

# Sprawozdanie z listy drugiej

Karolina Bąk

Listopad 2019

## 1 Zadanie 1

Zadanie polegało na powtórzeniu zadania 5 z poprzedniej listy dla nieco zaburzonych danych. Ponownie obliczyłam iloczyn skalarny dla wektora:

$$x = (2.718281828, -3.141592654, 1.414213562, 0.5772156649, 0.3010299957)$$

oraz:

$$y = (1486.2497, 878366.9879, -22.37492, 4773714.647, 0.000185049)$$

gdzie usunęłam ostatnie 9 z  $x_4$  oraz ostatnie 7 z  $x_5$ . Uzyskałam następujące wyniki:

	Float32	Float64
1	-0.4999443	-0.004296342739891585
2	-0.4543457	-0.004296342998713953
3	-0.5	-0.004296342842280865
4	-0.5	-0.004296342842280865

Poprzednie wyniki wyglądały następująco:

	Float32	Float64
1	-0.4999443	$1.0251881368296672 \cdot 10^{-10}$
2	-0.4543457	$-1.5643308870494366 \cdot 10^{-10}$
3	-0.5	0
4	-0.5	0

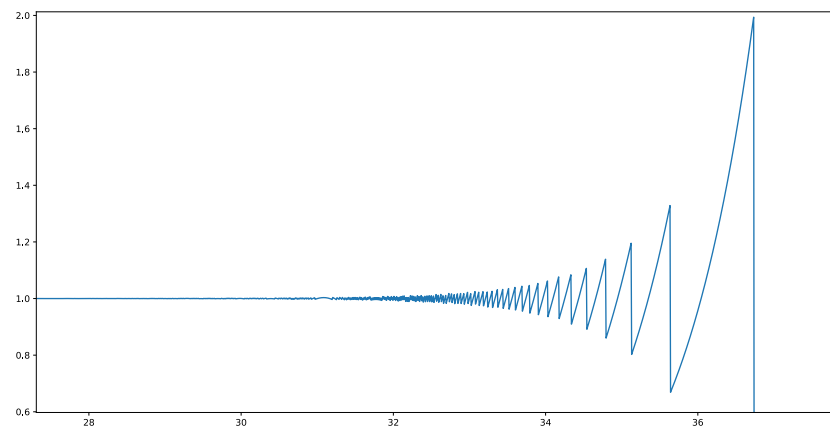
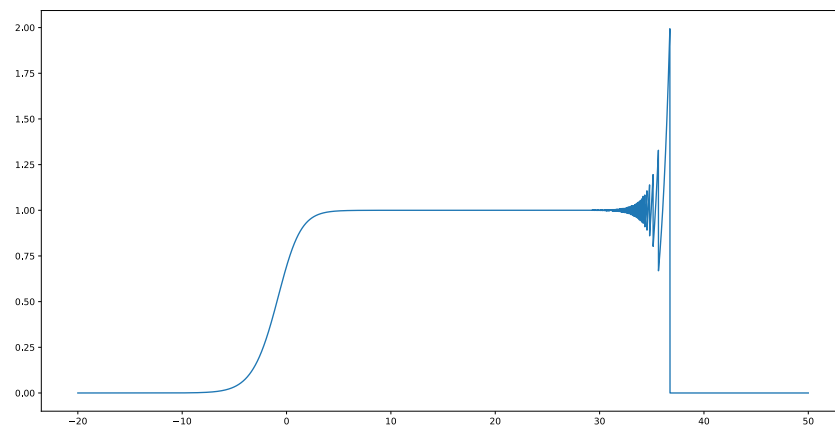
Zmniejszenie liczby cyfr w części dziesiętnej  $x$  spowodowało znaczną utratę bliskości do poprawnego wyniku przy Float64 ( $-1.00657107000000 \cdot 10^{-11}$ ). Niewielkie zmiany rzędu  $10^{-10}$  spowodowały ogromną zmianę ( $10^7$  razy większy wynik). Potwierdza to wniosek z pierwszej listy, że zadanie jest źle uwarunkowane dla danych tego typu (różne znaki przy współrzędnych).

