Relatório de Pesquisa - MAP2212 - EP1

Cálculo do π através do método de Monte Carlo

Nome: Guilherme Dias Vianna (NUSP: 9301429)
Nome: Michel Csillag Finger (NUSP: 5490532)
Data de Entrega: 16/03/2024
Professor: Julio Michael Stern
Unidade: IME - USP

1 Objetivo

O objetivo deste exercício é estimar a constante π com uma precisão dada (sem conhecimento prévio do valor) utilizando um algoritmo de Monte Carlo.
O algoritmo em questão utilizado será o de gerar n pontos aleatórios pertencentes ao quadrado de lado 2:

$$\{(x_i, y_i)\}_{i=1}^{n} \in [-1, 1] \times [-1, 1]$$

E observar quantos se encontrarão dentro do círculo unitário:

$$\mathcal{C}_1 = \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leqslant 1\}$$

Pela **Lei dos Grandes Números**(GUT, 2005), é esperado que a razão do número de pontos dentro do círculo em relação ao total de pontos se aproxime da razão entre as áreas:

$$\frac{\text{Número de pontos em } \mathcal{C}_1}{\text{Número total de pontos gerados}} \xrightarrow{n \to +\infty} \frac{\pi}{4}$$

Resta agora saber qual será um bom n para que nossa estimativa seja, mesmo sem saber *a priori* o valor de π, para que a precisão de nossa estimativa seja menor que um dado ϵ.

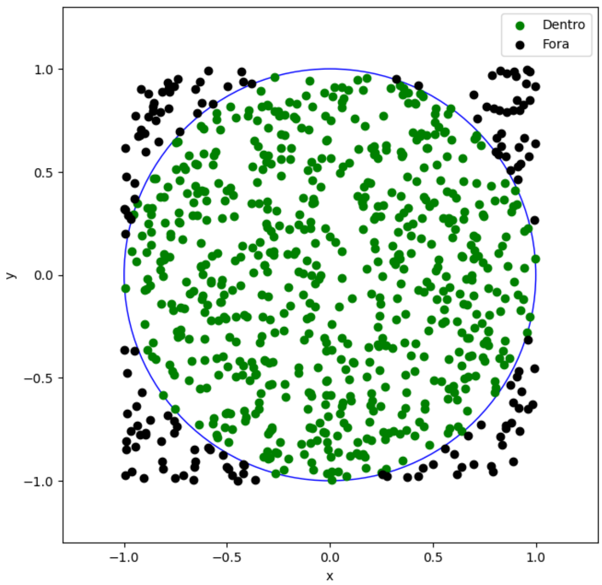

Figura 1: Ideia do experimento.

2 Preparativos

2.1 Uma primeira aproximação

Primeiramente é importante ter alguma noção de uma cota inferior e superior para π.

- Como a área do quadrado é 4, a área do círculo é π e o círculo está contido no quadrado, temos que:

$$\pi < 4$$

- Como dentro de uma circunferência temos[1] um dodecágono regular, composto de 12 triângulos regulares de área 1/4, temos então que:

$$\pi > 12 \cdot \frac{1}{4} = 3$$

  Combinado ambas temos que:

$$3 < \pi < 4$$

Isso já nos dá uma primeira aproximação e um intervalo para se manter em mente ao lançar mão de alguma outra técnica ou método de estimação.

2.2 Uma descrição estatística da simulação

O algoritmo de Monte Carlo utilizado consistirá de:

1. Gerar n valores $X_i \sim \mathcal{U}(-1, 1)$ e n valores $Y_i \sim \mathcal{U}(-1, 1)$ independentes
2. Calculamos a razão entre o número de pontos $P_i = (X_i, Y_i) \in \mathcal{C}_1$, sobre o número total:

$$\mathcal{R} = \frac{\text{Número de pontos em } \mathcal{C}_1}{n}$$

Veja que, com isso, o numerador de $\mathcal{R}$ é dado pela variável aleatória $S_n \sim \text{Bin}(n, \frac{\pi}{4})$, já que para cada ponto P_i, a probabilidade de estar no interior do círculo é dada por uma variável $p_i \sim \text{Ber}(\frac{\pi}{4})$.
Veja também que o valor esperado de S_n é:

$$\mathbb{E}[S_n] = n\frac{\pi}{4}$$

E com isso:

$$\mathbb{E}[\mathcal{R}] = \frac{1}{n}\mathbb{E}[S_n] = \frac{\pi}{4}$$

2.3 Estimativa do n necessário

Uma estimativa possível de n para obter a precisão desejada pode ser obtida pela **Desigualdade de Chebyshev** (ROUSSAS, 2005):

$$\mathbb{P}\left(|X - \mathbb{E}[X]| > \epsilon\right) \leqslant \frac{\mathbb{V}\text{ar}[X]}{\epsilon^2}$$

Tomando X como $4\mathcal{R}$, ficamos com:

$$\mathbb{P}\left(4\left|\frac{S_n - \mathbb{E}[S_n]}{n}\right| > \epsilon\right) \leqslant \frac{\mathbb{V}\text{ar}[4\frac{S_n}{n}]}{\epsilon^2}$$

