

Projekt TAIo 2014 - dokumentacja testów

Zespół: Jakub Gocławski, Janusz Lewandowski, Michał Wójcik

Spis treści

- [1. Testy na danych syntetycznych](#)
 - [1.1. Metodyka przeprowadzania testów](#)
 - [1.2. Domyślne parametry obowiązkowe](#)
 - [1.3. Automat deterministyczny \(etap I\)](#)
 - [1.3.1. Efektywność optymalizacji \(błąd\)](#)
 - [1.3.2. Efektywność optymalizacji \(czas\)](#)
 - [1.4. Automat niedeterministyczny \(etap II\)](#)
 - [1.4.1. Efektywność optymalizacji \(błąd\)](#)
 - [1.4.2. Efektywność optymalizacji \(czas\)](#)
 - [1.5. Automat rozmyty \(etap III\)](#)
 - [1.5.1. Efektywność optymalizacji \(błąd\)](#)
 - [1.5.2. Efektywność optymalizacji \(czas\)](#)
 - [1.6. Wyniki na wykresach](#)
- [2. Testy na danych rzeczywistych](#)
 - [2.1. Automat deterministyczny](#)
 - [2.2. Automat niedeterministyczny](#)
 - [2.3. Automat rozmyty](#)
 - [2.4. Otrzymany najlepszy klasyfikator](#)
 - [2.4.1. Wybór typu automatu](#)
 - [2.4.2. Wpływ parametrów uruchomieniowych na wyniki](#)

1. Testy na danych syntetycznych

1.1. Metodyka przeprowadzania testów

Dla każdego parametru uruchomieniowego uruchamiane były testy celem sprawdzenia jak zmiana danego parametru wpływa na efektywność znajdowania rozwiązania, zarówno biorąc pod uwagę błąd klasyfikacji jak i czas obliczeń (tworzenia klasyfikatora).

W przypadku testowania danego parametru, pozostałe parametry przyjmowały stałą wartość, nie zmieniającą się w trakcie testów. Badany parametr natomiast przyjmował kilka różnych wartości, celem znalezienia różnic w obliczeniach. Ze względu na losowy charakter algorytmu optymalizującego, dla każdej wybranej wartości badanego parametru, testy były uruchamiane 10 razy, a ich wynik został uśredniony.

1.2. Domyślne parametry obowiązkowe

W trakcie testowania wpływu danego parametru na obliczenia, przyjęto pewne domyślne wartości wymaganych parametrów. Obowiązują one we wszystkich testach, chyba, że zaznaczono inaczej lub test dotyczy jednego w obowiązkowych parametrów.

-dyskretyzacja 4 -iloscKlas 10 -iloscCech 5 -iloscPowtorzenWKlasie 50
-PSOmaxit 20 -PSOs 50

1.3. Automat deterministyczny (etap I)

1.3.1. Efektywność optymalizacji (błąd)

a) Liczba instancji treningowych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
20	45.41 %	41.66 %
50	45.16 %	46.33 %
100	39.5 %	37.5 %
150	42.75 %	45.44 %

b) Liczba klas

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
3	1.38 %	1.11 %
5	16.51 %	18.13 %
7	32.86 %	37.14 %
10	49.32 %	53.05 %

c) Dyskretyzacja

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
2	57.67 %	59.33 %
4	41.75 %	48.12 %
6	40.17 %	43.34 %
10	44.75 %	45.67 %

d) Zaszumienie losowanych danych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
0.5	41.25 %	46.33 %
1	39.58 %	37.33 %
2	45.91 %	52 %
3	50.41 %	43.33 %

e) Liczba iteracji algorytmu optymalizującego

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
20	39.16 %	41.66 %
40	41.5 %	45.66 %
60	41.75 %	47.33 %
80	39.33 %	36.33 %

f) Stopień zrównoleglenia optymalizacji

Nasze rozwiązanie nie wspiera zrównoleglenia.

