Raport - Zaawansowane metody klasyfikacji oraz analiza skupień – algorytmy grupujące i hierarchiczne

Filip Michewicz 282239
Wiktor Niedźwiedzki 258882

18 czerwca 2025 Anno Domini

Spis treści

Spis wykresów

Spis tabel

1 Zaawansowane metody klasyfikacji

W pierwszej części zadania zastosujemy algorytmy *ensemble learning* (bagging, boosting i random forest) w celu poprawy dokładności cech klasyfikacyjnych. W drugiej natomiast poznamy i ocenimy nową metodę klasyfikacji - metodę wektorów nośnych (SVM).

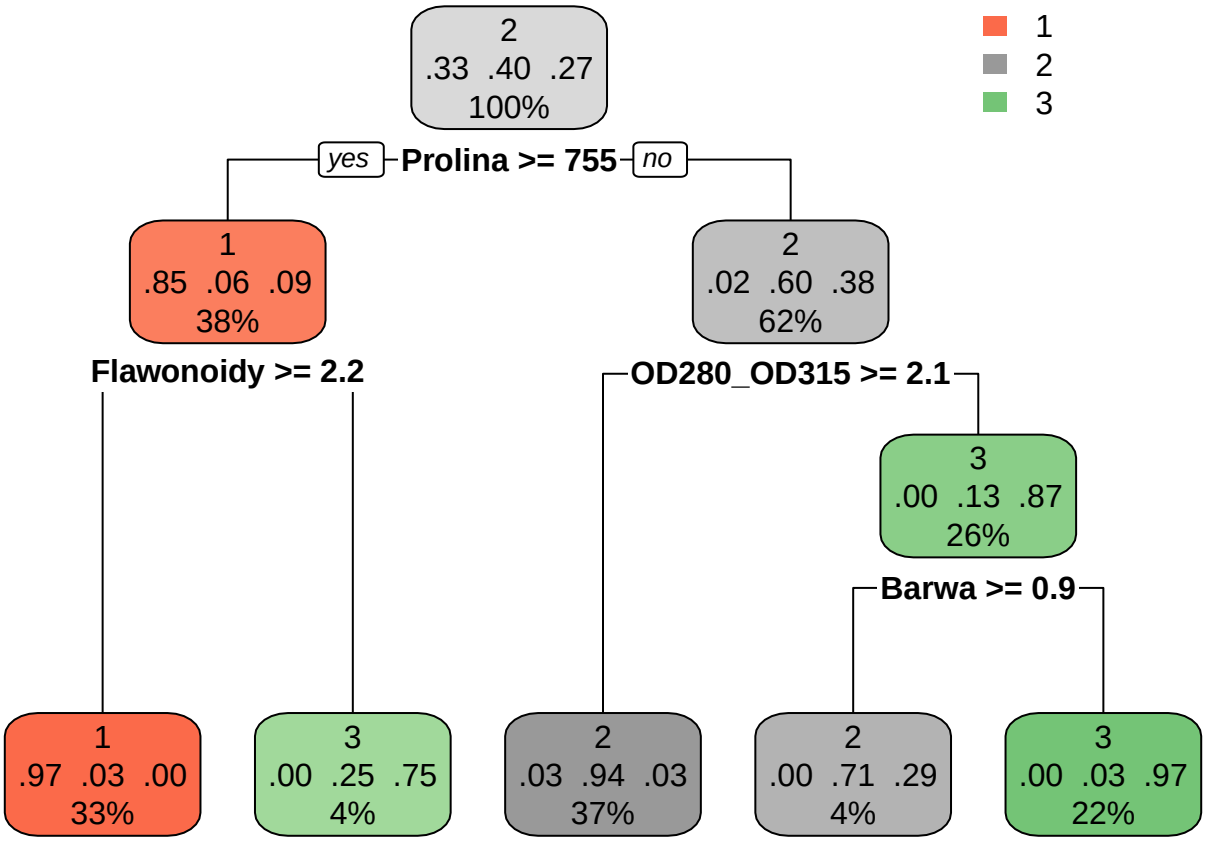
Zadanie zostanie wykonane na zbiorze danych *wine*, którego szczegółowy opis znajduje się w poprzednim raporcie.

1.1 Rodziny klasyfikatorów/uczenie zespołowe

Wyróżniamy trzy algorytmy uczenia zespołowego (ang. ensemble learning):

- **Bagging** - generujemy B-bootstrapowych replikacji zbioru uczącego, na podstawie których tworzymy B klasyfikatorów. Następnie łączymy je w klasyfikator zagregowany, który przydziela dane cechy do klas za pomocą reguły "głosowania większości" (w przypadku remisu wybiera losowo). Każdy klasyfikator powstaje niezależnie (w sensie takim, że wyniki poprzednich nie mają wpływu na generowanie nowych).
- **Boosting** - podobnie jak w bagging, tworzymy klasyfikator zagregowany złożony z wielu pojedynczych klasyfikatorów. Jednak różnica jest taka, że klasyfikatory powstają sekwencyjnie. Na początku każda cecha w zbiorze ma przypisaną taką samą wagę. Z każdą kolejną iteracją natomiast waga zwiększa się dla uprzednio źle sklasyfikowanych przypadków.
- **Random forest** (dla drzew klasyfikacyjnych) - metoda podobna do bagging z tą różnicą, że klasyfikatory powstają na podstawie różnych m-elementowych podzbiorach cech (m mniejsze bądź równe wszystkim cechom).

