

**AKADEMIA GÓRNICZO-HUTNICZA**  
WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI  
KIERUNEK INFORMATYKA



ALGORYTMY GEOMETRYCZNE

---

**Laboratorium 1**  
Ćwiczenie wprowadzające

---

Kyrylo Iakymenko  
Czwartek 13:00 - 14:30 tydzień B

Kraków, 16 października 2023

# 1 Wprowadzenie

## 1.1 Cel ćwiczenia

To ćwiczenie ma na celu zapoznanie się z metodami generacji losowych punktów oraz badanie metod klasyfikacji położenia punktów na płaszczyźnie względem prostej.

## 1.2 Położenie punktu względem prostej

Położenie punktu względem prostej będziemy wyznaczać obliczając dane wyznaczniki. Wyznaczniki pozwalają określić położenie punktu  $c$  względem prostej która jest wyznaczona przez punkty  $a$  i  $b$ . Jeżeli wyznacznik jest większy od 0 to punkt znajduje się z lewej strony prostej, jeżeli jest mniejszy od 0 to punkt znajduje się po prawej stronie prostej, a jeżeli wartość wyznacznika jest równa 0 (lub jej wartość bezwzględna  $< \varepsilon$ ) to punkt leży na prostej.

Pomimo, że powyższe wyznaczniki są sobie równoważne to na skutek niedoskonałości reprezentacji liczb rzeczywistych w komputerze wyniki mogą się różnić w zależności od użytego wyznacznika.

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

$$(2) \det(a, b, c) = \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix}.$$

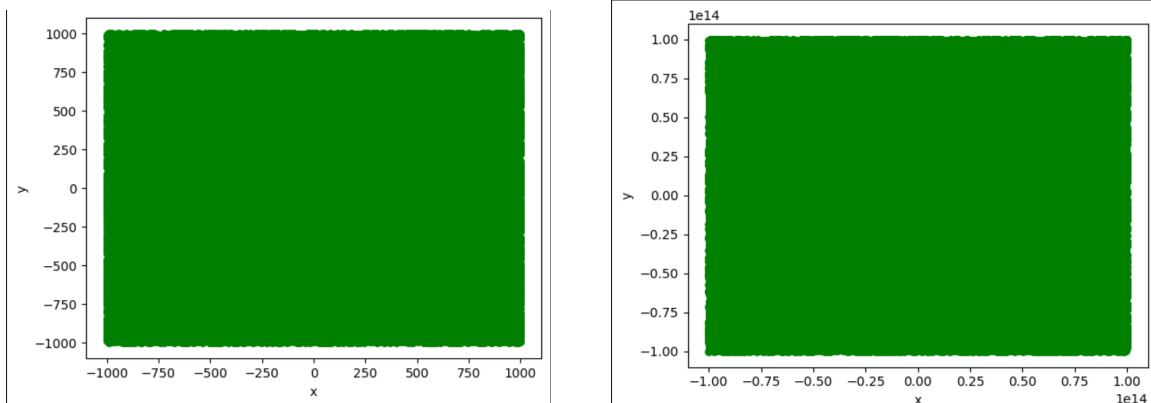
## 2 Zbiory testowe

Na potrzeby ćwiczenia wygenerujemy 4 zbiory punktów losowych.