g) Parametry PSO - rozmiar roju

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10	60.33 %	58 %
30	51.41 %	53.33 %
50	47.75 %	46 %
70	38.91 %	44 %

h) Błąd dla klas prawidłowych względem liczby elementów obcych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10%	53.63 %	60 %
20%	55.55 %	71.94 %
30%	56.66 %	57.17 %
40%	69.16 %	69.76 %

1.3.2. Efektywność optymalizacji (czas)

a) Liczba instancji treningowych

Wartość parametru	Średni czas obliczeń [s.]
20	2.54
50	5.81
100	11.44
150	17.07

b) Liczba klas

Wartość parametru	Średni czas obliczeń [s.]
3	3.13
5	5.10
7	7.94
10	11.81

e) Liczba iteracji algorytmu optymalizującego

Wartość parametru	Średni czas obliczeń [s.]
20	5.85
40	11.42
60	16.88
80	22.49

f) Stopień zrównoleglenia optymalizacji

Nasze rozwiązanie nie wspiera zrównoleglenia.

g) Parametry PSO - rozmiar roju

Wartość parametru	Średni czas obliczeń [s.]
10	1.18
30	3.52
50	5.92
70	8.25

1.4. Automat niedeterministyczny (etap II)

1.4.1. Efektywność optymalizacji (błąd)

a) Liczba instancji treningowych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
20	77.29 %	77.5 %
50	78.83 %	83.33 %
100	81.83 %	78.83 %
150	79.16 %	80.11 %

b) Liczba klas

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
3	-	-
5	9.16 %	17.33 %
7	15.71 %	20.95 %
10	75.91 %	78.66 %

c) Dyskretyzacja

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
2	82.16 %	82.33 %
4	84.41 %	81 %
6	78.83 %	76 %
10	79.91 %	82.33 %

d) Zaszumienie losowanych danych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
0.5	80.66 %	81.66 %
1	80.66 %	80 %
2	82.75 %	84.66 %
3	79.66 %	84.33 %

e) Liczba iteracji algorytmu optymalizującego

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
20	79.5 %	79.66 %
40	83.41 %	87.0 %
60	77.5 %	78.66 %
80	81.08 %	82.33 %

f) Stopień zrównoleglenia optymalizacji

Nasze rozwiązanie nie wspiera zrównoleglenia.

g) Parametry PSO - rozmiar roju

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10	80.91 %	82.33 %
30	76.91 %	82.66 %
50	81.0 %	84.0 %
70	80.58 %	80.33 %

h) Błąd dla klas prawidłowych względem liczby elementów obcych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10%	81.74 %	83.03 %
20%	84.93 %	82.77 %
30%	85 %	86.15 %
40%	84.64 %	82.14 %

i) Ograniczenie niedeterminizmu

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10	42.41 %	46.33 %
15	44.08 %	44.0 %
20	79 %	80.33 %
25	79.33 %	79.33 %

1.4.2. Efektywność optymalizacji (czas)

a) Liczba instancji treningowych

Wartość parametru	Średni czas obliczeń [s.]
20	8.06
50	19.28
100	38.89
150	56.92

b) Liczba klas

Wartość parametru	Średni czas obliczeń [s.]
3	-
5	7.99
7	11.81
10	18.66

e) Liczba iteracji algorytmu optymalizującego

Wartość parametru	Średni czas obliczeń [s.]
20	18.55
40	35.99
60	54
80	89.96

f) Stopień zrównoleglenia optymalizacji

Nasze rozwiązanie nie wspiera zrównoleglenia.

