

Sprawozdanie laboratorium numer 1:

Predykaty Geometryczne

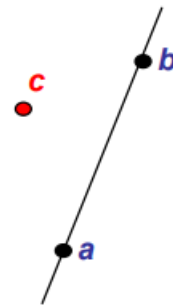
1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z podstawowymi predykatami geometrycznymi, ich implementacją oraz przetestowanie i zwizualizowanie wyników.

2. Wstęp teoretyczny:

$$\det(a, b, c) = \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix}$$

$$\det(a, b, c) = \begin{vmatrix} a_x - c_x & a_y - c_y \\ b_x - c_x & b_y - c_y \end{vmatrix}$$



Rysunek 2.1

Do sprawdzenia, po której stronie prostej znajduje się punkt, wykorzystujemy wyznacznik macierzy stworzonej z 3 punktów. Można to zrobić na dwa sposoby, tworząc macierz 3x3 lub 2x2. Znak wyznacznika określa położenie punktu c.

$\det(a, b, c) > 0 \Rightarrow$ punkt c leży po lewej stronie prostej ab

$\det(a, b, c) < 0 \Rightarrow$ punkt c leży po prawej stronie prostej ab

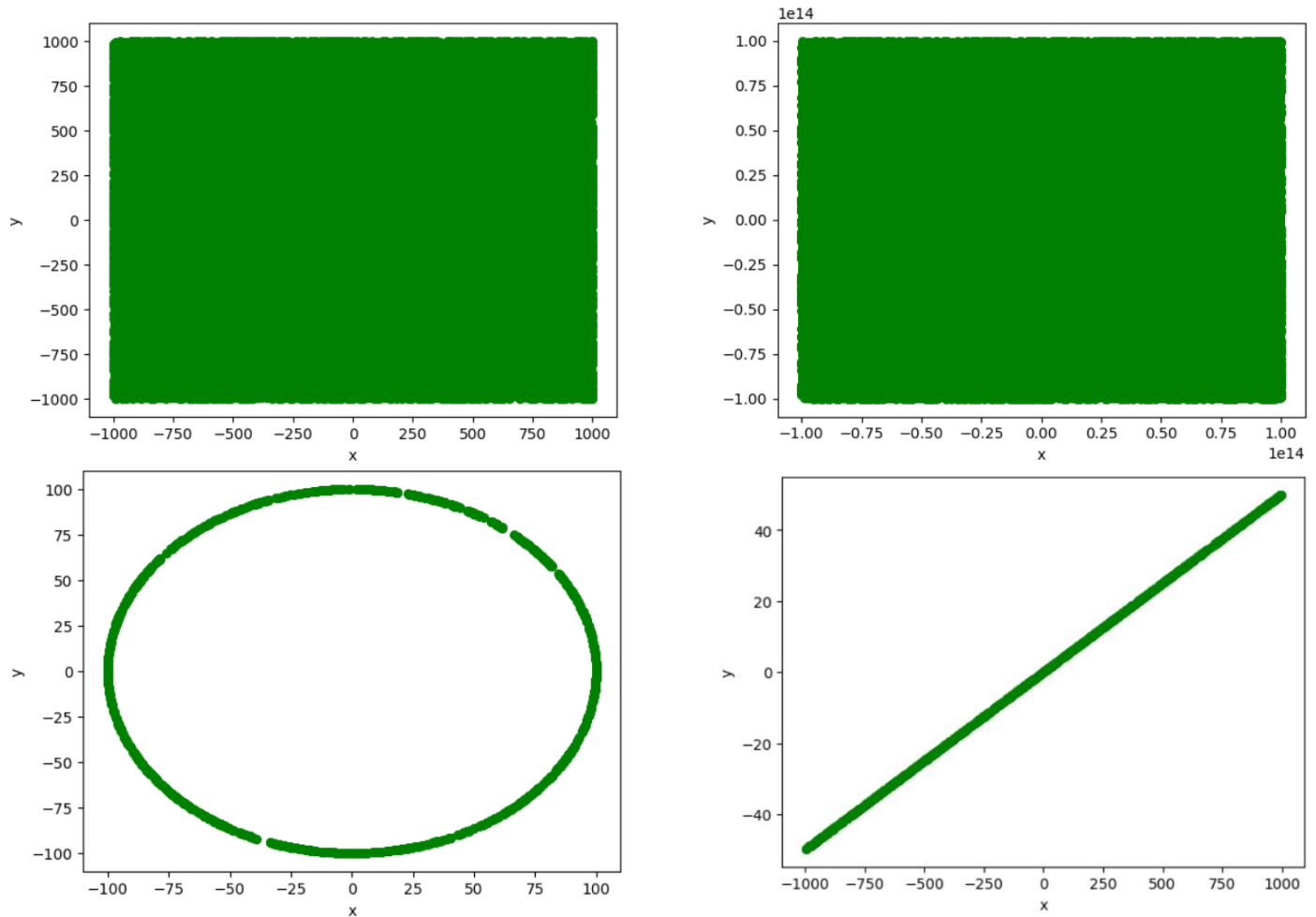
$\det(a, b, c) = 0 \Rightarrow$ punkt c leży na prostej ab

