

SME0602 - Cálculo Numérico

Roberto F. Ausas

rfausas@icmc.usp.br

www.lmacc.icmc.usp.br/~ausas/

Lista sobre programação em python: Exercícios para entregar.

- Antes de começar a realizar o tarefa se recomenda estudar com as Jupyter Notebook apresentadas pelo professor.
- A tarefa e o relatório serão feitos em grupo (máximo 3 integrantes).
- O relatório será feito na própria Jupyter Notebook desenvolvida com algumas explicações e os resultados obtidos ao rodar.
- **NÃO ENTREGAR ARQUIVOS .zip OU QUALQUER OUTRO FORMATO QUE NÃO SEJA O DA JUPYTER NOTEBOOK .ipynb, POIS SERÃO DESCONSIDERADOS.**
- **Todos os exercícios devem estar no mesmo arquivo e as células devem ter sido executadas para que o professor possa ver os resultados.**
- Cada aluno deverá colocar o relatório no escaninho.
- Na jupyter notebook deverá constar o nome de todos os participantes.
- A data de entrega será até às 6am do dia 03/04/2023 no escaninho do Tidia.

Exo. A. Fazer em python uma função que:

- (a) Pega dois vetores randômicos \mathbf{a} e \mathbf{b} de dimensão n , e dois escalares randômicos α e β e calcula um vetor \mathbf{c} tal que

$$\mathbf{c} = \alpha \mathbf{a} + \beta \mathbf{b}$$

- (b) Pega uma matriz randômica \mathbf{A} de $n \times n$ e calcula a sua m -essima potência

$$\mathbf{A}^m = \underbrace{\mathbf{A} \dots \mathbf{A}}_{m \text{ vezes}}$$

(tomar valores de $m = 2, 3, 4$).

Em todos os casos medir o tempo necessário para realizar as operações para diferentes dimensões n . Plotar o tempo de cálculo como função da dimensão n usando a escala linear padrão e a escala `loglog`. No segundo ponto, colocar no mesmo gráfico os resultados para os diferentes valores de m . Tirar conclusões.

Exo. B. Considerar uma função escalar f e o seu gradiente:

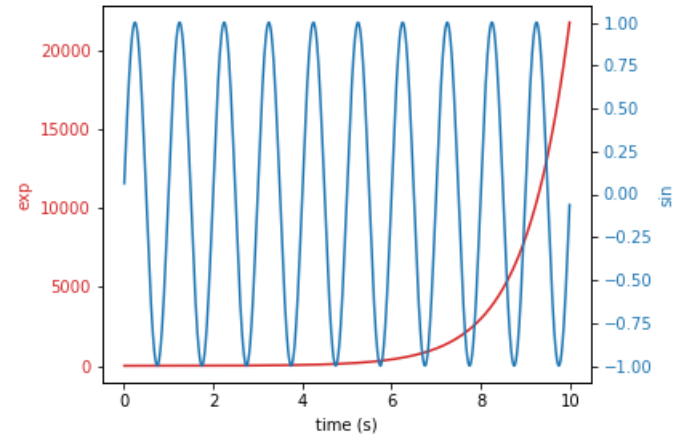
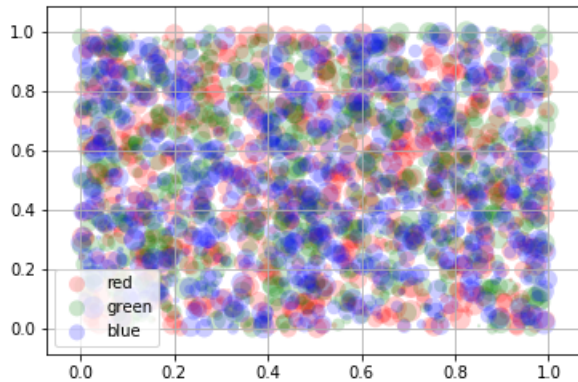
$$f(x, y) = 3 \cos(x) \cos(y), \quad \vec{v}(x, y) = \nabla f(x, y) = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]^T$$

Usando a biblioteca `matplotlib`, plotar a função f como superfície, as curvas de nível usando a função `plt.contour` e o campo de vetores \vec{v} usando a função `plt.quiver`.

- Pesquisar o comportamento dessas funções e gerar vários gráficos para ilustrar o comportamento da função numa região do plano de interesse.

- Graficar separadamente
- Grafica tudo no mesmo gráfico.

Exo. C. Utilizando a biblioteca `matplotlib` fazer os seguintes plots:



- No caso da esquerda, trata-se de pontos distribuídos aleatoriamente na região $[0, 1] \times [0, 1]$, feitos com 3 cores diferentes. Usar a função `scatter`.
- No caso da direita, trata-se dos gráficos da função $\sin(2\pi t)$ e $\exp(t)$ no intervalo $[0, 10]$. Notar que a escala da direita se aplica à primeira função e a escala da esquerda à segunda função.

Ajustar os parâmetros necessários para tentar que o resultado se pareça o mais possível com as imagens.

Exo. D. Considerar a sequência de valores gerados pelo mapeo logístico.

$$x_n = a x_{n-1} (1 - x_{n-1})$$

Considerar $x_0 = 0.1$ e diferentes valores de a entre 0 e 4 (p.e., $a = 1, 2, 3.8$ e 4).

- Para cada caso, gerar a sequência de valores indo desde $n = 1$ até $n = 10000$.
- Plotar a sequência de valores usando `matplotlib`. Plotar numa janela de valores interessante para visualizar o comportamento em cada caso.
- Gerar um histograma de valores usando a função `hist`.
- Calcular a média, a variância da sequência

$$\bar{x} = \frac{1}{N} \sum_{i=0}^N x_i, \quad \sigma = \frac{1}{N-1} \sum_{i=0}^N (x_i - \bar{x})^2$$

Pode programar-lo na mão ou usar uma função de `numpy` já pronta.

- Gerar o gráfico de bifurcações, que seria um gráfico com os valores que toma a sequência como função do parâmetro a . O resultado deveria ser algo do tipo:

