

# Laboratorium 7 - Pierwiastki równań nieliniowych

Piotr Karamon

29.04.2024r.

## Treści zadań

**1. Napisz iteracje wg metody Newtona do rozwiązywania każdego z następujących równań nieliniowych:**

1.  $x \cos(x) = 1$
2.  $x^3 - 5x - 6 = 0$
3.  $e^{-x} = x^2 - 1$

**2.**

1. Pokaż, że iteracyjna metoda

$$x_{k+1} = \frac{x_{k-1}f(x_k) - x_kf(x_{k-1})}{f(x_k) - f(x_{k-1})}$$

Matematycznie jest równoważna z metodą siecznych przy rozwiązywaniu skalarne go nieliniowego równania  $f(x) = 0$

2. Jeśli realizujemy obliczenia w arytmetyce zmiennoprzecinkowej o skończonej precyzji, jakie zalety i wady ma wzór podany w podpunkcie a), w porównaniu ze wzorem dla metody siecznych podanym poniżej?

$$x_{k+1} = x_k - f(x_k) \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})}$$

**3. Zapisz iteracje Newtona do rozwiązywania następującego układu równań nieliniowych.**

$$\begin{aligned}x_1^2 + x_1x_2^3 &= 9 \\ 3x_1^2x_2 - x_2^3 &= 4\end{aligned}$$

## Zadanie 1

Będziemy korzystali z następującego wzoru

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Możemy zapisać procedurę w języku Python realizującą metodę Newtona. Procedurę będziemy kończyli po ustalonej ilości iteracji  $n$ .

```
def newton(x, next_x, n = 10):  
    for k in range(n):  
        x = next_x(x)  
    return x
```

a)  $x \cos x = 1$

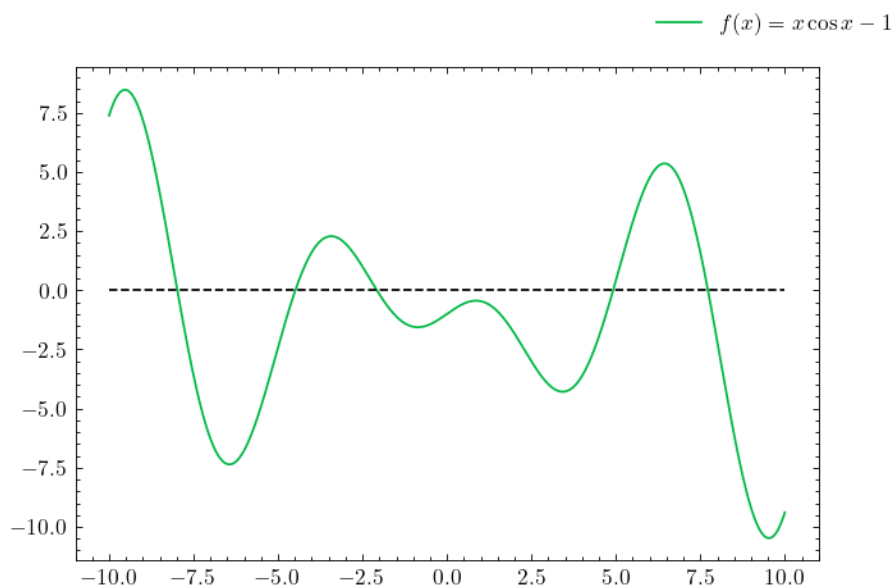
Przekształcamy:

$$f(x) = x \cos x - 1 = 0$$

$$f'(x) = \cos x - x \sin x$$

$$x_{k+1} = x_k - \frac{x_k \cos x_k - 1}{\cos x_k - x_k \sin x_k} = -\frac{x_k^2 \sin x_k - 1}{\cos x_k - x_k \sin x_k}$$

Rysujemy wykres.



**Rysunek 1:** Wykres funkcji  $f(x) = x \cos x - 1$ .

Przetestujemy procedurę na dla kilku punktów startowych nazwanych  $s_i$ .