3. Środowisko i użyte narzędzia

Do wykonania ćwiczenia zostało wykorzystane narzędzie jupyter notebook oraz język programowania python. Do wygenerowania pseudolosowych punktów oraz obliczania wyznaczników macierzy, użyta została biblioteka numpy, natomiast do wizualizacji otrzymanych wyników- narzędzie stworzone przez koło naukowe BIT, bazujące na bibliotece matplotlib. Funkcje były uruchamiane na procesorze Intel Core I7 8550U i systemie Windows 11.

4. Plan ćwiczenia

Pierwszym zadaniem było wygenerowanie i zwizualizowanie 4 różnych zbiorów punktów:

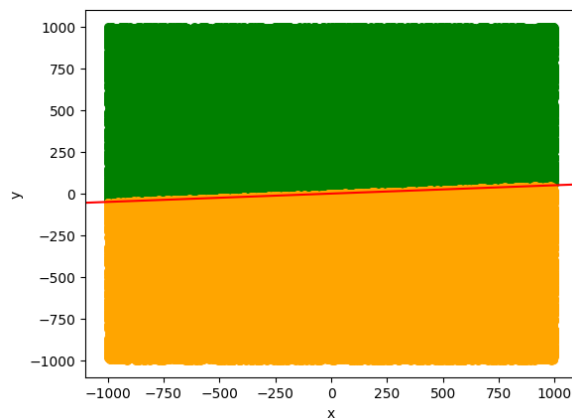


Następnie tworzymy prostą na bazie wektora ab , gdzie $a = [1.0, 0.0]$ i $b = [1.0, 0.1]$ i sprawdzamy liczbę punktów leżących nad, pod i na prostej ab . Położenie punktu będzie obliczane za pomocą wyznacznika z uwzględnieniem różnych tolerancji wobec zera (ϵ) oraz dwóch precyzji typu float.

