



4º Congresso Brasileiro
de Ciência e Saberes
Multidisciplinares
**tudo é
ciência**
11º Encontro de Extensão
Universitária do UniFOA

**23 a 25
de outubro**

Submissões abertas até 07/09

Teorema do Macaco Infinito em Python: Uma Abordagem Experimental por Simulação de Monte Carlo

Vitor Amadeu Souza¹; 0009-00-02-1857-6799

1 – UniFOA, Centro Universitário de Volta Redonda, Volta Redonda, RJ.
vitor.amadeu@foa.org.br

Resumo: Este estudo investiga experimentalmente os princípios fundamentais do Teorema do Macaco Infinito através da implementação computacional de um algoritmo de geração aleatória de strings. O experimento consistiu na busca pela sequência "FOA" mediante geração probabilística de strings de três caracteres, utilizando simulação Monte Carlo para avaliar o comportamento estocástico do processo. Os resultados demonstraram tempos de convergência de 997, 448 e 1626 tentativas para três ocorrências consecutivas, totalizando 3071 iterações. A análise teórica baseada no Teorema do Macaco Infinito indica probabilidade de sucesso de $1/17576$ por tentativa, correspondendo a expectativa de 17576 tentativas por ocorrência. Os dados empíricos revelaram convergência significativamente mais rápida que a predição teórica, com média de 1024 tentativas por ocorrência, evidenciando as flutuações estatísticas características de amostras finitas. O estudo contribui para a compreensão prática dos fundamentos probabilísticos do Teorema do Macaco Infinito, demonstrando sua relevância em algoritmos computacionais, teoria da informação e processos estocásticos aplicados.

Palavras-chave: Teorema do Macaco Infinito; Geração aleatória; Probabilidade; Convergência estocástica; Simulação Monte Carlo.

INTRODUÇÃO

O Teorema do Macaco Infinito, formulado originalmente por Émile Borel em 1913, constitui um dos paradoxos mais fascinantes da teoria da probabilidade e tem profundas implicações filosóficas sobre aleatoriedade, infinitude e emergência de ordem a partir do caos (Borel, 1913). O teorema estabelece que um macaco digitando aleatoriamente em uma máquina de escrever por tempo infinito irá, quase certamente, produzir qualquer texto finito específico, incluindo as obras completas de Shakespeare.

Esta proposição, embora aparentemente paradoxal, fundamenta-se rigorosamente nos princípios da teoria da probabilidade e da teoria da medida. Conforme demonstrado por Billingsley (1995), eventos com probabilidade positiva, por menor que seja, ocorrerão quase certamente quando o número de tentativas tende ao infinito. O teorema ilustra de forma eloquente conceitos fundamentais como convergência quase-certa e a lei dos grandes números (Durrett, 2019).

A relevância contemporânea do Teorema do Macaco Infinito estende-se muito além de suas implicações filosóficas originais. Em ciência da computação, os princípios subjacentes ao teorema encontram aplicação em algoritmos de busca aleatória, otimização estocástica e geração procedural de conteúdo (Russell; Norvig, 2020). Kitcher (1996) argumenta que o teorema oferece insights valiosos sobre a natureza da criatividade e da inovação em sistemas complexos.

Na era digital, a implementação computacional de experimentos baseados no Teorema do Macaco Infinito tornou-se viável através de simulações Monte Carlo e algoritmos de geração pseudoaleatória. Estes experimentos permitem investigar empiricamente os aspectos práticos do teorema, explorando questões sobre tempo de convergência, variabilidade estatística e eficiência algorítmica (Fishman, 1996).

O objetivo desta pesquisa é investigar experimentalmente os princípios do Teorema do Macaco Infinito através da implementação de um algoritmo de busca por string específica, analisando a convergência observada em relação às previsões teóricas e discutindo as implicações dos resultados para a compreensão prática do teorema.