**Tablica 1:** Wyniki dla  $x \cos x - 1 = 0$ 

$k$	$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$
0	-6.0000000000	-5.0000000000	-3.0000000000	-2.0000000000	-1.0000000000	4.0000000000
1	-3.4357659615	-4.5237936402	-1.6061669252	-2.0750450612	-6.1144173135	5.5228453452
2	-61.8761802541	-4.4881078762	-2.1811051027	-2.0739330107	-2.6230217442	4.8602948096
3	-61.1580199134	-4.4876696715	-2.0751788409	-2.0739328091	-2.0336239769	4.9175653053
4	-61.2451319509	-4.4876696033	-2.0739330620	-2.0739328091	-2.0742335171	4.9171859390
5	-61.2447280874	-4.4876696033	-2.0739328091	-2.0739328091	-2.0739328239	4.9171859253
6	-61.2447280834	-4.4876696033	-2.0739328091	-2.0739328091	-2.0739328091	4.9171859253
7	-61.2447280834	-4.4876696033	-2.0739328091	-2.0739328091	-2.0739328091	4.9171859253
8	-61.2447280834	-4.4876696033	-2.0739328091	-2.0739328091	-2.0739328091	4.9171859253
9	-61.2447280834	-4.4876696033	-2.0739328091	-2.0739328091	-2.0739328091	4.9171859253
$f(x_9)$	-0.0000000000	-0.0000000000	-0.0000000000	-0.0000000000	-0.0000000000	-0.0000000000

Widzimy szybką zbieżność metody, oraz jej efektywność. Zaskakujący jest wynik dla  $-6.0$ , ponieważ metoda znalazła miejsce zerowe bardzo oddalone od najbliższego (patrzac na wykres  $\approx -8.5$ ).

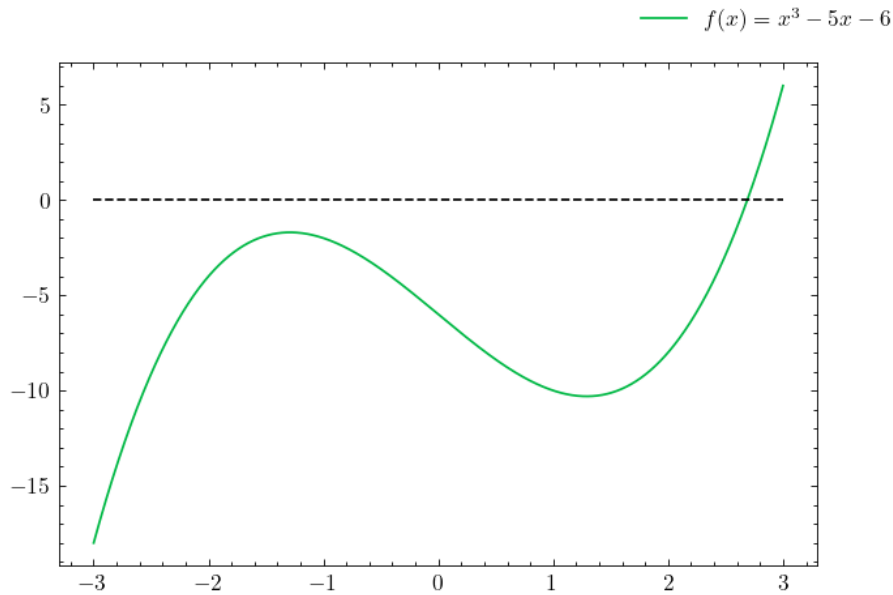
b)  $x^3 - 5x - 6 = 0$

Przekształcamy:

$$f'(x) = 3x^2 - 5$$

$$x_{k+1} = x_k - \frac{x_k^3 - 5x_k - 6}{3x_k^2 - 5} = \frac{2x_k^3 + 6}{3x_k^2 - 5}$$

Rysujemy wykres.

**Rysunek 2:** Wykres funkcji  $f(x) = x^3 - 5x - 6$ .

Przetestujemy procedurę na dla kilku punktów startowych nazwanych  $s_i$ .

