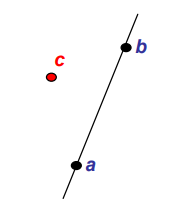
Franciszek Jawor grupa nr 2 15:00 - 16:30 PN

Sprawozdanie laboratorium numer 1: Predykaty Geometryczne

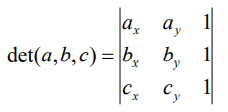
# Cel ćwiczenia

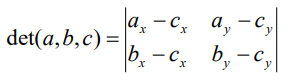
Celem ćwiczenia jest zapoznanie się z podstawowymi predykatami geometrycznymi, ich implementacją oraz przetestowanie i zwizualizowanie wyników.

# Wstęp teoretyczny:



Rysunek 2.1 Przykładowa sytuacja





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 => punkt c leży po lewej stronie prostej ab

det(a,b,c) < 0 => punkt c leży po prawej stronie prostej ab

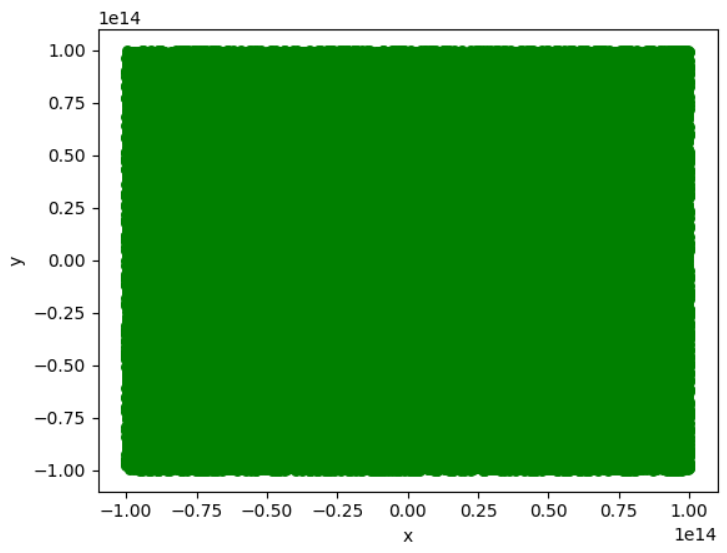
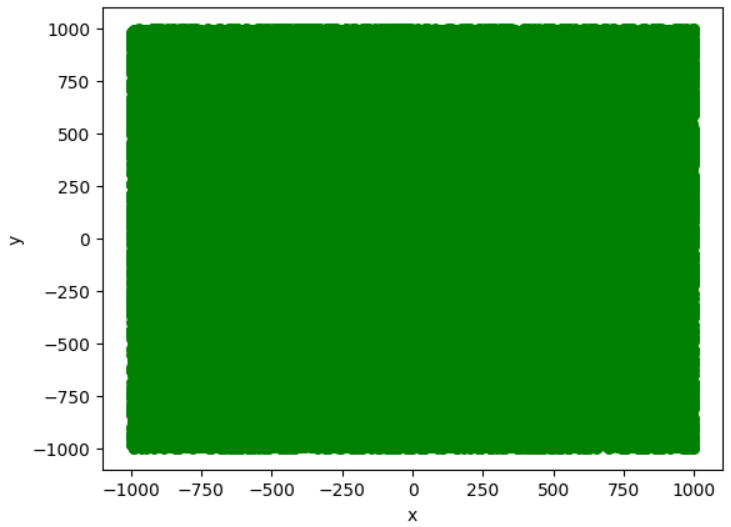
det(a,b,c) = 0 => punkt c leży na prostej ab

# Środowisko i użyte narzędzia

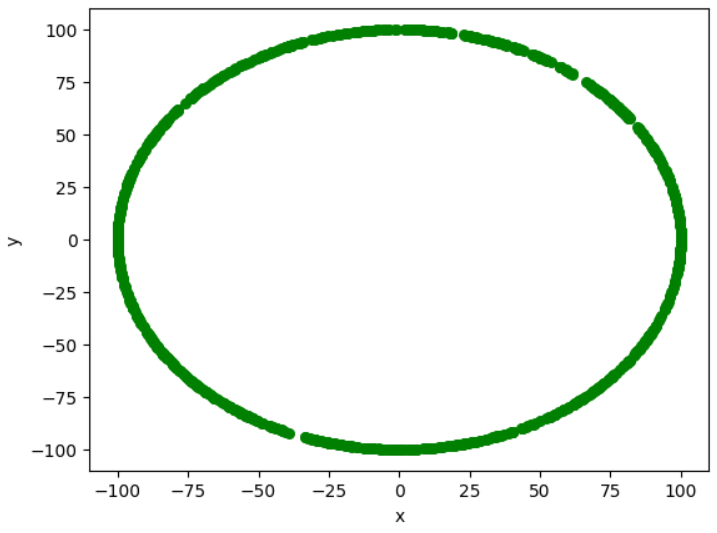
Do wykonania ćwiczenia zostało wykorzystane narzędzie jupyter notebook oraz język programowania python. Do wygenerowania pseudolosowych punktów oraz obliczania wyznaczników macierzy, użyta została biblioteka numpy, natomiast do wizualizacji otrzymanych wyników- narzędzie stworzone przez koło naukowe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

