

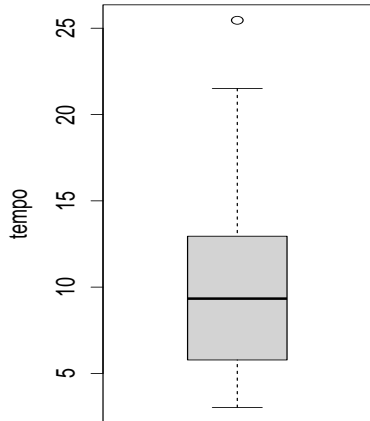
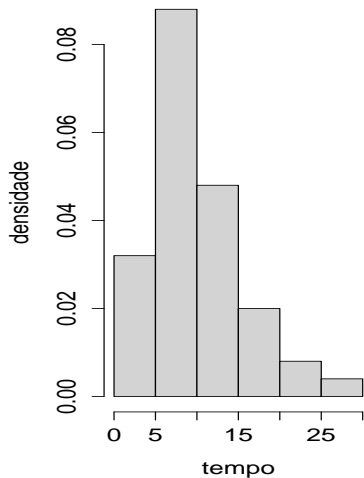
Métodos analíticos e numéricos para a obtenção de distribuições à posteriori: exemplos univariados

Prof. Caio Azevedo

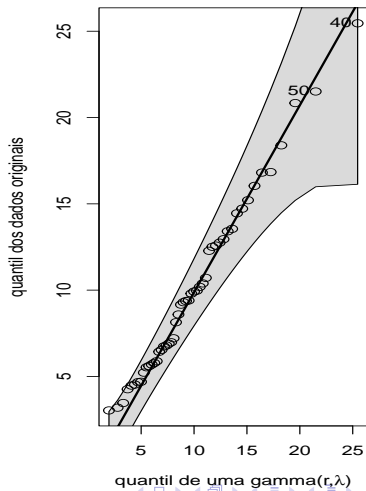
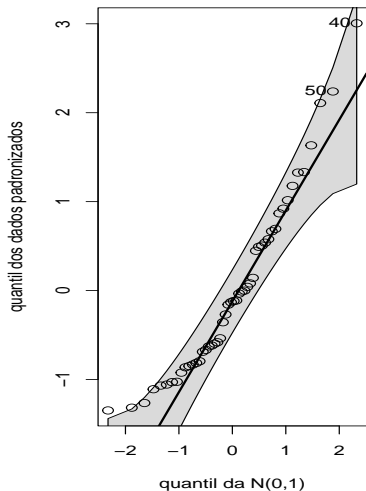
Exemplo: dados da potência de turbina de aviões

- Conjuntos de dados relativos ao desempenho de 5 tipos de turbina de avião. Tempo (em unidade de milhões de ciclos) até a perda de velocidade.
- Objetivo inicial: comparar os tipos de turbinas.
- Por enquanto, vamos considerar que as turbinas são de um único tipo.
- Seja x_i o tempo (observado) até a perda de velocidade da i -ésima turbina.
- Objetivo: fazer inferência sobre o tempo médio até a perda da velocidade. Veja também: [aqui](#) e [aqui](#).

Histograma e box-plot dos tempos



QQplot [N(0,1) e gama(r, λ)] e dos tempos



Análise descritiva

- Os parâmetros usados no QQplot da distribuição gama, correspondem à estimativa pelo **método dos momentos** de uma distribuição $\text{gama}(r, \lambda)$, cuja esperança é igual à $r\lambda$. Tais estimativas possuem solução analítica.
- Podemos considerar as **estimativas de máxima verossimilhança** que, nesse caso, não tem solução analítica. Precisamos utilizar métodos numéricos (discutiremos mais à frente e também veja [aqui](#) e [aqui](#)).
- A distribuição é assimétrica positiva, com curtoso um pouco acima da normal e a presença de um outlier. A suposição de normalidade não parece razoável, ocorrendo o contrário para a suposição de distribuição gama.

Análise descritiva

tempo	medida resumo
média	9,98
var.	26,54
dp	5,15
1°Q	5,80
med.	9,34
3°Q	12,90
CV(%)	51,64
min.	3,03
max.	25,46
ca	0,91
curt.	3,42

Análise descritiva

- O CV da distribuição $gamma(r, \lambda)$ (na parametrização adotada) é
$$CV = 100 \times \frac{\sqrt{r\lambda}}{r\lambda} = \frac{100}{\sqrt{r}}.$$
- Usando $\tilde{r}_{MM} \approx 3,83$. Assim, $CV_{MM} \approx 51,12$. Esse valor é próximo daquele observado, (51,64), que é mais um indício à favor da utilização da distribuição gama.

Modelagem

- Vamos assumir que $X_i|r \stackrel{i.i.d}{\sim} \text{gama}(r, \lambda)$, $i = 1, 2, \dots, n = 50$, com λ conhecido (por enquanto). Neste caso $\mathcal{E}(X_i|r) = r\lambda$.
- Assim

$$p(x|r) = \frac{1}{\lambda^r \Gamma(r)} e^{-x/\lambda} x^{r-1} \mathbb{1}_{(0, \infty)}(x)$$

- Verossimilhança:

$$\begin{aligned} p(\mathbf{x}|r) &= \frac{1}{\lambda^{nr} \Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} \\ &\propto \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} \end{aligned}$$

- Qual é a família conjugada de priors?

