

Porównanie działania wybranych algorytmów minimalizacji stochastycznej

Kacper Feliks, Maciej Wiśniewski

28-01-2025

Cel projektu

Projekt polega na prostym opracowaniu statystycznym wyników porównania działania wybranych algorytmów minimalizacji stochastycznej. Zdecydowaliśmy się do porównania użyć następujących algorytmów:

- **Poszukiwanie przypadkowe (Pure Random Search, PRS)**

- **Metoda wielokrotnego startu (multi-start, MS)**

Opis algorytmów

Poszukiwanie przypadkowe (Pure Random Search, PRS)

Algorytm PRS polega na losowym przeszukiwaniu przestrzeni rozwiązań, w której minimalizowana funkcja jest zdefiniowana. Działa w następujący sposób:

1. **Losowanie punktów:** Losujemy kolejne punkty w przestrzeni poszukiwań z rozkładu jednostajnego. Jeżeli dziedzina poszukiwań jest kostką wielowymiarową, to każdą współrzędną punktu losujemy z odpowiedniego jednowymiarowego rozkładu jednostajnego.
Na przykład, jeśli dziedzina poszukiwań to kostka trójwymiarowa $[0, 1] \times [-2, 2] \times [100, 1000]$, losowanie współrzędnych wygląda następująco:
 - pierwsza współrzędna: $U(0, 1)$,
 - druga współrzędna: $U(-2, 2)$,
 - trzecia współrzędna: $U(100, 1000)$.
2. **Porównanie wartości funkcji:** Wartość funkcji w każdym wylosowanym punkcie porównujemy z aktualnie zapamiętanym minimum. Jeśli wartość funkcji w nowym punkcie jest mniejsza, zapamiętujemy ten punkt jako nowe minimum.
3. **Wynik:** Wartość funkcji w ostatnim zapamiętanym punkcie stanowi wynik algorytmu.

Metoda wielokrotnego startu (Multi-Start, MS)

Algorytm **MS** łączy losowe przeszukiwanie przestrzeni z metodami optymalizacji lokalnej. Jego kroki są następujące:

1. **Losowanie punktów:** Podobnie jak w **PRS**, losujemy zadany zbiór punktów startowych z rozkładu jednostajnego w przestrzeni poszukiwań.
2. **Uruchomienie optymalizacji lokalnej:** Dla każdego wylosowanego punktu startowego uruchamiana jest metoda optymalizacji lokalnej.
3. **Porównanie wyników:** Dla każdego startu zapisujemy wartość funkcji w zwróconym punkcie lokalnego minimum. Wynikiem algorytmu jest minimalna wartość funkcji spośród wszystkich punktów końcowych.

Do porównania należało wybrać dwie z funkcji dostępnych w pakiecie **smoof**, które są skalarne (single-objective) i mają wersje dla różnej liczby wymiarów (akceptują parametr **dimensions**).

W celu sprawdzenia dostępnych algorytmów wykonaliśmy następujący algorytm, który znajdował dostępne funkcje o wymaganych parametrach:

```
library(smoof)

scalar_dimensional_functions <- Filter(function(fn_name) {
  fn <- get(fn_name, envir = asNamespace('smoof'))
  is.function(fn) &&
  'dimensions' %in% names(formals(fn)) &&
  inherits(try(fn(2), silent = TRUE), 'smoof_function') &&
  getNumberOfObjectives(fn(2)) == 1
}, ls('package:smoof'))

print(scalar_dimensional_functions)
```

Do porównania wybraliśmy dwie funkcje:

- **Griewank**
- **Schwefel**

Nasz wybór padł dokładnie na te funkcje ze względu na nich odmiennność, trudność w optymalizacji oraz niebanalną implementację.

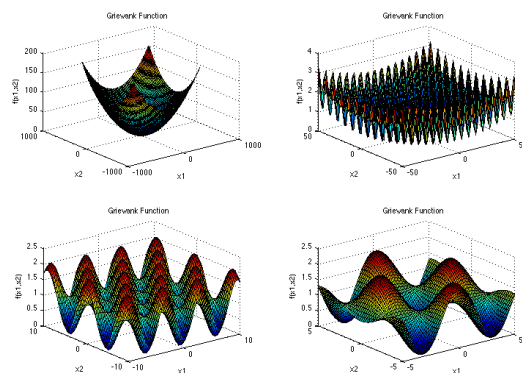
Funkcja Griewanka

Funkcja Griewanka ma wiele szeroko rozpowszechnionych minimów lokalnych, które są regularnie dystrybuowane. Wzór funkcji:

$$f(\mathbf{x}) = \sum_{i=1}^d \frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

gdzie przez d rozumiemy ilość wymiarów. Funkcja jest zazwyczaj definiowana na hiperszkiełach $x_i \in [-600, 600]$, dla każdego $i = 1, \dots, d$.

Minimum globalne $f(\mathbf{x}^*) = 0$, dla $\mathbf{x}^* = (0, \dots, 0)$



Wizualizacja funkcji Griewanka w 3D

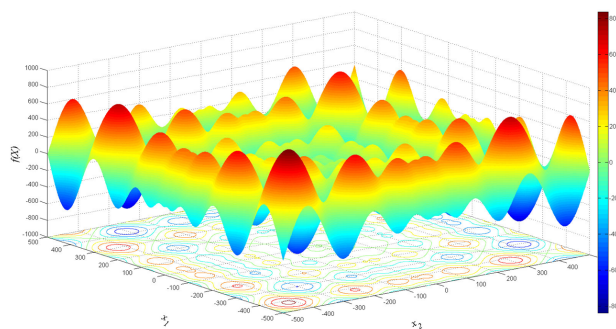
Funkcja Schwefela

Funkcja Schwefela jest złożoną funkcją, posiadającą wiele minimów lokalnych. Wzór funkcji:

$$f(\mathbf{x}) = \sum_{i=1}^n -x_i \sin\left(\sqrt{|x_i|}\right)$$

gdzie przez d rozumiemy ilość wymiarów. Funkcja jest zazwyczaj definiowana na hiperszkiełach $x_i \in [-500, 500]$, dla każdego $i = 1, \dots, d$.