Na wykresie przedstawiono pojedyncze drzewo klasyfikacyjne.



Wykres 1: Pojedyńcze drzewo klasyfikacyjne

Błąd estymowany metodą bootstrap .632+ wynosi 9.1%. Na jego podstawie określimy poprawę modelu po zastosowaniu metod uczenia zespołowego. Poprawę błędu klasyfikacji będziemy definiować jako stosunek różnicy błędu osiągniętego na jednym drzewie oraz błędu po zastosowaniu danej metody, do błędu na pojedynczym drzewie właśnie.

W tej części analizy do oceny wydajności modelu zostanie wykorzystana wyłącznie metoda .632+, ponieważ koryguje ona obciążenie estymatora (bias) i dostarcza bardziej wiarygodnej oceny błędu generalizacji, zwłaszcza przy ryzyku przeuczenia i ograniczonej liczbie próbek.

Tabela 1: Poprawa dokładności klasyfikacji za pomocą drzewa klasyfikacyjnego, z podziałem na algorytmy uczenia zespołowego oraz liczbę replikacji

Algorytm uczenia zespołowego	Liczba replikacji							
	1	5	10	20	30	40	50	100
Bagging	41.52	25.72	42.43	63.67	56.94	63.24	65.60	65.03
Random Forest	87.65	83.70	86.83	89.45	82.73	88.72	82.30	84.78
Boosting	72.95	67.31	67.28	73.75	63.21	67.37	70.96	71.30

Z Tabeli 1. wynika, że wszystkie rozważane metody — las losowy, boosting oraz bagging — znacząco poprawiły błąd klasyfikacji względem samego drzewa, osiągając redukcję błędu przekraczającą 60%. Świadczy to o wysokiej skuteczności zastosowanych technik ensemble do zwiększenia skuteczności pojedynczego drzewa klasyfikacyjnego.

1.2 Metoda wektorów nośnych (SVM)

W tej części przeprowadzona będzie klasyfikacja na podstawie metody wektorów nośnych, z podziałem na różne funkcje jądrowe.

Metoda SVM jest jedną z najczęściej stosowanych technik uczenia maszynowego w zadaniach klasyfikacyjnych. Jej podstawowym celem jest wyznaczenie hiperpłaszczyzny maksymalnie oddzielającej obserwacje należące do różnych klas, przy jednoczesnym maksymalizowaniu marginesu między klasami.

Dzięki zastosowaniu funkcji jądrowych (kernel functions), SVM umożliwia również skuteczną klasyfikację danych nieliniowo separowalnych poprzez odwzorowanie ich do przestrzeni o wyższej liczbie wymiarów. W niniejszej analizie zostaną porównane różne funkcje jądrowe, w tym liniowa, wielomianowa oraz radialna (RBF), w kontekście ich wpływu na jakość klasyfikacji.

1.2.1 Jądro liniowe

Przeanalizowano skuteczność klasyfikacyjną z zastosowaniem **jądra liniowego**, zarówno **bez skalowania danych**, jak i **po ich skalowaniu**.

Porównanie wyników dla obu wariantów (ze skalowaniem i bez) pozwala ocenić wpływ przeskalowania zmiennych na jakość klasyfikacji. Ponieważ SVM opiera się na obliczeniach odległości i iloczynów skalarnych, skalowanie danych może znacząco wpłynąć na działanie algorytmu, szczególnie gdy zmienne wejściowe różnią się skalą lub jednostką.

Za każdym razem skuteczność modelu oceniano trzema metodami: **wielokrotnego podziału**, **bootstrapu** oraz **kroswalidacji**. W pierwszych dwóch przypadkach stosowano podział danych w stosunku **2:1** (czyli 2/3 zbioru do nauki, 1/3 do testowania), natomiast **kroswalidację** przeprowadzono z użyciem **10 zbiorów** (10-fold), przy czym model trenowano na 9 częściach, a testowano na jednej. Wszystkie podawane wartości dokładności odnoszą się do skuteczności modelu na zbiorze testowym.