Modelagem

■ Posteriori :

$$\begin{aligned} p(r|\mathbf{x}) &= \frac{\frac{1}{\lambda^{nr}\Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} p(r)}{\int_0^\infty \frac{1}{\lambda^{nr}\Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} p(r) dr} \\ &= \frac{\frac{1}{\lambda^{nr}\Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r)}{\int_0^\infty \frac{1}{\lambda^{nr}\Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r) dr} = \frac{\frac{1}{\lambda^{nr}\Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r)}{p(\mathbf{x})}. \end{aligned}$$

Modelagem

- Independentemente da priori escolhida não é possível obter a posteriori de forma analítica (exata).
- Constante de normalização:

$$p(\mathbf{x}) = \int_0^\infty \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r) dr$$

- A integral correspondente à constante de normalização não tem solução analítica exata, independentemente da escolha da priori.

Modelagem

- Pode-se obter aproximações:
 - Para a posteriori (sem se obter momentos e quantis): obtenção numérica somente da constante de normalização.
 - Somente para os momentos e/ou quantis (sem se obter aproximação para a posteriori): [MML](#) (Marginal Maximum Likelihood), [MMAP](#) (Maximum Marginal a Posteriori).
 - Posteriori, momentos e quantis ([ABC](#), [INLA](#), [MCMC](#)).
 - Veja [aqui](#) também: Caps. 5, 7, 9 e 10.

Obtenção das posterioris (marginais)

- Métodos Analíticos:
 - Aproximação à distribuição normal ([link](#)).
 - Método de Laplace ([link](#), Cap. 7).
- Métodos Numéricos:
 - Integração/Maximização numéricas ([link](#)).
 - Métodos de simulação estocástica não iterativos (como nos problemas de comparação das médias de: 1) duas dist. de [Poisson](#) e 2) duas dist. [normais](#) com variâncias desconhecidas e diferentes).
 - Approximate Bayesian Computation (ABC).
 - Integrated Nested Laplace Approximation (INLA).
 - Métodos de simulação estocástica de Monte Carlo via Cadeias de Markov (MCMC).

Características (Métodos Analíticos)

- Aproximação à distribuição normal: fornece, simultaneamente, aproximação da própria posterior, bem como dos momentos e quantis.
- Aproximação de Laplace: fornece aproximação da posteriori, mas os momentos e quantis tem de ser calculados posteriormente.

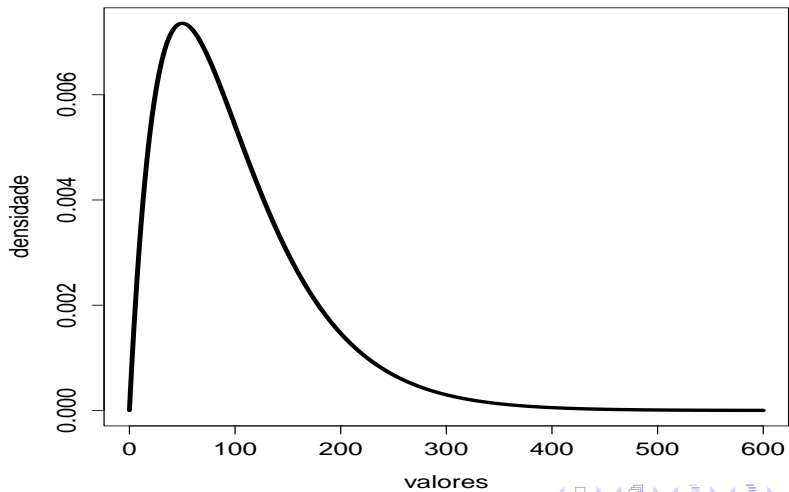
Características (Métodos Numéricos)

- Integração/Maximização numéricas: fornecem aproximação da posteriori mas (dependendo da abordagem) os momentos e quantis tem de ser calculados posteriormente.
- Métodos de simulação estocástica (não iterativos e MCMC): fornecem uma amostra aleatória das posteriores e, de posse desta, pode-se estimar (não parametricamente) as posteriores e calcular medidas -resumo para estimar os momentos e quantis.
- As vezes, tem-se de usar métodos analíticos e numéricos em conjunto.

Voltando ao exemplo

- Suponha que o pesquisador (responsável pelo experimento) tenha fornecido as seguintes informações:
 - De experimentos anteriores: $\tilde{\lambda} \approx 2,66$.
 - À priori: $\mathcal{E}(r) = 100$ e $\mathcal{V}(r) = 5000$.
- Assim, vamos assumir que : $r \sim \text{gama}(\alpha, \gamma)$, ($\alpha = 2, \gamma = 50$).

