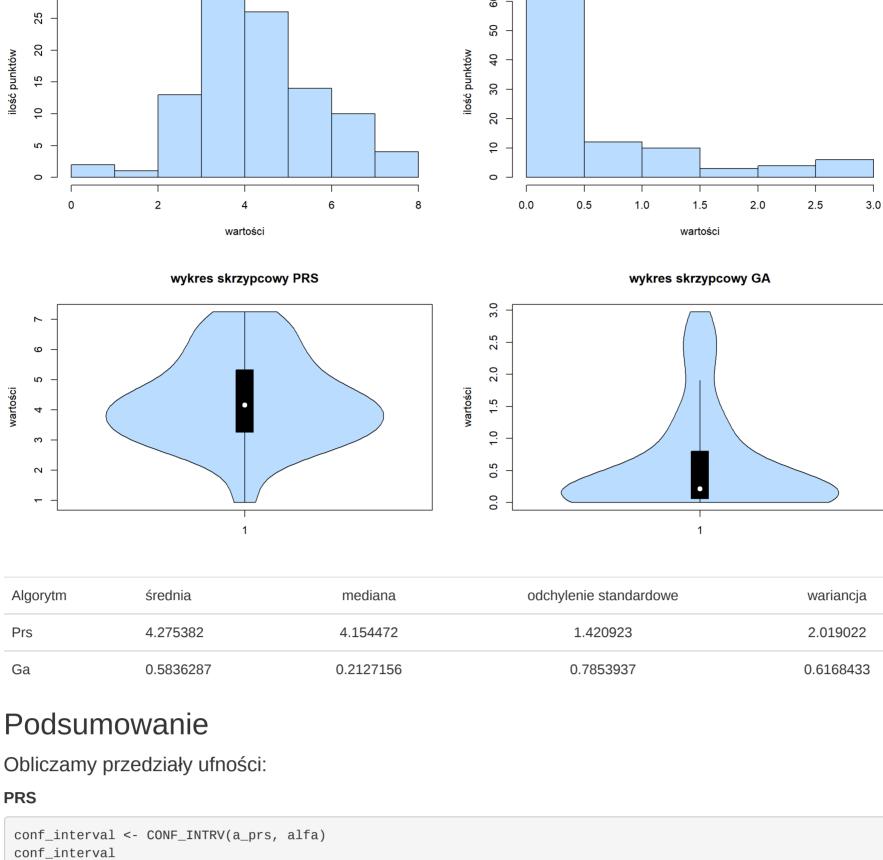
Statystyka - Projekt Mateusz Marczyk & Karol Sewiło 2023-01-26 Wstęp Celem projektu jest porównanie działania dwóch algorytmów minimalizacji stochastycznej. Do porównania wybraliśmy algorytm poszukiwania przypadkowego (PRS) oraz algorytm genetyczny (GA). Zmienne globalne budget <- 1000 cost <- 100 alfa <- 0.05 Implementacje wybranych algorytmów Algorytm Pure Random Search (PRS) PRS <- function(lower\_bound, upper\_bound, f,n) { if(length(lower\_bound) != length(upper\_bound)) stop("Bounds must be equal dimensions") p <- integer(length(lower\_bound))</pre> fmin <- Inf **for** (x **in** 1:n) { for(i in seq\_along(p)) { p[i] <- runif(1, lower\_bound[i], upper\_bound[i])</pre> fmin <- min(fmin, f(p))</pre> return(fmin) Algorytm Genetyczny (GA) GA <- function(f, n){ result <- ga( type = "real-valued", fitness = function(x) (-1)\*f(x), lower = getLowerBoxConstraints(f), upper = getUpperBoxConstraints(f), maxiter = 40,popSize = 25)return(-result@fitnessValue) } Funckja obliczania przedziału ufności CONF\_INTRV <- function(vector, alfa)</pre> return(unname(c(mean(vector)-quantile(vector, 1-(alfa/2))\*sd(vector)/sqrt(length(vector)), mean(vector)+quantil e(vector, 1-(alfa/2))\*sd(vector)/sqrt(length(vector))))) Wybrane funkcje Do testów wybraliśmy funkcję Akcley'a oraz Rastrigin'a. Obie są skalarnem wielomodalne oraz akceptują dane w różnych wymiarach. Opracowanie danych Na początu analizować będziemy wyniki obu algorytmów dla funkcji Akcley'a w odpowiednio: 2, 10 oraz 20 wymiarach. Następnie przeprowdzimy analogiczną analizę dla funkcji Rastrigin'a. Hipoteza zerowa - Algorytmy nie różnią się efektywnością szukania minimum funkcji. Hipotezy alternatywne 1. Algorytm PRS jest efektywniejszy w szukaniu minimum funkcji. 2. Algorytm genetyczny jest efektywniejszy w szukaniu minimum funkcji. Funkcja Ackley'a Wzór funkcji:  $f(\mathbf{x}) = -a \cdot \exp\Bigl(-b\sqrt{rac{1}{d}\sum_{i=1}^d x_i^2})\Bigr) - \exp\Bigl(rac{1}{d}\sum_{i=1}^d \cos(cx_i)\Bigr) + a + \exp(1)$ Minimum globalne:  $f(\mathbf{x}^*) = 0$ , at  $\mathbf{x}^* = (0, \dots, 0)$ Zalecana dziedzina poszukiwań:  $x_i \in [-32.768, \ 32.768], \ orall_{1 \leq i \leq d}, ext{ gdzie } d ext{ to wymiar przestrzeni}$ Domyślne stałe:  $a = 20, \ b = 0.2, \ c = 2\pi$ Ackley Function f(x1,x2) 5 40 20 20 -20 -20 40 x2 $\times 1$ 2 wymiary ackley <- makeAckleyFunction(2)</pre> a\_prs <- replicate(n = cost,</pre> PRS(getLowerBoxConstraints(ackley),