Tabela 2: Jądro liniowe - bez skalowania

Metoda	Współczynnik kary - C						
	0.001	0.01	0.1	1	10	100	1000
Wielokrotny podział	40.00	98.17	97.00	95.50	96.50	96.33	95.67
Cross-validation	45.08	57.14	92.88	93.63	93.63	93.63	93.63
Bootstrap	37.37	95.86	97.21	97.47	95.47	96.75	96.23
Średnio	40.82	83.72	95.70	95.53	95.20	95.57	95.18

Tabela 3: Jądro liniowe - ze skalowaniem

Metoda	Współczynnik kary - C						
	0.001	0.01	0.1	1	10	100	1000
Wielokrotny podział	38.50	97.50	97.17	96.50	96.00	96.50	96.00
Cross-validation	32.71	50.63	93.26	93.07	93.07	93.07	93.07
Bootstrap	38.65	96.34	97.28	95.46	97.25	96.45	96.80
Średnio	36.62	81.49	95.90	95.01	95.44	95.34	95.29

Z analizy Tabeli 2. oraz Tabeli 3. wynika, że optymalną wartością współczynnika kary - **C** w klasyfikacji SVM z jądrem liniowym jest **0.1**. Dla tej wartości obserwuje się najwyższą średnią skuteczność klasyfikacji zarówno bez skalowania, jak i po skalowaniu danych. Ponadto, zastosowanie skalowania cech nieznacznie poprawia wyniki, co wskazuje na korzystny wpływ normalizacji na efektywność modelu.

Wyższe wartości parametru **C** nie przekładają się na istotną poprawę skuteczności klasyfikacji, a w niektórych przypadkach powodują nawet jej nieznaczny spadek. Wynika to z faktu, że zbyt duża wartość **C** powoduje nadmierne dopasowanie modelu do danych treningowych (overfitting). W konsekwencji, mimo że model stara się minimalizować błędy na zbiorze treningowym, jego efektywność na danych testowych nie ulega poprawie, co potwierdzają uzyskane wyniki.

1.2.2 Jądro wielomianowe

Przeanalizowano skuteczność klasyfikacyjną z zastosowaniem **jądra wielomianowego**, zarówno **bez skalowania danych**, jak i **po ich skalowaniu**.

Podobnie jak w przypadku jądra liniowego, porównanie wyników dla obu wariantów pozwala ocenić wpływ przeskalowania zmiennych na jakość klasyfikacji. W przypadku jądra wielomianowego, oprócz parametru **C**, uwzględniany jest także dodatkowy parametr **gamma**, który wpływa na działanie funkcji jądrowej i może modyfikować złożoność granicy decyzyjnej. Skalowanie cech ma zatem istotne znaczenie również ze względu na większą liczbę parametrów wrażliwych na różnice w skali danych.

Tabela 4: Jądro wielomianowe - wielokrotny podział, bez skalowania

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	36.33	41.33	95.67	95.00	95.00	95.50	94.33	94.00	95.83	95.83	95.33	96.83
0.01	39.67	38.00	94.50	95.17	93.67	95.17	95.33	96.33	94.67	95.17	92.50	95.50
0.1	38.33	84.17	96.33	94.83	94.50	96.00	95.50	94.67	97.33	94.17	95.17	95.83
1	40.00	95.33	94.67	95.67	95.17	95.67	94.83	94.50	94.33	95.17	96.00	94.83
10	41.17	96.50	95.67	95.83	96.17	96.17	96.50	95.67	96.83	96.00	94.83	94.33

Tabela 4: Jądro wielomianowe - wielokrotny podział, bez skalowania (kontynuacja)

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
100	81.50	95.67	96.00	95.17	96.00	95.00	95.33	94.33	93.67	95.67	95.83	95.83
1000	96.50	94.17	96.33	94.50	96.00	96.17	96.50	96.33	96.17	96.50	95.33	95.50

Tabela 5: Jądro wielomianowe - wielokrotny podział, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	38.83	41.50	95.33	93.50	95.67	95.50	94.33	96.50	95.00	95.50	94.83	96.17
0.01	38.50	42.50	95.17	95.50	94.50	96.17	95.67	95.67	95.17	96.17	95.33	96.50
0.1	33.50	88.17	95.33	96.00	95.83	96.00	95.33	96.33	95.83	95.17	95.50	95.17
1	39.17	95.17	94.67	94.83	95.17	95.17	94.67	95.67	94.33	96.17	94.50	94.00
10	44.67	95.17	96.00	96.00	94.00	94.17	95.83	95.00	95.17	96.00	95.00	93.83
100	81.33	95.50	94.50	96.17	94.17	93.67	95.83	95.33	96.50	95.00	96.17	95.33
1000	96.00	94.17	95.33	96.17	95.17	96.33	95.50	94.33	94.33	93.67	96.33	95.50

Tabela 6: Jądro wielomianowe - cross-validation, bez skalowania

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	39.93	39.93	96.63	95.52	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11
0.01	39.93	43.89	96.67	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11
0.1	39.93	87.06	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11
1	39.93	96.63	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11
10	43.89	96.67	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11
100	87.06	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11
1000	96.63	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11	96.11

W Tabeli 6. widać efekt plateau - dla wielu kombinacji wartości **C** i **gamma** dokładność utrzymuje się na poziomie 96,11%. Oznacza to, że w badanym zakresie hiperparametrów model osiągnął nasycenie: dalsze zwiększanie **C** czy **gamma** nie poprawia dopasowania ani dokładności. Problem plateau utrudnia dalszą optymalizację, ponieważ w "płaskim obszarze" przestrzeni parametrów zmiana wartości nie przynosi korzyści.