Minimum globalne $f(\mathbf{x}^*) = 418.9829 * d$, dla $\mathbf{x}^* = (420.9687, \dots, 420.9687)$



Reprezentacja funkcji Schwefela w przestrzeni 3D

Specyfikacja sprzętu

Obliczenia i testy zostały wykonane na komputerze o następujące specyfikacji:

- system Windows 10
- procesor Intel Core i7-6700HQ 2.60 GHz.
- pamięć Ram 16Gb

WYNIKI

Funkcja Griewanka 2D

Miara	MS	PRS
Średnia	0.6055442	0.6434122
Wartość najmniejsza	0.007396	0.0373963
Wartość największa	4.4481658	1.7034647
Mediana	0.3253765	0.6161477
Dolny kwartył (25%)	0.1257573	0.4292943
Górny kwartył (75%)	0.8310025	0.8541051

Funkcja Griewanka 10D

Miara	MS	PRS
Średnia	$5.3433198 \times 10^{-11}$	51.7091663
Wartość najmniejsza	$1.3820056 \times 10^{-12}$	12.0229676
Wartość największa	$2.1964297 \times 10^{-10}$	83.9943267
Mediana	$4.1341264 \times 10^{-11}$	51.6668436
Dolny kwartył (25%)	$2.7560176 \times 10^{-11}$	43.7718444
Górny kwartył (75%)	$7.4529827 \times 10^{-11}$	61.4505366

Funkcja Griewanka 20D

Miara	MS	PRS
Średnia	$8.1357054 \times 10^{-11}$	222.7744787
Wartość najmniejsza	$1.3866686 \times 10^{-13}$	138.0348086
Wartość największa	$1.7841773 \times 10^{-10}$	271.2005696
Mediana	$7.5553341 \times 10^{-11}$	223.940142
Dolny kwartył (25%)	$5.6181226 \times 10^{-11}$	207.9649149
Górny kwartył (75%)	$1.0998349 \times 10^{-10}$	239.8048899

Funkcja Schwefela 2D

Miara	MS	PRS
Średnia	-836.7813912	-797.1918759
Wartość najmniejsza	-837.9657745	-836.5685042
Wartość największa	-719.5274399	-643.5621982
Mediana	-837.9657745	-804.2280449
Dolny kwartyl (25%)	-837.9657745	-823.5638115
Górny kwartyl (75%)	-837.9657745	-781.6197479

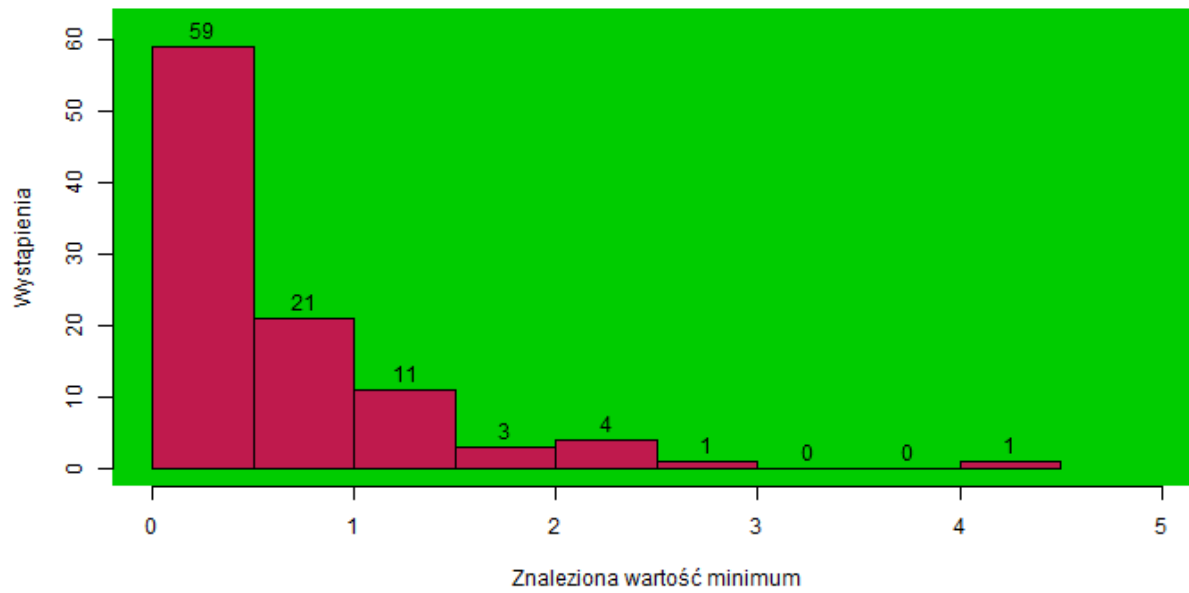
Funkcja Schwefela 10D

Miara	MS	PRS
Średnia	-3262.1275223	-1990.2290063
Wartość najmniejsza	-3716.0755343	-2648.5365291
Wartość największa	-2923.4470726	-1671.7495182
Mediana	-3259.0250766	-1977.0998535
Dolny kwartyl (25%)	-3378.9804707	-2101.5557137
Górny kwartyl (75%)	-3142.8476519	-1880.3428427