40 getUpperBoxConstraints(ackley), ackley, budget)) a\_ga <- replicate(n = cost, GA(ackley, budget))</pre> hist(a\_prs, main="histogram PRS", xlab = "wartości", ylab = "ilość punktów", col = "#BADDFF") hist(a\_ga, main="histogram GA", xlab = "wartości", ylab = "ilość punktów", col = "#BADDFF") vioplot(a\_prs, main="wykres skrzypcowy PRS", ylab = "wartości", col = "#BADDFF") vioplot(a\_ga, main="wykres skrzypcowy GA", ylab = "wartości", col = "#BADDFF") histogram PRS histogram GA 30 9 25 20 20 ilość punktów ilość punktów 40 15 30 20 2 10



## [1] 3.274548 5.276216

## [1] 0.3756612 0.7915962

10 wymiarów

19

18

17

Podsumowanie

## [1] 16.62230 19.56473

## [1] 5.796658 10.185640

Obliczamy przedziały ufności:

conf\_interval <- CONF\_INTRV(a\_prs, alfa)</pre>

conf\_interval <- CONF\_INTRV(a\_ga, alfa)</pre>

wartości

**PRS** 

GA

conf\_interval

conf\_interval

19.0

20.2

Algorytm

Ga

**PRS** 

GA

conf\_interval

conf\_interval

19.2

19.4

średnia

19.90033605

11.549499

conf\_interval <- CONF\_INTRV(a\_prs, alfa)</pre>

conf\_interval <- CONF\_INTRV(a\_ga, alfa)</pre>

standardowe jest stosunkowo niewielkie.

Funkcja Rastrigin'a

Podsumowanie

## [1] 19.41127 20.38940

## [1] 9.519652 13.579346

Obliczamy przedziały ufności:

19.8

wartości

wykres skrzypcowy PRS

20.0

20.2

mediana

19.92528026

11.712324

conf\_interval <- CONF\_INTRV(a\_ga, alfa)</pre>

odchylenie standardowe jest znaczące.