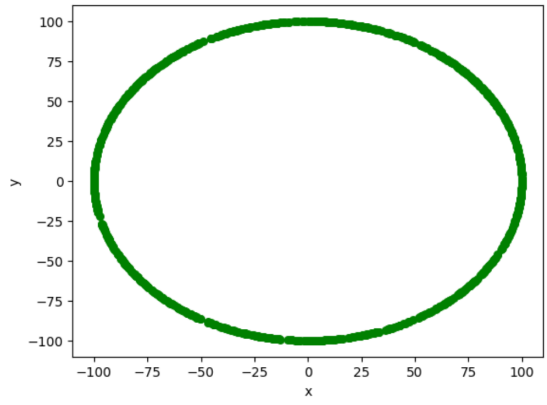
- $10^5$  losowych punktów  $(x, y)$  w przestrzeni  $\mathbb{R}^2$ , gdzie  $(x, y) \in [-1000, 1000]^2$ .
- $10^5$  losowych punktów  $(x, y)$  w przestrzeni  $\mathbb{R}^2$ , gdzie  $(x, y) \in [-10^{14}, 10^{14}]^2$ .
- 1000 losowych punktów w przestrzeni  $\mathbb{R}^2$  leżących na okręgu o środku  $O = (0, 0)$  i promieniu  $R = 100$ .
- 1000 losowych punktów w przestrzeni  $\mathbb{R}^2$  dla  $x \in \langle -1000, 1000 \rangle$  leżących na prostej wyznaczonej przez wektor  $\overrightarrow{ab}$ . Gdzie  $a = (-1.0, 0.0)$ ,  $b = (1.0, 0.1)$ .

## 3 Wykresy i algorytmy generacji zbiorów

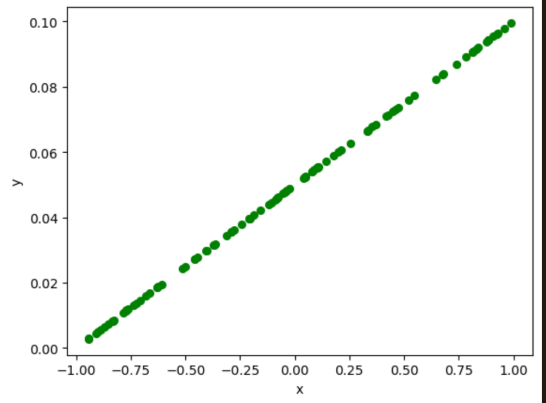
### 3.1 Wykresy wygenerowanych zbiorów



Rys. 1: (a)  $10^5$  losowych punktów  $(x, y) \in [-1000, 1000]^2$ . Rys. 2: (b)  $10^5$  losowych punktów  $(x, y) \in [-10^{14}, 10^{14}]^2$ .



Rys. 3: (c) 1000 losowych punktów leżących na okręgu.



Rys. 4: (d) 1000 losowych punktów na prostej.

### 3.2 Algorytmy generacji zbiorów

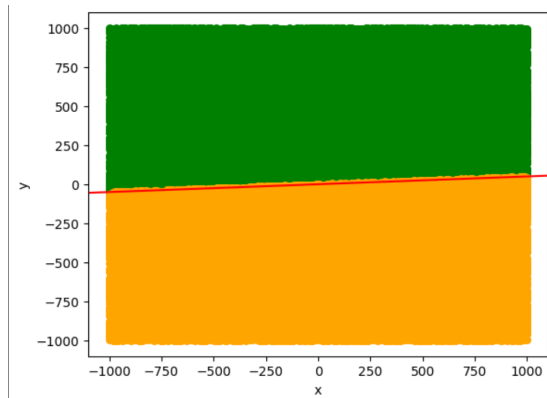
1. Dla zbiorów a i b. Osobna generacja każdego z  $10^5$  losowych punktów.
2. Dla zbioru c. Parametryzacja punktów na okręgu za pomocą funkcji trygonometrycznych sin i cos.
3. Dla zbioru d. Przekształcenie odcinka do postaci parametrycznej

$$l : \begin{cases} x = x_0 + tv_x \\ y = y_0 + tv_y \end{cases} \text{ dla } t \in [0, 1].$$

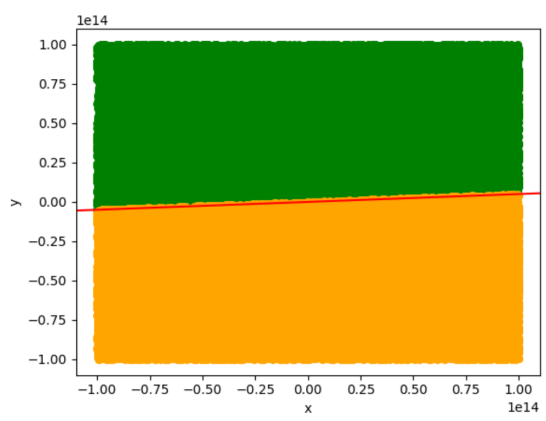
A potem generowanie  $t$  w podanym przedziale i dodanie odpowiednich punktów.