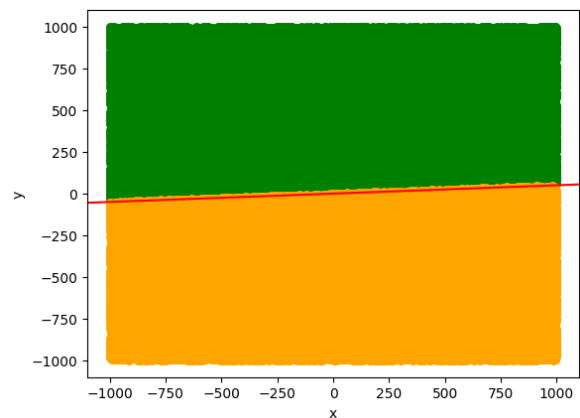
5. Analiza wyników

5.1 Zbiór punktów $(x, y) \in [-1000, 1000]^2$

Precyzja	Funkcja do wyznacznika	Liczba punktów po lewej		Liczba punktów po prawej		Liczba punktów na prostej	
		float32	float64	float32	float64	float32	float64
$\epsilon = 10^{-8}$	mat_det_3x3	50003	50003	49997	49997	0	0
	mat_det_3x3_lib	50003	50003	49997	49997	0	0
	mat_det_2x2	50003	50003	49997	49997	0	0
	mat_det_2x2_lib	50003	50003	49997	49997	0	0
$\epsilon = 10^{-10}$	mat_det_3x3	50003	50003	49997	49997	0	0
	mat_det_3x3_lib	50003	50003	49997	49997	0	0
	mat_det_2x2	50003	50003	49997	49997	0	0
	mat_det_2x2_lib	50003	50003	49997	49997	0	0
$\epsilon = 10^{-12}$	mat_det_3x3	50003	50003	49997	49997	0	0
	mat_det_3x3_lib	50003	50003	49997	49997	0	0
	mat_det_2x2	50003	50003	49997	49997	0	0
	mat_det_2x2_lib	50003	50003	49997	49997	0	0
$\epsilon = 10^{-14}$	mat_det_3x3	50003	50003	49997	49997	0	0
	mat_det_3x3_lib	50003	50003	49997	49997	0	0
	mat_det_2x2	50003	50003	49997	49997	0	0
	mat_det_2x2_lib	50003	50003	49997	49997	0	0
$\epsilon = 0$	mat_det_3x3	50003	50003	49997	49997	0	0
	mat_det_3x3_lib	50003	50003	49997	49997	0	0
	mat_det_2x2	50003	50003	49997	49997	0	0
	mat_det_2x2_lib	50003	50003	49997	49997	0	0



Rysunek 5.1 Wykres zbioru A dla precyzji float32

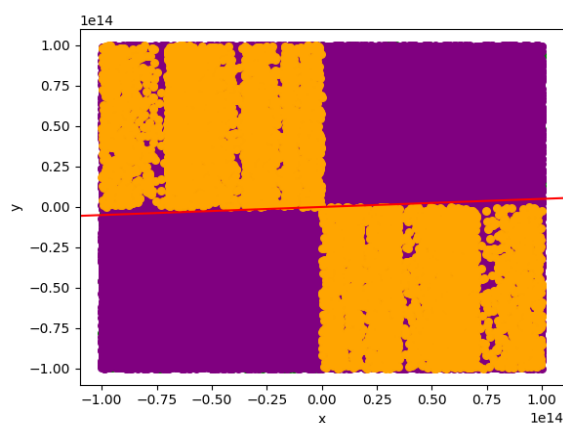


Rysunek 5.2 Wykres zbioru A dla precyzji float64

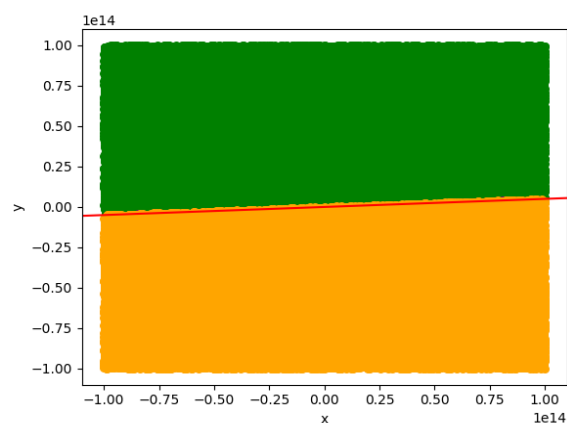
W tym przypadku nie ma żadnych różnic, pomiędzy precyzjami, tolerancjami i metodami obliczania wyznaczników.

