

Problemas em equipe 07

Estudantes: Eduardo Eiji Goto, Gustavo Hammerschmidt, João Vitor Andrioli de Souza.

Parte 1 – Distribuição gamma

- 1) Um grupo de mini-VANs está esperando passageiros no aeroporto. O tempo entre duas chegadas consecutivas de passageiros é distribuído exponencialmente com média de 3 passageiros por minuto. Uma mini-VAN parte assim que tem 4 passageiros ou assim que se passe 10 minutos desde que o primeiro passageiro entrou na mini-VAN. Suponha que você foi o primeiro passageiro a entrar na mini-VAN. Qual é a probabilidade que você tenha que esperar 10 minutos até a saída da mini-VAN?

```
import scipy.stats as st

print(1-st.gamma.cdf(10, a=3, scale=3))
```

Output:

0.35277615643394045

$P[X > 10] = 0.35277 = 35.27\%$

Parte 2 – Simulação das propriedades da distribuição exponencial

Implementação da função pExpMenor

As propriedades da distribuição exponencial podem ser verificadas por simulação. O arquivo `propriedadesExp.py` contém a implementação da função `pExpMenor`, que simula essa propriedade e retorna os valores teóricos e simulados.

A função recebe como parâmetro os valores das médias das duas variáveis aleatórias exponenciais, sorteia as exponenciais em arrays de tamanho `nSim`, contando quantas vezes a exponencial `x1` foi menor do que `x2`, e retorna a proporção de vezes que isso ocorreu. A probabilidade teórica (`probT`) é calculada pela equação deduzida em sala de aula.

Execute a função diversas vezes para observar que os valores simulados ficam bem próximos do valor teórico, quando `nSim` é grande (maior do que 10.000).

Implementação da função somaExpCDFI

O arquivo `propriedadesExp.py` contém a implementação da função `somaExpCDFI`, que calcula, com um algoritmo iterativo, a probabilidade de a soma de `N` variáveis aleatórias exponenciais ser menor que certo valor `x`.

Recebe como argumento o valor de `x`, a média das exponenciais que serão somadas, a quantidade (`N`) de exponenciais somadas, e a quantidade de simulações (`nSim`).

- Iniciar a variável *deuCerto* com zero.
- Sortear o array *EXP* com *N* variáveis aleatórias com média *MU*.
- Calcular o array *soma* contendo o somatório de *EXP*, ou seja, a soma de *N* variáveis aleatórias exponenciais com média *MU*.
- Incrementar *deuCerto* se a soma for menor do que o valor *x* para o qual queremos calcular a CDF (passado com argumento).
- Retornar *deuCerto* dividido por *nSim*.

Tarefa:

Programar a função `somaExpCDFV` usando apenas vetores e comparar o desempenho das duas funções.

Copie o seu código aqui.

```
def somaExpCDFV(x, N, MU, nSim):  
  
    # sorteia matrizes EXP  
    matrix = np.random.exponential(MU, [nSim, N])  
  
    # soma as linhas  
    lines = np.sum(matrix, axis=1)
```

```
# quais linhas tem soma menor do que x
less = lines < x

return np.count_nonzero(less) / nSim
```