Tabela 7: Jądro wielomianowe - cross-validation, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	39.93	39.93	96.63	96.08	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52
0.01	39.93	44.41	96.08	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52
0.1	39.93	87.06	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52
1	39.93	96.63	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52
10	44.41	96.08	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52
100	87.06	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52
1000	96.63	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52	95.52

Tabela 8: Jądro wielomianowe - bootstrap, bez skalowania

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	35.81	37.52	94.48	94.53	94.11	93.19	95.66	94.64	93.19	94.33	93.78	95.21
0.01	37.16	34.65	94.05	95.78	91.71	93.47	94.83	95.41	94.49	94.61	93.94	93.09
0.1	36.71	87.61	93.02	95.61	94.76	94.78	95.60	95.35	95.84	94.42	93.71	94.30
1	34.36	94.07	95.21	93.50	94.66	93.88	93.89	92.39	96.10	94.99	94.39	95.29
10	36.32	91.83	95.82	95.22	92.77	92.92	92.04	94.26	94.09	94.33	93.76	94.08
100	83.91	95.12	95.35	93.04	94.43	93.98	93.58	95.22	93.68	93.68	94.87	94.19
1000	94.12	94.96	94.29	94.78	95.04	94.96	95.38	94.04	95.54	94.87	95.46	93.61

Tabela 9: Jądro wielomianowe - bootstrap, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	34.82	36.49	95.66	94.02	93.86	94.69	93.93	94.63	94.69	93.19	95.26	92.95
0.01	37.58	41.03	93.14	95.32	94.89	94.41	95.24	93.90	94.48	95.39	95.36	95.32
0.1	37.63	82.98	95.92	92.58	93.71	93.71	94.57	94.48	94.48	94.87	93.29	94.45
1	36.87	95.87	94.57	94.92	94.19	92.49	95.86	94.61	94.57	95.01	94.76	94.72
10	39.23	94.27	94.97	96.14	94.26	94.16	93.73	95.79	93.64	94.56	93.49	94.47
100	81.48	95.35	95.05	94.53	93.41	93.85	93.70	92.34	93.89	94.89	93.80	94.24
1000	94.02	94.86	94.32	92.73	94.32	94.62	94.35	94.66	93.74	94.86	95.62	94.75

Analiza Tabel 4–9 prowadzi do wniosku, że najwyższą dokładność klasyfikacji uzyskano dla kombinacji hiperparametrów: **gamma = 10** oraz **C = 0.1**. Ponadto zauważono, że zastosowanie skalowania cech miało korzystny wpływ na wyniki – w większości przypadków prowadziło do poprawy skuteczności klasyfikatora.

Dodatkowo przeanalizowano wpływ stopnia wielomianu jądra na jakość klasyfikacji.

Tabela 10: Dokładność klasyfikacji w zależności od stopnia wielomianu — metoda wielokrotnego podziału z użyciem najlepszej kombinacji parametrów gamma i C

	Stopień wielomianu								
	2	3	4	5	6	7	8	9	10
Dokładność	87.5	94.83	85.5	90	80.17	81.17	73	79.5	69.67

Z Tabeli 10. wynika, że najwyższą dokładność klasyfikacji uzyskano dla wielomianu stopnia **3** – czyli tego samego, który został zastosowany przy wcześniejszym doborze optymalnych wartości parametrów **gamma** i **C**. Potwierdza to trafność wyboru tego stopnia jako podstawy do dalszej optymalizacji modelu.

1.2.3 Jądro radialne (RBF)

Przeanalizowano skuteczność klasyfikacyjną z zastosowaniem jądra radialnego, zarówno **bez skalowania danych**, jak i **po ich skalowaniu**.

Podobnie jak w przypadku jąder liniowego i wielomianowego, porównanie wyników dla obu wariantów umożliwia ocenę wpływu przeskalowania zmiennych na jakość klasyfikacji. W przypadku jądra radialnego, oprócz parametru **C**, kluczową rolę odgrywa również parametr **gamma**, który kontroluje zasięg wpływu pojedynczych obserwacji treningowych.