## 2 Zadanie 2

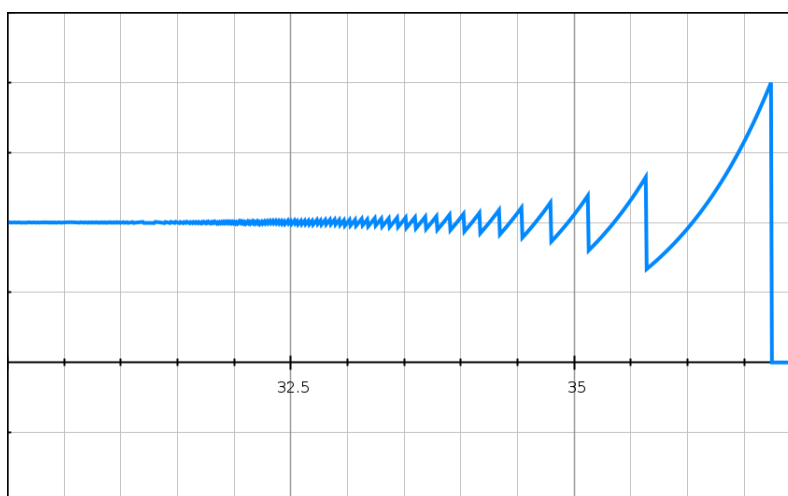
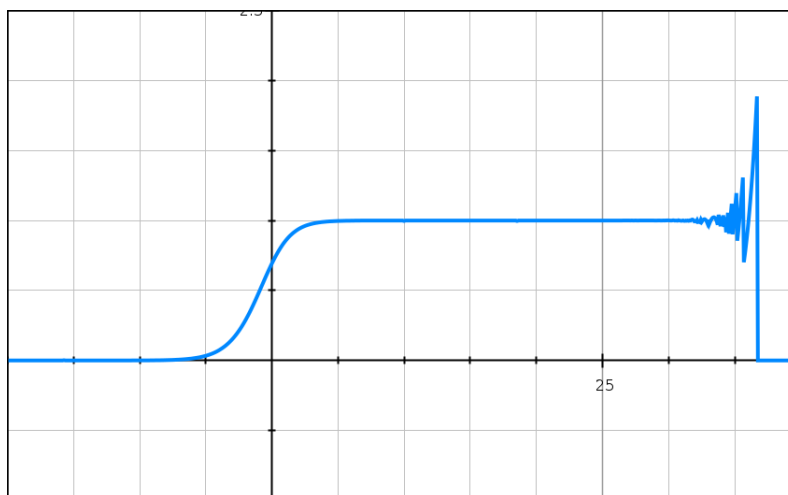
Celem drugiego zadania było przedstawienie graficznie wykresu funkcji:

$$f(x) = e^x * \ln(1 + e^{-x})$$

w co najmniej dwóch programach do wizualizacji. Jeden wykres wykonałam przy użyciu Pythona i paczki matplotlib, a drugi on-line na stronie graphsketch.com. Poniżej przedstawiam otrzymane wykresy w Pythonie:



Wykresy z graphsketch.com:



Granicą powyższej funkcji jest 1. Przybliżając wykresy oraz testując dokładnie funkcję dla odpowiednich wartości ustaliłam, że zaburzenia pojawiły się około  $x = 16$ , a około  $x = 18$  zaczęły przekraczać 1. Im dalej tym większe było zaburzenie, aż funkcja zmieniła się w stałe 0. Stało się tak, ponieważ działałam na bardzo dużych i bardzo małych liczbach. Mnożąc je ze sobą, stale zwiększałam błąd w obliczeniach, który rósł aż  $e(-x)$  zostało pochłonięte przez 1, co wyzerowało logarytm. Dlatego każda następna wartość była równa 0.