PRS(getLowerBoxConstraints(ackley), getUpperBoxConstraints(ackley),

wartości

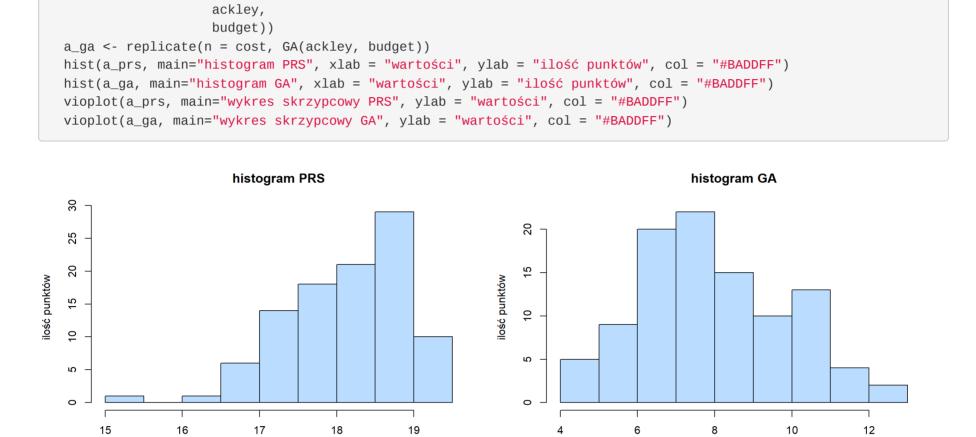
wykres skrzypcowy PRS

ackley <- makeAckleyFunction(10)</pre>

a\_prs <- replicate(n = cost,</pre>

GA

conf\_interval



wartości

wykres skrzypcowy GA

W przypadku porównywania 2 wymiarów dla funkcji Ackley'a sprawdza się hipoteza alternatywna 2.

Natomiast dla algorytmu genetycznego średnia jest niemal trzykrotnie większa od mediany, a

Dla algorytmu PRS średnia jest bardzo zbliżona do mediany, a odchylenie standardowe jest niewielkie.

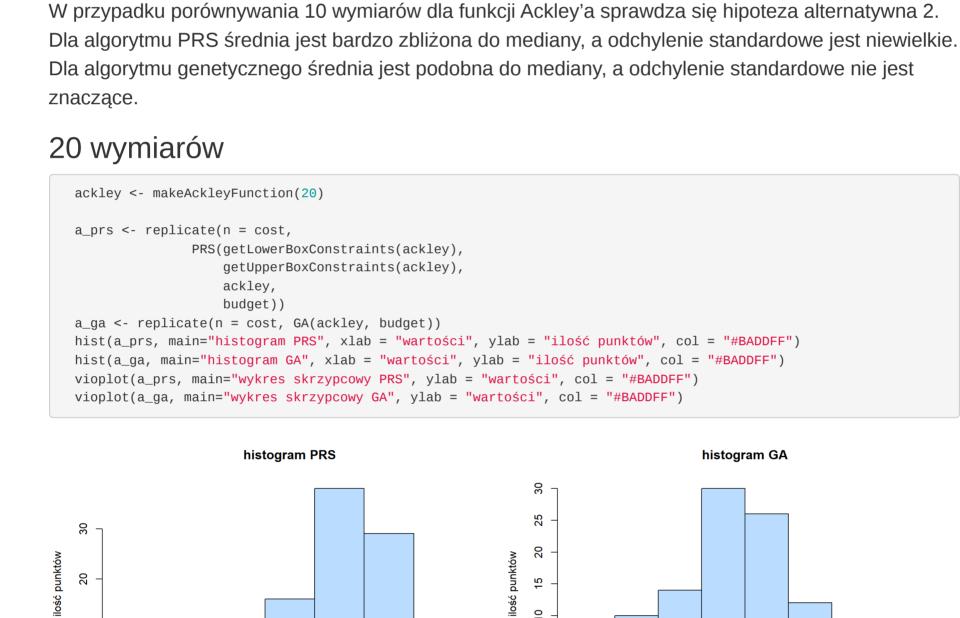
16 9 Algorytm średnia mediana odchylenie standardowe wariancja Prs 18.0935143 18.2096402 0.7707545 0.5940626 1.900478 3.611818 Ga 7.991149 7.778349