Tabela 11: Jądro radialne - wielokrotny podział, bez skalowania

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	39.90	39.90	39.90	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
0.01	39.90	39.90	39.90	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
0.1	80.75	97.16	39.90	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
1	97.75	98.27	61.73	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
10	97.16	98.27	67.39	41.01	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
100	96.05	98.27	67.39	41.01	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
1000	96.05	98.27	67.39	41.01	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9

Tabela 12: Jądro radialne - wielokrotny podział, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
0.01	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
0.1	80.39	96.67	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
1	97.22	97.78	61.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
10	97.78	98.33	64.77	41.05	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
100	96.08	98.33	64.77	41.05	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
1000	96.08	98.33	64.77	41.05	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93

Tabela 13: Jądro radialne - cross-validation, bez skalowania

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	39.90	39.90	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
0.01	39.90	39.90	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
0.1	77.03	96.67	39.90	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
1	98.33	98.33	64.08	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
10	97.78	98.33	68.07	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
100	97.22	98.33	68.07	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9
1000	97.22	98.33	68.07	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9	39.9

Tabela 14: Jądro radialne - cross-validation, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
0.01	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
0.1	80.39	96.63	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
1	97.78	98.33	63.07	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
10	97.19	98.33	68.10	41.05	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
100	96.63	98.33	68.10	41.05	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93
1000	96.63	98.33	68.10	41.05	39.93	39.93	39.93	39.93	39.93	39.93	39.93	39.93

W Tabelach 11–14. obserwujemy, że zarówno dla danych bez skalowania, jak i po ich skalowaniu, dokładność utrzymuje się na poziomie około 39,9% dla wielu kombinacji parametrów **gamma** i **C**. Odwołując się do Raportu 3. "Gdybyśmy przypisali wszystkie obserwacje do najczęściej występującej klasy, uzyskalibyśmy dokładność na poziomie 39.89%."

Oznacza to, że model w tych ustawieniach przypisuje wszystkie obserwacje do dominującej klasy, co wskazuje na underfitting.

Fakt, że skalowanie nie poprawiło wyniku, sugeruje, iż problem nie wynika wyłącznie z różnic w skali cech, lecz także z nieoptymalnego zakresu hiperparametrów, możliwej nierównowagi klas lub niewystarczającej reprezentacji cech do separacji. Efekt plateau w obu wariantach (ze skalowaniem i bez) oznacza, że w badanym zakresie dalsze zmiany **gamma** i **C** nie wpływają na poprawę dokładności, ponieważ model nie „widzi" struktur rozróżniających klasy.

Tabela 15: Jądro radialne - bootstrap, bez skalowania

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	32.45	36.58	37.49	35.99	37.60	39.60	37.18	36.82	36.45	36.70	38.89	30.52
0.01	36.95	33.78	36.73	35.22	36.07	33.77	35.94	30.85	38.46	36.02	35.84	36.47
0.1	58.83	93.26	37.78	38.52	40.83	38.32	34.96	37.33	38.94	38.94	39.15	33.07
1	97.67	97.07	52.80	36.71	39.72	39.55	36.21	36.85	38.90	36.52	37.67	39.68

Tabela 15: Jądro radialne - bootstrap, bez skalowania (kontynuacja)

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
10	97.61	97.19	55.85	40.59	37.69	39.33	37.71	36.88	38.67	40.81	38.39	36.35
100	96.70	97.74	56.09	36.87	37.43	37.79	39.96	34.46	36.23	36.83	36.79	40.34
1000	97.22	98.09	55.66	37.24	40.70	39.07	35.26	37.26	39.72	34.70	36.69	35.08

Tabela 16: Jądro radialne - bootstrap, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma											
	0.01	0.1	1	2	3	4	5	6	7	8	9	10
0.001	38.78	38.84	38.00	38.23	34.25	34.59	37.64	38.99	36.31	35.66	36.80	36.55
0.01	36.69	41.22	37.57	36.96	41.96	35.29	37.49	40.31	37.38	36.52	37.90	38.72
0.1	44.36	93.31	36.56	40.24	37.92	38.26	39.22	37.49	37.56	37.15	35.53	37.95
1	97.57	97.01	53.57	39.56	35.95	36.26	38.66	37.87	36.26	36.90	35.80	38.72
10	97.05	97.66	56.69	40.11	38.27	37.09	36.94	35.16	35.45	37.59	37.00	34.46
100	96.24	97.42	52.50	38.45	38.86	37.32	40.19	37.09	38.50	38.34	38.59	34.87
1000	96.56	97.96	57.47	37.30	39.01	38.35	36.89	36.75	38.49	37.93	39.61	38.05

Tabela 15. i 16. nie wykazują efektu plateau, jednak uzyskane wyniki są niskie. Zaobserwowano, że najlepsze rezultaty pojawiają się przy niskich wartościach **gamma** i wysokim współczynniku kary **C**, co sugeruje, że taka konfiguracja sprzyja lepszej separacji klas. W celu dalszej poprawy dokładności przeprowadzono dodatkowe testy z użyciem metody bootstrap po skalowaniu danych, modyfikując zakres parametrów **gamma** i **C**.