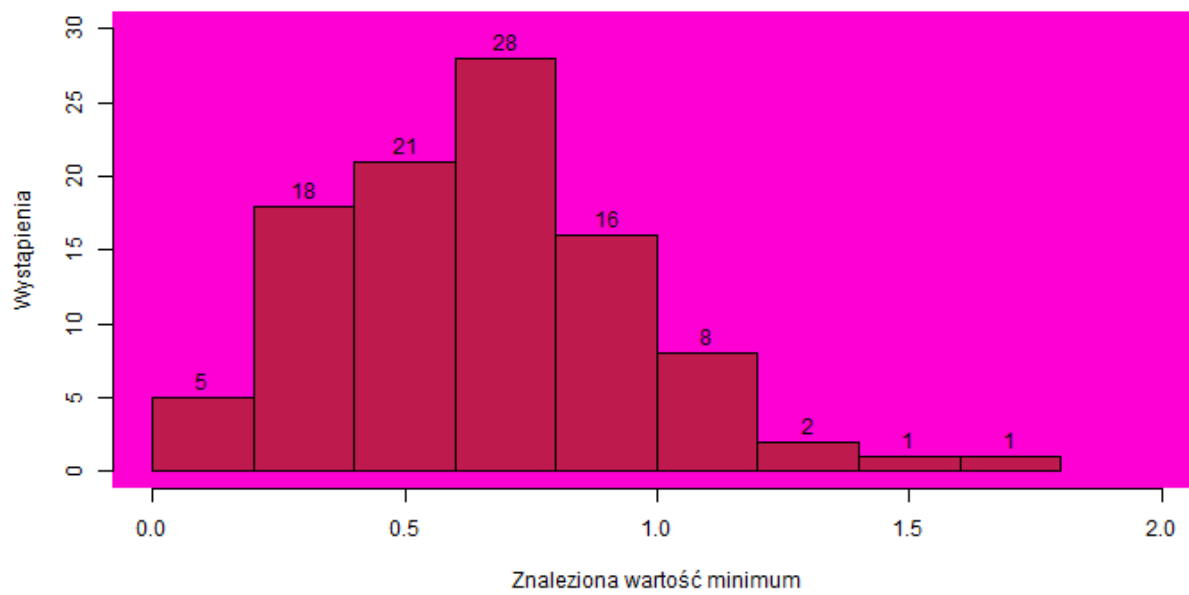
Funkcja Schwefela 20D

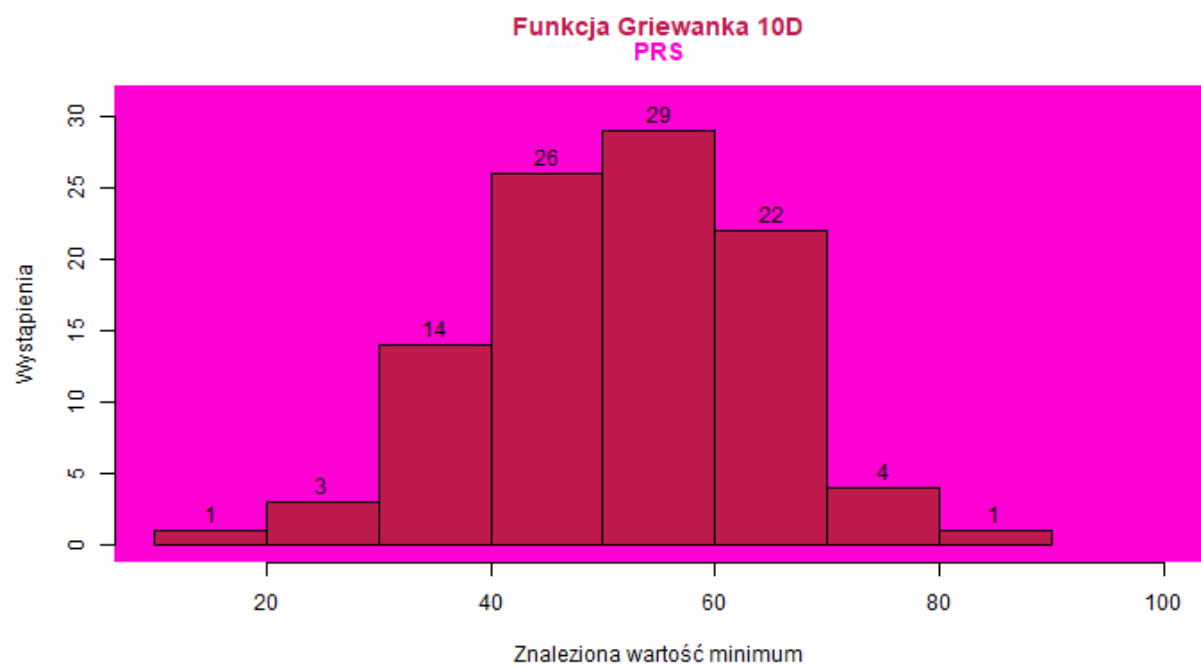
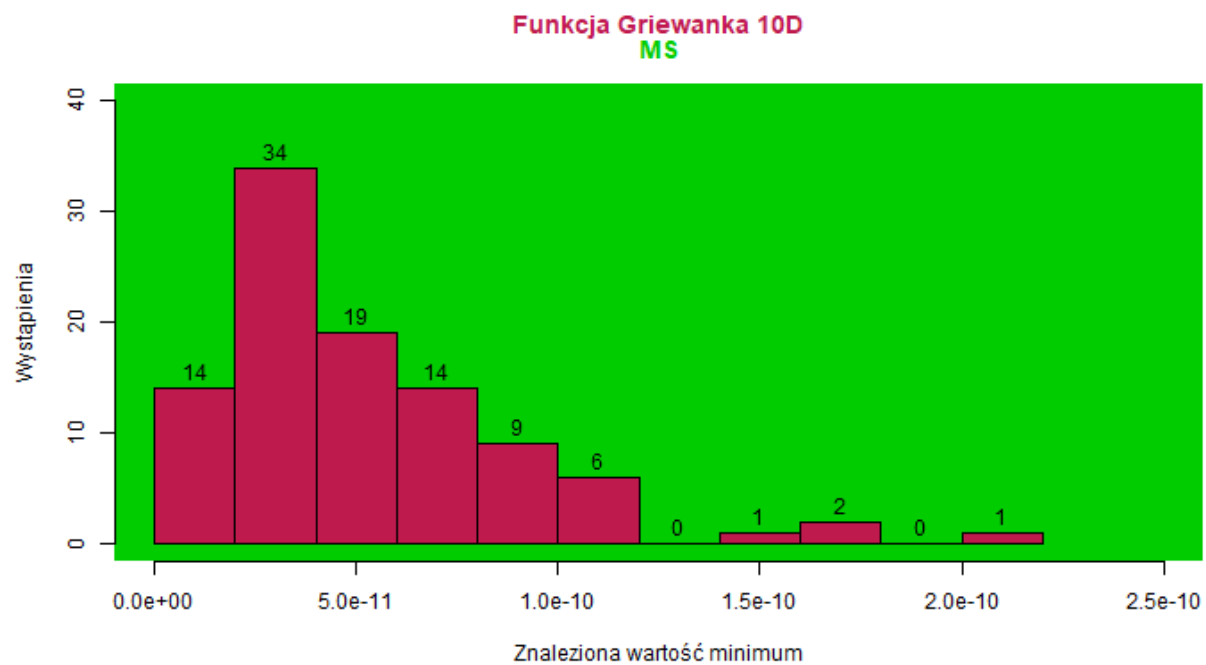
Miara	MS	PRS
Średnia	-5930.2842622	-2871.2092706
Wartość najmniejsza	-6721.5020158	-3883.630182
Wartość największa	-5415.4216596	-2404.6476755
Mediana	-5927.3537852	-2845.6522752
Dolny kwartyl (25%)	-6073.1127276	-3018.7275952
Górny kwartyl (75%)	-5732.1016643	-2683.1582378

