.mcmc.inf', 'theta.hpd.mcmc.sup', 'icc.hpd.mcmc.inf', 'icc.hpd.mcmc
.sup', 'icc2.hpd.mcmc.inf', 'icc2.hpd.mcmc.sup', 'vara.hpd.mcmc.inf', 'vara.hp
d.mcmc.sup',
'vari.hpd.mcmc.inf', 'vari.hpd.mcmc.sup', 'vare.hpd.mcmc.inf', 'vare.hpd.mcmc
.sup', 'varinter.hpd.mcmc.inf', 'varinter.hpd.mcmc.sup',
'icc.moda.mcmc', 'icc2.moda.mcmc', 'theta.moda.mcmc', 'vari.moda.mcmc', 'vara.
moda.mcmc', 'varinter.moda.mcmc', 'vare.moda.mcmc')

matestmean.MCMC =
matest.MCMC[c("theta.mean.mcmc", "vari.mean.mcmc", "vara.mean.mcmc", "vare.me
an.mcmc", "varinter.mean.mcmc", "icc.mean.mcmc", "icc2.mean.mcmc"), ]

matestmedian.MCMC =
matest.MCMC[c("theta.median.mcmc", "vari.median.mcmc", "vara.median.mcmc", "v
are.median.mcmc", "varinter.median.mcmc", "icc.median.mcmc", "icc2.median.mcm
c"), ]

matestmoda.MCMC =
matest.MCMC[c("theta.moda.mcmc", "vari.moda.mcmc", "vara.moda.mcmc", "vare.mo
da.mcmc", "varinter.moda.mcmc", "icc.moda.mcmc", "icc2.moda.mcmc"), ]

matestpont.MCMC = rbind(matestmean.MCMC, matestmedian.MCMC,
matestmoda.MCMC)

matestsd.MCMC =
matest.MCMC[c("theta.sd.mcmc", "vari.sd.mcmc", "vara.sd.mcmc",
"varinter.sd.mcmc", "vare.sd.mcmc", "icc.sd.mcmc", "icc2.sd.mcmc"), ]

matestsd.MC.error_MCMC =
matest.MCMC[c("theta.MC_error.mcmc", "vari.MC_error.mcmc", "vara.MC_error.mc
mc",
"varinter.MC_error.mcmc", "vare.MC_error.mcmc", "icc.quantil2.5.mcmc", "icc.M
C_error.mcmc", "icc2.MC_error.mcmc"), ]

matestic1.MCMC =
matest.MCMC[c("theta.quantil2.5.mcmc", "vari.quantil2.5.mcmc", "vara.quantil
2.5.mcmc", "varinter.quantil2.5.mcmc", "vare.quantil2.5.mcmc", "icc.quantil2.
5.mcmc", "icc2.quantil2.5.mcmc",
'theta.hpd.mcmc.inf', 'vari.hpd.mcmc.inf', 'vara.hpd.mcmc.inf', 'varinter.hpd
.mcmc.inf', 'vare.hpd.mcmc.inf', 'icc.hpd.mcmc.inf', 'icc2.hpd.mcmc.inf'), ]

matestic2.MCMC =
matest.MCMC[c("theta.quantil97.5.mcmc", "vari.quantil97.5.mcmc", "vara.quant
il97.5.mcmc", "varinter.quantil97.5.mcmc", "vare.quantil97.5.mcmc", "icc.quan
til97.5.mcmc", "icc2.quantil97.5.mcmc",
'theta.hpd.mcmc.sup', 'vari.hpd.mcmc.sup', 'vara.hpd.mcmc.sup', 'varinter.hpd
.mcmc.sup', 'vare.hpd.mcmc.sup', 'icc.hpd.mcmc.sup', 'icc2.hpd.mcmc.sup'), ]

# Calculando Medidas resumo
media.MCMC = apply(matestpont.MCMC, 1, mean)
vicio.MCMC = media.MCMC - rep(params, 3)
varian.MCMC = apply(matestpont.MCMC, 1, var)
eqm.MCMC = varian.MCMC + vicio.MCMC^2

params2 = rep(params, 2)

cober.MCMC = rep(0, 14)
ampl.MCMC = rep(0, 1)

for(j in 1:14){
  cober.MCMC[j] =
100*sum(matestic1.MCMC[j, ]<=params2[j]&params2[j]<=matestic2.MCMC[j, ])/rep
1

```

```

    ampl.MCMC[j] = mean(matestic2.MCMC[j,]-matestic1.MCMC[j,])
  }
names(cober.MCMC) =
c("theta.cobertura.mcmc.central", "vari.cobertura.mcmc.central", "vara.cober
tura.mcmc.central", "varinter.cobertura.mcmc.central", "vare.cobertura.mcmc.
central", "icc.cobertura.mcmc.central", "icc2.cobertura.mcmc.central",
"theta.cobertura.mcmc.hpd", "vari.cobertura.mcmc.hpd", "vara.cobertura.mcmc.
hpd", "varinter.cobertura.mcmc.hpd", "vare.cobertura.mcmc.hpd", "icc.cobertur
a.mcmc.hpd", "icc2.cobertura.mcmc.hpd")
names(ampl.MCMC) =
c("theta.cobertura.mcmc.central", "vari.cobertura.mcmc.central", "vara.cober
tura.mcmc.central", "varinter.cobertura.mcmc.central", "vare.cobertura.mcmc.
central", "icc.cobertura.mcmc.central", "icc2.cobertura.mcmc.central",
"theta.cobertura.mcmc.hpd", "vari.cobertura.mcmc.hpd", "vara.cobertura.mcmc.
hpd", "varinter.cobertura.mcmc.hpd", "vare.cobertura.mcmc.hpd", "icc.cobertur
a.mcmc.hpd", "icc2.cobertura.mcmc.hpd")

# Salvando tabela de resultados

Quadro.MCMC.1 =
data.frame(media=media.MCMC, vicio=vicio.MCMC, variancia=varian.MCMC, eqm=eqm
.MCMC)
Quadro.MCMC.2 = data.frame(amplitude=ampl.MCMC, cobertura=cober.MCMC)

```