Tabela 17: Jądro radialne - bootstrap, ze skalowaniem - zmienione parametry C i gamma

Współcznnik kary - C	Współczynnik - gamma						
	1e-06	1e-05	1e-04	0.001	0.01	0.1	1
1	36.07	35.80	37.55	71.98	98.08	97.75	51.10
10	39.90	36.76	67.52	97.43	97.76	97.29	58.29
100	33.99	69.35	97.69	96.08	97.89	97.22	52.98
1000	72.96	96.58	96.63	96.87	97.54	97.96	57.70
10000	96.93	97.18	97.03	96.37	97.21	97.94	53.00
1e+05	96.75	97.26	97.04	97.06	96.81	98.56	56.69
1e+06	96.92	96.22	97.31	96.24	97.39	98.38	53.51

Analiza Tabeli 17. pokazuje, że bardzo niskie wartości **gamma** i **C** skutkują niską dokładnością na poziomie już spotkanego przypisania wszystkich obserwacji do najliczniejszej klasy. Bardzo duże **C** wymuszają surową karę za błędy, co zwiększa dopasowanie do danych, ale może prowadzić do przeuczenia. Natomiast małe **gamma** powoduje, że wpływ pojedynczych punktów jest szeroki, co wygładza granicę decyzyjną i zapobiega nadmiernemu dopasowaniu. Połączenie dużego **C** z małym lub umiarkowanym **gamma** daje najlepsze wyniki, zapewniając równowagę między dopasowaniem a uogólnianiem modelu.

Na podstawie tego stwierdzamy że najlepsze wyniki dla jądra radialnego osiągane są dla **C** wynoszącego 1 000 000 i **gamma** wynoszącego 0.0001.

1.2.4 Jądro sigmoidalne

Przeanalizowano skuteczność klasyfikacyjną z zastosowaniem jądra sigmoidalnego, zarówno **bez skalowania danych**, jak i **po ich skalowaniu**.

Podobnie jak w przypadku jąder liniowego, wielomianowego i radialnego, porównanie wyników dla obu wariantów umożliwia ocenę wpływu przeskalowania zmiennych na jakość klasyfikacji.

Tabela 18: Jądro sigmoidalne - wielokrotny podział, bez skalowania

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
0.1	34.83	39.67	97.17	90.17	86.17	87.17	88.33
1	33.33	97.67	97.17	85.33	80.17	82.83	82.83
10	97.17	98.17	94.83	83.00	85.00	80.33	83.17
100	98.33	97.17	94.00	79.33	81.50	80.33	81.67
1000	95.33	96.50	92.67	83.17	81.33	83.67	82.83
10000	96.33	97.33	93.67	80.83	80.83	79.33	82.67
1e+05	96.83	98.00	90.67	82.67	81.33	80.33	80.67

Z Tabeli 18. można wyciągnąć interesującą obserwację. Dla niskich wartości parametrów **C** oraz **gamma** klasyfikator SVM wykazuje bardzo niską skuteczność – na tyle niską, że jego działanie jest gorsze niż proste przypisanie wszystkich obserwacji do najczęściej występującej klasy w zbiorze danych.

Oznacza to, że przy zbyt małej karze za błąd klasyfikacji (**C**) oraz zbyt małym zasięgu wpływu pojedynczych obserwacji (**gamma**), model nie jest w stanie uchwycić żadnych istotnych wzorców w danych. W rezultacie uzyskana reguła klasyfikacyjna staje się praktycznie bezużyteczna i wykazuje się efektywnością niższą niż przypisanie wszystkich przypadków do jednej, dominującej klasy.

Tabela 19: Jądro sigmoidalne - wielokrotny podział, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
0.1	44.67	38.67	96.33	91.00	89.67	85.67	86.00
1	41.33	98.00	96.50	84.17	80.83	80.67	81.67
10	97.83	97.17	94.67	85.17	81.50	82.50	85.17
100	98.00	96.17	92.67	82.67	80.67	81.67	82.00
1000	96.83	96.67	93.50	81.33	81.33	83.67	80.50
10000	96.17	97.17	91.83	81.67	80.17	81.83	82.00
1e+05	96.67	95.83	91.17	79.83	81.33	83.83	79.33

Tabela 20: Jądro sigmoidalne - cross-validation, bez skalowania

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
0.1	39.77	42.03	97.19	88.17	85.36	85.36	85.95
1	42.03	98.30	96.60	81.90	80.85	82.52	81.96
10	98.30	98.30	93.79	81.93	80.16	83.07	82.55
100	98.30	96.63	94.90	82.52	81.31	83.07	81.96
1000	96.63	96.63	94.90	81.96	79.05	83.63	81.96
10000	96.63	96.63	94.35	81.96	76.24	83.63	82.52
1e+05	96.63	96.63	94.35	80.82	76.80	84.22	82.52

Tabela 21: Jądro sigmoidalne - cross-validation, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
0.1	39.87	42.68	97.19	89.35	86.01	86.01	87.09
1	42.68	98.30	97.19	82.06	78.73	76.96	81.96
10	98.30	98.86	94.90	79.84	75.95	77.52	82.52
100	98.86	94.93	93.20	80.95	75.95	76.96	82.52
1000	94.93	95.49	94.35	79.84	74.84	77.52	82.52
10000	95.49	95.49	93.76	79.84	76.50	76.96	82.52
1e+05	95.49	95.49	93.79	82.61	75.92	76.96	82.52

