

Integrais pelo Metodo de Monte Carlo

Francisco Rosa Dias de Miranda- N° USP 4402962

16 de Setembro de 2020

1 Descrição do EP2:

Este exercício programa foi projetado em R e teve como objetivo calcular numericamente $\int_0^1 f(t)dt$, sendo

$$f(t) = e^{(-0.2506(1+t)(0.5)(1+\cos 0.6440)t)}\gamma(t)$$

com $\gamma(t) \approx 1$. Como parâmetros deste programa, temos a precisão desejada, que foi estipulada em 3 casas depois da vírgula. A saída é impressa na tela com os valores da integral por cada método, sua respectiva precisão e o número de pontos utilizando os métodos:

- Monte Carlo Cru
- Hit-Or-Miss
- Importance Sampling
- Função quadrática como variável de controle

A seguir, descreveremos brevemente cada método:

1.1 Monte Carlo Cru

Considerado o mais simples, este método consiste em gerar números aleatórios que possuam distribuição uniforme dentro do intervalo $[a, b]$ do domínio da função $f(t)$ que queremos integrar. A estimativa da integral é dada por:

$$\int_0^1 f(t)dt \approx E[f(t)] = \frac{1}{n} \sum_{k=1}^n f(t_k)$$

1.2 Hit-Or-Miss

Para estimar a integral através do método hit-or-miss, primeiramente determinamos um retângulo de base $(b - a)$ e altura $h = 1$, pois em nosso caso $f(t) < 1, \forall t \in [0, 1]$. Geramos então N pares de números aleatórios (t_i, y_i) , onde $t_i \sim U([a, b])$, $y_i \sim U([0, h])$ e temos a integral estimada por:

$$\int_0^1 f(t)dt \approx A \frac{n}{N},$$

onde $A = (a-b)h$ é a área do retângulo que contém o gráfico de f , n é o número de pontos em que $y_i < f(t_i)$ e N é o número total de pontos gerados.

1.3 Importance Sampling

Neste método, iremos gerar números aleatórios em regiões onde $f(t)$ é maior, para aumentar a eficiência. Usamos uma distribuição $g(t)$ de tal forma que $\frac{f(t)}{g(t)} \approx C$, com C constante. Temos então um estimador para $\int f(t)dt$:

$$\int f(t)dt = \int \frac{f(t)}{g(t)}g(t)dt = \int \frac{f(t)}{g(t)}dg \approx \frac{1}{n} \sum_{k=1}^n \frac{f(x_k)}{g(x_k)},$$

com $x_k \sim g$. Neste EP utilizamos a distribuição beta com parâmetros $a = 1$ e $b = 1,5$ como função g .

1.4 Monte Carlo com função quadrática como variável de controle

Neste caso, a variável de controle será uma função $h(t)$ "próxima" de $f(t)$ e facilmente integrável analiticamente. Fazendo

$$\int_0^1 f(t)dt = \int_0^1 (f(t) + h(t) - h(t))dt = \int_0^1 h(t)dt + \int_0^1 (f(t) - h(t))dt$$

teremos $\int_0^1 (f(t) - h(t))dt \approx 0$, pois $f(t) \approx h(t)$. Como $\int_0^1 h(t)dt$ pode ser calculado de maneira exata analiticamente, teremos um estimador para a integral dado por

$$\int_0^1 f(t)dt \approx \int_0^1 g(t)dt + \frac{1}{n} \sum_{k=1}^n (f(t_k) - h(t_k))$$

No caso de $h(t) \approx f(t)$, teremos $\frac{1}{n} \sum_{k=1}^n (f(t_k) - h(t_k)) \approx 0$, o que irá diminuir a variância deste método em relação ao método de Monte Carlo Cru, tornando-o mais eficiente. No caso deste EP, foi utilizada uma função quadrática $h(t) = at^2 + bt + c$. A determinação das constantes a , b e c é discutida adiante.

2 Número de pontos e critério de parada

Neste EP, o critério de parada utilizado foi o desvio padrão da média σ_n . Para um valor ϵ desejado, o algoritmo executa o cálculo da integral, inicialmente com 1000 pontos. Se $\sigma_n > \epsilon I$, onde I é o valor estimado para $\int_0^1 f(t)dt$, acrescentamos mais 1000 pontos e repetimos o processo sucessivamente até que $\sigma_n < \epsilon I$. Desta forma obtemos um erro relativo da mesma ordem de grandeza de ϵ .

Método	n	I	ϵ
Crude	602000	0.6981666	8.998307e-04
Hit or Miss	261000	0.6991188	8.977463e-04
Importance Sampling	137000	0.6974152	8.983215e-04
Variável de Controle	2000	0.6981370	2.783546e-06

Tabela 1: Exemplo de resultados obtidos usando o programa do EP2

3 Determinação dos coeficientes a , b e c utilizados no método de Monte Carlo por Variável de Controle

Sabemos que qualquer função quadrática em R é unicamente determinada por seu valor em 3 pontos. Para encontrar uma variável de controle que se aproxime de f , precisamos de um polinômio de grau 2 que interpole f o mais próximo possível. Como queremos interpolar f no intervalo $[0, 1]$, utilizamos os pontos $1/6$, $1/2$, $5/6$, de forma a garantir que nenhum ponto do domínio fique a uma distância maior do que $1/6$ de um ponto onde a aproximação é exata. Logo, queremos obter o polinômio $ax^2 + bx + c$ que passe pelos pontos $(\frac{1}{6}, f(\frac{1}{6}))$, $(\frac{1}{2}, f(\frac{1}{2}))$ e $(\frac{5}{6}, f(\frac{5}{6}))$. Para tal, basta resolver o sistema linear a seguir para a , b e c :

$$\begin{cases} a(1/6)^2 + b(1/6) + c = f(1/6) \\ a(1/2)^2 + b(1/2) + c = f(1/2) \\ a(5/6)^2 + b(5/6) + c = f(5/6) \end{cases}$$

Ao resolvê-lo, obtemos $a \approx 0.05770457$, $b \approx -0.19983003$ e $c \approx 0.77882512$. Colocando $g(t)$ em forma de uma soma de monômios, temos

$$g(t) \approx 0.05770457t^2 - 0.19983003t + 0.77882512$$

4 Testes numéricos e discussão:

A saída do programa gerou resultados conforme esperávamos. O número de pontos foi menor conforme o grau de refinamento do método, sendo que o de Variável de Controle foi o mais rápido apesar de necessitar de bem menos amostras, pois a escolha de $g(t)$ aproximou-se de $f(t)$ em 6 casas decimais em $[a, b]$. O método mais demorado foi o Importance Sampling, possivelmente devido aos cálculos envolvendo a função beta. Um exemplo dos resultados obtidos é exibido na tabela 1.