Priori



Aproximação à distribuição normal (univariada)

- Lembrando: $p(\theta)$: priori, $p(\mathbf{x}|\theta)$: verossimilhança e $p(\theta|\mathbf{x})$: posteriori.
- Sob certas condições de regularidade (expansão em [séries de Taylor](#)), pode-se aproximar a distribuição à posteriori de interesse como:

$$p(\theta|\mathbf{x}) \approx \frac{\sqrt{h_n}}{\sqrt{2\pi}} \exp \left\{ -\frac{h_n}{2} (\theta - m_n)^2 \right\},$$

em que $h_n = h_0 + h(\tilde{\theta})$, $m_n = h_n^{-1} (h_0 m_0 + h(\tilde{\theta})\tilde{\theta})$.

Cont.

- Em que:
 - m_0 é a moda da distribuição à priori (calculada, eventualmente, de modo numérico).
 - $\tilde{\theta}$ é a estimativa de máxima verossimilhança de θ (calculada, eventualmente, de modo numérico).
 - Além disso,

$$h_0 = -\left. \frac{d^2 \ln p(\theta)}{d\theta^2} \right|_{\theta=m_0} \text{ e } h(\tilde{\theta}) = -\left. \frac{d^2 \ln p(\mathbf{x}|\theta)}{d\theta^2} \right|_{\theta=\tilde{\theta}}$$

- Dessa forma, temos que $\hat{\theta}_{EAP} = \hat{\theta}_{MeAP} = \hat{\theta}_{MAP} \approx m_n$ e $VAP(\theta) \approx h_n^{-1}$.

Característica da AN (aproximação à distribuição normal)

- Relativamente rápida.
- Útil quando as obtenções da moda da priori e do estimador de MV são simples.
- Os estimadores (EAP,MAP,MeAP, EPAP) e intervalos de credibilidade não, necessariamente, respeitarão o espaço paramétrico.
- Quanto mais distante da distribuição normal for a posteriori, maior tem de ser o tamanho da amostra para que a aproximação seja satisfatória.

Voltando ao exemplo

- Lembremos que $r \sim \text{gama}(\alpha, \gamma)$, ou seja,

$$p(r) = \frac{1}{\gamma^\alpha \Gamma(\alpha)} e^{-r/\gamma} r^{\alpha-1} \mathbb{1}_{(0, \infty)}(r).$$

- Assim

$$-\frac{d^2 \ln p(r)}{dr^2} = \frac{(\alpha - 1)}{r^2},$$

$$-\frac{d^2 \ln p(\mathbf{x}|r)}{dr^2} = \frac{n}{(\Gamma(r))^2} \left[\Gamma''(r)\Gamma'(r) - (\Gamma'(r))^2 \right] = n\Psi'(r),$$

em que $\Gamma^{(\cdot)}(r)$ é a derivada de ordem (\cdot) da função gama $\Gamma(r)$,

$$\Psi(r) = \frac{d \ln \Gamma(r)}{dr} \text{ (função digama) e } \Psi'(r) = \frac{d\Psi(r)}{dr}.$$

Voltando ao exemplo e usando o programa R

- Neste caso, $m_0 = \gamma(\alpha - 1)$. contudo, a estimativa de mv de r não é obtenível analiticamente. Assim, alguma método numérico de maximização deve ser empregado.
- A função `optim`, do programa R, permite maximizar funções utilizando algum algoritmo numérico (otimizador). Particularmente, pode-se usar o método de otimização BFGS ([aqui](#)).
- Tal função já vem pré instalada no R.
- Para utilizar os algoritmos de Newtom-Raphson ou Escore de Fisher, pode-se recorrer ao pacote `maxLik` (usando a função `maxLik`).

Comandos no R: obtenção da estimativa de MV

```
# função optim
fr<- function(r,lambda,n,x){
logl<- -n*lgamma(r)-n*r*log(lambda) + (r-1)*sum(log(x))
return(-logl)}

# Aproximação pela normal
resultlik<-optim((mean(vx)^2/var(vx)),fr,lambda=lambda,n=n,
                 x=vx,method="BFGS",hessian=T)

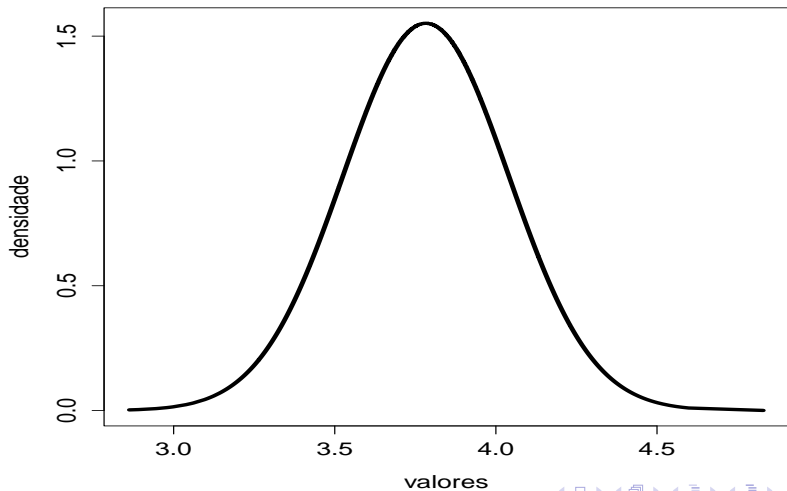
rmv <- resultlik$par
hr <- resultlik$hessian
hn<- h0 + hr

mn <- solve(hn)*(h0*m0 + hr*rmv)
```

Aplicação da aproximação à normal

- Neste caso, $m_0 = 50$, $\tilde{r} = 3,78$, $h_0 = 0,0004$, $h(\tilde{r}) = 15,12$
- Assim, $r|\mathbf{x} \approx N(3,78, 0,07)$.
- Portanto, $\tilde{r}_{EAP} \approx \tilde{r}_{MAP} \approx \tilde{r}_{MeAP} \approx 3,78$, $EPAP(r) \approx 0,26$.
- $IC_B(r, 95\%) \approx [3,28; 4,29]$.