**Tablica 2:** Wyniki dla  $x^3 - 5x - 6 = 0$

$k$	$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$
0	-1.0000000000	0.0000000000	1.0000000000	2.5000000000	3.2500000000	4.0000000000
1	-2.0000000000	-1.2000000000	-4.0000000000	2.7090909091	2.7974238876	3.1162790698
2	-1.4285714286	-3.7411764706	-2.8372093023	2.6892858026	2.6943566798	2.7565563488
3	0.1506493506	-2.6690494517	-2.0720209708	2.6890953412	2.6891086506	2.6911929055
4	-1.2179526353	-1.9563101329	-1.4964181414	2.6890953236	2.6890953237	2.6890974467
5	-4.3409774479	-1.3845952388	-0.4085222751	2.6890953236	2.6890953236	2.6890953236
6	-3.0583465816	0.9199560167	-1.3032262191	2.6890953236	2.6890953236	2.6890953236
7	-2.2207887225	-3.0707116109	16.5260018265	2.6890953236	2.6890953236	2.6890953236
8	-1.6237139465	-2.2290261666	11.0923495414	2.6890953236	2.6890953236	2.6890953236
9	-0.8804988876	-1.6303879712	7.5129223823	2.6890953236	2.6890953236	2.6890953236
$f(x_9)$	-2.2801372349	-2.1819002820	380.4947994886	-0.0000000000	-0.0000000000	-0.0000000000

Procedura działa, ale tylko gdy wybierzemy “dobry” punkt startowy. Przykład pokazuje to jak ważny jest odpowiedni dobór  $x_0$ .

c)  $e^{-x} = x^2 - 1$

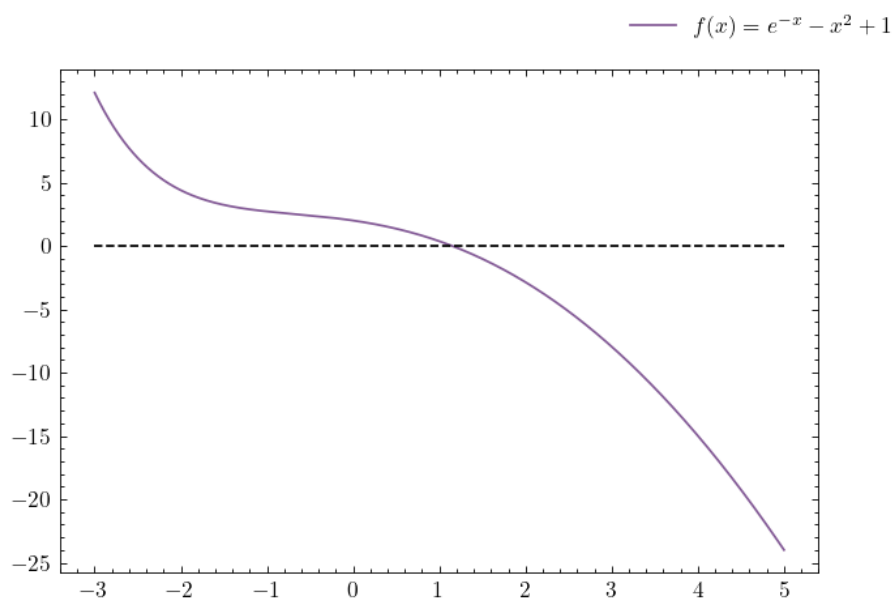
Przekształcamy:

$$f(x) = e^{-x} - x^2 + 1$$

$$f'(x) = -e^{-x} - 2x$$

$$x_{k+1} = x_k - \frac{e^{-x_k} - x_k^2 + 1}{-e^{-x_k} - 2x_k}$$

Rysujemy wykres.



**Rysunek 3:** Wykres funkcji  $f(x) = e^{-x} - x^2 + 1$ .

Przetestujemy procedurę na dla kilku punktów startowych nazwanych  $s_i$ .