12

10

 $\infty$ 

wartości



20.0 19.6 19.8 wartości wartości 12 10

10

4

8

10

odchylenie standardowe

0.24156182

1.454063

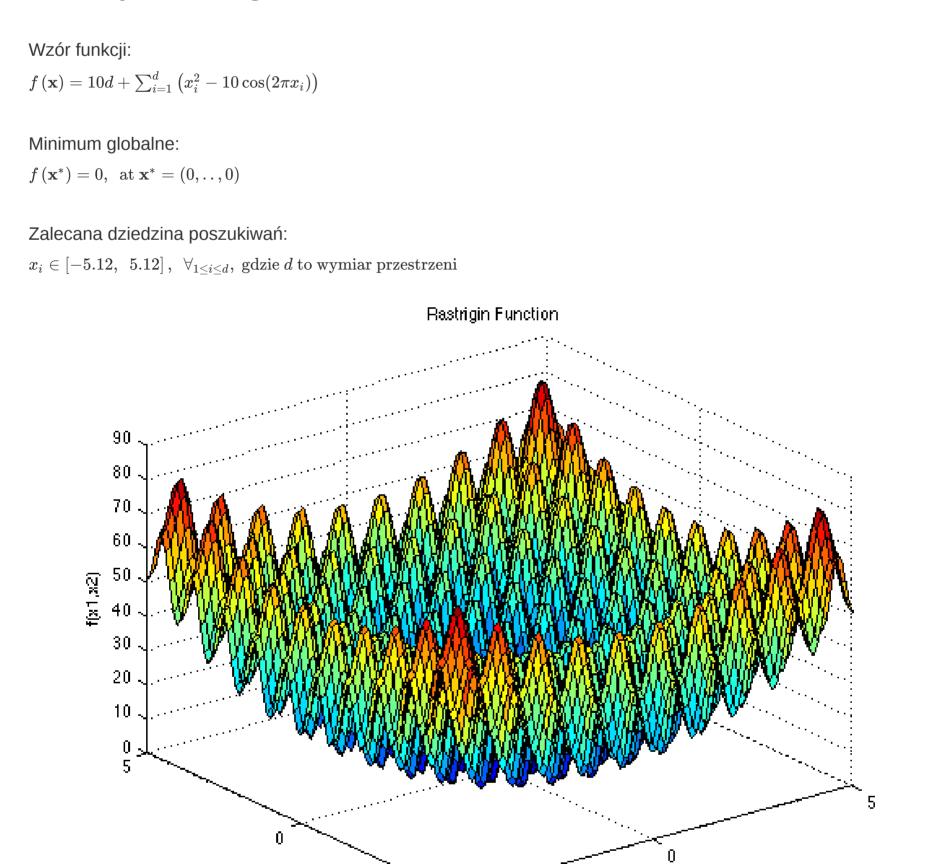
12 wartości

wykres skrzypcowy GA

wariancja

0.05835211

2.114299



-5

hist(r\_prs, main="histogram PRS", xlab = "wartości", ylab = "ilość punktów", col = "#BADDFF") hist(r\_ga, main="histogram GA", xlab = "wartości", ylab = "ilość punktów", col = "#BADDFF")

50

40

30

20

ilość punktów

vioplot(r\_prs, main="wykres skrzypcowy PRS", ylab = "wartości", col = "#BADDFF") vioplot(r\_ga, main="wykres skrzypcowy GA", ylab = "wartości", col = "#BADDFF")

 $\times 1$ 

histogram GA

 $\times 2$ 

PRS(getLowerBoxConstraints(rastrigin), getUpperBoxConstraints(rastrigin),

rastrigin <- makeRastriginFunction(2)</pre>

rastrigin, budget)) r\_ga <- replicate(n = cost, GA(rastrigin, budget))</pre>

histogram PRS

r\_prs <- replicate(n = cost,</pre>

2 wymiary

30

25

15

10

25

20

15

10

2

100

90

80

70

9

Algorytm

Prs

Ga

wartości

60

70

średnia

84.674428

22.178742

Podsumowanie

standardowe jest znaczące.