# Plan ćwiczenia

Pierwszym zadaniem było wygenerowanie i zwizualizowanie 4 różnych zbiorów punktów:

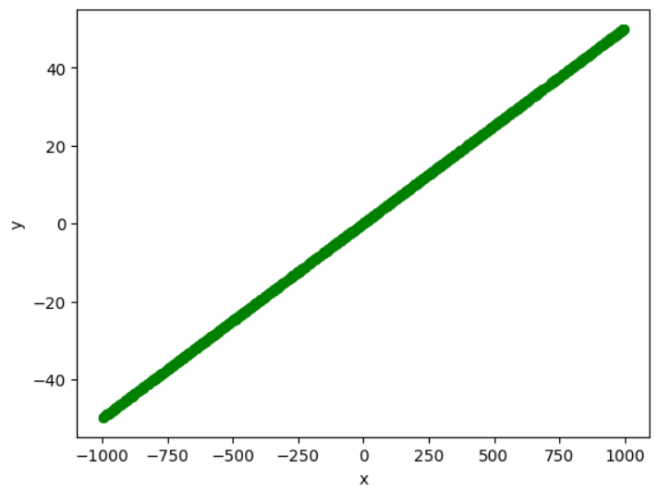


Rysunek 4.2 Wizualizacja zbioru B



Rysunek 4.3 Wizualizacja zbioru C

Rysunek 4.1 Wizualizacja zbioru A



Rysunek 4.4 Wizualizacja zbioru D

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.

Legenda oznaczeń punktów:

- Punkt znajduje się po lewej stronie prostej

- Punkt znajduje się po prawej stronie prostej

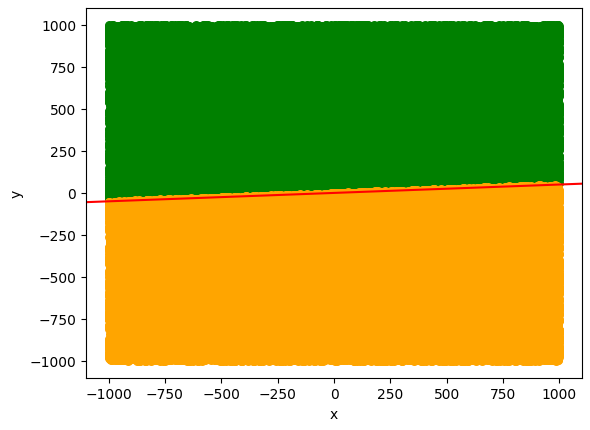
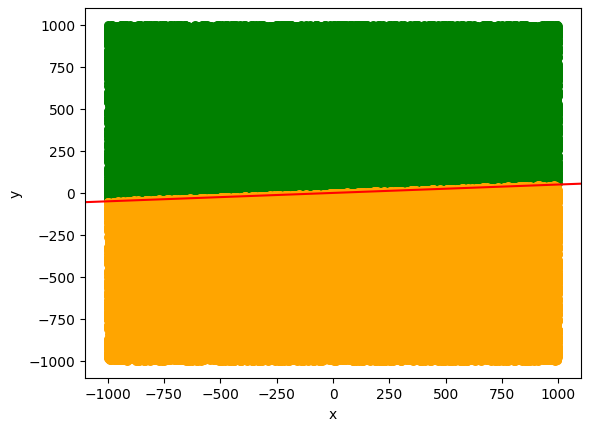
- Punkt znajduje się na prostej

# Analiza wyników

## 5.1 Zbiór punktów (x, y) ∈ [-1000,1000]2

Tabela 5‑1 Rozkład punktów dla zbioru A

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Precyzja** | **Funkcja do wyznacznika** | **Liczba punktów po lewej** | | **Liczba punktów po prawej** | | **Liczba punktów na prostej** | |
| float32 | float64 | float32 | float64 | float32 | float64 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 5.2 Rozkład punktów ze zbioru A dla precyzji float64