5.2 Zbiór punktów $(x, y) \in [-10^{14}, 10^{14}]^2$

Precyzja	Funkcja do wyznacznika	Liczba punktów po lewej		Liczba punktów po prawej		Liczba punktów na prostej	
		float32	float64	float32	float64	float32	float64
$\epsilon = 10^{-8}$	mat_det_3x3	49987	49987	50013	50013	0	0
	mat_det_3x3_lib	49987	49987	50013	50013	0	0
	mat_det_2x2	0	49982	0	50008	100000	10
	mat_det_2x2_lib	6737	49986	6652	50011	86611	3
$\epsilon = 10^{-10}$	mat_det_3x3	49987	49987	50013	50013	0	0
	mat_det_3x3_lib	49987	49987	50013	50013	0	0
	mat_det_2x2	0	49982	0	50008	100000	10
	mat_det_2x2_lib	6737	49986	6652	50011	86611	3
$\epsilon = 10^{-12}$	mat_det_3x3	49987	49987	50013	50013	0	0
	mat_det_3x3_lib	49987	49987	50013	50013	0	0
	mat_det_2x2	0	49982	0	50008	100000	10
	mat_det_2x2_lib	6737	49986	6652	50011	86611	3
$\epsilon = 10^{-14}$	mat_det_3x3	49987	49987	50013	50013	0	0
	mat_det_3x3_lib	49987	49987	50013	50013	0	0
	mat_det_2x2	0	49982	0	50008	100000	10
	mat_det_2x2_lib	6737	49986	6652	50011	86611	3
$\epsilon = 0$	mat_det_3x3	49987	49987	50013	50013	0	0
	mat_det_3x3_lib	49987	49987	50013	50013	0	0
	mat_det_2x2	0	49982	0	50008	100000	10
	mat_det_2x2_lib	6737	49986	6652	50011	86611	3



Rysunek 5.2 Wykres zbioru B dla precyzji float 32

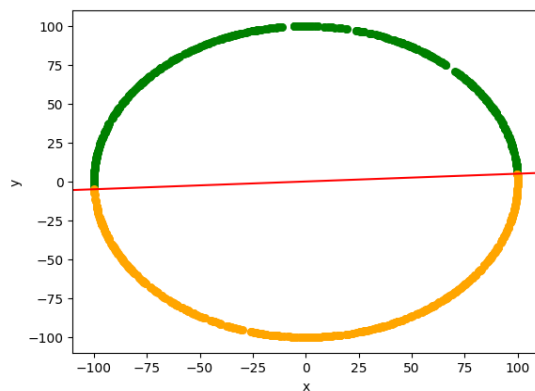


Rysunek 5.2 Wykres zbioru B dla precyzji float64

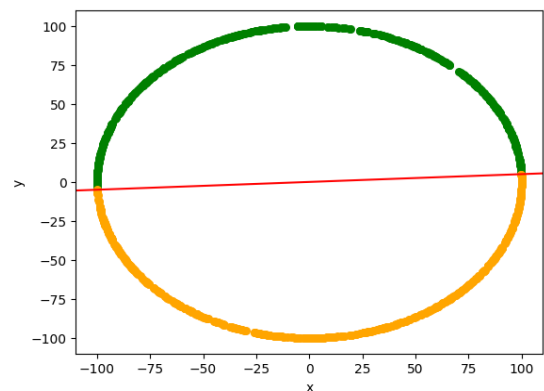
W tym przypadku obserwujemy drastyczną różnicę w obliczonym położeniu punktów. Dla float32 wyniki są przypadkowe, ponieważ zakres danych wykracza poza maksymalne wartości dla tego typu. Natomiast dla float64, wyniki są bardzo podobne do poprzedniego podpunktu. Jedyne różnice, które występują, spowodowane są różnymi metodami obliczania wyznacznika macierzy.