## 4 Testy klasyfikacyjne dla różnych wartości $\varepsilon$

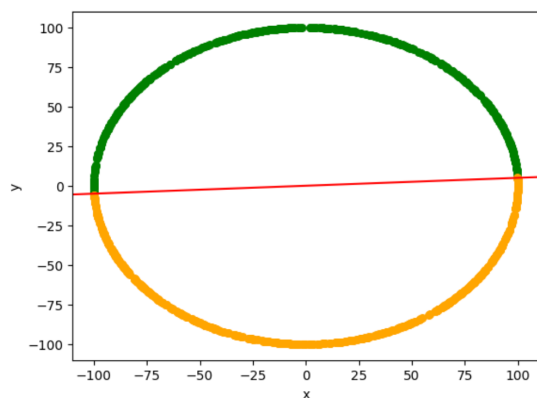
### 4.1 Przykładowe wykresy dla $\varepsilon = 10^{-4}$ .



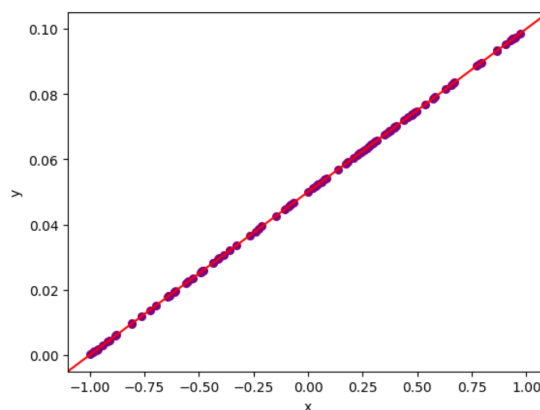
Rys. 5: (a)  $10^5$  losowych punktów  $(x, y) \in [-1000, 1000]^2$ .



Rys. 6: (b)  $10^5$  losowych punktów  $(x, y) \in [-10^{14}, 10^{14}]^2$ .



Rys. 7: (c) 1000 losowych punktów leżących na okręgu.



Rys. 8: (d) 1000 losowych punktów na prostej.

## 4.2 Tabele

Poniżej przedstawione są tabele ilości zaklasyfikowanych punktów ze zbiorów (a, b, c, d) ze względu na ich położenie od prostej,  $\varepsilon$  tolerancje wartości bliskich zera oraz funkcje użytą dla obliczenia wyznacznika.

eps			-4	-8	-12	-14	-16	-18	-20
0	Własny 2x2	Punkty po lewej	50167	50167	50167	50167	50167	50167	50167
1		Punkty na prostej	0	0	0	0	0	0	0
2		Punkty po prawej	49833	49833	49833	49833	49833	49833	49833
3	Numpy 2x2	Punkty po lewej	50167	50167	50167	50167	50167	50167	50167
4		Punkty na prostej	0	0	0	0	0	0	0
5		Punkty po prawej	49833	49833	49833	49833	49833	49833	49833
6	Własny 3x3	Punkty po lewej	50167	50167	50167	50167	50167	50167	50167
7		Punkty na prostej	0	0	0	0	0	0	0
8		Punkty po prawej	49833	49833	49833	49833	49833	49833	49833
9	Numpy 3x3	Punkty po lewej	50167	50167	50167	50167	50167	50167	50167
10		Punkty na prostej	0	0	0	0	0	0	0
11		Punkty po prawej	49833	49833	49833	49833	49833	49833	49833

Tabela 1: Wyniki dla zbioru a.