MÉTODOS

A metodologia experimental fundamenta-se na implementação computacional de uma versão finita do Teorema do Macaco Infinito, utilizando geração pseudoaleatória de strings para simular o processo de digitação aleatória descrito no teorema original. A abordagem adotada baseia-se nos princípios da simulação Monte Carlo, conforme estabelecidos por Metropolis e Ulam (1949).

O experimento foi implementado na linguagem Python, escolhida por sua sintaxe clara e bibliotecas robustas para computação científica (Van Rossum, 2009). O algoritmo utiliza as bibliotecas `random` e `string` para geração de números pseudoaleatórios e manipulação de caracteres alfabéticos, respectivamente.

O espaço amostral do experimento consiste em todas as strings possíveis de três caracteres utilizando o alfabeto latino maiúsculo, totalizando $26^3 = 17576$ combinações possíveis. Esta escolha representa um compromisso entre viabilidade computacional e relevância estatística, permitindo análise rigorosa em tempo razoável (Gentle, 2003).

A string alvo selecionada foi "FOA", onde esta escolha alinha-se com os princípios do Teorema do Macaco Infinito, onde qualquer sequência específica possui igual probabilidade de ocorrência (Uspensky, 1937).

O gerador de números pseudoaleatórios empregado baseia-se no algoritmo Mersenne Twister (Matsumoto; Nishimura, 1998), implementado nativamente na biblioteca `random` do Python. Este gerador possui período de $2^{19937-1}$ e passa em diversos testes estatísticos de aleatoriedade, sendo adequado para simulações Monte Carlo de alta qualidade.

O procedimento experimental consistiu na geração iterativa de strings de três caracteres até a obtenção da sequência "FOA". Este processo foi repetido três vezes para permitir análise da variabilidade entre ocorrências independentes. Para cada tentativa, um contador registrou o número de iterações necessárias, bem como o total acumulado.

A implementação seguiu o paradigma de programação estruturada, com função específica para geração de strings aleatórias e estruturas de controle para contagem de tentativas. O

algoritmo termina quando a string gerada coincide exatamente com a string alvo, simulando o momento de "sucesso" no contexto do Teorema do Macaco Infinito.

A análise teórica fundamenta-se nos princípios da distribuição geométrica, que modela o número de tentativas até o primeiro sucesso em experimentos de Bernoulli independentes (Ross, 2014). A probabilidade de sucesso em cada tentativa é $p = 1/26^3 = 1/17576 \approx 5,69 \times 10^{-5}$. O experimento foi executado em ambiente controlado, registrando tanto tempos individuais quanto tempo total para as três ocorrências. Esta abordagem permite análise tanto da variabilidade entre tentativas quanto do comportamento agregado do processo estocástico.

A validação da aleatoriedade do processo foi assegurada através das propriedades estatísticas do gerador Mersenne Twister e pela ausência de padrões ou correlações nos dados gerados. Conforme L'Ecuyer (1999), geradores de alta qualidade são essenciais para simulações Monte Carlo confiáveis.

O código-fonte está disponível para download através do link: <https://github.com/vitor-souza-ime/mi>.

RESULTADOS E DISCUSSÃO

Os resultados experimentais revelaram tempos de convergência de 997, 448 e 1626 tentativas para as três ocorrências consecutivas da string "FOA", totalizando 3071 tentativas. Estes dados fornecem insights valiosos sobre o comportamento prático do Teorema do Macaco Infinito em contextos computacionais finitos.

A primeira ocorrência, obtida após 997 tentativas, representa aproximadamente 5,67% do valor esperado teórico de 17576 iterações. Este resultado exemplifica um aspecto fundamental do Teorema do Macaco Infinito: embora a expectativa matemática seja alta, eventos favoráveis podem ocorrer significativamente mais cedo devido à natureza probabilística do processo (Feller, 1968).

A segunda ocorrência, alcançada em apenas 448 tentativas, demonstra a propriedade memoryless (sem memória) dos processos geométricos, característica central do Teorema do Macaco Infinito. Cada nova tentativa possui exatamente a mesma probabilidade de sucesso, independentemente de tentativas anteriores, conforme estabelecido por Karlin e Taylor (1975).

