# ERRO AMOSTRAL PYTHON

Para calcular o erro amostral em Python, você pode usar a biblioteca estatística "statistics". Ela fornece

uma função chamada "stdev()" ou, a biblioteca “numpy” que pode ser usadas para calcular a desvio padrão da amostra.

## Exemplo:1

“**ddof=1” -> significa "Delta Degrees of Freedom" e é usado quando calculamos o desvio padrão em Python usando a função "numpy.std()".**

**O DDOF é usado para ajustar o desvio padrão para refletir a precisão do erro amostral. O valor padrão de DDOF é 0, o que calcula o desvio padrão populacional. No entanto, quando DDOF é definido como 1, o desvio padrão é calculado como se fosse amostral, ou seja, como se estivesse sendo calculado a partir de uma amostra da população. Isso é feito ajustando o divisor usado no cálculo do desvio padrão de (n) para (n-ddof)**

**Exemplo de como calcular o desvio padrão amostral com dddof=1:**

**import numpy as np**

**sample = [1, 2, 3, 4, 5]**

**std\_amostral = np.std(sample, ddof=1)**

**print(std\_amostral)**

**Neste exemplo, o desvio padrão será calculado com ddof=1, ou seja, considerando que a amostra é uma amostra da população, e não a população inteira.**

**O número 1.96 é usado para calcular o intervalo de confiança para a média amostral. O intervalo de confiança é uma faixa de valores estimados que, com uma determinada probabilidade (geralmente 95%), incluirá a verdadeira média da população.**

**A razão pela qual 1.96 é usado é devido ao Teorema do Limite Central, que afirma que a distribuição das médias amostrais se aproxima de uma distribuição normal, à medida que o tamanho da amostra aumenta. Com base nessa distribuição normal, sabemos que 95% das médias amostrais estarão dentro de 1.96 desvios padrões da média da população.**

***Então, se o erro amostral é dado por 1.96 vezes o desvio padrão da amostra dividido pelo raiz quadrada do tamanho da amostra, então podemos ter certeza de que a média amostral estará dentro de um intervalo de confiança de 95%.***

import statistics

**#De forma direta com “.stdev”**

populacao = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

amostra = [2, 3, 4, 5, 6]

desvio\_padrao\_amostra = stdev(amostra)

desvio\_padrao\_populacao = stdev(populacao)

## Exemplo:2

**#Com Numpy detalhado**

import numpy as np

**# Dados amostrais**

sample\_data = np.array([1, 2, 3, 4, 5])

**# Cálculo da média amostral**

sample\_mean = np.mean(sample\_data)

**# Cálculo do desvio padrão amostral**

sample\_std = np.std(sample\_data, ddof=1)**#ddof=1**

**# Cálculo do erro amostral amplo**

sample\_margin\_of\_error = 1.96 \* (sample\_std / np.sqrt(len(sample\_data**))) #####A razão do 1.96**

print("Média amostral: ", sample\_mean)

print("Desvio padrão amostral: ", sample\_std)

print("Erro amostral amplo: +/-", sample\_margin\_of\_error)

***O erro amostral é uma medida da diferença entre a medida estatística (como a média) de uma amostra e a verdadeira medida estatística da população. A fórmula para o erro amostral é dada por:***

***Erro amostral = Média da população / Média amostral***

***Note que o erro amostral é uma medida aleatória, pois depende da amostra selecionada. Portanto, é comum expressar o erro amostral em termos de sua variância ou desvio padrão.***

**ChatGPT:**

**Este exemplo assume que a população inteira é conhecida e que a amostra é uma amostra aleatória da população. O erro amostral é então calculado como o desvio padrão da população dividido pelo desvio padrão da amostra.**

**Obs: É importante lembrar que esse é apenas um exemplo ilustrativo, e que a realidade pode ser diferente dependendo da sua amostra e população.**