$$x_{k+1} = x_k - \frac{e^{-x_k} - x_k^2 + 1}{-e^{-x_k} - 2x_k}$$

**Tablica 3:** Wyniki dla  $x^3 - 5x - 6 = 0$

$k$	$s_0$	$s_1$	$s_2$	$s_3$	$s_4$	$s_5$
0	-5.0000000000	-4.0000000000	-3.0000000000	0.0000000000	1.0000000000	5.0000000000
1	-4.1011464523	-3.1502205559	-2.1419896175	2.0000000000	1.1553624035	2.6022893594
2	-3.2470479300	-2.3041718323	-0.9775765094	1.3072714736	1.1477761807	1.5228900533
3	-2.4056946141	-1.2488535111	2.8672502670	1.1553178500	1.1477576323	1.1855247471
4	-1.4018661438	1.7114946614	1.6301877271	1.1477759644	1.1477576321	1.1482080665
5	1.0583735615	1.2262505719	1.2072983206	1.1477576323	1.1477576321	1.1477576975
6	1.1504541735	1.1496634656	1.1488648023	1.1477576321	1.1477576321	1.1477576321
7	1.1477599702	1.1477588005	1.1477580266	1.1477576321	1.1477576321	1.1477576321
8	1.1477576321	1.1477576321	1.1477576321	1.1477576321	1.1477576321	1.1477576321
9	1.1477576321	1.1477576321	1.1477576321	1.1477576321	1.1477576321	1.1477576321
$f(x_9)$	-0.0000000000	-0.0000000000	-0.0000000000	-0.0000000000	-0.0000000000	-0.0000000000

Dla wszystkich testowanych punktów startowych procedura znalazła miejsce zerowe funkcji.

## Wnioski

Metoda działa bardzo dokładnie oraz szybko, jeżeli wybierzemy dobry punkt startowy  $x_0$ . Dla niektórych funkcji wybór tego punktu jest o wiele ważniejszy i wrażliwszy niż dla innych. Również samo zachowanie metody jest trudne do dokładnego przewidzenia, ( $x_0 = -6.0$  w a). Co pokazuje, że napisanie uniwersalnej procedury szukania miejsc zerowych jest zadaniem bardzo trudnym i wymaga wykrywania “pułapek” i próbowania im odpowiednio zapobiec.

## Zadanie 2

### a) Dowód

W obu metodach

$$x_0 = a$$

$$x_1 = b$$

Wychodzimy z równania dla metody siecznych:

$$\begin{aligned} x_{k+1} &= x_k - f(x_k) \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} = \frac{x_k f(x_k) - x_k f(x_{k-1})}{f(x_k) - f(x_{k-1})} - f(x_k) \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} = \\ &= \frac{x_k f(x_k) - x_k f(x_{k-1}) - f(x_k) x_k + f(x_k) x_{k-1}}{f(x_k) - f(x_{k-1})} = \frac{x_{k-1} f(x_k) - x_k f(x_{k-1})}{f(x_k) - f(x_{k-1})} \quad \blacksquare \end{aligned}$$

Z czego wniosek, że oba wzory są matematycznie równoważne.

## b) Wady i zalety

To jest wzór  $a$

$$x_{k+1} = \frac{x_{k-1}f(x_k) - x_k f(x_{k-1})}{f(x_k) - f(x_{k-1})}$$

To jest wzór  $b$

$$x_{k+1} = x_k - f(x_k) \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})}$$

Choć są matematycznie równoważne to jednak mogą inaczej się zachowywać w przypadku arytmetyki zmiennoprzecinkowej.

Oba wzory zawierają ryzyko dzielenia przez zero, gdy wartości  $f(x_k)$  i  $f(x_{k-1})$  są bardzo bliskie. Wzór (a) wymaga dwóch mnożeń i jednego dzielenia. Wzór (b) wymaga jednego mnożenia i jednego dzielenia. Dodawań oraz odejmowań nie wliczam ponieważ są o wiele tańsze obliczeniowo. Zatem wzór (a) może być marginalnie szybszy przy bardzo czasowo-wrażliwych zastosowaniach.

Problemy zaczynają się w miarę zbieżności  $x_k$  do  $x^*$ ,  $f(x^*) = 0$ . Wtedy licznik i mianownik we wzorze (a) są bardzo małymi liczbami bliskimi zera. W wyniku dzielenia takich liczb powstają wyraźne błędy.

Wzór (b) ma bardzo przyjazną formę:

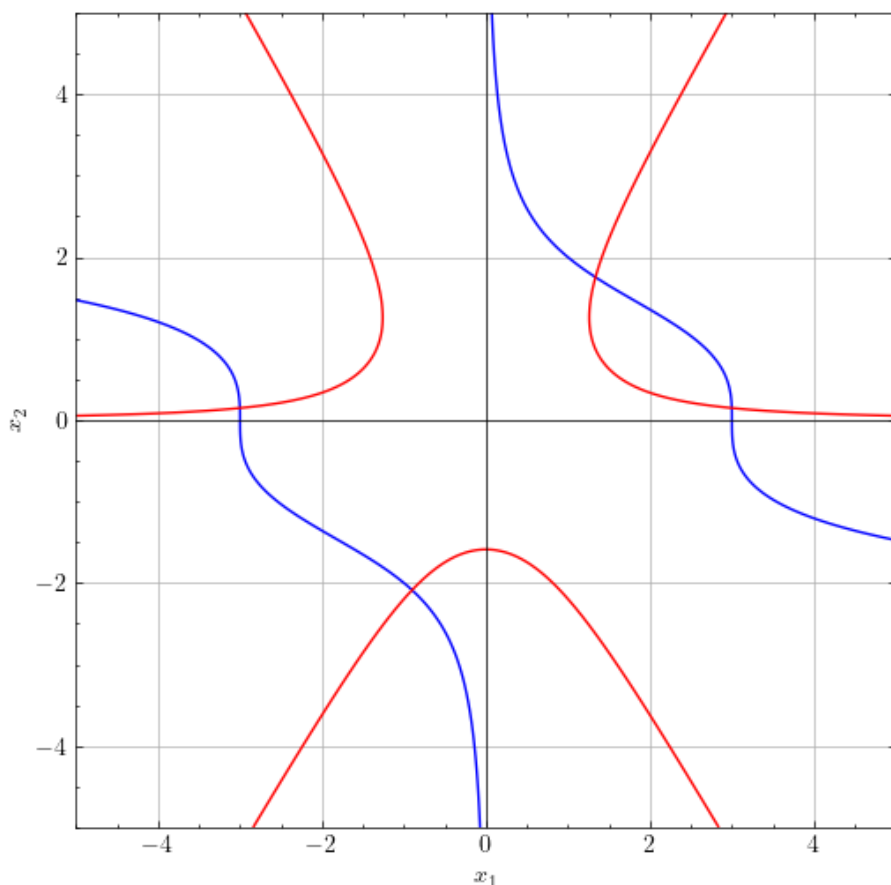
$$\text{ulepszone przybliżenie} = \text{dobre przybliżenie} + \text{mała korekcja}$$

Fakt takiego rozbicia broni nas przed nadmiernym “zepsuciem” obecnego przybliżenia.

## Zadanie 3

$$\begin{aligned}x_1^2 + x_1 x_2^3 - 9 &= 0 \\ 3x_1^2 x_2 - x_2^3 - 4 &= 0\end{aligned}$$

Każde z powyższych równań możemy narysować w układzie współrzędnych ( $x = x_1$ ,  $y = x_2$ ).



**Rysunek 4:** Niebieski :  $x_1^2 + x_1x_2^3 - 9 = 0$ , czerwony  $3x_1^2x_2 - x_2^3 - 4 = 0$ .

Niech

$$F(x_1, x_2) = (x_1^2 + x_1x_2^3 - 9, 3x_1^2x_2 - x_2^3 - 4)$$

Znalezienie rozwiązania układu jest równoważne znalezieniu takiej pary argumentów dla których  $F(x_1, x_2) = (0, 0)$  inaczej  $F(\mathbf{x}) = \mathbf{0}$ . Jak w “zwykłej” metodzie Newtona mamy iteracyjny wzór na nowe przybliżenie miejsca zerowego,

$$\mathbf{x}_{k+1} = \mathbf{x}_k - J_F(\mathbf{x}_k)^{-1}F(\mathbf{x}_k)$$

By nie obliczać odwrotności macierzy Jacobiego, przekształcamy równanie do postaci

$$J_F(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) = -F(\mathbf{x}_k)$$

Macierz Jacobiego  $J_F$ :

$$J_F(x_1, x_2) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{pmatrix}$$

gdzie pochodne cząstkowe są równe:

$$\begin{aligned}\frac{\partial f_1}{\partial x_1} &= 2x_1 + x_2^3 \\ \frac{\partial f_1}{\partial x_2} &= 3x_1x_2^2 \\ \frac{\partial f_2}{\partial x_1} &= 6x_1x_2 \\ \frac{\partial f_2}{\partial x_2} &= 3x_1^2 - 3x_2^2\end{aligned}$$