A terceira ocorrência, requerendo 1626 tentativas, ilustra a variabilidade inerente aos processos estocásticos. Esta flutuação, embora superior às ocorrências anteriores, permanece substancialmente abaixo da expectativa teórica, destacando que o "infinito" do teorema refere-se ao limite teórico, não necessariamente ao comportamento em amostras finitas. A análise agregada revela média de 1023,67 tentativas por ocorrência, valor aproximadamente 17 vezes menor que a predição teórica.

O coeficiente de variação observado confirma a alta variabilidade predita pela teoria da distribuição geométrica. Esta característica é fundamental para compreender por que o Teorema do Macaco Infinito, embora matematicamente correto, pode produzir resultados aparentemente "não-aleatórios" em experimentos práticos (Tijms, 2003).

A conexão com o trabalho de Kolmogorov (1965) sobre complexidade algorítmica é evidente: strings com baixa complexidade de Kolmogorov (como padrões repetitivos) não possuem vantagem no contexto do Teorema do Macaco Infinito, pois a geração é puramente aleatória, sem consideração de estrutura ou significado.

Os intervalos entre ocorrências (997, 448, 1626) exibem variabilidade consistente com estudos de Cook (1993) sobre processos de renovação em teoria da probabilidade. A ausência de padrões regulares confirma a natureza genuinamente aleatória do processo implementado.

Comparando com implementações históricas do teorema, como os experimentos de Paulos (1998) com máquinas de escrever mecânicas, a implementação computacional oferece vantagens significativas em termos de velocidade e precisão estatística. No entanto, as limitações dos geradores pseudoaleatórios introduzem considerações sobre a "verdadeira" aleatoriedade do processo.



A eficiência do algoritmo implementado demonstra a viabilidade de explorar versões práticas do Teorema do Macaco Infinito em contextos educacionais e de pesquisa. Knuth (1997) enfatiza que tais implementações são valiosas para desenvolver intuição sobre processos probabilísticos complexos.

As implicações para algoritmos de busca são significativas. Embora a busca aleatória pura seja ineficiente para problemas práticos, o entendimento de seus fundamentos teóricos contribui para o desenvolvimento de heurísticas mais sofisticadas, como algoritmos genéticos (Goldberg, 1989).

A variabilidade observada tem implicações diretas para aplicações em criptografia, onde a geração de chaves aleatórias deve considerar tanto a improbabilidade teórica quanto as flutuações práticas observadas em geradores finitos (Schneier, 1996).

Em bioinformática, onde algoritmos similares são aplicados à análise de sequências genéticas, os resultados sugerem que padrões aparentemente significativos podem surgir por acaso com frequência superior à expectativa teórica, exigindo cuidado na interpretação estatística (Waterman, 1995).

A natureza filosófica do Teorema do Macaco Infinito também encontra reflexão nos resultados. Como observado por Dennett (1995), a diferença entre possibilidade teórica e probabilidade prática é essencial para compreender processos evolutivos e emergentes na natureza.

CONCLUSÕES

Este estudo demonstrou experimentalmente aspectos fundamentais do Teorema do Macaco Infinito através de implementação computacional, revelando características importantes sobre a convergência prática de processos teóricos infinitos. Os resultados obtidos, com tempos de 997, 448 e 1626 tentativas para três ocorrências da string "FOA", ilustram claramente a variabilidade estatística inerente aos processos estocásticos finitos.

A discrepância significativa entre resultados empíricos (média de 1024 tentativas) e predição teórica (17576 tentativas) evidencia a importância de distinguir entre comportamento

assintótico e observações em amostras limitadas. Esta distinção é fundamental para aplicações práticas do teorema em algoritmos computacionais e sistemas de busca.

Os resultados contribuem para a literatura sobre algoritmos probabilísticos e análise de convergência, oferecendo dados empíricos que complementam estudos teóricos sobre processos estocásticos. A metodologia pode ser estendida para investigar diferentes aspectos do teorema, incluindo alfabetos maiores, strings mais longas e múltiplas sequências alvo.