Aproximação à normal



Aproximação de Laplace (AL)

- Suponha que desejamos calcular $\int_a^b e^{Mf(x)} dx$. Admita que exista um único máximo global para $f(\cdot)$, e denote-o por x_0 . Então (expansão em Séries de Taylor):

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2 + R,$$

$R = O[(x - x_0)^3] \rightarrow R < k(x - x_0)^3$, para algum $k > 0$ e $f'(x_0) = 0$, $f''(x_0) < 0$, $f(\cdot)$, é derivada de ordem (\cdot) ([link](#)).

- Portanto:

$$f(x) \approx f(x_0) - \frac{1}{2}|f''(x_0)|(x - x_0)^2.$$

Cont.

- Aproximação de Laplace:

$$\begin{aligned}\int_a^b e^{Mf(x)} dx &\approx e^{Mf(x_0)} \int_a^b e^{-M|f''(x_0)|(x-x_0)^2/2} dx \\ &= \sqrt{2\pi} e^{Mf(x_0)} \sigma \left\{ \Phi\left(\frac{b-x_0}{\sigma}\right) - \Phi\left(\frac{a-x_0}{\sigma}\right) \right\},\end{aligned}$$

em que $\sigma = \sqrt{\sigma^2}$, $\sigma^2 = (M|f''(x_0)|)^{-1}$, $\Phi(\cdot)$ é a fda da normal padrão.

Característica da AL

- Relativamente rápida.
- Apropriada quando o integrando é unimodal e o máximo pode ser obtido facilmente (ainda que seja necessário utilizar métodos numéricos).
- Inapropriada: em integrais múltiplas à medida que a dimensão aumenta e/ou o máximo é complicado de ser obtido.
- Os **estimadores** (EAP,MAP,MeAP, EPAP) e **intervalos de credibilidade** não, necessariamente, respeitarão o espaço paramétrico.

Cont. do exemplo da distribuição gama

- Nosso objetivo é utilizar a aproximação de Laplace para calcular:

$$\int_0^{\infty} g(r) dr = \int_0^{\infty} e^{f(r)} dr, \text{ em que } f(r) = \ln g(r), M = 1, \text{ ou seja:}$$

$$p(\mathbf{x}|r)p(r) = g(r) = \frac{1}{\lambda^{nr}\Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} \frac{\gamma^\alpha}{\Gamma(\alpha)} e^{-r/\gamma} r^{\alpha-1}$$

- Note que, neste caso, o ponto de máximo (r_0) corresponde à moda da posteriori (MAP) que, neste caso, será obtida numericamente (`optim`).

- Além disso, $f''(r) = - \left[n\Psi'(r) + \frac{\alpha - 1}{r} \right]$.

Aproximação de Laplace no R

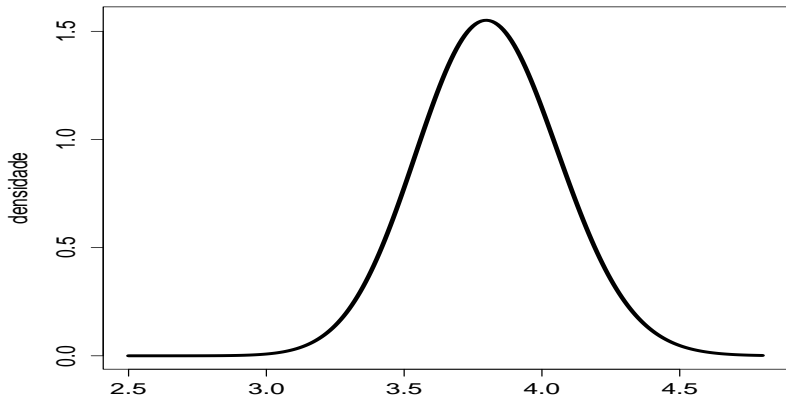
```
# Obtendo o MAP
fpost<- function(r,lambda,n,vx,alphap,gammap)
{logl<- -n*lgamma(r)-n*r*log(lambda) - n*mean(vx)/lambda +
  (r-1)*sum(log(vx)) -r/gammap +(alphap-1)*log(r) -
  alphap*log(gammap) - lgamma(alphap)}return(-logl)}
# extraindo os resultados
resultpost<-optim((mean(vx)^2/var(vx)), fpost, lambda=lambda,
n=n,vx=vx, alphap=alphap,gammap=gammap,method="BFGS",
hessian=T)
r0 <- resultpost$par
hr0 <- resultpost$hessian
```