g) Parametry PSO - rozmiar roju

Wartość parametru	Średni czas obliczeń [s.]
10	3.73
30	11.09
50	18.57
70	32.37

1.5. Automat rozmyty (etap III)

1.5.1. Efektywność optymalizacji (błąd)

a) Liczba instancji treningowych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
20	46.04 %	45.83 %
50	53.08 %	59 %
100	56.91 %	59.33 %
150	47.30 %	46.55 %

b) Liczba klas

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
3	5.55 %	11.11 %
5	19.5 %	13.33 %
7	27.26 %	29.04 %
10	47.5 %	47 %

c) Dyskretyzacja

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
2	71.41 %	72 %
4	41.83 %	52.33 %
6	57.33 %	59.33 %
10	45.91 %	40.33 %

d) Zaszumienie losowanych danych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
0.5	50.58 %	45 %
1	49.25 %	51 %
2	63 %	61.66 %
3	61.66 %	69 %

e) Liczba iteracji algorytmu optymalizującego

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
20	61.66 %	58.0 %
40	42.75 %	42.66 %
60	49.58 %	48.66 %
80	46.33 %	50.66 %

f) Stopień zrównoleglenia optymalizacji

Nasze rozwiązanie nie wspiera zrównoleglenia.

g) Parametry PSO - rozmiar roju

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10	77.66 %	78.66 %
30	54.58 %	59.0 %
50	50.33 %	56.66 %
70	53.66 %	52.33 %

h) Błąd dla klas prawidłowych względem liczby elementów obcych

Wartość parametru	Średni błąd obliczeń	
	Zbiór uczący	Zbiór testowy
10%	58.63 %	68.18 %
20%	55.90 %	60.27 %
30%	60.64 %	60.76 %
40%	59.94 %	61.19 %

1.5.2. Efektywność optymalizacji (czas)

a) Liczba instancji treningowych

Wartość parametru	Średni czas obliczeń [s.]
20	17.42
50	43.13
100	85.6
150	127.82

b) Liczba klas

Wartość parametru	Średni czas obliczeń [s.]
3	11.54
5	19.71
7	28.52
10	42.38

e) Liczba iteracji algorytmu optymalizującego

Wartość parametru	Średni czas obliczeń [s.]
20	42.99
40	85.41
60	127.38
80	167.77

f) Stopień zrównoleglenia optymalizacji

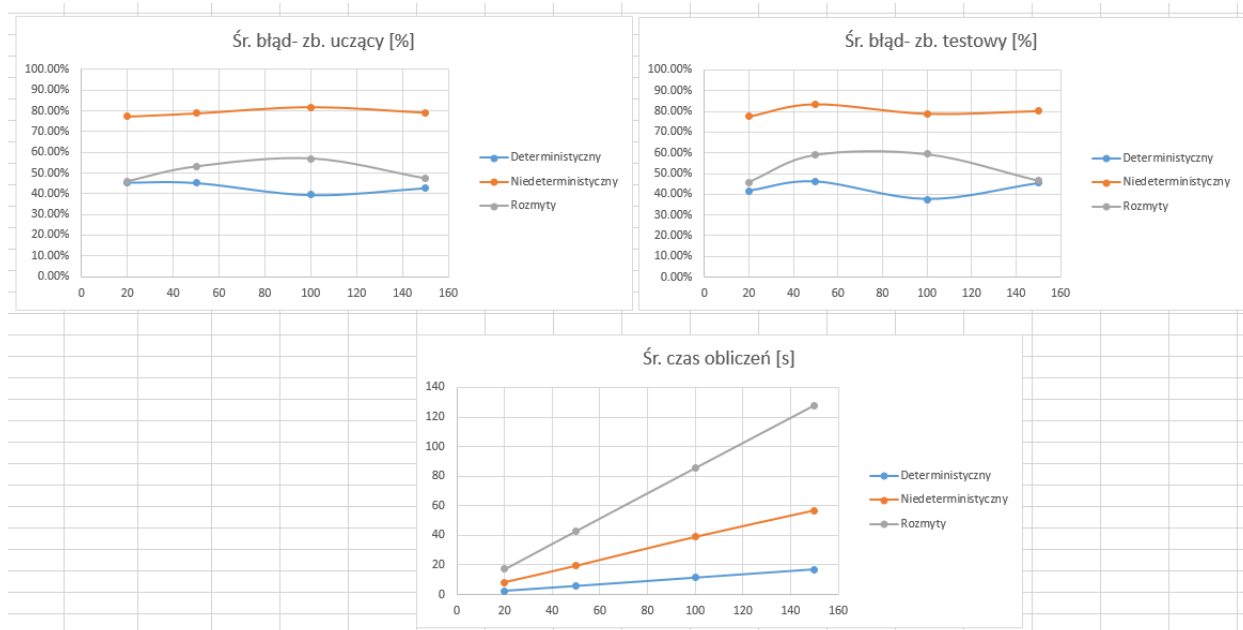
Nasze rozwiązanie nie wspiera zrównoleglenia.