r\_prs <- replicate(n = cost,</pre>

rastrigin <- makeRastriginFunction(20)</pre>

rastrigin, budget)) r\_ga <- replicate(n = cost, GA(rastrigin, budget))</pre>

histogram PRS

PRS(getLowerBoxConstraints(rastrigin), getUpperBoxConstraints(rastrigin),

hist(r\_prs, main="histogram PRS", xlab = "wartości", ylab = "ilość punktów", col = "#BADDFF") hist(r\_ga, main="histogram GA", xlab = "wartości", ylab = "ilość punktów", col = "#BADDFF")

vioplot(r\_prs, main="wykres skrzypcowy PRS", ylab = "wartości", col = "#BADDFF") vioplot(r\_ga, main="wykres skrzypcowy GA", ylab = "wartości", col = "#BADDFF")

20 wymiarów

35

30

190

Algorytm

Prs

Ga

**PRS** 

średnia

228.4766

74.55114

Podsumowania całości i wnioski

Podsumowanie

Obliczamy przedziały ufności:

ilość punktów

80

wartości

wykres skrzypcowy PRS

90

100

mediana

86.327949

21.533745

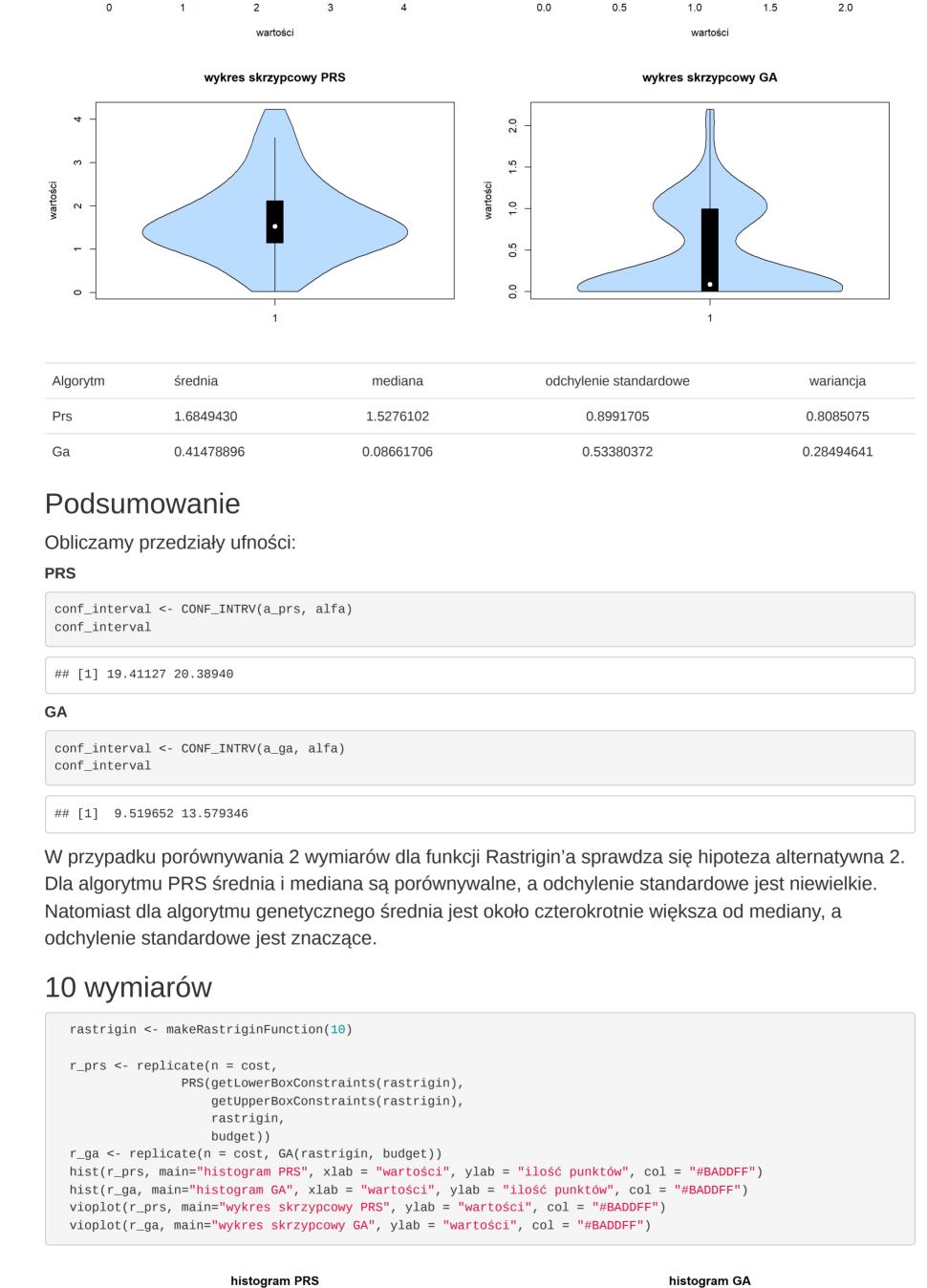
ilość punktów

ilość punktów

W przypadku porównywania 20 wymiarów dla funkcji Ackley'a sprawdza się hipoteza alternatywna 2.

małe. Dla algorytmu genetycznego średnia jest bardzo zbliżona do mediany, a odchylenie

Dla algorytmu PRS średnia jest prawie identyczna jak mediana, a odchylenie standardowe jest bardze



30

25

20

15

10

40

35

30

25

20

15 10 10

odchylenie standardowe

9.533503

7.570933

20

30

wartości

wykres skrzypcowy GA

histogram GA

40

wariancja

90.887679

57.319031

ilość punktów

Pousumowanie
Obliczamy przedziały ufności:
PRS
<pre>conf_interval &lt;- CONF_INTRV(a_prs, alfa) conf_interval</pre>
## [1] 19.41127 20.38940
GA
<pre>conf_interval &lt;- CONF_INTRV(a_ga, alfa) conf_interval</pre>
