Avaliação

Estatística Computacional

Paulo Ricardo Seganfredo Campana

18 de setembro de 2023

Questão 1. Responda as seguintes questões:

a) Explique, de modo geral, qual a importância de processos de geração de números pseudo-aleatórios e dê exemplos de aplicações.

De modo geral, são usadas para imitar o comportamento não determinístico das variáveis aleatórias para simulações estatísticas, como no Bootstrap onde é selecionada amostras aleatórias do conjunto de dados, ou na integração de monte carlo, onde é preciso números aleatórios dentro de um intervalo.

b) Defina matematicamente os geradores congruenciais linear e misto. Explique o funcionamento desses geradores.

São geradores da forma $x_{n+1} = ax_n \mod m$ e $x_{n+1} = (ax_n + b) \mod m$ respectivamente, o comportamento do próximo número na sequência é baseado no anterior seguindo uma relação linear aplicada no resto da divisão por algum número. Quando não há elemento anterior na sequência, se usa uma semente, a qualidade destes geradores dependem muito da escolha das constantes a, b e m.

c) Defina o método da transformação inversa para geração de números pseudo-aleatórios.

Baseia-se em utilizar a função quantílica da distribuição de interesse, obtida pela inversa da função de distribuição acumulada, e aplicar um número pseudo-aleatório de distribuição uniforme $[0, 1]$ nesta função, o resultado será um número pseudo-aleatório da distribuição de interesse pois:

$$F(x) : S_X \longrightarrow [0, 1]$$
$$F^{-1}(x) : [0, 1] \longrightarrow S_X$$

d) Usando o método da transformação inversa, obtenha uma expressão para o gerador de números pseudo-aleatórios para uma variável aleatória com função de distribuição $F(x) = x^\theta,\ 0 < x < 1,\ \theta > 0$.

Obtendo a função quantílica da distribuição temos que $F^{-1}(x) = \exp\left\{\frac{1}{\theta}\log x\right\}$:

$$\begin{aligned} F(x) &= x^\theta \\ \log F(x) &= \theta \log x \\ \frac{1}{\theta}\log F(x) &= \log x \\ \exp\left\{\frac{1}{\theta}\log F(x)\right\} &= x \end{aligned}$$

Então o algorítimo de geração para esta distribuição será gerar números pseudo-aleatórios da distribuição uniforme em $[0, 1]$ e aplicar-los na função quantílica.

```
rnovo <- function(n, theta) {
    U <- runif(n, 0, 1)
    exp(log(U) / theta)
}

rnovo(48, 0.94)
##  [1] 0.603671433 0.983589303 0.448283962 0.359813395 0.217896925 0.079708041
##  [7] 0.757985435 0.133185313 0.492266196 0.325387698 0.413817395 0.134290590
## [13] 0.857682520 0.942970536 0.191400894 0.751755529 0.007257753 0.648117294
## [19] 0.701182034 0.197298837 0.806990639 0.577142342 0.336734730 0.272972164
## [25] 0.949344649 0.429985475 0.052932252 0.371343923 0.819537577 0.096902712
## [31] 0.730396660 0.343885197 0.224105193 0.836397266 0.238420959 0.397923293
## [37] 0.157223034 0.559000344 0.774844894 0.529628988 0.732741808 0.656296956
## [43] 0.759721390 0.437783486 0.397007577 0.791654067 0.570314300 0.094684322
```

e) Explique o método da aceitação e rejeição para geração de números pseudo-aleatórios.

Com o objetivo de gerar números de uma distribuição X com densidade $f(x)$, gere números de uma distribuição conhecida Y com densidade $g(y)$, aceitamos esse número gerado Y como número pseudo aleatório da distribuição X dependendo de uma constante e o valor de uma V.A. uniforme:

$$\text{Se } \frac{f(Y)}{cg(Y)} > U, \text{ tome } Y \text{ como número gerado da distribuição } X$$

f) Use o método da aceitação e rejeição para obter um método de geração de números pseudo-aleatórios de uma variável aleatória com função densidade de probabilidade $f(x) = 20x(1-x)^3,\ 0 < x < 1$.