Ostatecznie  $J_F$  jest równe:

$$J_F(x_1, x_2) = \begin{pmatrix} 2x_1 + x_2^3 & 3x_1x_2^2 \\ 6x_1x_2 & 3x_1^2 - 3x_2^2 \end{pmatrix}$$

Algorytm zapiszemy w języku Python wykorzystując bibliotekę Numpy.

```
import numpy as np

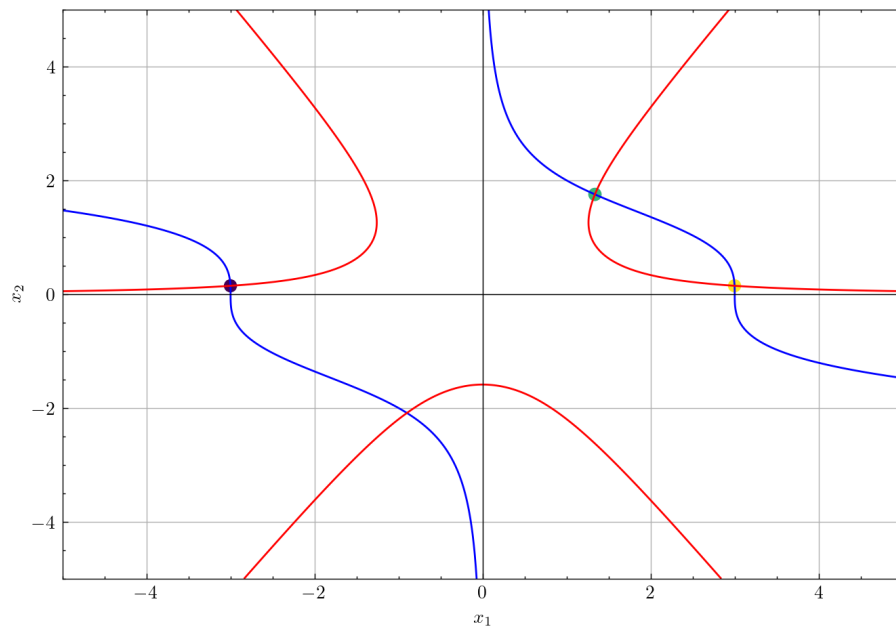
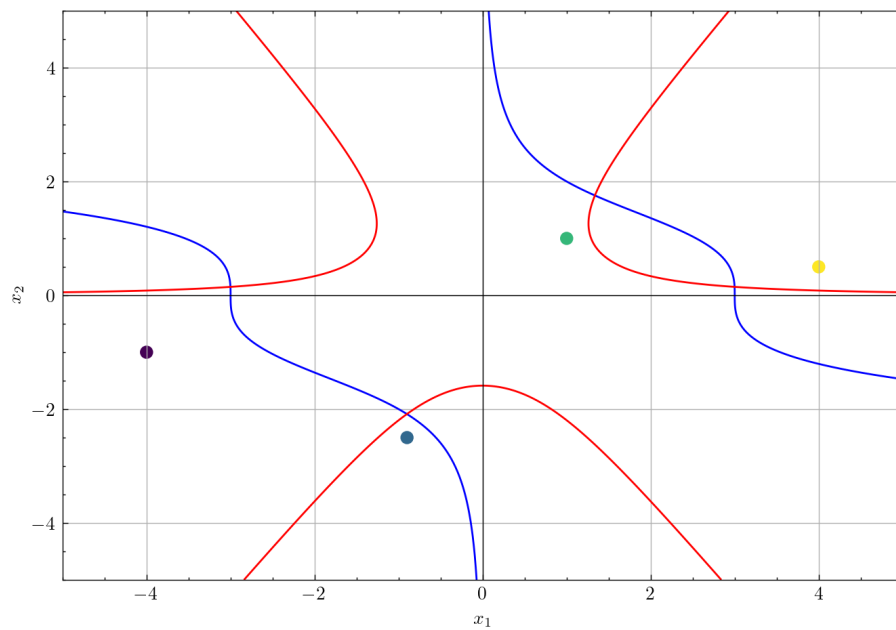
def F(X):
    x1, x2 = X
    return np.array([