g) Parametry PSO - rozmiar roju

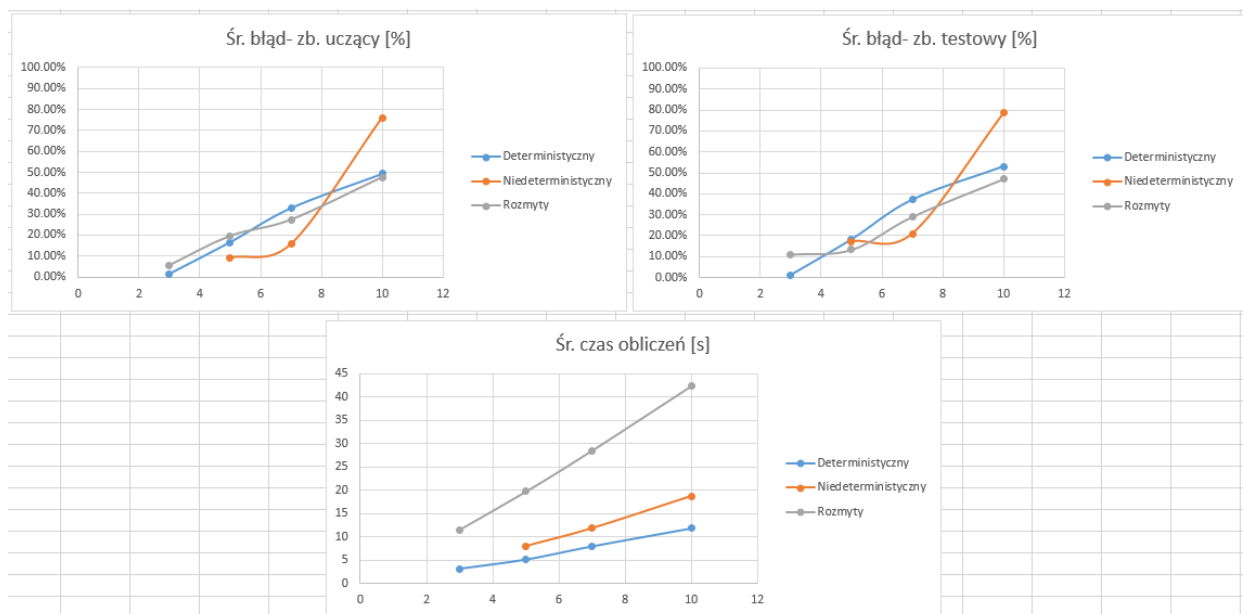
Wartość parametru	Średni czas obliczeń [s.]
10	9.05
30	26.93
50	43.98
70	62.31