Tabela 22: Jądro sigmoidalne - bootstrap, bez skalowania

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
0.1	36.34	37.31	96.34	85.68	86.02	86.06	86.21
1	37.51	96.89	96.22	81.55	81.58	81.76	79.85
10	96.42	96.64	93.05	80.86	75.98	81.31	83.65
100	96.13	96.84	92.18	81.41	83.09	80.02	82.80
1000	95.34	96.27	90.46	81.05	78.46	82.53	85.28
10000	96.38	96.16	89.79	79.34	80.83	81.81	80.99
1e+05	96.23	96.64	89.23	81.60	79.03	78.91	80.03

Tabela 23: Jądro sigmoidalne - bootstrap, ze skalowaniem

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
0.1	33.70	39.10	97.22	86.30	86.64	86.34	86.13

Tabela 23: Jądro sigmoidalne - bootstrap, ze skalowaniem (kontynuacja)

Współcznnik kary - C	Współczynnik - gamma						
	0.001	0.01	0.1	1	10	100	1000
1	37.32	97.30	95.53	82.20	78.83	77.81	81.28
10	97.48	97.05	93.58	77.13	79.98	79.50	81.63
100	96.52	96.62	91.13	80.82	79.20	82.18	81.96
1000	96.76	96.69	89.91	81.88	79.70	79.15	83.34
10000	97.58	96.65	90.59	81.15	79.59	82.85	79.70
1e+05	96.01	96.41	90.21	80.68	81.31	80.91	81.73

Analizując wyniki przedstawione w Tabelach 18–23. dla jądra sigmoidalnego, można zauważyć, że model osiąga najlepsze wyniki klasyfikacyjne przy wartościach parametru **C** mieszczących się w przedziale od 100 do 1000 oraz **gamma** w zakresie od 0.001 do 0.01.

1.3 Porówanie metod

Tabela 24: Dokładność klasyfikacji za pomocą drzewa klasyfikacyjnego, z podziałem na algorytmy uczenia zespołowego oraz liczbę replikacji

Algorytm uczenia zespołowego	Liczba replikacji							
	1	5	10	20	30	40	50	100
Bagging	94.68	93.24	94.76	96.69	96.08	96.65	96.87	96.82
Random Forest	98.88	98.52	98.80	99.04	98.43	98.97	98.39	98.62
Boosting	97.54	97.03	97.02	97.61	96.65	97.03	97.36	97.39

Z Tabeli 24. możemy odczytać, że wszystkie trzy metody uczenia zespołowego skutecznie poprawiły klasyfikację przypadków. Najlepsze wyniki osiągnęła metoda RandomForest, której dokładność dla dowolnej liczby klasyfikatorów wyniosła powyżej 98% (w tym pojedyczny wynik powyżej 99%, najlepszy ze wszystkich). Najgorsze wyniki natomiast osiągnęła metoda Bagging, dla liczby replikacji boostrapowych poniżej 20. Możemy również zauważyć, że ten sposób poprawy klasyfikacji na tym zbiorze nie osiągnął jednak dużo mniejszych wyników od Boosting. Zatem jeżeli zależy nam na jak najmniejszej złożoności obliczeniowej, warto z Boosting zrezygnować na rzecz pozostałych metod.

Dla metody wektorów nośnych zauważyć mogliśmy, że dla różnych jąder różnie działają kombinacje parametrów kary oraz (dla odpowiednich) gammy. Jadro liniowe oraz wielomianowe osiągnęły największą dokładnośc klasyfikacji dla $\mathbf{C} = 0.1$ wielomianowe również dla stopnia równego 3 i $\mathbf{gamma} = 10$. Jądra radialne oraz sigmoidalne wymagały o wiele większej różnicy między parametrami. W pierwszym różnica jest około 10000-krotna, a dla drugiej, aż 10^{10}.

Najlepsze parametry dla poszczególnych SVM dają wyniki klasyfikacji porównywalne do metod boosting oraz bagging. Jednak najlepsze wyniki osiągamy dla metody RandomForest.

1.4 Wnioski

Podsumowując, w tej części raportu zbadaliśmy dokładność klasyfikacji osiąganą za pomocą metod wektorów nośnych (SVM) oraz metod uczenia zespołowego.

Dla odpowiednio dobranych parametrów kary oraz (z wyjątkiem dla jądra liniowego) gamma, SVM dało wyniki przekraczające ponad 94%. Jednak dla każdego jądra dobór parametrów różnił się, jednak najlepsze wyniki otrzymywała para kary i co najmniej 100-krotnie mniejszej gammy.