Desejamos que nossa estimativa de $4\mathcal{R}$ (representada aqui por r) seja tal que $\left|\frac{r-\pi}{\pi}\right| < 0,05\%$, como a quantidade dentro do módulo será proporcional a $|r - \pi|$ nas simulações, escolhemos então $\epsilon = 0,05\pi\%$, substituindo os valores esperados e variâncias e fazendo algumas manipulações:

[1] Como o foco deste trabalho não é sobre geometria plana, apenas indicamos a demonstração desse fato em (DOLCE; POMPEO, 2013)

$$\mathbb{P}\left(\left|\frac{4\frac{S_n}{n}-\pi}{\pi}\right| > 0,05\%\right) \leqslant \frac{16}{n^2\epsilon^2}\mathbb{V}\mathrm{ar}[S_n] = \frac{4\pi(1-\frac{\pi}{4})}{n(0,005\pi)^2} = \frac{16000000}{n}\frac{1-\frac{\pi}{4}}{\pi}$$

Analisemos a seguinte função $f : \mathbb{R}_+^* \to \mathbb{R}$, dada por:

$$f(x) = \frac{1-\frac{x}{4}}{x}$$

Veja que:

$$\frac{\mathrm{d}f}{\mathrm{d}x} = -\frac{1}{x^2} < 0$$

Daonde tiramos que f é estritamente descrescente. Como sabemos que $\pi > 3$, então temos que $f(\pi) < f(3)$ Com isso podemos escrever então que:

$$\frac{16000000}{n}\frac{1-\frac{\pi}{4}}{\pi} < \frac{16000000}{n}\frac{1-\frac{3}{4}}{3} = \frac{4000000}{3n}$$

Com isso obtivemos uma cota superior para a probabilidade da nossa estimativa:

$$\mathbb{P}\left(\left|\frac{4\frac{S_n}{n}-\pi}{\pi}\right| > 0,05\%\right) \leqslant \frac{4000000}{3n}$$

Agora precisamos arbitrar o n necessário para que a probabilidade seja baixa (já que gostaríamos que a probabilidade de nossa estimativa ter um erro relativo maior que $0,05\%$ fosse baixa).
Escolhendo que a probabilidade em questão seja 5%:

$$0,05 \leqslant \frac{4000000}{3n} \Rightarrow n \geqslant 26666666,67$$

Veja que com isso, podemos então concluir que $n^* = 26666667$ é um tamanho de amostra suficiente para a precisão requisitada. Porém, como serão gerados pontos com duas coordenadas aleatórias, na realidade serão necessários $2n^* = 53333334$ números sorteados.

3 Implementação da simulação via `Python`

- Geração de Pontos Aleatórios:

  Utiliza-se a função `np.random.uniform(-1, 1, (n, 2))` para gerar `n` pontos aleatórios. Cada ponto tem coordenadas `(x, y)`, com `x` e `y` distribuídos uniformemente entre -1 e 1, assegurando que os pontos estejam dentro do quadrado de lado 2 centrado na origem.

- Determinação de Pontos Dentro da Circunferência:

  Para cada ponto, calcula-se $\texttt{x}^2 + \texttt{y}^2$ para determinar se o ponto está dentro da circunferência de raio 1. Se $\texttt{x}^2 + \texttt{y}^2 \leq 1$, o ponto está dentro da circunferência.

- Cálculo da Proporção e Estimativa de π:

  A proporção de pontos dentro da circunferência (`proporcao_dentro`) é calculada dividindo-se o número de pontos dentro pelo total de pontos `n`. Esta proporção é usada para estimar π, multiplicando-a por 4 (`pi_estimado`).

- Avaliação da Precisão da Estimativa:

  A precisão da estimativa de π é avaliada calculando-se a diferença percentual entre `pi_estimado` e o valor real de π, usando `100*np.abs(pi_estimado - np.pi)/np.pi`, resultando em `diferenca_para_pi`.

- Condição de Aceitação:

  O programa verifica se a diferença percentual (`diferenca_para_pi`) excede 0,05%. Se exceder, indica que a precisão da estimativa não está dentro do critério estabelecido. Caso contrário, considera-se que a estimativa possui uma precisão aceitável.

- Execução Repetida:

  O programa executa `m` vezes a função `pontos_dentro_circunferencia_e_figura(n)`, onde `m` é definido como 10000. Para cada execução, `n` é fixado em 26666667, o valor obtido pela desigualdade de Chebyshev para alcançar a precisão desejada na estimativa de π.

4 Resultados e conclusão

Dentre as 10000 execuções do programa, todas ficaram dentro da margem de tolerância permitida. As possíveis razões para isso são:

- Provavelmente se deve a ter uma majoração maior do que a necessária.
- A quantidade de pontos é muito alta e como são números pseudo-aleatórios, o efeito de aleatoriedade simulada é perdido.

Referências

DOLCE, O.; POMPEO, J. *Fundamentos de matemática elementar - Volume 9: Geometria plana*. [S.l.]: Atual Didáticos, 2013.

GUT, A. *Probability: A Graduate Course*. [S.l.]: Springer, 2005.

ROUSSAS, G. *Introduction to measure-theoretic probability*. [S.l.]: Elsevier, 2005.