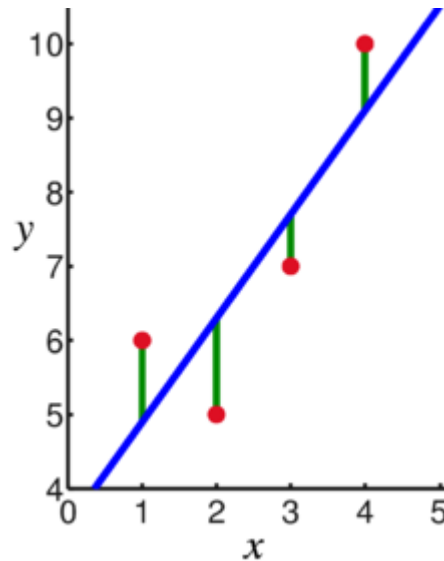


Laborator 2

Exercitii ipywidgets:

- Generati o lista de 20 de perechi de valori $\{x_i, y_i\}_{i=0,19}$ in intervalul $[0, 1]$, afisati aceste valori pe un grafic, impreuna cu o dreapta definita de o functie liniara $y = a \cdot x + b$. Intr-un alt plot afisati, ca histograma, distanta dintre un punct de coordonate (x_i, y_i) si punctul de intersectie a verticalei duse prin x_i cu dreapta data. Dreapta trebuie sa fie controlabila din widgets, prin cei doi coeficienti. Constatati modificarea histogramei in functie de pozitia dreptei si calculati suma: $\sum_{i=0}^{19} (y_i - (a \cdot x_i + b))^2$, adica suma patratelor lungimilor segmentelor verzi de mai jos.



Indicatii:

- Pentru generare de valori distribuite uniform in intervalul $[0, 1]$ puteti folosi functia `numpy.random.uniform` (<https://docs.scipy.org/doc/numpy/reference/generated/numpy.random.uniform.html>).
 - Puteti opta sa returnati cele 20 de puncte sub forma `vector_x`, `vector_y`.
- (Optional) Incarcati fisierul `yacht_hydrodynamics.csv` (adaptare dupa [Carbon Nanotubes Data Set](http://archive.ics.uci.edu/ml/datasets/Carbon+Nanotubes) (<http://archive.ics.uci.edu/ml/datasets/Carbon+Nanotubes>)). In functie de alegerile exprimate de un utilizator, afisati intr-un grafic 2D coloanele numerice alese (de exemplu, coloana 0 si coloana 2).

Indicatii/optiuni:

- Incarcarea de date se poate face cu numpy, functia `loadtxt` (<https://docs.scipy.org/doc/numpy/reference/generated/numpy.loadtxt.html>). Specificati faptul ca se sare peste prima linie din fisier (header).
- Numarul de coloane din setul de date incarat este 8 si se poate afla cu `data.shape[1]`.
- Puteti descarca fisierul direct de pe net, de exemplu cu functia `urlopen` din pachetul `urllib.request` sau cu pachetul `pandas` (https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_csv.html).
- Pentru cele doua alegeri puteti sa instantiati doua obiecte `Dropdown` (<https://ipywidgets.readthedocs.io/en/stable/examples/Widget%20List.html#Dropdown>) sau `Select` (<https://ipywidgets.readthedocs.io/en/stable/examples/Widget%20List.html#Select>).
- Daca in variabila `data` sunt datele incarcate, selectarea coloanei `i` se face cu `data[:, i]`.

Ciclari, siruri de caractere

Se recomanda ca urmatoarele exercitii sa le lucrati in Jupyter notebook/lab. Incercati parametrizarea functiilor cu ipywidgets.

1. (fizz-buzz test) Sa se scrie numerele de la 1 la n ; pentru fiecare multiplu de 3 se va scrie in locul numarului 'Fizz', pentru multiplu al lui 5 se va scrie 'Buzz'; daca numarul este multiplu de 15 se va scrie in locul lui 'FizzBuzz'.
2. Sa se verifice conjectura Collatz (https://en.wikipedia.org/wiki/Collatz_conjecture) pentru numerele intre 1 si 1000. Optional: reprezentati intr-un scatter plot numarul de pasi necesari pentru stabilizare, precum aici (https://en.wikipedia.org/wiki/Collatz_conjecture#/media/File:Collatz-stopping-time.svg).
3. Sa se creeze o functie care preia un numar n si returneaza un alt numar pe baza cifrelor lui n , astfel: se calculeaza numarul cifrelor pare din n (posibil sa fie si 0), numarul de cifre impare (poate fi si 0), suma lor si se formeaza numarul din acestea 3; daca exista vreun zero nesemnificativ, acesta se va ignora. Numarul obtinut se va supune aceleiasi transformari. Exemplu: 3->11->22->202->303->123->123->123... Verificati ca dupa un numar finit de transformari se ajunge la numarul 123; faceti aceste verificari pentru numerele din intervalul 1, 1000.
4. * Se pleaca de la un numar intreg. Fiecare cifra a sa se scrie cu litere, in limba engleza (de exemplu 5->five). Clati numarul total de caractere rezultate, iar pentru numarul obtinut repetati procedura. Verificati pentru numerele de la 1 la n ca se obtine intr-un numar finit de pasi numarul 4. Exemplu: 123->onetwothree->11->oneone->6->six->3->three->5->five->4->four->4->four....
5. Este comportamentul de mai sus valabil si pentru transcriere in limba romana, eventual pentru alt cuvnt?
6. Se pleaca de la un numar n ; se scriu toti divizorii sai, inclusiv 1 si n ; se aduna *cifrele* tuturor acestor divizori; pentru numarul obtinut se aplica acelasi procedeu. Verificati ca procesul se stabilizeaza in numarul 15. Exemplu: 20->1, 2, 4, 5, 10, 20 -> suma cifrelor: 15->1, 3, 5, 15-> suma cifrelor: 15...