        x1**2 + x1*x2**3 - 9,
        3*x1**2*x2 - x2**3 - 4
    ])

def J_F(X):
    x1, x2 = X
    return np.array([
        [2*x1+x2**3, 3*x1*x2],
        [6*x1*x2, 3*x1**2 - 3*x2**2]
    ])

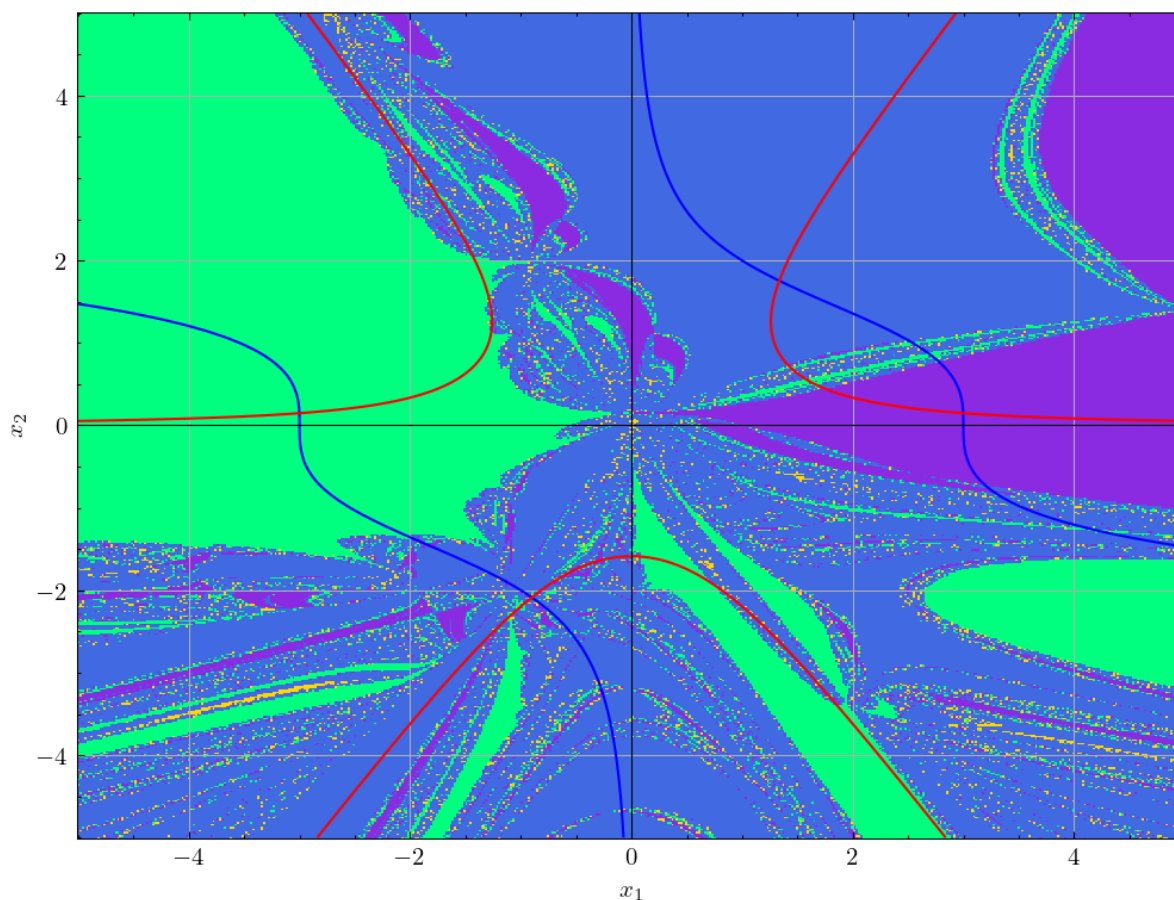
def newton(X, F, J_F, n=10):
    for i in range(n):
        X = X + np.linalg.solve(J_F(X), -F(X))
    return X
```