5.3 Zbiór punktów leżących na okręgu o środku $O = (0,0)$ i promieniu $R = 100$

Precyzja	Funkcja do wyznacznika	Liczba punktów po lewej		Liczba punktów po prawej		Liczba punktów na prostej	
		float32	float64	float32	float64	float32	float64
$\epsilon = 10^{-8}$	mat_det_3x3	480	480	520	520	0	0
	mat_det_3x3_lib	480	480	520	520	0	0
	mat_det_2x2	480	480	520	520	0	0
	mat_det_2x2_lib	480	480	520	520	0	0
$\epsilon = 10^{-10}$	mat_det_3x3	480	480	520	520	0	0
	mat_det_3x3_lib	480	480	520	520	0	0
	mat_det_2x2	480	480	520	520	0	0
	mat_det_2x2_lib	480	480	520	520	0	0
$\epsilon = 10^{-12}$	mat_det_3x3	480	480	520	520	0	0
	mat_det_3x3_lib	480	480	520	520	0	0
	mat_det_2x2	480	480	520	520	0	0
	mat_det_2x2_lib	480	480	520	520	0	0
$\epsilon = 10^{-14}$	mat_det_3x3	480	480	520	520	0	0
	mat_det_3x3_lib	480	480	520	520	0	0
	mat_det_2x2	480	480	520	520	0	0
	mat_det_2x2_lib	480	480	520	520	0	0
$\epsilon = 0$	mat_det_3x3	480	480	520	520	0	0
	mat_det_3x3_lib	480	480	520	520	0	0
	mat_det_2x2	480	480	520	520	0	0
	mat_det_2x2_lib	480	480	520	520	0	0



Rysunek 5.3 Wykres zbioru C dla precyzji float32

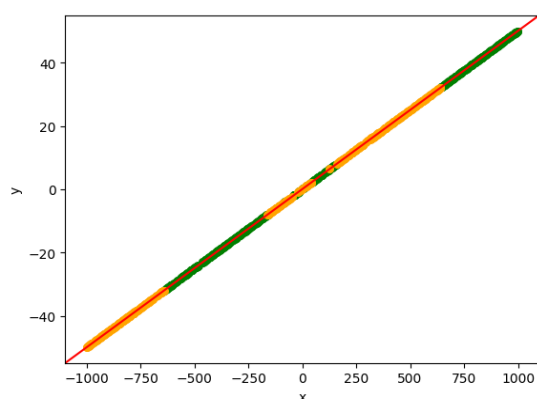


Rysunek 5.3 Wykres zbioru C dla precyzji float64

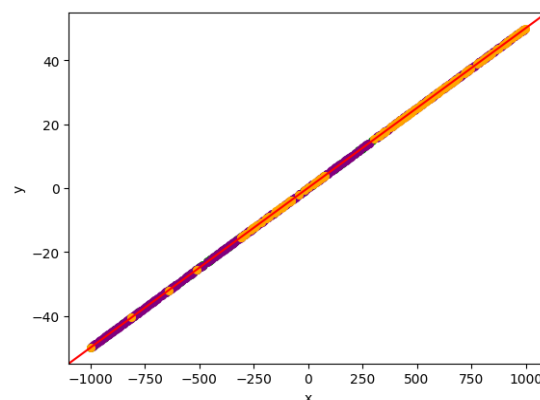
W tym przypadku nie ma żadnych różnic, pomiędzy precyzjami, tolerancjami i metodami obliczania wyznaczników.