## [1] 9.519652 13.579346
W przypadku porównywania 10 wymiarów dla funkcji Rastrigin'a sprawdza się hipoteza alternatywna 2. Dla algorytmu PRS średnia jest bardzo zbliżona do mediany, a odchylenie standardowe jest niewielkie. Dla algorytmu genetycznego średnia jest porównywalna do mediany, a odchylenie

200 260 60 180 220 240 40 80 100 120 wartości wartości wykres skrzypcowy PRS wykres skrzypcowy GA 120 100 wartości 80

mediana

230.6898

74.01480

9

40

odchylenie standardowe

13.2615

18.86128

wariancja

175.8673

355.74783

20

15

ilość punktów 10

<pre>conf_interval &lt;- CONF_INTRV(a_prs, alfa) conf_interval</pre>
## [1] 19.41127 20.38940
GA
<pre>conf_interval &lt;- CONF_INTRV(a_ga, alfa) conf_interval</pre>
## [1] 9.519652 13.579346
W przypadku porównywania 20 wymiarów dla funkcji Rastrigin'a sprawdza się hipoteza alternatywna 2. Dla algorytmu PRS średnia jest bardzo bliska medianie, a odchylenie standardowe jest bardzo niewielkie. Natomiast dla algorytmu genetycznego średnia praktycznie pokrywa się z miedaną, a odchylenie standardowe jest znaczące.

Jak widać algorytm genetyczny daje zdecydowanie lepsze wyniki niż algorytm PRS. Możemy jednak zaobserwować, iż wraz ze wzrostem liczby wymiarów algorytm PRS dogania algorytm genetyczny oraz jego odchylenie standardowe jest procentowo niższe niż w przypadku GA. Ze względu na nierównomierne rozłożenie wyników algorytmu genetycznego, jego odchylenie standardowe oraz wariancja są stosunkowo wysokie. Dla każdego z testowanych wymiarów średnia oraz mediana algorytmu PRS były do siebie bardzo zbliżone, a w przypadku GA miało to miejsce dopiero dla większej ilości wymiarów (tj. 10 i więcej).