### 3 Zadanie 3

Zadanie polegało na rozwiązaniu układu równań liniowych

$$Ax = b$$

gdzie  $A$  jest daną macierzą współczynników  $A \in \mathbb{R}^{n \times n}$ , a  $b \in \mathbb{R}^n$  jest wektorem prawych stron. Macierz  $A$  jest generowana na dwa sposoby. Pierwszy to macierz Hilberta o danym stopniu  $n$  generowana przez funkcję  $\text{hilb}(n)$ . Drugim sposobem jest generowanie losowej macierzy o zadanym wskaźniku uwarunkowania  $c$  przez funkcję  $\text{matcond}(n, c)$ . Wektor  $b$  jest generowany przez powyższe równanie, gdzie  $x = (1, \dots, 1)^T$ . Znane więc są dokładne  $A$  oraz  $b$ .

Test w zadaniu polega na rozwiązaniu układu na dwa różne sposoby: eliminacją Gaussa  $x = A \backslash b$  oraz  $x = A^{-1}b$ . Dla macierzy Hilberta uzyskałam poniższe wyniki.

Gauss:

n	$\frac{\ x - xp\ }{\ x\ }$	cond(A)	rank(A)
1	0.0	1.0	1
2	$5.661048867003676 \cdot 10^{-16}$	19.28147006790397	2
3	$8.022593772267726 \cdot 10^{-15}$	524.0567775860644	3
4	$4.137409622430382 \cdot 10^{-14}$	15513.73873892924	4
5	$1.6828426299227195 \cdot 10^{-12}$	476607.25024259434	5
6	$2.618913302311624 \cdot 10^{-10}$	$1.4951058642254665 \cdot 10^7$	6
7	$1.2606867224171548 \cdot 10^{-8}$	$4.75367356583129 \cdot 10^8$	7
8	$6.124089555723088 \cdot 10^{-8}$	$1.5257575538060041 \cdot 10^{10}$	8
9	$3.8751634185032475 \cdot 10^{-6}$	$4.931537564468762 \cdot 10^{11}$	9
10	$8.67039023709691 \cdot 10^{-5}$	$1.6024416992541715 \cdot 10^{13}$	10
11	0.00015827808158590435	$5.222677939280335 \cdot 10^{14}$	10
12	0.13396208372085344	$1.7514731907091464 \cdot 10^{16}$	11
13	0.11039701117868264	$3.344143497338461 \cdot 10^{18}$	11
14	1.4554087127659643	$6.200786263161444 \cdot 10^{17}$	11
15	4.696668350857427	$3.674392953467974 \cdot 10^{17}$	12
16	54.15518954564602	$7.865467778431645 \cdot 10^{17}$	12
17	13.707236683836307	$1.263684342666052 \cdot 10^{18}$	12
18	9.134134521198485	$2.2446309929189128 \cdot 10^{18}$	12
19	9.720589712655698	$6.471953976541591 \cdot 10^{18}$	13
20	7.549915039472976	$1.3553657908688225 \cdot 10^{18}$	13

Odwrotna macierz:

n	$\frac{\ x-xp\ }{\ x\ }$	cond(A)	rank(A)
1	0.0	1.0	1
2	$1.404333874306803*10^{-15}$	19.28147006790397	2
3	0.0	524.0567775860644	3
4	0.0	15513.73873892924	4
5	$3.3544360584359632*10^{-12}$	476607.25024259434	5
6	$2.0163759404347654*10^{-10}$	$1.4951058642254665*10^7$	6
7	$4.713280397232037*10^{-9}$	$4.75367356583129*10^8$	7
8	$3.07748390309622*10^{-7}$	$1.5257575538060041*10^{10}$	8
9	$4.541268303176643*10^{-6}$	$4.931537564468762*10^{11}$	9
10	0.0002501493411824886	$1.6024416992541715*10^{13}$	10
11	0.007618304284315809	$5.222677939280335*10^{14}$	10
12	0.258994120804705	$1.7514731907091464*10^{16}$	11
13	5.331275639426837	$3.344143497338461*10^{18}$	11
14	8.71499275104814	$6.200786263161444*10^{17}$	11
15	7.344641453111494	$3.674392953467974*10^{17}$	12
16	29.84884207073541	$7.865467778431645*10^{17}$	12
17	10.516942378369349	$1.263684342666052*10^{18}$	12
18	7.575475905055309	$2.2446309929189128*10^{18}$	12
19	12.233761393757726	$6.471953976541591*10^{18}$	13
20	22.062697257870493	$1.3553657908688225*10^{18}$	13

Dla macierzy losowych uzyskałam poniższe wyniki.