1.6. Wyniki na wykresach

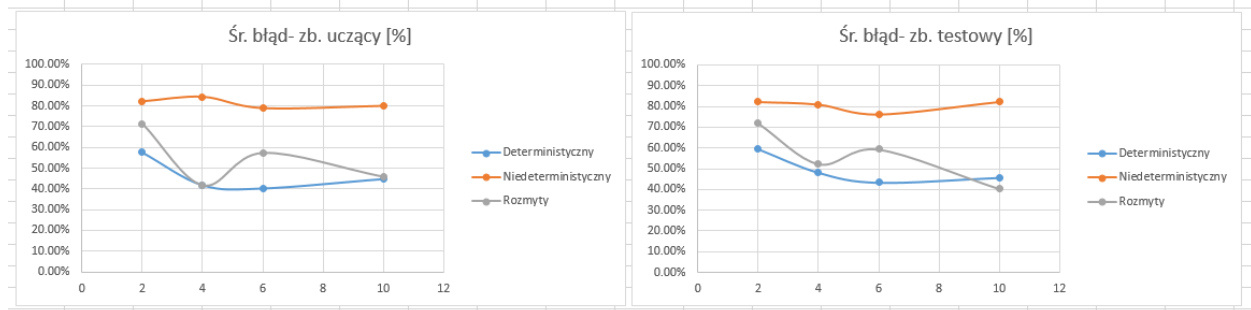
a) Liczba instancji treningowych



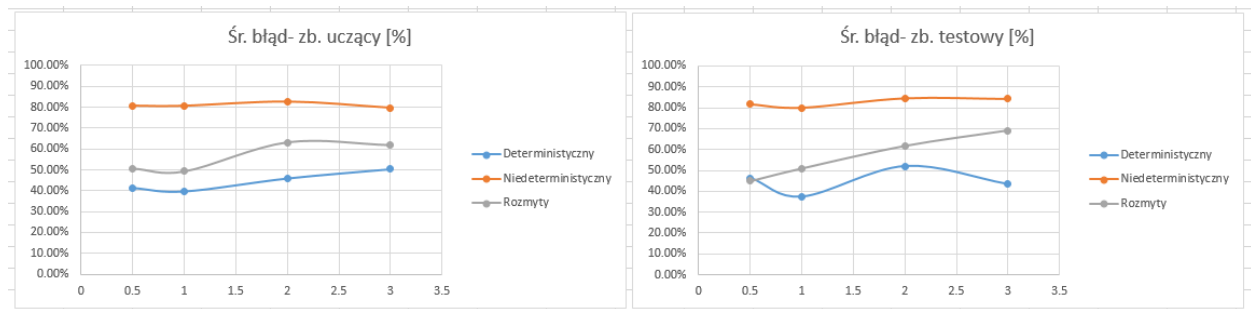
b) Liczba klas