eps			-4	-8	-12	-14	-16	-18	-20
0	Własny 2x2	Punkty po lewej	50269	50269	50269	50269	50269	50269	50269
1		Punkty na prostej	5	5	5	5	5	5	5
2		Punkty po prawej	49726	49726	49726	49726	49726	49726	49726
3	Numpy 2x2	Punkty po lewej	50270	50270	50270	50270	50270	50270	50270
4		Punkty na prostej	7	7	7	7	7	7	7
5		Punkty po prawej	49723	49723	49723	49723	49723	49723	49723
6	Własny 3x3	Punkty po lewej	50272	50272	50272	50272	50272	50272	50272
7		Punkty na prostej	0	0	0	0	0	0	0
8		Punkty po prawej	49728	49728	49728	49728	49728	49728	49728
9	Numpy 3x3	Punkty po lewej	50272	50272	50272	50272	50272	50272	50272
10		Punkty na prostej	0	0	0	0	0	0	0
11		Punkty po prawej	49728	49728	49728	49728	49728	49728	49728

Tabela 2: Wyniki dla zbioru *b*.

eps			-4	-8	-12	-14	-16	-18	-20
0	Własny 2x2	Punkty po lewej	477	477	477	477	477	477	477
1		Punkty na prostej	0	0	0	0	0	0	0
2		Punkty po prawej	523	523	523	523	523	523	523
3	Numpy 2x2	Punkty po lewej	477	477	477	477	477	477	477
4		Punkty na prostej	0	0	0	0	0	0	0
5		Punkty po prawej	523	523	523	523	523	523	523
6	Własny 3x3	Punkty po lewej	477	477	477	477	477	477	477
7		Punkty na prostej	0	0	0	0	0	0	0
8		Punkty po prawej	523	523	523	523	523	523	523
9	Numpy 3x3	Punkty po lewej	477	477	477	477	477	477	477
10		Punkty na prostej	0	0	0	0	0	0	0
11		Punkty po prawej	523	523	523	523	523	523	523

Tabela 3: Wyniki dla zbioru *c*.

eps			-4	-8	-12	-14	-16	-18	-20
0	Własny 2x2	Punkty po lewej	0	0	0	0	0	25	26
1		Punkty na prostej	100	100	100	100	100	45	43
2		Punkty po prawej	0	0	0	0	0	30	31
3	Numpy 2x2	Punkty po lewej	0	0	0	0	0	30	32
4		Punkty na prostej	100	100	100	100	100	34	32
5		Punkty po prawej	0	0	0	0	0	36	36
6	Własny 3x3	Punkty po lewej	0	0	0	0	0	28	28
7		Punkty na prostej	100	100	100	100	100	47	47
8		Punkty po prawej	0	0	0	0	0	25	25
9	Numpy 3x3	Punkty po lewej	0	0	0	0	0	18	18
10		Punkty na prostej	100	100	100	100	100	82	82
11		Punkty po prawej	0	0	0	0	0	0	0

Tabela 4: Wyniki dla zbioru  $d$ .

### 4.3 Analiza wyników

1. W zbiorze  $a$  niezależnie od  $\varepsilon$  widzimy brak zmian w ilości zaklasyfikowanych do różnych zbiorów punktów.

Żeby zweryfikować nasze wyniki policzmy prawdopodobieństwo, że punkt w naszym zbiorze zostanie wylosowany w miejscu dla którego będzie zaklasyfikowany, jako należący do prostej. W naszych obliczeniach dla prostoty pominiemy niepewności związane z obliczeniami komputerowymi.

Klasyfikujemy punkty do odpowiedniej grupy obliczając iloczyn wektorowy  $\vec{ab} \times \vec{ac}$ , gdzie  $c = (x, y)$  jest punktem, dla którego poszukujemy wiadomości o lokalizacji względem prostej przechodzącej przez punkty  $a$  i  $b$ . Metoda ta jest równoznaczna z obliczeniem wyznacznika macierzy  $2 \times 2$ :

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

Wiemy także, że iloczyn wektorowy  $\vec{A} \times \vec{B}$  można także obliczyć ze wzoru:

$$\vec{A} \times \vec{B} = \|A\| \cdot \|B\| \cdot \sin \theta.$$

Gdzie  $\sin \theta$  to kąt pomiędzy wektorami  $\vec{A}$  i  $\vec{B}$ .

