

Estima o Valor da Constante π (Pi) Utilizando o Método de Monte Carlo

Este projeto implementa o **Método de Monte Carlo** para estimar o valor da constante matemática π (Pi) através de simulação computacional com visualização gráfica.

Informações da Atividade

- Autor:** Diogo da Silva Rego
- Matrícula:** 20240045381
- Disciplina:** Estatística Computacional
- Professor:** Pedro Rafael Diniz Marinho
- Data:** 19/09/2025

Objetivo

Implementar um algoritmo que utiliza o Método de Monte Carlo para aproximar o valor de π através da geração de pontos aleatórios e análise geométrica da relação entre a área de um círculo e um quadrado.

Metodologia

O método baseia-se no seguinte princípio:

- Geração de Pontos:** Gerar pontos aleatórios dentro de um quadrado de lado 2 (coordenadas de -1 a 1)
- Verificação:** Determinar se cada ponto está dentro do círculo unitário inscrito no quadrado
- Cálculo:** Usar a proporção de pontos dentro do círculo para estimar π

Fórmula Matemática

Plain Text

$$\pi \approx 4 \times (\text{pontos_dentro_do_círculo} / \text{total_de_pontos})$$

Esta fórmula deriva da relação entre as áreas:

- Área do círculo:** $\pi \times r^2$

- **Área do quadrado:** $(2r)^2$
- **Razão:** $\pi/4$

Requisitos

Dependências Python

Bash

```
pip install matplotlib numpy
```

Versão Python

- Python 3.6 ou superior

Como Executar

1. Clone o repositório:

Bash

```
git clone [URL_DO_SEU_REPOSITORIO]
cd [NOME_DO_REPOSITORIO]
```

1. Instale as dependências:

Bash

```
pip install matplotlib numpy
```

1. Execute o script:





Bash

```
python monte_carlo_pi.py
```

Características da Visualização

O script gera um gráfico com as seguintes características:

-  **Pontos Azuis:** Pontos que caem dentro do círculo unitário

-  **Pontos Vermelhos:** Pontos que caem fora do círculo
-  **Círculo Verde:** Referência do círculo unitário (raio = 1)
-  **Quadrado Preto:** Área total de simulação
-  **Título Informativo:** Mostra a estimativa de π , precisão e informações do autor

Resultados Esperados

O script executa simulações com diferentes números de pontos para demonstrar como a precisão melhora com o aumento da amostra:

Pontos	Precisão Típica
1.000	~95-98%
10.000	~98-99%
100.000	~99-99.5%
1.000.000	~99.5-99.9%

Estrutura do Projeto

Plain Text

```
.
├── monte_carlo_pi.py           # Script principal
├── README.md                  # Este arquivo
└── monte_carlo_pi_visualization.png # Gráfico gerado (após execução)
```

Funcionalidades

Função Principal: `estimar_pi_com_visualizacao()`

- Gera pontos aleatórios
- Classifica pontos (dentro/fora do círculo)
- Calcula estimativa de π
- Cria visualização gráfica

Função de Visualização: `criar_visualizacao()`

- Plota pontos coloridos por categoria
- Desenha círculo e quadrado de referência
- Adiciona informações estatísticas
- Salva gráfico em alta resolução

Função Completa: `executar_simulacao_completa()`

- Executa múltiplas simulações
- Compara precisão com diferentes amostras
- Gera relatório detalhado
- Exibe explicação metodológica



Conceitos Abordados

- Métodos de Monte Carlo
- Simulação Computacional
- Estatística Computacional
- Visualização de Dados
- Análise de Precisão
- Programação Científica em Python



Contexto Acadêmico

Este projeto foi desenvolvido como parte da disciplina de Estatística Computacional, demonstrando a aplicação prática de métodos numéricos para resolver problemas matemáticos através de simulação computacional.



Licença

Este projeto é desenvolvido para fins acadêmicos como parte do curso de Estatística Computacional.

Universidade Federal da Paraíba (UFPB) Departamento de Estatística

Introdução aos Softwares Estatísticos.