Funkcja Griewanka 2D
MS

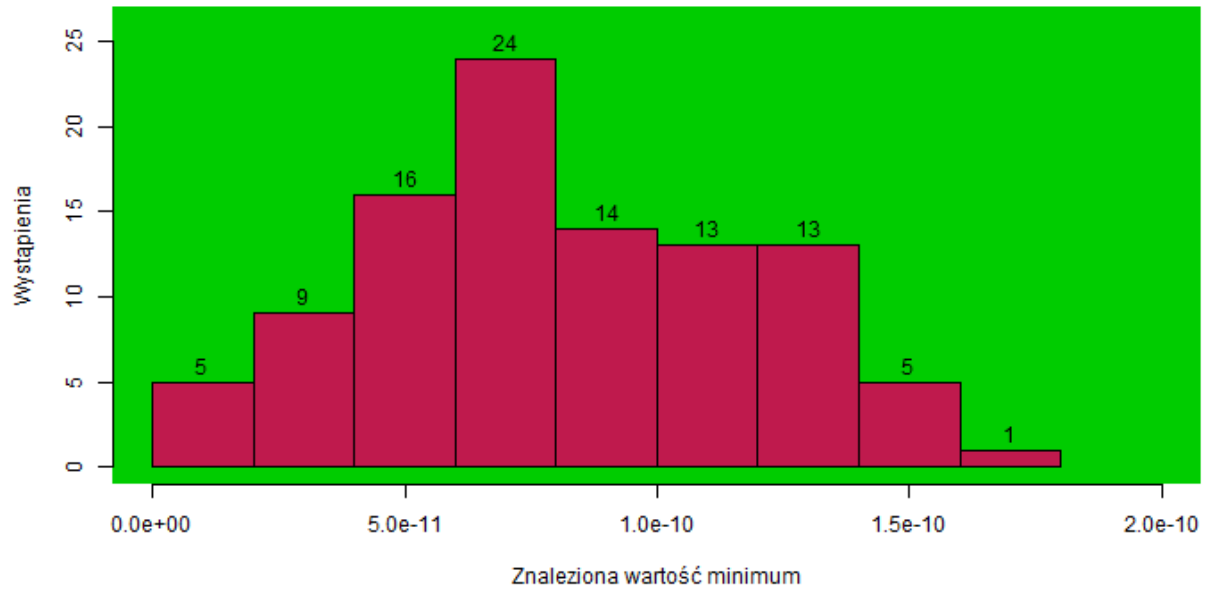


Funkcja Griewanka 2D
PRS

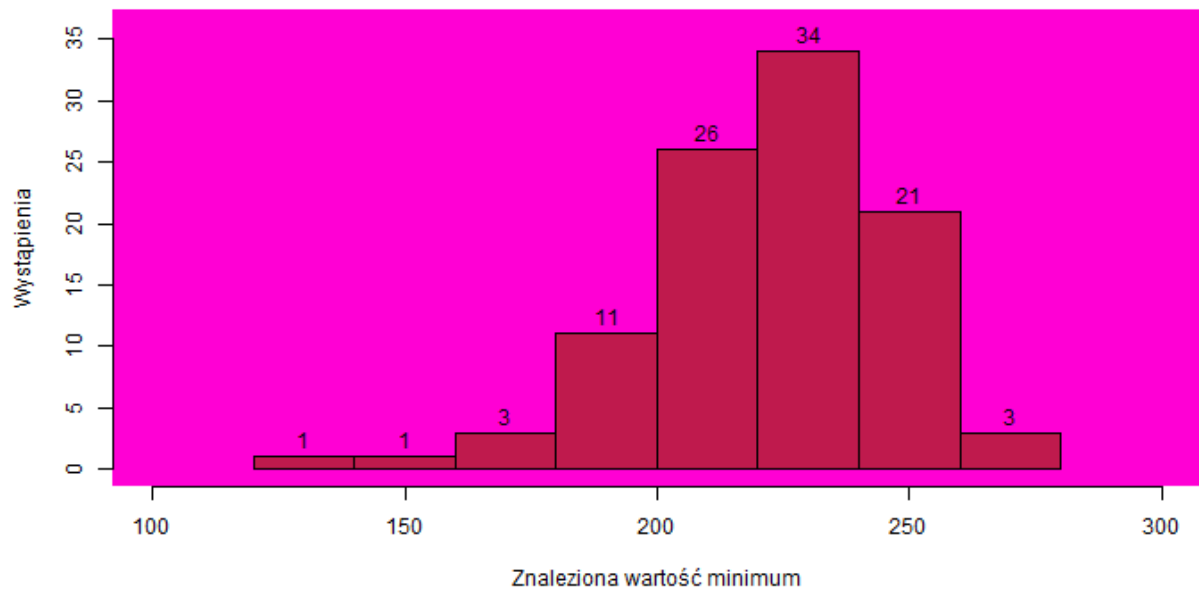




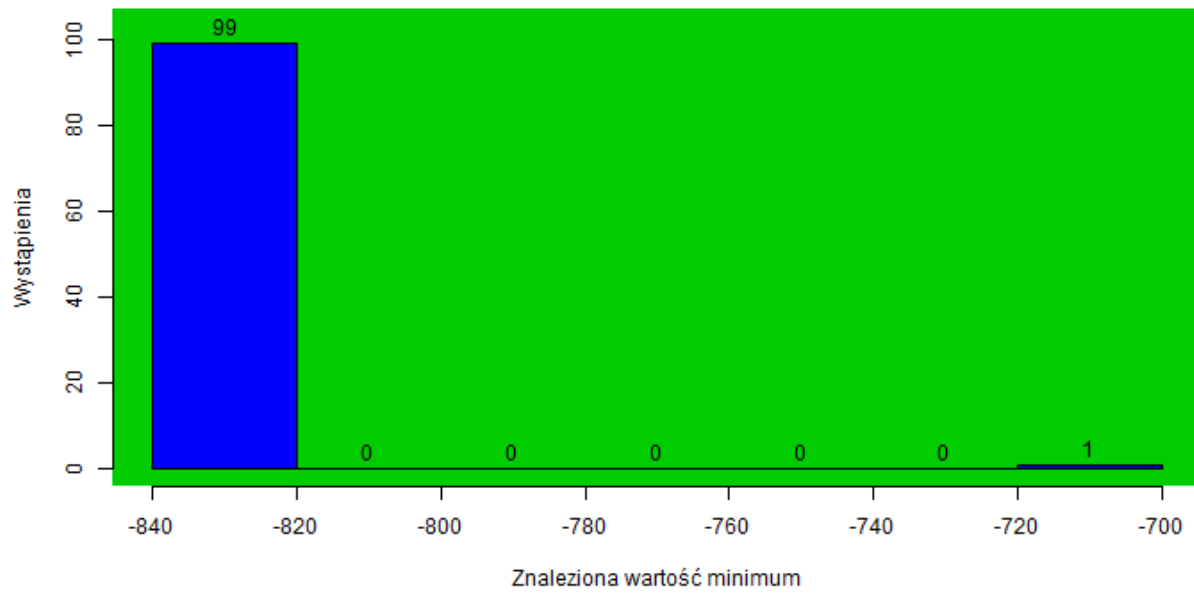
Funkcja Griewanka 20D
MS



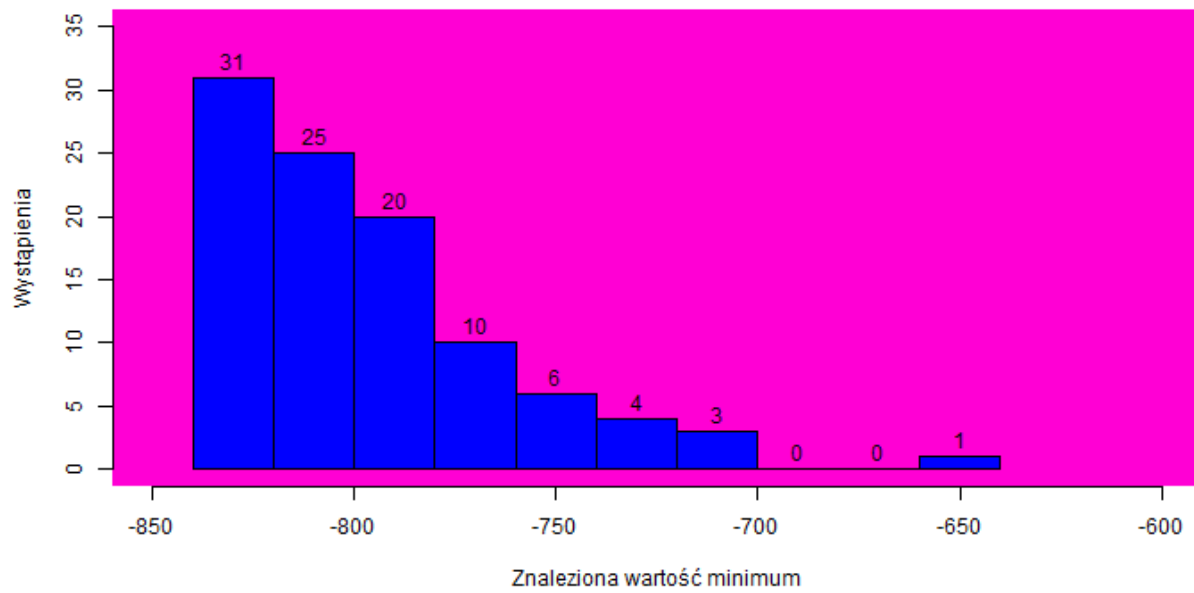
Funkcja Griewanka 20D
PRS



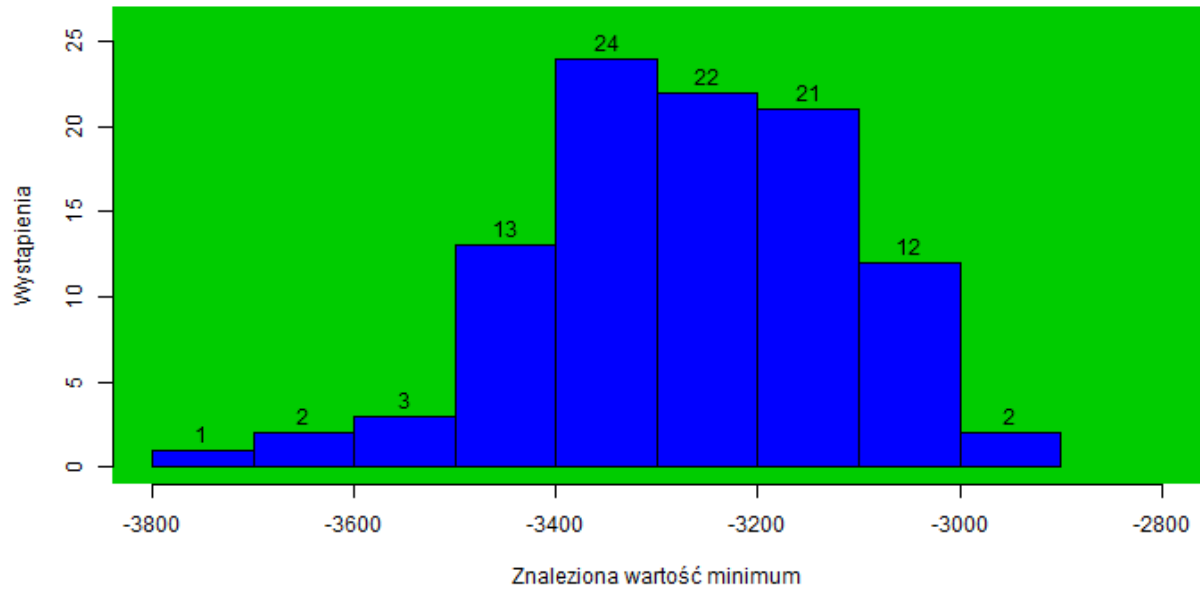
Funkcja Schwefela 2D
MS



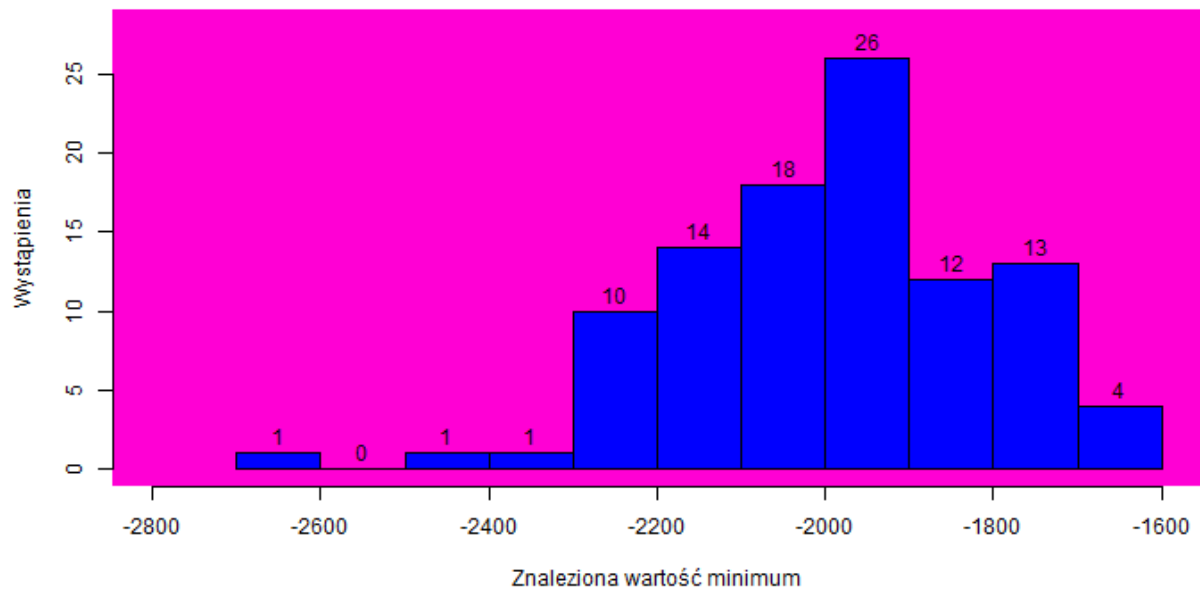
Funkcja Schwefela 2D
PRS



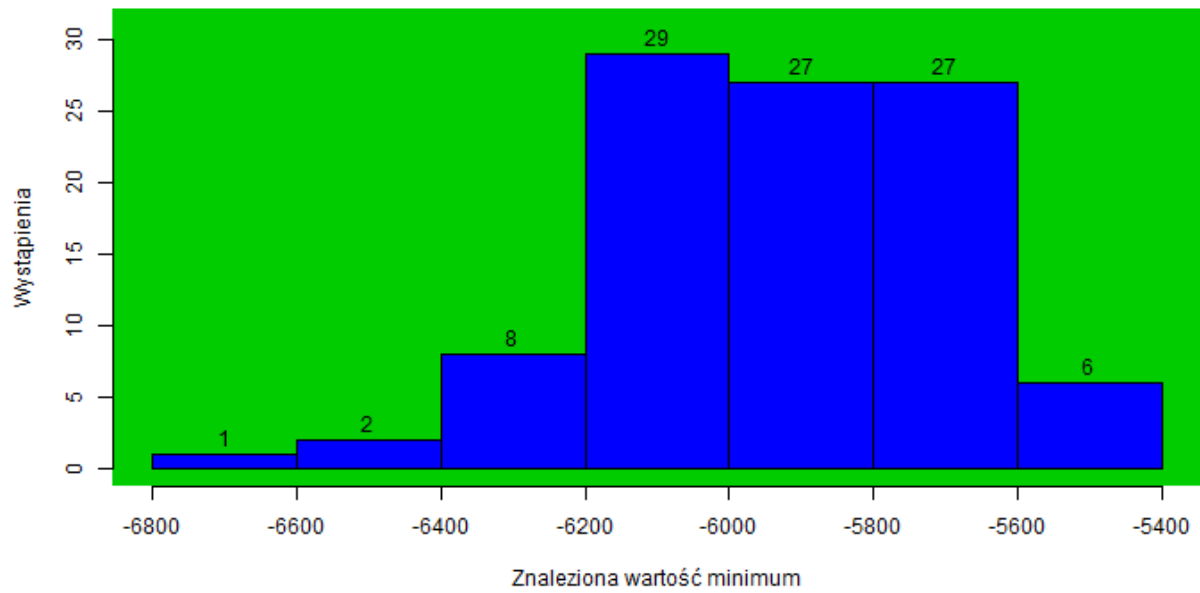
Funkcja Schwefela 10D
MS



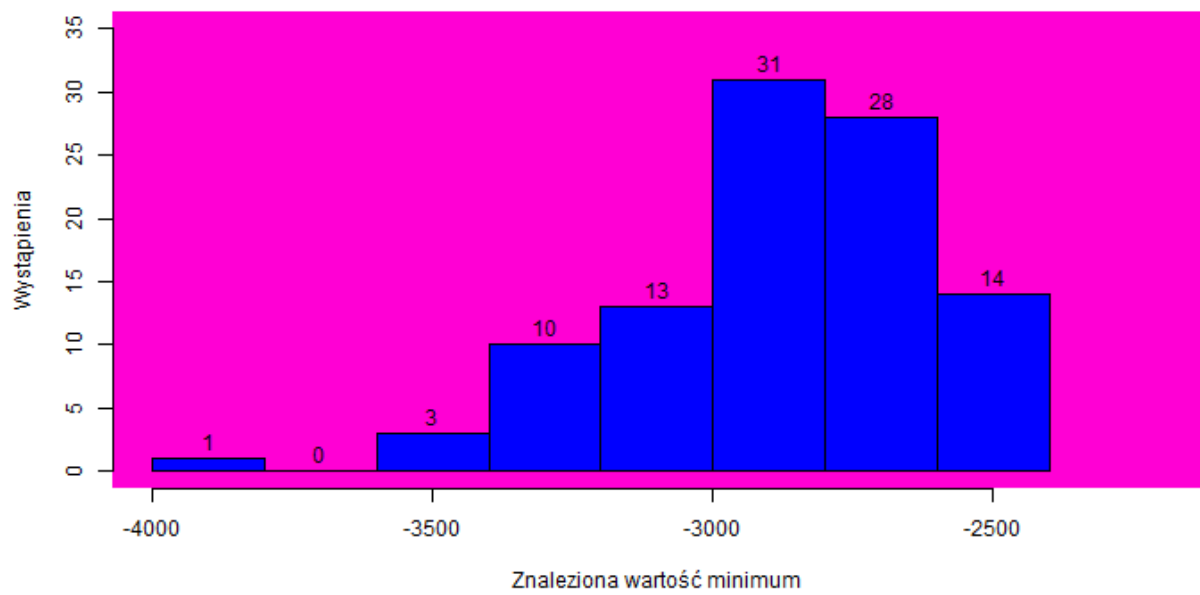
Funkcja Schwefela 10D
PRS



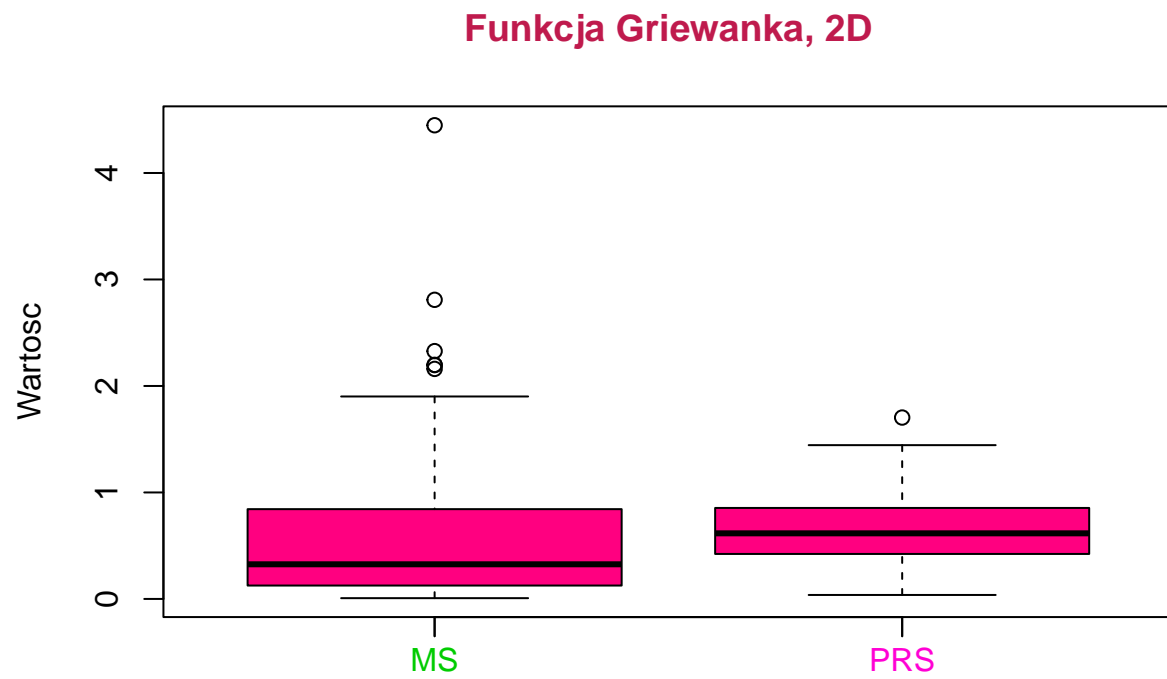
Funkcja Schwefela 20D
MS



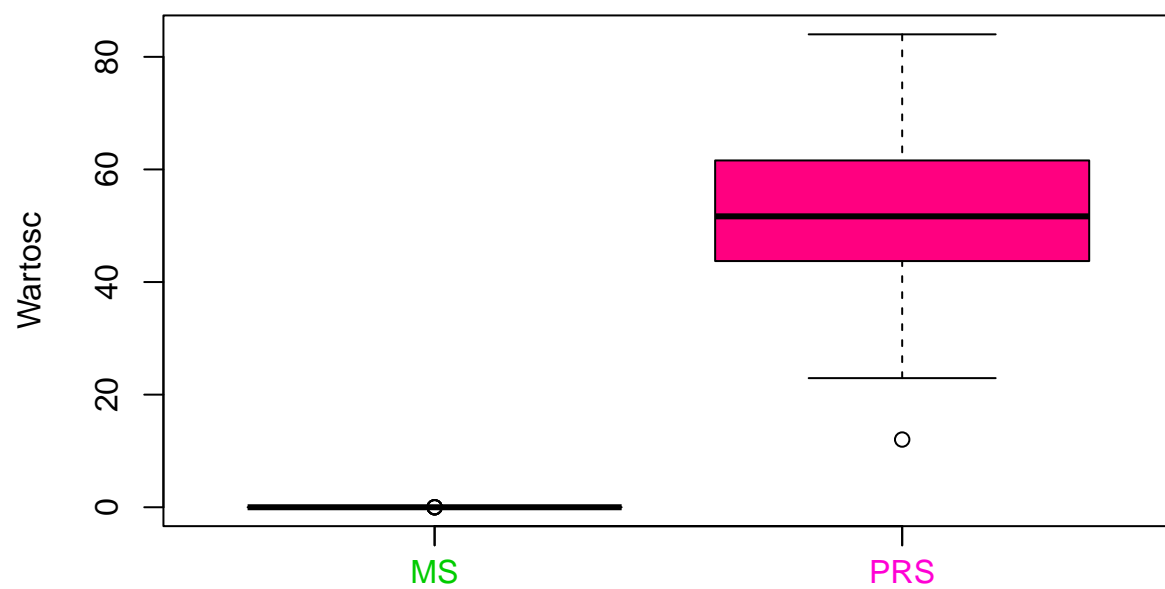
Funkcja Schwefela 20D
PRS



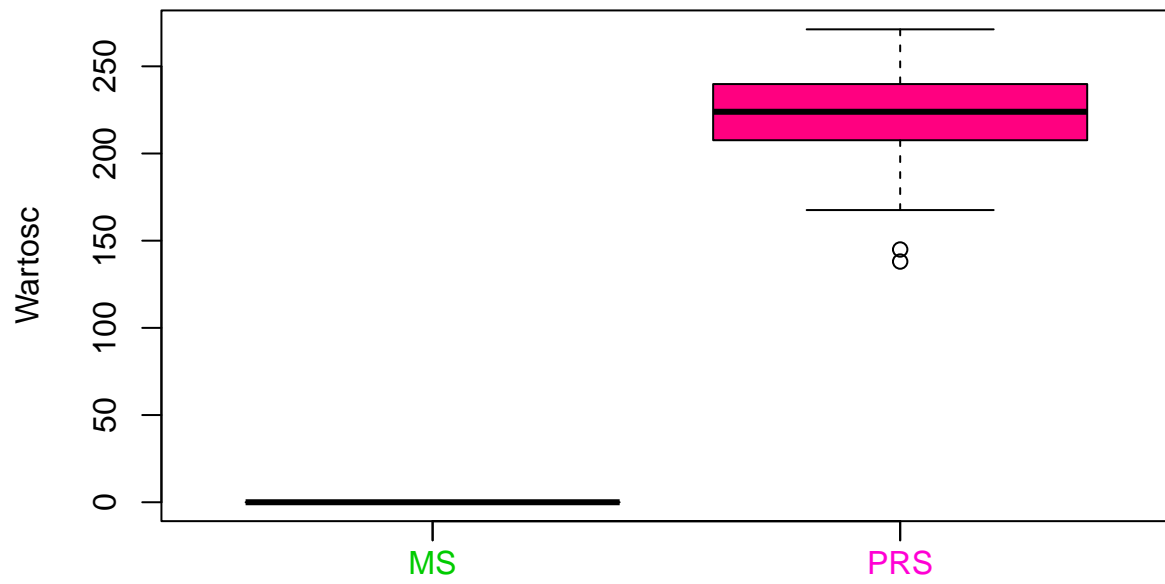
Wykresy pudełkowe