Wybermy teraz dla konkretnego punktu  $c$  takie  $a$  i  $b$  na prostej, żeby

- (a) Dla  $A = \vec{ab}$ ,  $\|A\| = 1$ .
- (b)  $\sin \theta = 1$ , gdzie  $\theta$  jest kątem pomiędzy  $\vec{ab}$  i  $\vec{ac}$  (wektor  $\vec{ac}$  jest prostopadły do naszej prostej i do wektora  $\vec{ab}$ ).

Wtedy nasz iloczyn

$$\vec{A} \times \vec{B} = \|A\| \cdot \|B\| = 1 \cdot \|B\| = \|B\|.$$

I pytanie klasyfikacji punktu jako należącego do prostej sprowadza się do sprawdzenia czy  $\|B\| < \varepsilon$ . Dla zbioru  $a$  długość prostej o równaniu  $y = \frac{1}{20}x - 1$  na całym zbiorze  $(-1000, 1000)$  jest równa

$$\sqrt{(y(1000) - y(-1000))^2 + (1000 + 1000)^2} = \sqrt{10^4 + 4 \cdot 10^6} = \sqrt{4,01 \cdot 10^6} \approx 2 \cdot 10^3.$$

Pole obszaru na którym punkty będą klasyfikowane jako należące do prostej zatem wynosi

$$2 \cdot 2 \cdot 10^3 \cdot \varepsilon = 4 \cdot 10^3 \cdot \varepsilon.$$

Teraz w porównaniu do całkowitego pola naszego zbioru

$$\frac{4 \cdot 10^3 \cdot \varepsilon}{(1000 - (-1000))^2} = \frac{4 \cdot 10^3 \cdot \varepsilon}{4 \cdot 10^6} = 10^{-3} \cdot \varepsilon.$$

Patrząc na ten wynik dla  $\varepsilon = 10^{-4}$  nie jest zaskakujące to, że żaden z  $10^5$  punktów nie został zakwalifikowany jako leżący na prostej. Gdyż prawdopodobieństwo wylosowania takiego punktu to tylko  $10^{-7}$ .

2. Dla zbioru b widzimy, że dla wszystkich  $10^{-20} < \varepsilon < 10^5$  dostajemy ten sam wynik, czego też warto było się spodziewać gdyż stosując metodę opisaną w poprzednim podpunkcie dostajemy prawdopodobieństwo wylosowania punktu leżącego na prostej

$$\frac{4 \cdot 10^{14} \cdot \varepsilon}{(10^{14} - (-10^{14}))^2} = \frac{4 \cdot 10^{14} \cdot \varepsilon}{4 \cdot 10^{28}} = 10^{-14} \cdot \varepsilon.$$

Czyli nawet dla  $\varepsilon = 10^5$  prawdopodobieństwo, że co najmniej jeden z  $10^5$  punktów zostanie zaklasyfikowany do prostej to około  $10^{-4}$ .

Natomiast obserwujemy też specyficzne dla wyznacznika  $2 \times 2$  zachowanie, a mianowicie niezależnie od  $\varepsilon$  pewna stała ilość punktów zostaje zaklasyfikowana jako leżąca na prostej. Prawdopodobnie to jest związane z tym, że wyznacznik  $2 \times 2$  jest mniej precyzyjny w porównaniu do wyznacznika  $3 \times 3$ .

3. Dla zbioru c widzimy podobną sytuację. Dla  $\varepsilon > 10^{-2}$  każde zmniejszenie  $\varepsilon$  powoduje ogromne zmiany w wynikach, ale podalsze zwiększenie precyzji nie ma żadnego efektu na wynikach.
4. Dla zbioru d zmniejszenie  $\varepsilon$  ma największe znaczenie ze wszystkich. Nawet przejście pomiędzy  $\varepsilon = 10^{-16}$  do  $\varepsilon = 10^{-18}$  ma ogromne znaczenie na ilości punktów zakwalifikowanych do prostej.