5.4 Zbiór punktów o współrzędnej $x \in [-1000, 1000]$, leżących na prostej wyznaczonej przez wektor ab , gdzie $a = (-1.0, 0.0)$, $b = (1.0, 0.1)$

Precyzja	Funkcja do wyznacznika	Liczba punktów po lewej		Liczba punktów po prawej		Liczba punktów na prostej	
		float32	float64	float32	float64	float32	float64
$\epsilon = 10^{-8}$	mat_det_3x3	306	0	322	0	372	1000
	mat_det_3x3_lib	396	0	425	0	179	1000
	mat_det_2x2	163	0	155	0	682	1000
	mat_det_2x2_lib	507	0	490	0	3	1000
$\epsilon = 10^{-10}$	mat_det_3x3	307	0	322	0	371	1000
	mat_det_3x3_lib	396	0	426	0	178	1000
	mat_det_2x2	163	0	156	0	681	1000
	mat_det_2x2_lib	508	0	492	0	0	1000
$\epsilon = 10^{-12}$	mat_det_3x3	307	0	322	0	371	1000
	mat_det_3x3_lib	396	0	426	0	178	1000
	mat_det_2x2	163	74	156	73	681	853
	mat_det_2x2_lib	508	111	492	119	0	770
$\epsilon = 10^{-14}$	mat_det_3x3	307	0	322	0	371	1000
	mat_det_3x3_lib	401	20	437	95	162	885
	mat_det_2x2	163	131	156	136	681	733
	mat_det_2x2_lib	508	151	492	164	0	685
$\epsilon = 0$	mat_det_3x3	307	168	322	395	371	437
	mat_det_3x3_lib	468	372	480	337	52	291
	mat_det_2x2	163	134	156	144	681	722
	mat_det_2x2_lib	508	156	492	171	0	673



Rysunek 5.4 Wykres dla zbioru D dla precyzji float32



Rysunek 5.4 Wykres dla zbioru D dla precyzji float64

W tym przypadku widać ewidentne różnice, sama precyzja typu float zmienia bardzo dużo. Dla float32 większość punktów nie leży na prostej (mimo, że w praktyce leżą), najlepsze rezultaty daje tutaj użycie funkcji mat_det_2x2, która to ma skuteczność około 68%, inne funkcje mają skuteczność na poziomie 37%, 16% i 0%, niezależnie od tolerancji dla zera. Natomiast dla typu float64, uzyskujemy lepsze wyniki, dla wysokich tolerancji, mamy skuteczność 100%, jednak wraz ze zmniejszaniem tolerancji, ta skuteczność spada nawet do 29%. Ciekawą obserwacją jest różne zachowanie funkcji liczących wyznacznik macierzy, przykładowo dla tolerancji $\epsilon = 10^{-14}$ najlepiej

sprawdza się `mat_det_3x3` (100% skuteczności), natomiast dla tolerancji $\epsilon = 0$ najlepsze rezultaty daje funkcja `mat_det_2x2`.

6. Wnioski

Analiza wyników przeprowadzonych testów pozwala stwierdzić, że dla zbiorów A, B i C dostosowanie precyzji zmiennoprzecinkowej, tolerancji na zero oraz metody wyliczania wyznacznika nie miało dużego wpływu na rozróżnienie punktów (w przypadku zbiorów A i C rezultaty były identyczne, a dla zbioru B zastosowanie wyznacznika 2x2 prowadziło do nieznacznych różnic). Odmienna sytuacja miała miejsce dla zbioru D, który składa się z punktów położonych na prostej. Zbiór ten ujawnił ograniczenia komputera w kwestii przechowywania liczb rzeczywistych – wynikające z jego skończonej precyzji. Problemy te były szczególnie zauważalne przy niższej precyzji, gdzie większość punktów nie była rozpoznawana jako leżące na prostej. Nawet przy wyższej precyzji nie uzyskano idealnych rezultatów, gdy tolerancja na zero była zbyt mała.

Podsumowując, zadanie pokazało, jak ważne jest ostrożne podejście do operacji na liczbach rzeczywistych w środowisku komputerowym, ponieważ ich reprezentacja (która w pewnym sensie jest ograniczona przez możliwości sprzętowe – komputer nie radzi sobie z nieskończonymi rozwinięciami dziesiętnymi) może istotnie wpływać na dokładność wyników i prowadzić do błędnych wniosków.