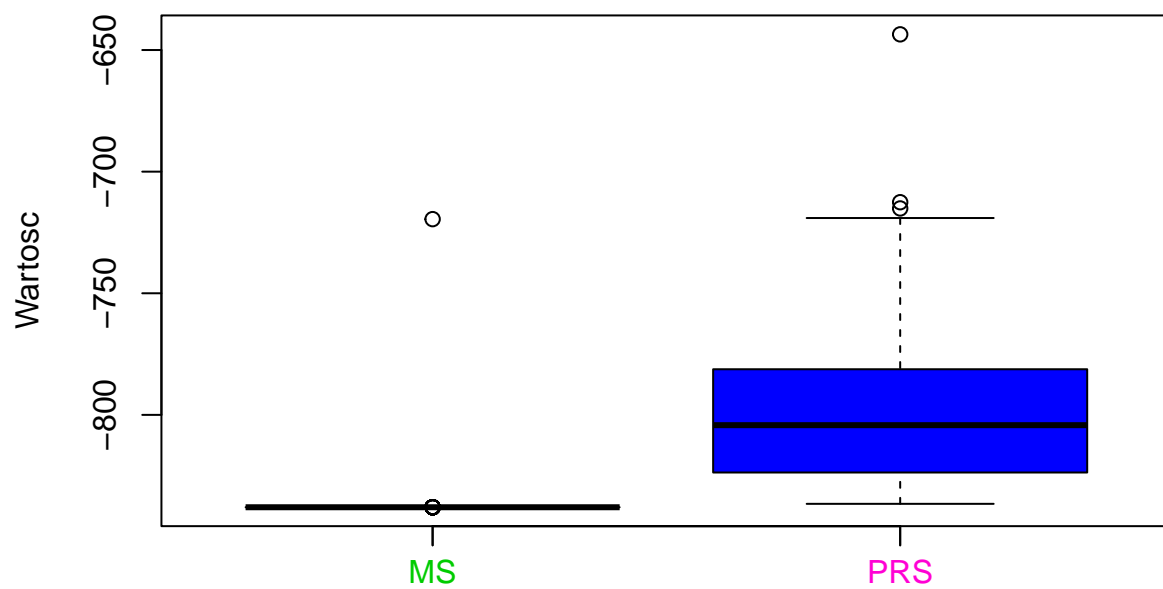
Funkcja Griewanka, 10D



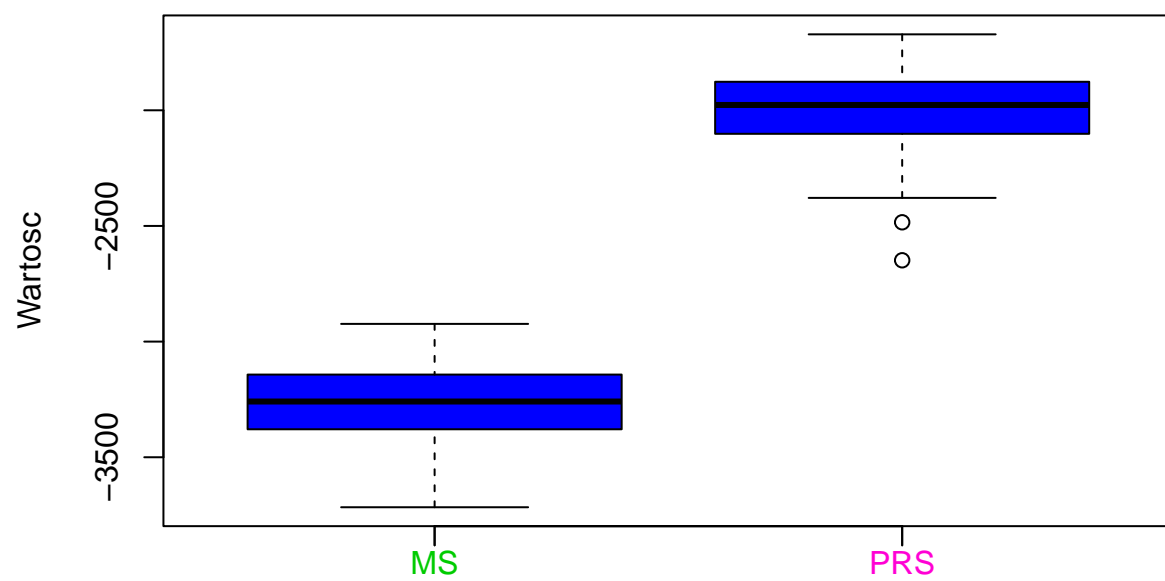
Funkcja Griewanka, 20D



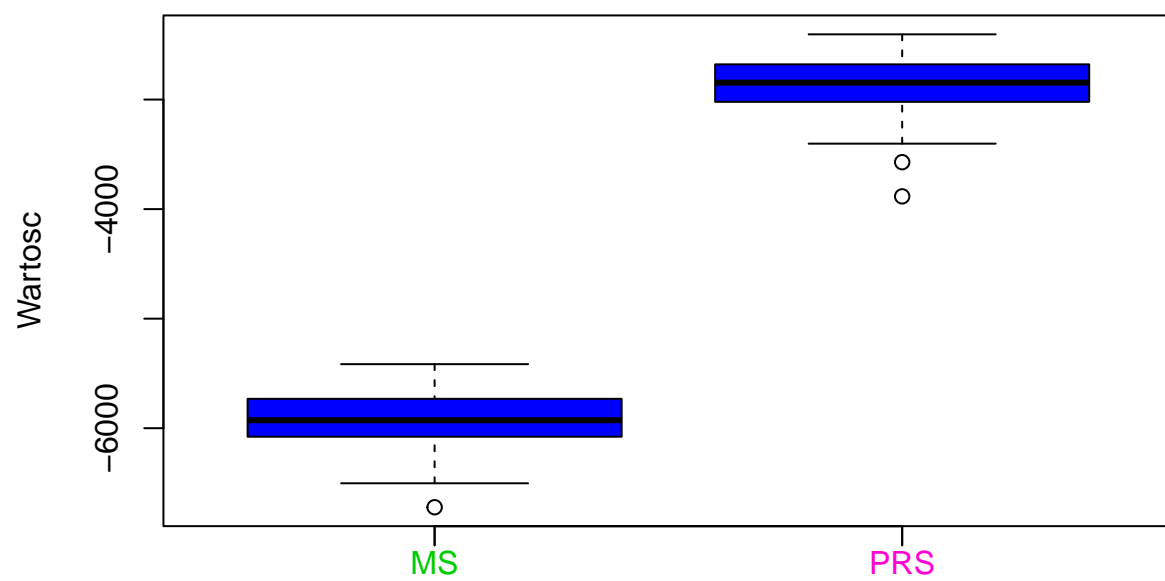
Funkcja Schwefela, 2D



Funkcja Schwefela, 10D



Funkcja Schwefela, 20D



T testy

Dla hipotezy zerowej twierdzącej, że średnie są sobie równe

Funkcja Griewanka, 2D

```
##
## Paired t-test
##
## data:  G2PRS and G2MS
## t = 0.51955, df = 99, p-value = 0.6045
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -0.1067528  0.1824889
## sample estimates:
## mean difference
##      0.03786802
```

Funkcja Griewanka, 10D

```
##
## Paired t-test
##
## data:  G10PRS and G10MS
## t = 40.75, df = 99, p-value < 2.2e-16
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  49.19135 54.22699