Aproximação de Laplace no R

```
#####  
# Aproximação da constante de normalização  
sigma <- solve(sqrt(abs(hr0)))  
cn <- as.numeric(sqrt(2*pi)*  
exp(-fpost(r0,lambda,n,vx,alphap,gammap))*sigma*  
(1-pnorm(-r0/sigma)))  
auxpost <- seq(2.5,4.8,0.01)  
fpostAL <- exp(-fpost(auxpost,lambda,n,vx,alphap,gammap))/cn  
plot(auxpost,fpostAL,type="l",xlab="valores",ylab="densidade",  
main="",cex=1.3,cex.lab=1.3,cex.axis=1.3,lwd=4)
```

Cont. do exemplo da distribuição gama

- Assim $\int_0^{\infty} g(r) dr \approx e^{f(r_0)} \sqrt{2\pi} (f''(r_0))^{-1} = 1,526686e - 67$ e



Comentários

- Note que obtivemos apenas uma aproximação para a constante de normalização.
- Lembre, ainda, que

$$\mathcal{E}(h(r)|\mathbf{x}) = \frac{\int_0^{\infty} h(r) \frac{1}{\lambda^{nr} \Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} p(r) dr}{\int_0^{\infty} \frac{1}{\lambda^{nr} \Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} p(r) dr} = \frac{\int_0^{\infty} h(r) g(r) dr}{\int_0^{\infty} g(r) dr}$$

- Assim, para obtermos, por exemplo, o EAP, teríamos que também aproximar $\int_0^{\infty} r g(r) dr$. Para obter o DPAP teríamos também que aproximar $\int_0^{\infty} r^2 g(r) dr$.

Comentários

- Posteriori:

$$p(r|\mathbf{x}) = \frac{\frac{1}{\lambda^{nr}\Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} p(r)}{\int_0^{\infty} \frac{1}{\lambda^{nr}\Gamma(r)^n} e^{-n\bar{x}/\lambda} \prod_{i=1}^n x_i^{r-1} p(r) dr} = \frac{g(r)}{\int_0^{\infty} g(r) dr}$$

- Para obtermos o estimador MAP, teríamos que maximizar $g(r)$.

Comentários

- Para obtermos a MeAP, temos que encontrar o valor “q” , solução da seguinte equação

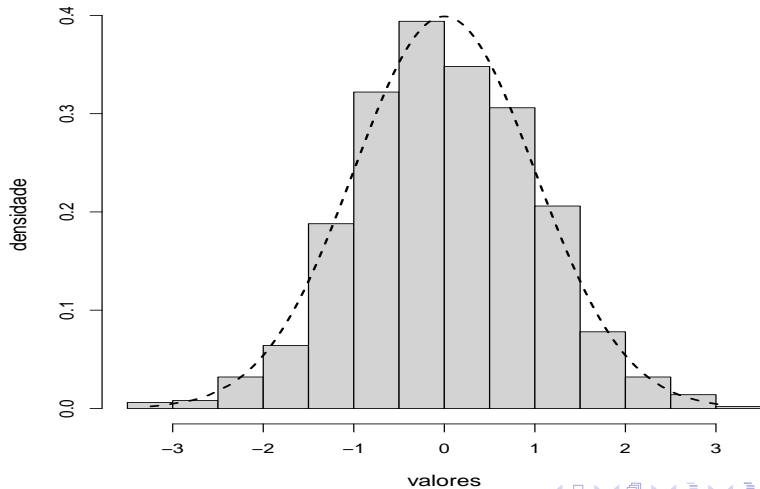
$$F(q|\mathbf{x}) = 1/2 \leftrightarrow \frac{\int_0^q g(r)dr}{\int_0^\infty g(r)dr} = 1/2 \leftrightarrow \frac{\int_0^q g(r)dr}{\int_0^\infty g(r)dr} - \frac{1}{2} = 0$$

- Nesse caso, métodos numéricos devem ser empregados para se obter o MAP e o MdAP. Falaremos sobre eles adiante.
- Exercícios: obter, numericamente, EAP, DPAP, MAP, MdAP.

Integração por Quadratura

- Substituir (no caso univariado) o cálculo da área sob a curva pela soma das áreas de um número finito de retângulos.
- Substituir (no caso multivariado) o cálculo da área sob a curva pela soma das áreas de um número finito de paralelepípedos.
- Ponto-chave: Definir os pontos e pesos de quadratura.
- Formas de cálculo dos pontos e pesos:
 - Não-adaptativa: mantem-se fixo os pontos e pesos.
 - Adaptativa: muda-se os pontos e/ou os pesos.

Graficamente



Características

- Em geral, na Estatística (frequentista ou bayesiana), deseja-se calcular

$$\mathcal{E}(w(X)) = \int_a^b w(x)f(x)dx,$$

onde $f(\cdot)$ é alguma fdp, $-\infty \leq a < b \leq \infty$.

- Sendo assim, $f(\cdot)$ terá massa relevante em apenas um subconjunto de $[a, b]$.
- Define-se um conjunto de pontos, x_1, x_2, \dots, x_m e os respectivos pesos associados $A_1, A_2, \dots, A_m, A_i = A(x_i)$. Via de regra, os pesos correspondem aos comprimentos dos intervalos determinados pelos pontos de quadraturas.

Cont.