Gauss:

n	$\frac{\ x-xp\ }{\ x\ }$	cond(A)	rank(A)
5	$1.7901808365247238*10^{-16}$	1.0000000000000009	5
5	$1.4043333874306804*10^{-16}$	9.999999999999998	5
5	$1.3136335981433191*10^{-16}$	1000.0000000000316	5
5	$3.635564360697878*10^{-10}$	$1.0000000004173718*10^7$	5
5	$9.930136612989092*10^{-17}$	$9.99966763751934*10^{11}$	5
5	0.22407027119013165	$1.1171692878820258*10^{16}$	4
10	$3.1985215122904827*10^{-16}$	1.000000000000001	10
10	$4.749367485114549*10^{-16}$	10.000000000000007	10
10	$3.1044345184083204*10^{-14}$	1000.0000000000381	10
10	$7.312885725957515*10^{-11}$	$9.9999999487321*10^6$	10
10	$3.5736334640683613*10^{-6}$	$9.999705522584362*10^{11}$	10
10	0.007886044944081804	$2.852771852948684*10^{16}$	9
20	$5.093734210850115*10^{-16}$	1.0000000000000013	20
20	$7.199349044417091*10^{-16}$	9.999999999999991	20
20	$1.9013586298626663*10^{-14}$	999.999999999523	20
20	$8.344401704472224*10^{-11}$	$1.0000000003527017*10^7$	20
20	$4.029043035753852*10^{-5}$	$1.0000489898080726*10^{12}$	20
20	0.12546327027700901	$6.405885370975909*10^{15}$	19

Odwrócona macierz:

n	$\frac{\ x-xp\ }{\ x\ }$	cond(A)	rank(A)
5	$2.0471501066083611*10^{-16}$	1.0000000000000007	5
5	$2.2752801345137457*10^{-16}$	9.999999999999998	5
5	$1.53220431207386*10^{-14}$	1000.00000000000089	5
5	$1.0165530953424155*10^{-10}$	$9.99999993366444*10^6$	5
5	$1.7481138973067626*10^{-5}$	$9.9993293808524*10^{11}$	5
5	0.17936451951089713	$5.668621505285428*10^{15}$	4
10	$2.7866376757248753*10^{-16}$	1.0000000000000009	10
10	$4.550560269027491*10^{-16}$	10.000000000000014	10
10	$3.2824171342942656*10^{-14}$	1000.00000000000537	10
10	$4.020386326707943*10^{-10}$	$9.99999998237088*10^6$	10
10	$9.19195614625663*10^{-6}$	$9.999112458768989*10^{11}$	10
10	0.12357124133296675	$1.0162651221978986*10^{16}$	9
20	$3.394814396577995*10^{-16}$	1.0000000000000013	20
20	$5.489713268447767*10^{-16}$	10.000000000000005	20
20	$2.3174107273898665*10^{-14}$	999.9999999999737	20
20	$2.3524392971717174*10^{-10}$	$1.000000006275691*10^7$	20
20	$1.0904881120709677*10^{-5}$	$1.0000824146741567*10^{12}$	20
20	0.04533606607016923	$8.635757378668062*10^{15}$	19

Z otrzymanych danych wynika, że im większy wskaźnik uwarunkowania tym większy błąd względny, czyli jeśli wskaźnik uwarunkowania jest duży, to zadanie jest źle uwarunkowane. Przy macierzy Hilberta z rosnącym n bardzo szybko zwiększa się cond(A), przez co uzyskane wyniki szybko zaczynają być absurdalne. Od n=10 w macierzy Hilberta można zauważyć, że rząd zaczyna rosnąć coraz wolniej. Ilość tych samych rank() pod rząd rośnie co 1 (2 razy 10, 3 razy 11, 4 razy 12, itd). W losowych macierzach również można zauważyć spadek rzędu na cond() $=10^{16}$ .

## 4 Zadanie 4