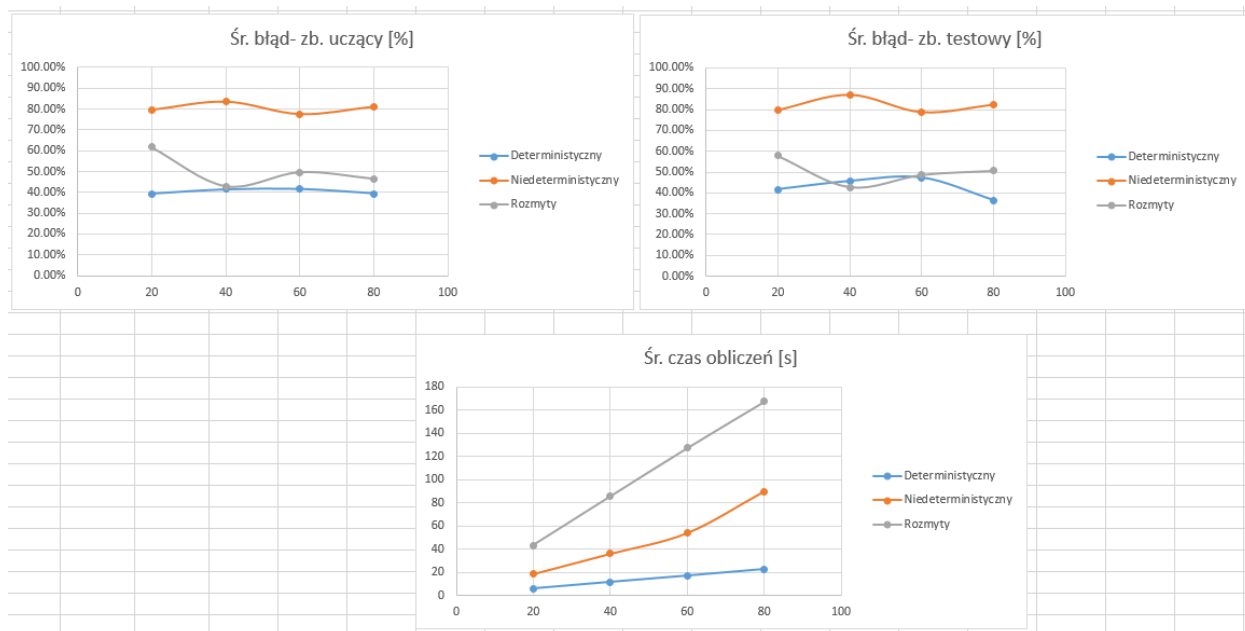
c) Dyskretyzacja



d) Zaszumienie losowanych danych



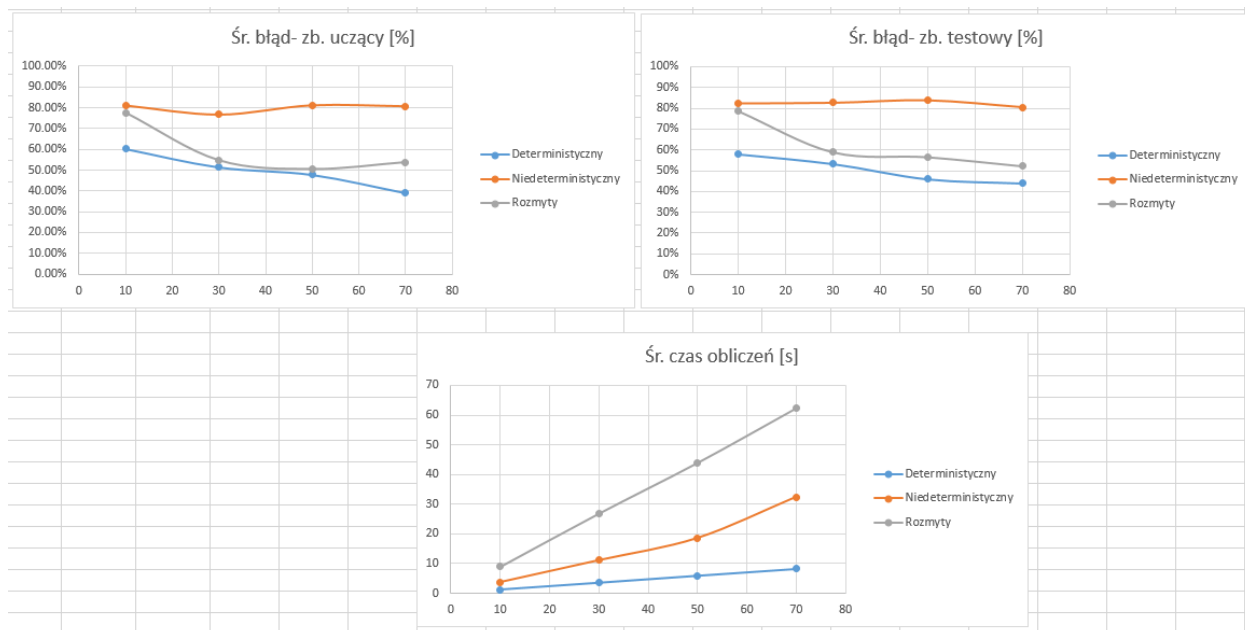
e) Liczba iteracji algorytmu optymalizującego