- Pode-se considerar que cada ponto corresponde ao valor médio de intervalos de mesmo comprimento.
- Assim

$$\int_a^b w(x)f(x)dx \approx \sum_{i=1}^m A_i w(x_i)f(x_i)$$

- Existem várias formas de se determinar os pontos e pesos. Em geral, isto é feito a partir do máximo (moda) de $f(\cdot)$. Pontos ótimos, em muitos casos, são obtidos através de certos polinômios.
- Polinômios de Gauss-Hermite, Gauss-Legendre, Jacobi, Legendre, Gauss-Laguerre etc. Veja [Abramowitz & Stegun \(1972\)](#).

Quadratura Adaptativa

- Atualização dos pontos e/ou dos pesos de quadratura.
- Nosso interesse continua sendo calcular $h = \int_a^b w(x)f(x)dx$.
- Algoritmo
 - Calcule duas aproximações para h ,
$$S_1 = S(f(x); a, b) = \sum_{i=1}^m A_i w(x_i) f(x_i)$$
 e
$$S_2 = S(f(x); a, (a+b)/2) + S(f(x); (a+b)/2, b)$$
 - Calcule $\epsilon = |S_1 - S_2|$.
 - Se $\epsilon < \tau$, pare, caso contrário, aplique o Passo 1, recursivamente, até que a precisão requerida seja atingida.

Quadratura Adaptativa no R

- Função **integrate** (já vem instalada nas versões mais recentes).
- Permite calcular integrais, usando a quadratura adaptativa, para funções pré-definidas pelo usuário, em um dado intervalo.
- Admite limites infinitos.
- O integrando pode ser definido pelo usuário.

Voltando ao exemplo

- Posteriores :

$$p(r|\mathbf{x}) = \frac{\frac{1}{\lambda^{nr}\Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r)}{\int_0^{\infty} \frac{1}{\lambda^{nr}\Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r) dr}$$

Voltando ao exemplo

- Momentos :

$$\mathcal{E}(h(R)|\mathbf{x}) = \frac{\int_0^{\infty} h(r) \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r) dr}{\int_0^{\infty} \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r) dr}$$

Voltando ao exemplo

- Numericamente

$$\mathcal{E}(h(R)|\mathbf{x}) \approx \frac{\sum_{j=1}^m h(r_j) \frac{1}{\lambda^{nr_j} \Gamma(r_j)^n} \prod_{i=1}^n x_i^{r_j-1} p(r_j) A_j}{\sum_{j=1}^m \frac{1}{\lambda^{nr_j} \Gamma(r_j)^n} \prod_{i=1}^n x_i^{r_j-1} p(r_j) A_j}$$

Função integrate: comandos no R

```
# Quadratura adaptativa
fpost<- function(r,lambda,n,vx,alphap,gammap)
{logl<- -n*lgamma(r)-n*r*log(lambda) + (r-1)*sum(log(vx))
-r/gammap +(alphap-1)*log(r) - (alphap-1)*log(gammap)
  return(exp(logl))}
# Constante de normalização
cn<-integrate(fpost,0,Inf,lambda,n,vx,alphap,gammap)$value
auxpost <- seq(2.5,4.8,0.01)
fpostQA <- (fpost(auxpost,lambda,n,vx,alphap,gammap))/cn
plot(auxpost,fpostQA,type="l",xlab="valores", ylab="densidade",
main="", cex=1.3, cex.lab=1.3, cex.axis=1.3, lwd=4)
```

Função integrate

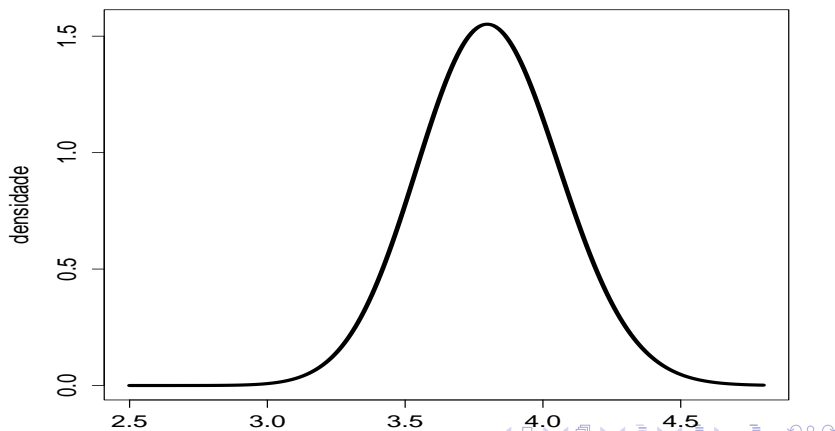
```
# EAP
fpostEAP<- function(r,lambda,n,vx,alphap,gammap)
{logl<- -n*lgamma(r)-n*r*log(lambda) + (r-1)*sum(log(vx))
-r/gammap +(alphap-1)*log(r) - (alphap-1)*log(gammap)
  return(r*exp(logl))}
EAP<-integrate(fpostEAP,0,Inf,lambda,n,vx,alphap,gammap)
$value/cn
```

Função integrate

```
# EPAP
fpostEAP2<- function(r,lambda,n,vx,alphap,gammap)
{logl<- -n*loggamma(r)-n*r*log(lambda) + (r-1)*sum(log(vx))
-r/gammap +(alphap-1)*log(r) - (alphap-1)*log(gammap)
  return((r^2)*exp((logl)))}
EPAP<-sqrt(integrate(fpostEAP2,0,Inf,lambda,n,vx,alphap,gammap)
$value/cn -EAP^2)
```

Função integrate: posteriori, EAP e DPAP

- $\tilde{r}_{EAP} \approx 3,81$, $EPAP(r) \approx 0,26$.