## 5 Porównywanie czasów klasyfikacji dla różnych funkcji obliczających wyznacznik

Poniżej przedstawiona jest tabela średniego czasu klasyfikacji na zbiorze testowym specjalnie dla tego celu (zbiór ma ogromną ilość punktów).  $10^6$  losowych punktów  $(x, y)$  w przestrzeni  $\mathbb{R}^2$ , gdzie  $(x, y) \in [-10^{14}, 10^{14}]^2$ .

```
test_set = generate_uniform_points(-10^4, 10^4, 10 ** 6)
```

	Numpy 2x2	Numpy 3x3	Własny 2x2	Własny 3x3
<b>0</b> Średni czas klasyfikacji [s]	5.84	5.74	0.58	0.46

Tabela 5: Czasy klasyfikacji na zbiorze testowym.

Widzimy ogromne różnice w czasach klasyfikacji naszego zbioru dla różnych wyznaczników. Najbardziej oczywistą różnicą jest różnica pomiędzy wyznacznikami napisanymi ręcznie, a funkcjami bibliotecznymi, z których ostatnie demonstrują dużo gorsze czasy. Prawdopodobnie możemy zaobserwować takie zjawisko dlatego, że funkcje biblioteczne implementują dużo większy zakres narzędzi, przez co pod czas wielokrotnego wywołania tej samej funkcji są wolniejsze w porównaniu do kilku prostych operacji arytmetycznych, którymi są nasze funkcje liczące wyznacznik.

## 6 Testy precyzji float64 i float32

Na poniższym wykresie jest przedstawiona tabela ilości zaklasyfikowanych do prostej punktów w zależności od precyzji zminnej (float32, float64) i  $\varepsilon$ . zbiorze testowym specjalnie dla tego celu (zbiór punktów na prostej).  $10^4$  losowych punktów w przestrzeni  $\mathbb{R}^2$  dla  $x \in \langle -1000, 1000 \rangle$  leżących na prostej wyznaczonej przez wektor  $\overrightarrow{ab}$ . Gdzie  $a = (-1.0, 0.0)$ ,  $b = (1.0, 0.1)$ .

	eps	-4	-8	-12	-14	-16	-18	-20	-22	-24	-26
0	float64	10000	10000	10000	10000	10000	4067	3855	3855	3855	3855
1	float32	10000	10000	1522	1522	1522	362	361	361	361	361

Tabela 6: Porównanie precyzji float64 i float 32.

Patrząc na porównanie precyzji float64 z float32, możemy dojść do wniosku, że float64 daje (jak wypadało się spodziewać) dużo większą precyzję obliczeń. Widoczna jest poważna różnica w obliczeniach już dla  $\varepsilon = 10^{-12}$ . Dlatego na potrzeby algorytmów geometrycznych zalecane jest używanie float64.

## 7 Podsumowanie

Porównując ilość zaklasyfikowanych punktów dla różnych  $\varepsilon$  dostaliśmy różne wyniki dla różnych metod obliczenia wyznacznika. Klasyfikacja punktów ze zbiorów a oraz c dawała podobne wyniki niezależnie od użytej funkcji liczenia wyznacznika czy  $\varepsilon$ . Spowodowane jest to tym, że istnieje bardzo małe prawdopodobieństwo, iż punkt będzie znajdował się dokładnie na prostej, z dokładnością do użytej tolerancji. W podpunkcie b wybór epsilon nie miał znaczenia jednak wybór wyznacznika już tak. Jest to spowodowane dużymi odległościami pomiędzy punktami. Operacja klasyfikacji punktów wymaga jednak dużej precyzji, która jest tracona z powodu dużych wartości które są generowane. Z tego możemy dojść do wniosku, że użycie wyznaczników  $3 \times 3$  daje nam większą precyzję w obliczeniach, a też patrząc na wyniki porównania wydajności funkcji liczących wyznacznik, są wykonywane tak samo szybko, jak i funkcje liczenia wyznacznika  $2 \times 2$ .

We wszystkich testach i porównaniach statystycznych dostaliśmy wyniki, które zgadzały się z wartościami teoretycznymi. Na tej podstawie możemy dojść do wniosku, że w obliczeniach nie było poważnych błędów i ćwiczenie było wykonane poprawnie.