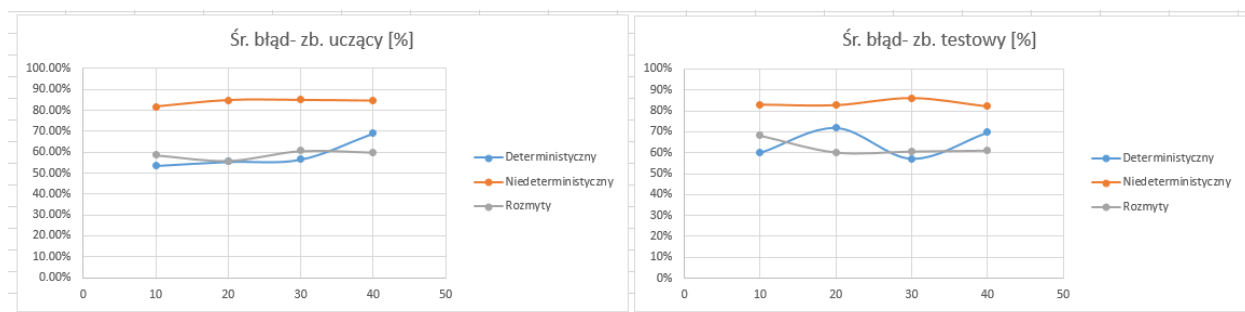
f) Stopień zrównoleglenia optymalizacji

Nasze rozwiązanie nie wspiera zrównoleglenia.

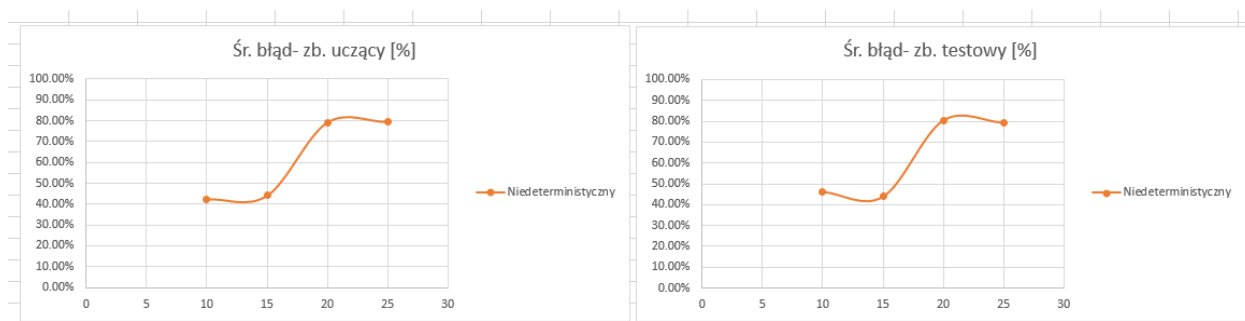
g) Parametry PSO - rozmiar roju



h) Błąd dla klas prawidłowych względem liczby elementów obcych



i) Ograniczenie niedeterminizmu



2. Testy na danych rzeczywistych

Stosowane ustawienia programu:

- dyskretyzacja 10
- procRozmTest 20
- ograniczNietermin 20 (tylko dla niedeterministycznego)
- PSO maxit 20
- PSO s 50

2.1. Automat deterministyczny

Zbiór uczący (rozmiar = 8000)

Liczba błędnych przyporządkowań: 6606 (82.575 %)

Zbiór testowy (rozmiar = 2000)

Liczba błędnych przyporządkowań: 1686 (84.3 %)

Czas działania: 58m

2.2. Automat niedeterministyczny

Zbiór uczący (rozmiar = 8000)

Liczba błędnych przyporządkowań: 6732 (84.15 %)

Zbiór testowy (rozmiar = 2000)