Continuação: Moda a Posteriori

- Maximização numérica da posteriori (para obter o MAP) usando a função $\ln p(r|\mathbf{x})$. Note que, é equivalente a maximizar o log da posteriori, ou seja:

$$\ln p(r|\mathbf{x}) = \ln p(\mathbf{x}|r) + \ln p(r) + \ln cn$$

cn = constante de normalização.

- Pode-se usar $-\left(\frac{d^2 \ln p(r|\mathbf{x})}{dr^2}\right)^{-1/2} \Big|_{r=r_0}$ como medida de precisão associada ao MAP, em que r_0 é a estimativa MAP.
- $\tilde{r}_{MAP} \approx 3,80$.

Função optim (obtenção da moda à posteriori)

```
fpost<- function(r,lambda,n,vx,alphap,gammap)
{logl<- -n*lgamma(r)-n*r*log(lambda) - n*mean(vx)/lambda +
  (r-1)*sum(log(vx)) -r/gammap +(alphap-1)*log(r) -
  alphap*log(gammap) - lgamma(alphap)
  return(-logl)}
resultpost<-optim((mean(vx)^2/var(vx)),fpost,lambda=lambda,n=n,
vx=vx,alphap=alphap,gammap=gammap,method="BFGS",hessian=T)
r0 <- resultpost$par
hr0 <- resultpost$hessian
```

Continuação: Mediana a Posteriori

- Por definição

$$F(r_{MeAP}|\mathbf{x}) = 0,5$$

- Podemos resolver a equação acima usando o algoritmo de Newton Raphson:

$$r^{(t)} = r^{(t-1)} - \left(p(r^{(t-1)}|\mathbf{x}) \right)^{-1} F(r^{(t-1)}|\mathbf{x}), t = 1, 2, \dots$$

- $\tilde{r}_{MdAP} \approx 3,80$.

Comandos no R: cálculo da mediana a posteriori (Ad Hoc)

```
fpost<- function(r,lambda,n,v.x,alphap,gammap)
{logl<- -n*lgamma(r)-n*r*log(lambda) + (r-1)*sum(log(v.x))
-r/gammap +(alphap-1)*log(r) - (alphap-1)*log(gammap)
return(exp(logl))}
cn<-integrate(fpost,0,Inf,lambda,n,vx,alphap,gammap)$value
fda <-integrate(fpost,0,4,lambda,n,vx,alphap,gammap)$value/cn
q10<-4;iter <-1; while(iter<=10){gr1 <-
integrate(fpost,0,q10,lambda,n,vx,alphap,gammap)$value/cn -0.5
hr1 <- fpost(q10,lambda,n,vx,alphap,gammap)/cn
q1t <- q10 - solve(hr1)*gr1
iter <- iter + 1;q10 <- q1t}
```

Integração por Monte Carlo

- **Lei Forte dos Grandes Números:** Seja X_1, X_2, \dots uma sequência i.i.d. (condicionada em um vetor de parâmetros θ), tal que $\mathcal{E}(X_i|\theta) = \mu, |\mu| < \infty, \forall i$. Então

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{q.c.} \mu.$$

- Integração por Monte Carlo.

- 1 Gere um conjunto de m variáveis aleatórias i.i.d, digamos

x_1, x_2, \dots, x_m .

- 2 Calcule $h \approx \frac{1}{m} \sum_{i=1}^m w(x_i)$.

Integração por Monte Carlo

- Em outras palavras, as integrais de interesse devem ser escritas como o valor esperado de alguma variável aleatória com distribuição conhecida.
- Voltando ao exemplo. Obtendo a constante de normalização:

$$\begin{aligned} I &= \int_0^\infty \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} p(r) dr = \int_0^\infty \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} \frac{e^{-r/\gamma} r^{\alpha-1}}{\Gamma(\alpha) \gamma^\alpha} dr \\ &\equiv \int_0^\infty \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} e^{-r/\gamma} r^{\alpha-1} \phi e^{r/\phi} \left(\frac{e^{-r/\phi}}{\phi} \right) dr \\ &= \int_0^\infty w(r) \frac{e^{-r/\phi}}{\phi} dr = \mathcal{E}(W(R)|\phi) \end{aligned}$$

Integração por Monte Carlo

- Em que $\phi = 3,78$ (estimativa de mv) e

$$w(r) = \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} \phi e^{-r(1/\gamma+1/\phi)} r^{\alpha-1}.$$