Como a V.A. de interesse é contínua e com suporte $[0,1]$, utilize $Y \sim U(0,1),\ g(y) = 1$. A constante c pode ser obtida maximizando o quociente $f(y)/g(y)$

```
f <- function(x) 20 * x * (1 - x)^3
g <- function(y) 1

c <- optimise(
    \(y) f(y) / g(y),
    lower = 0, upper = 1,
    maximum = TRUE
)$objective
c
## [1] 2.109375
```

```
rnovo2 <- function(n) {
    sapply(
        1:n,
        \(i) while (TRUE) {
            y <- runif(1, 0, 1)
            u <- runif(1, 0, 1)
            if (f(y) / g(y) / c > u) return (y)
        }
    )
}

rnovo2(48)
##  [1] 0.50958226 0.21776899 0.13699996 0.18348576 0.16762616 0.17931658
##  [7] 0.70353362 0.37938274 0.11536759 0.43622864 0.51377716 0.13826144
## [13] 0.69040623 0.14065386 0.27453404 0.24359479 0.14789163 0.21012952
## [19] 0.32781230 0.07897265 0.11379914 0.17392196 0.21271503 0.23730194
## [25] 0.38814811 0.28250657 0.42523220 0.46751644 0.39157101 0.43098697
## [31] 0.16148300 0.60727174 0.26913309 0.03194147 0.86206600 0.66850151
## [37] 0.76015533 0.39510988 0.23888620 0.22941464 0.29000261 0.42862613
## [43] 0.37618968 0.12912951 0.48788452 0.24727488 0.13890782 0.25656695
```

Questão 2. Responda as seguintes questões:

a) Defina matematicamente o método gradiente para maximização de uma função $g(\theta)$: $\Theta \longrightarrow \mathbb{R}$, em que Θ é um subespaço de $\mathbb{R}^p$.

É um método iterativo de busca pelo máximo de uma função, onde, a partir de um chute inicial para θ, a próxima iteração é dada por $\theta_{t+1} = \theta_t + \lambda_t \mathbf{M}_t \mathbf{g}_t$.

λ_t pode ser optimizado em cada passo de forma a maximizar θ_{t+1}. $\mathbf{M}_t$ é uma matriz $(p \times p)$ positiva definida qualquer, podemos tomar $\mathbf{M}_t = \mathbf{I}_p$, uma matriz constante por exemplo, ou optimizar $\mathbf{M}_t$ de alguma forma.

b) Defina os métodos quasi-Newton e explique as diferenças para os métodos de Newton?

O método de Newton de maximização para o caso univariado é definido como: $x_{t+1} = x_t - \frac{f'(x_t)}{f''(x_t)}$, de maneira similar, para o caso multivariado como: $\theta_{t+1} = \theta_t - \mathbf{H}_t^{-1}\mathbf{g}_t$, onde $\mathbf{g}_t$ é o vetor de primeiras derivadas da função e $\mathbf{H}_t$ a matriz de segundas derivadas.

Os métodos de quasi-Newton funcionam de maneira similar, porém não é usado diretamente a matriz $-\mathbf{H}_t^{-1}$ e sim uma sequência de matrizes $\mathbf{M}_t$ que se aproximam assintóticamente de $-\mathbf{H}_t^{-1}$ a partir de uma matriz inicial positiva definida $\mathbf{M}_0$.

Dessa forma, não é necessário obter a matriz de segundas derivadas $\mathbf{H}_t$ e invertê-la, além de que a sequência $\mathbf{M}_t$ com o método de quasi-Newton sempre será positiva definida ao contrário da matriz $-\mathbf{H}_t^{-1}$, essa propriedade é importante para o método ser válido.

Questão 3. Escreva sobre os métodos de Monte Carlo e sua importância para a estatística.

Quando algum resultado que desejamos obter não é possivel ser expresso de maneira analítica, recorremos aos métodos numéricos, entre eles, os métodos de Monte Carlo busca transformar o problema em uma esperança de variável aleatória, no qual pode ser aproximada pela média amostral de uma amostra aleatória dessa variável.

Por exemplo, para calcular a integral definida de uma função qualquer, podemos usar a distribuição uniforme para transformar este problema em uma esperança, que pode ser aproximada pela média amostral:

$$\int_a^b g(x)dx = (b-a)E[g(X)] \approx \frac{(b-a)}{m}\sum_{i=1}^{m} g(x_i)$$

Em que $X \sim U(a, b)$.

Questão 4. Escreva sobre o método bootstrap e sua importância para a estatística. Explique as diferenças entre o bootstrap paramétrico e o não-paramétrico.

Outro método numérico para estimar parâmetros e sua variância com base em um conjunto de dados. Consiste em gerar diversas amostras (réplicas bootstrap) e estimar pontualmente o parâmetro para cada amostra, assim podemos obter a média e a variância para o parâmetro a partir da distribuição desse parâmetro nas réplicas bootstrap.

Para o bootstrap paramétrico, as amostras são geradas aleatóriamente com base na distribuição amostral do conjunto de dados, enquanto que o bootstrap não-paramétrico as novas amostras são tomadas diretamente dos dados originais, com reposição.