Sprawdzimy teraz wizualnie działanie procedury dla kilku punktów startowych. Każdy punkt ma przypisany kolor, pierwszy rysunek jest przed wykonaniem metody Newtona, drugi po.





Widzimy, że jeden punkt startowy był bardzo blisko rozwiązania układu (około  $(-1.0, -2.0)$ ), a pomimo to otrzymaliśmy w wyniku algorytmu rozwiązanie o wiele bardziej odległe od niego.



**Rysunek 5:** Rysunek powstał przez stworzenie siatki punktów, a następnie sprawdzeniu do którego punktu przecięcia doprowadzi nas algorytm. Kolor zielony - punkt przecięcia blisko  $(-3, 0)$ , złoty -  $(-1, -2)$ , niebieski -  $(1.34, 1.75)$  i fioletowy  $(3, 0)$ . Podane punkty przecięcia są oczywiście grubymi przybliżeniami.

Otrzymany rysunek ma bardzo złożony kształt. Interesująco zachowują się punkty złote. Nie widać dużych ich skupisk, co oznacza że znalezienie odpowiedniego punktu startowego w celu obliczenia punktu przecięcia w okolicy  $(-1, -2)$  jest bardzo trudne. Widzimy, że nawet dla tak prostego układu równań znalezienie odpowiedniego punktu startowego, nawet gdy mamy wykres nie jest zadaniem oczywistym.

**Tablica 4:** Wyniki dla metody Newtona dla układów równań.

$k$	$s_0$	$s_1$	$s_2$	$s_3$
0	(-4.0000000, -1.0000000)	(-0.9000000, -2.5000000)	(1.0000000, 1.0000000)	(4.0000000, 0.5000000)
1	(-2.4025974, -0.7186147)	(-0.2102147, -1.5893320)	(1.3333333, 3.0000000)	(3.2461677, 0.2708145)
2	(-2.0594436, 0.0753021)	(-2.1640971, -2.1417363)	(0.8568302, 1.7798735)	(3.0450566, 0.1599434)
3	(-3.2324542, 0.2288266)	(-1.2880849, -2.4686882)	(1.3931174, 1.6687659)	(3.0010542, 0.1485345)
4	(-2.9909127, 0.1618511)	(1.5395674, 1.4925188)	(1.3416678, 1.8075097)	(2.9983866, 0.1484292)
5	(-2.9988516, 0.1483089)	(1.2869753, 1.9463698)	(1.3268279, 1.7256473)	(2.9983650, 0.1484310)
6	(-3.0015888, 0.1481113)	(1.2816874, 1.6678167)	(1.3407466, 1.7711935)	(2.9983654, 0.1484310)
7	(-3.0016243, 0.1481081)	(1.3520121, 1.8088970)	(1.3336127, 1.7447154)	(2.9983653, 0.1484310)
8	(-3.0016249, 0.1481080)	(1.3271672, 1.7249201)	(1.3378400, 1.7597541)	(2.9983653, 0.1484310)
9	(-3.0016249, 0.1481080)	(1.3408335, 1.7716416)	(1.3354775, 1.7510936)	(2.9983653, 0.1484310)
10	(-3.0016249, 0.1481080)	(1.3335379, 1.7444692)	(1.3368492, 1.7560425)	(2.9983653, 0.1484310)
11	(-3.0016249, 0.1481080)	(1.3378779, 1.7598986)	(1.3360694, 1.7532018)	(2.9983653, 0.1484310)
12	(-3.0016249, 0.1481080)	(1.3354543, 1.7510119)	(1.3365183, 1.7548282)	(2.9983653, 0.1484310)
13	(-3.0016249, 0.1481080)	(1.3368620, 1.7560896)	(1.3362617, 1.7538956)	(2.9983653, 0.1484310)
14	(-3.0016249, 0.1481080)	(1.3360619, 1.7531749)	(1.3364090, 1.7544299)	(2.9983653, 0.1484310)
$F(x_{14})$	(0.0000000, 0.0000000)	(-0.0154393, -0.0000227)	(0.0028346, -0.0000008)	(-0.0000000, -0.0000000)

Widzimy, że metoda charakteryzuje się szybką zbieżnością.

**Wnioski:** Rozwiązywanie nieliniowych układów równań metodą Newtona jest analogiczne do przypadku z jednym równaniem. Jedynym nietrywialnym elementem algorytmu jest policzenie macierzy Jacobiego. Metoda działa z bardzo dobrą zbieżnością w przypadku dobrego punktu startowego. Sam dobór tego punktu jest już zadaniem trudnym, nawet jeżeli mamy do dyspozycji rysunek równań.

## Bibliografia

- Marian Bubak, Katarzyna Rycerz: *Metody Obliczeniowe w Nauce i Technice. Równania nieliniowe*
- Newton's Method