- Então, $I = \mathcal{E}(W(R)|\phi)$, em que $R|\phi \sim \exp(\phi)$.
- Portanto, para aproximar a integral acima, podemos simular uma amostra aleatória de tamanho m de uma va $\exp(\phi)$, digamos r_1, \dots, r_m , e calcular

$$I \approx \phi \frac{1}{m} \sum_{j=1}^m \frac{1}{\lambda^{nr_j} \Gamma(r_j)^n} \prod_{i=1}^n x_i^{r_j-1} e^{-r_j(1/\gamma+1/\phi)} r_j^{\alpha-1}.$$

Integração por Monte Carlo

- Para o cálculo de $\mathcal{E}(R|\mathbf{x})$ e $\mathcal{E}(R^2|\mathbf{x})$, os desenvolvimentos são análogos. Mais especificamente.

$$I_1 = \mathcal{E}(R|\mathbf{x}) = \int_0^\infty w_1(r) \frac{e^{-r/\phi}}{\phi} dr = \mathcal{E}(W(R)|\phi),$$

$$I_2 = \mathcal{E}(R^2|\mathbf{x}) = \int_0^\infty w_2(r) \frac{e^{-r/\phi}}{\phi} dr = \mathcal{E}(W(R)|\phi),$$

$$\text{em que } w_1(r) = \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} \phi e^{-r(1/\gamma+1/\phi)} r^{\alpha-1} r$$

$$w_2(r) = \frac{1}{\lambda^{nr} \Gamma(r)^n} \prod_{i=1}^n x_i^{r-1} \phi e^{-r(1/\gamma+1/\phi)} r^{\alpha-1} r^2.$$

Integração por Monte Carlo

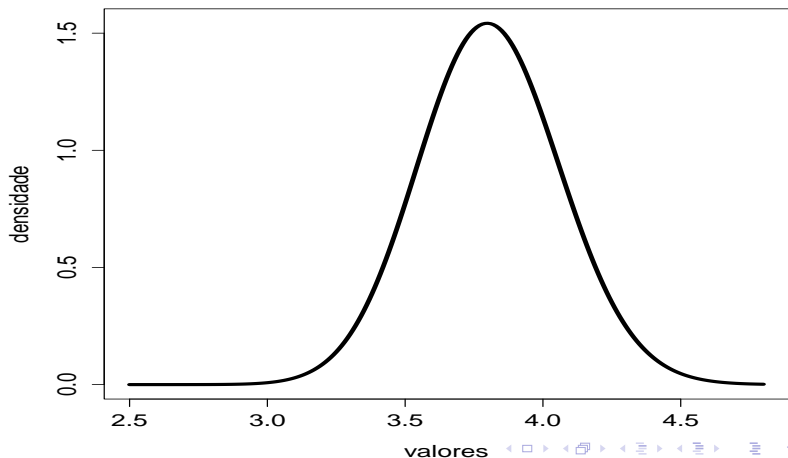
- Portanto, tais integrais podem ser aproximadas por, respectivamente

$$I_1 \approx \phi \frac{1}{m} \sum_{j=1}^m \frac{1}{\lambda^{nr_j} \Gamma(r_j)^n} \prod_{i=1}^n x_i^{r_j-1} e^{-r_j(1/\gamma+1/\phi)} r_j^{\alpha-1} r_j,$$

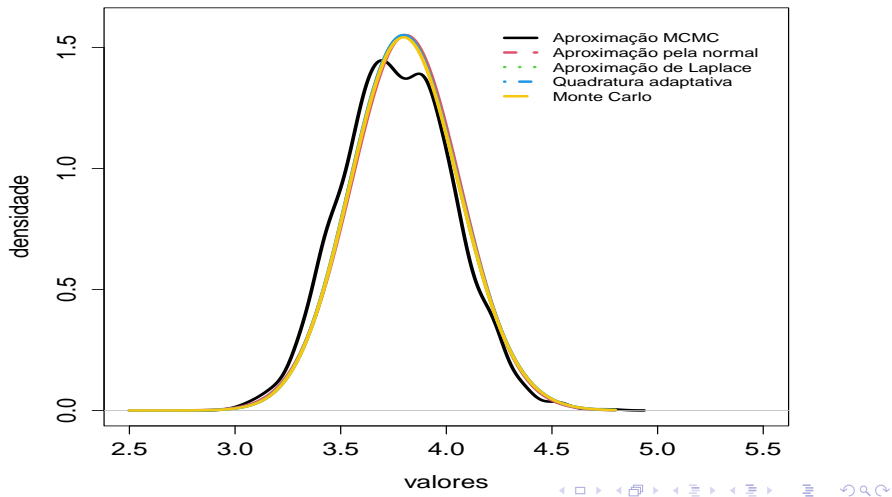
$$I_2 \approx \phi \frac{1}{m} \sum_{j=1}^m \frac{1}{\lambda^{nr_j} \Gamma(r_j)^n} \prod_{i=1}^n x_i^{r_j-1} e^{-r_j(1/\gamma+1/\phi)} r_j^{\alpha-1} r_j^2.$$

Integração por Monte Carlo : posteriori, EAP e DPAP

- $\tilde{r}_{EAP} \approx 3,80$, $EPAP(r) \approx 0,26$.



Comparação entre as aproximações da posteriori



Comparação entre o modelo gama ajustado e os dados (estimativas via quadratura adaptativa)