Liczba błędnych przyporządkowań: 1724 (86.2 %)

Czas działania: 3h 45m

2.3. Automat rozmyty

Zbiór uczący (rozmiar = 8000)

Liczba błędnych przyporządkowań: 5636 (70.45 %)

Zbiór testowy (rozmiar = 2000)

Liczba błędnych przyporządkowań: 1454 (72.7 %)

Czas działania: 51h 30m

2.4. Otrzymany najlepszy klasyfikator

2.4.1. Wybór typu automatu

Automat niedeterministyczny

Wygląda na najmniej skuteczny typ automatu, z czasem działania plasującym się zazwyczaj pomiędzy automatem deterministycznym oraz rozmytym. Większą skuteczność uzyskać można ograniczając niedeterminizm, co sugeruje, że w ogóle lepszym typem automatu jest automat deterministyczny.

Automat rozmyty

Cechuje się całkiem dobrymi wynikami, jednak kosztem znacząco dłuższego czasu wykonywania obliczeń, który rośnie szybko wraz ze zwiększaniem dokładności algorytmu za pomocą parametrów uruchomieniowych. Należy zauważyć, że wyniki uzyskane za pomocą tego typu automatu mają większą wartość, gdyż można z ich pomocą stosować wspomaganie wyników mechanizmu OCR metodą słownikową.

Warte odnotowania jest również to, iż ten typ automatu uzyskał najlepsze wyniki przy danych rzeczywistych.

Automat deterministyczny

Wydaje się najrozsądniejszym kompromisem pomiędzy skutecznością a czasem wykonywania obliczeń. Jego wyniki są porównywalne z wynikami automatu rozmytego (zazwyczaj lepsze dla danych syntetycznych), jednak czas obliczeń jest znacząco krótszy.

Podsumowanie

Trudno jest jednoznacznie wybrać najlepszy typ automatu. Jednak wydaje się, iż jest to automat deterministyczny. Jednakże warto zaznaczyć, że w pewnych przypadkach, np. gdy czas obliczeń nie jest aż tak istotny, rozsądnym rozwiązaniem jest również automat rozmyty. Przykładowo: dla danych rzeczywistych, automat rozmyty dał zauważalnie lepsze wyniki (72,7 %) niż automat deterministyczny (84,3 %). Jednak automat deterministyczny potrzebował jedynie 58 minut na znalezienie rozwiązania, podczas gdy automat rozmyty aż 51 godzin i 30 minut.

2.4.2. Wpływ parametrów uruchomieniowych na wyniki

Liczba instancji treningowych: oczywiście im większa tym wyniki są lepsze, ale rośnie również czas wykonania programu. Sugerowana wartość: w okolicy 100.

Liczba klas: im mniejsza, tym problem jest łatwiejszy.

Dyskretyzacja: z testów wynika, że im większa, tym lepsze wyniki. Sugerowana wartość: ok. 6.

Zaszumienie danych: z testów wynika, że nie wpływa znacząco na wyniki, choć lepsze są niższe wartości. Oczywiście zbyt wysoka wartość spowoduje wygenerowania trudnego zadania.

Liczba iteracji: nieznacznie poprawia wyniki, znacznie wydłuża czas działania. Sugerowana wartość: ok. 80.

Rozmiar roju PSO: nieznacznie poprawia wyniki, wydłuża czas działania. Wartość ta w ogólności powinna być proporcjonalna do rozmiaru problemu. W naszym przypadku wartość 40 stanowi rozsądny kompromis między skutecznością, a szybkością działania.

Liczba elementów obcych: powoduje utrudnienie zadania, jednak w ramach testowanych wartości, nie miała znaczącego wpływu na jakość wyników.

Ograniczenie niedeterminizmu: im mniejsza wartość, tym lepsze wyniki, co sugeruje, że automat deterministyczny jest lepszy niż niedeterministyczny.