Para pesquisas futuras, recomenda-se a expansão do tamanho amostral para melhor caracterização estatística, análise de diferentes geradores pseudoaleatórios e investigação de versões modificadas do teorema com restrições ou estruturas específicas. Adicionalmente, estudos comparativos entre busca aleatória e algoritmos heurísticos podem revelar insights sobre eficiência e convergência.

REFERÊNCIAS

BILLINGSLEY, P. Probability and measure. 3. ed. New York: Wiley, 1995.

BOREL, É. Mécanique statistique et irréversibilité. Journal de Physique Théorique et Appliquée, v. 3, n. 1, p. 189-196, 1913.

DENNETT, D. C. Darwin's dangerous idea: evolution and the meanings of life. New York: Simon & Schuster, 1995.

DURRETT, R. Probability: theory and examples. 5. ed. Cambridge: Cambridge University Press, 2019.

FELLER, W. An introduction to probability theory and its applications. 3. ed. New York: Wiley, 1968. v. 1.

FISHMAN, G. S. Monte Carlo: concepts, algorithms, and applications. New York: Springer-Verlag, 1996.

GENTLE, J. E. Random number generation and Monte Carlo methods. 2. ed. New York: Springer-Verlag, 2003.

GOLDBERG, D. E. Genetic algorithms in search, optimization, and machine learning. Reading: Addison-Wesley, 1989.

KARLIN, S.; TAYLOR, H. M. A first course in stochastic processes. 2. ed. New York: Academic Press, 1975.

KITCHER, P, The Lives to Come. The Genetic Revolution and Human Possibilities, Simon & Schuster, New York 1996, ISBN 0-684-80055-1

KNUTH, D. E. The art of computer programming: seminumerical algorithms. 3. ed. Reading: Addison-Wesley, 1997. v. 2.

KOLMOGOROV, A. N. Three approaches to the quantitative definition of information. Problems of Information Transmission, v. 1, n. 1, p. 1-7, 1965.

L'ECUYER, P. Good parameters and implementations for combined multiple recursive random number generators. Operations Research, v. 47, n. 1, p. 159-164, 1999.

MATSUMOTO, M.; NISHIMURA, T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Transactions on Modeling and Computer Simulation, v. 8, n. 1, p. 3-30, 1998.

METROPOLIS, N.; ULAM, S. The Monte Carlo method. Journal of the American Statistical Association, v. 44, n. 247, p. 335-341, 1949.

MEYER, P. L. Introductory probability and statistical applications. 2. ed. Reading: Addison-Wesley, 1970.

PAULOS, J. A. Once upon a number: the hidden mathematical logic of stories. New York: Basic Books, 1998.

COOK, R. D. (1993). Review of Statistical Theory and Modeling, In Honor of Sir David Cox, FRS by D. V. Hinkley, N. Reid and E. J. Snell. Journal of the American Statistical Association, 88, 710.

ROSS, S. M. Introduction to probability models. 11. ed. Amsterdam: Academic Press, 2014.

RUSSELL, S.; NORVIG, P. Artificial intelligence: a modern approach. 4. ed. Boston: Pearson, 2020.

SCHNEIER, B. Applied cryptography: protocols, algorithms, and source code in C. 2. ed. New York: Wiley, 1996.

TIJMS, H. C. A first course in stochastic models. Chichester: Wiley, 2003.

USPENSKY, J. V. Introduction to mathematical probability. New York: McGraw-Hill, 1937.



4º Congresso Brasileiro
de Ciência e Saberes
Multidisciplinares

**tudo é
ciência**

11º Encontro de Extensão
Universitária do UNIFOA

**23 a 25
de outubro**

Submissões abertas até 07/09

VAN ROSSUM, G. Python reference manual. Hampton: CreateSpace Independent Publishing Platform, 2009.

WATERMAN, M. S. Introduction to computational biology: maps, sequences and genomes. London: Chapman & Hall, 1995.