## sample estimates:
## mean difference
##      51.70917
```

Funkcja Griewanka, 20D

```
##
## Paired t-test
##
## data:  G20PRS and G20MS
## t = 87.939, df = 99, p-value < 2.2e-16
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  217.7479 227.8011
## sample estimates:
## mean difference
##      222.7745
```

Funkcja Schwefela, 2D

```
##
## Paired t-test
##
## data: S2PRS and S2MS
## t = 10.624, df = 99, p-value < 2.2e-16
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## 32.19550 46.98353
## sample estimates:
## mean difference
## 39.58952
```

Funkcja Schwefela, 10D

```
##
## Paired t-test
##
## data: S10PRS and S10MS
## t = 56.898, df = 99, p-value < 2.2e-16
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## 1227.543 1316.254
## sample estimates:
## mean difference
## 1271.899
```

Funkcja Schwefela, 20D

```
##
## Paired t-test
##
## data: S20PRS and S20MS
## t = 91.343, df = 99, p-value < 2.2e-16
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## 2992.623 3125.527
## sample estimates:
## mean difference
## 3059.075
```

Wnioski podsumowujące:

Funkcja Schwefela wykazuje większe różnice między algorytmami niż funkcja Griewanka. Dla funkcji Griewanka w 2D nie zaobserwowano istotnej różnicy między algorytmami, ale dla wyższych wymiarów (10D, 20D) MS osiąga znacząco lepsze wyniki niż PRS. W przypadku funkcji Schwefela, która charakteryzuje się trudnymi krajobrazami z wieloma lokalnymi minimami, metoda MS znacznie lepiej radzi sobie z eksplorowaniem przestrzeni poszukiwań i znajdowaniem głębszych minimów. Z kolei PRS wykazuje pewne trudności w bardziej złożonych przestrzeniach, co prowadzi do większych różnic w wynikach w wymiarach wyższych niż 2D.

W funkcji Schwefela dla 10D i 20D różnice między algorytmami stają się wyraźniejsze, ponieważ w tych wymiarach przestrzeń staje się bardziej złożona. Różnice w wynikach są statystycznie istotne, co wskazuje na większą efektywność MS w radzeniu sobie z trudnymi funkcjami optymalizacyjnymi w wyższych wymiarach. Wraz ze wzrostem liczby wymiarów, różnica między algorytmami staje się coraz bardziej wyraźna. Przewaga MS nad PRS rośnie wykładniczo wraz z wymiarowością. Testy statystyczne (t-test) oraz średnie różnice wskazują, że algorytmy różnią się w sposobie rozwiązywania problemu.