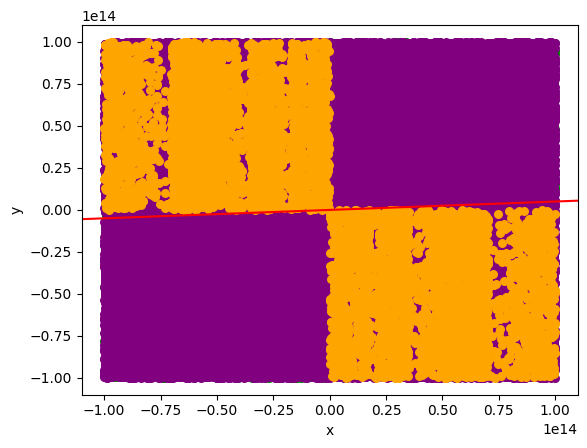
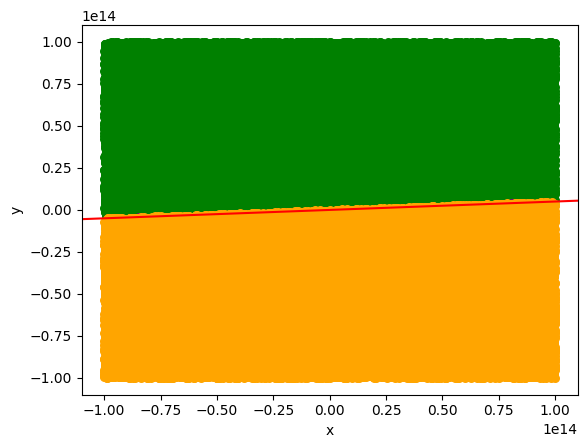
Rysunek 5.1 Rozkład punktów ze zbioru A dla precyzji float32

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

## 5.2 Zbiór punktów (x, y) ∈ [ -1014, 1014 ]2

Tabela 5‑2 Rozkład punktów dla zbioru B

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Precyzja** | **Funkcja do wyznacznika** | **Liczba punktów po lewej** | | **Liczba punktów po prawej** | | **Liczba punktów na prostej** | |
| float32 | float64 | float32 | float64 | float32 | float64 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 Rozkład punktów ze zbioru B dla precyzji float64

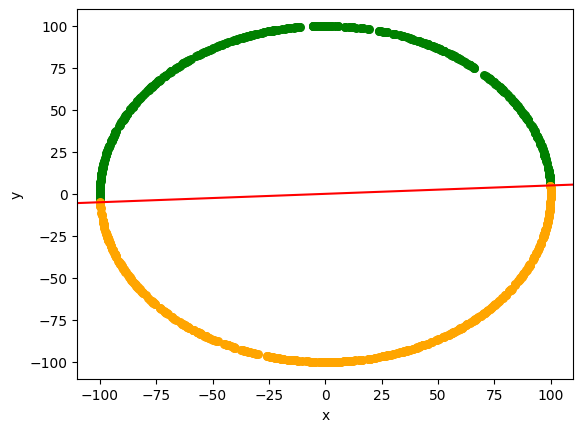
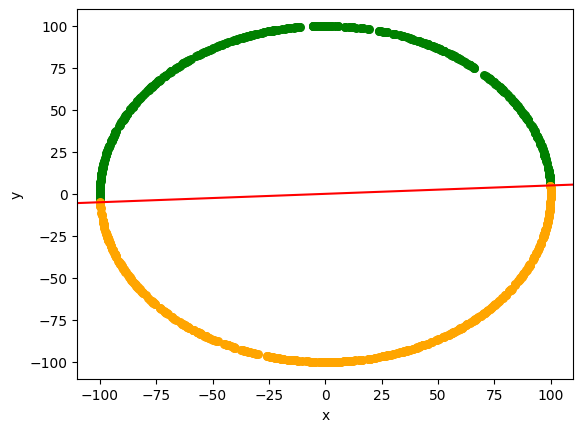
Rysunek 5.2 Rozkład punktów ze zbioru B dla precyzji float32

W tym przypadku obserwujemy drastyczną różnicę w obliczonym położeniu punktów. Dla float32 i wyznacznika 2x2 wyniki są przypadkowe, ponieważ zakres danych wykracza poza maksymalne wartości, dla tego typu. Odejmowanie dużej i małej wartości float, generuje duży błąd. Problem nie występuje w wyznaczniku 3x3, bo liczby najpierw są mnożone, przez co nie tracą, aż tak na precyzji. 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

