Exercício de familiarização - Python

O objetivo desse exercício é familiariza-se com a linguagem de programação Python. Alguns dos recursos que serão utilizados nesse exercício são a leitura de arquivos, exibição de dados em gráficos, operações utilizando vetores e matrizes e também a utilização de bibliotecas do Python para resolver sistema lineares.

Na diretório **slab** há uma série de arquivos de texto do modelo slab1.0 (Hayes et al., 2012) que é um modelo de geometria de zonas de subducção global. Os arquivos são um recorte do modelo completo para a região da América do Sul.

Os arquivos foram nomeados de acordo com a latitude, sendo o arquivo **slab.-1400**, por exemplo, o arquivo que contém os dados da latitude 14ºS. Abaixo está um exemplo dos dados contidos no arquivo **slab.-1400**. A primeira coluna representa a longitude (de 0º até 360º); a segunda coluna representa uma distância, em km, da longitude inicial de 270º; e a terceira coluna representa a profundidade, em km, da superfície da placa em subducção, no caso, a placa de Nazca.

```
282.44     1380.84     -4.70029
282.46     1383.06     -4.80106
282.48     1385.28     -4.90462
282.5      1387.5      -5.01151
282.52     1389.72     -5.12226
   ⋮          ⋮           ⋮
292.8      2530.8      -633.907
292.82     2533.02     -634.962
292.84     2535.24     -635.992
```

Às vezes é útil ter um domínio contínuo (não discretizado) para resolver um problema. O objetivo dessa prática é ler um dos arquivos de texto do modelo **slab1.0**, de qualquer uma das latitudes, e utilizar a biblioteca **numpy** para calcular um polinômio de grau k (equação 1) que melhor se ajuste aos dados e construir um gráfico de longitude por profundidade que vai conter os N pontos do modelo **slab1.0** e a curva melhor ajustada.

$$y = a_k x^k + \cdots + a_2 x^2 + a_1 x + a_0 \tag{1}$$

Utilizaremos o *solver* **numpy.linalg.solv()** para resolver o sistema $Ma = b$ e, dessa forma, calcular o polinômio de grau k que melhor se ajusta aos dados (M é uma matriz de tamanho $(k + 1) \times (k + 1)$ e b é o vetor de tamanho $(k + 1)$). Abaixo é mostrado como construir ambos M e b. Note que o máximo valor k do grau do polinômio deve ser $N - 1$.

$$M = \begin{bmatrix} \sum_{i=1}^{N} x_i^0 & \sum_{i=1}^{N} x_i^1 & \cdots & \sum_{i=1}^{N} x_i^k \\ \sum_{i=1}^{N} x_i^1 & \sum_{i=1}^{N} x_i^2 & \cdots & \sum_{i=1}^{N} x_i^{k+1} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=1}^{N} x_i^k & \sum_{i=1}^{N} x_i^{k+1} & \cdots & \sum_{i=1}^{N} x_i^{2k} \end{bmatrix} \text{ e } b = \begin{bmatrix} \sum_{i=1}^{N} x_i^0 y_i \\ \sum_{i=1}^{N} x_i^1 y_i \\ \vdots \\ \sum_{i=1}^{N} x_i^k y_i \end{bmatrix}$$

Referências

Hayes, Gavin P., David J. Wald, and Rebecca L. Johnson. "Slab1. 0: A three-dimensional model of global subduction zone geometries." *Journal of Geophysical Research: Solid Earth* 117.B1 (2012).