Tabela 5‑3 Rozkład punktów dla zbioru C

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Precyzja** | **Funkcja do wyznacznika** | **Liczba punktów po lewej** | | **Liczba punktów po prawej** | | **Liczba punktów na prostej** | |
| float32 | float64 | float32 | float64 | float32 | float64 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 Rozkład punktów ze zbioru C dla precyzji float64

Rysunek 5.3 Rozkład punktów ze zbioru C dla precyzji float32

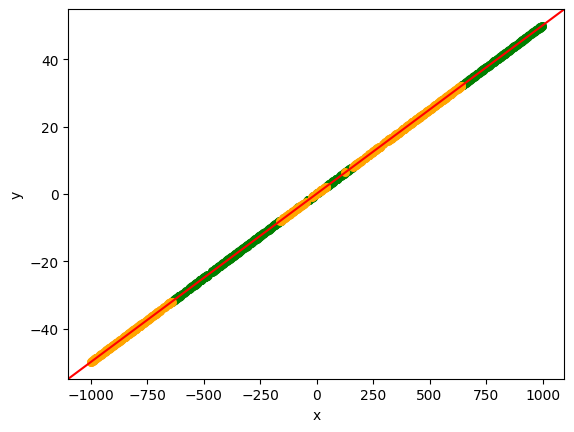
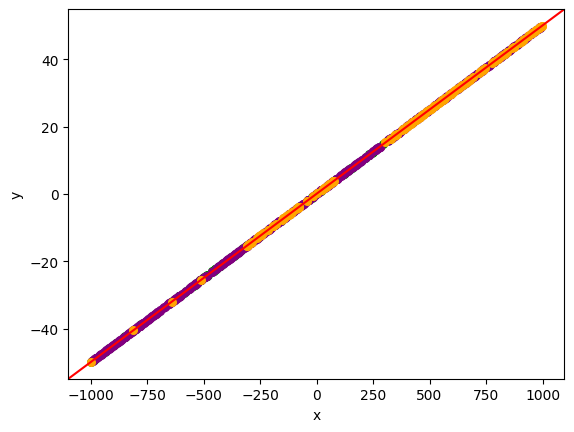
W tym przypadku nie ma żadnych różnic, pomiędzy precyzjami, tolerancjami i metodami obliczania wyznaczników.

## 5.4 Zbiór punktów o współrzędnej x ∈ [-1000, 1000], leżących na prostej

## wyznaczonej przez wektor ab, gdzie a = (-1.0, 0.0), b = (1.0, 0.1)

Tabela 5‑4 Rozkład punktów dla zbioru D

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Precyzja** | **Funkcja do wyznacznika** | **Liczba punktów po lewej** | | **Liczba punktów po prawej** | | **Liczba punktów na prostej** | |
| float32 | float64 | float32 | float64 | float32 | float64 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |
| **ε = 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 |

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 **ε = 10-14** najlepiej sprawdza się mat\_det\_3x3 ( 100% skuteczności), natomiast dla tolerancji **ε = 0** najlepsze rezultaty daje funkcja mat\_det\_2x2.

Rysunek 5.4 Rozkład punktów ze dla zbioru D dla precyzji float64

Rysunek 5.4 Rozkład punktów ze dla zbioru D dla precyzji float32

# Wnioski

Analizując otrzymane wyniki, widać, że precyzja, tolerancja, a nawet metoda obliczania wyznacznika, mają w niektórych przypadkach, kolosalne znaczenie. Funkcja mat\_det\_3x3 była ewidentnie najlepszą opcją, niezależnie od precyzji i zbioru. Najgorzej radziła sobie biblioteczna funkcja mat\_det\_2x2\_lib, dając czasem losowe rezultaty. Zbiory A, B i C miały stosunkowo podobne wyniki, natomiast zbiór D, pokazał jak niedoskonałe są liczby zmiennoprzecinkowe i ich arytmetyka na komputerach. Szczególnie dobrze, widać to było przy float32, gdzie większość tych punktów, nie była klasyfikowana poprawnie.

To ćwiczenie dobitnie pokazało jak bardzo ostrożnym trzeba być, wykonując operacje, na liczbach rzeczywistych. Trzeba być świadomym, że sama sposób ich reprezentacji zakłada niedokładności. Bardzo łatwo o błąd, który, jak pokazała historia może być tragiczny w skutkach.