Metody uczenia zespołowego zastosowane dla drzew klasyfikacyjnych znacznie poprawiły dokładność klasyfikacji, osiągając wyniki ponad 96% (z wyjątkiem 3 wartości dla bagging). Najlepszy okazał się las losow, który odznacza się mniejszą złożonością obliczeniową niż boosting oraz porównywalną do bagging.

2 Analiza skupień – algorytmy grupujące i hierarchiczne

W tym zadaniu zastosujemy i porównamy ze sobą metody analizy skupień - k-średnich i PAM jako algorytmy grupujące, oraz AGNES - algorytm hierarchiczny.

Zadanie zostanie wykonane na zbiorze danych *wine*, którego szczegółowy opis znajduje się w poprzednim raporcie.

To zadanie zostanie wykonane już na innym danych, którymi będzie zbiór *glass*.

2.1 Charakterystyka danych

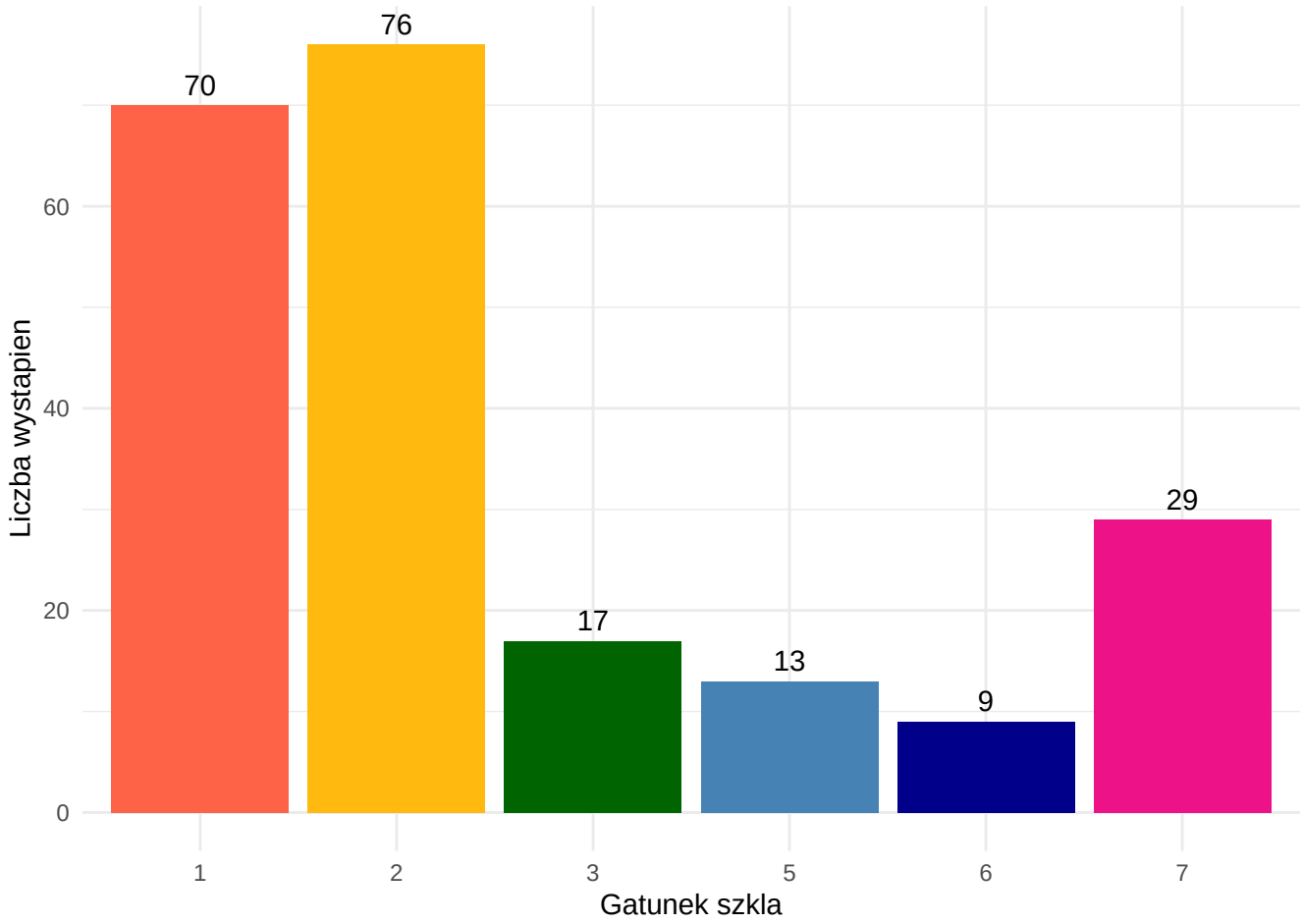
Zbiór danych glass zawiera **214** przypadków sześciu rodzajów szkła oraz **10** cech. Liczba brakujących danych wynosi **0**.

Znaczenie poszczególnych cech oraz ich typ przedstawiono w Tabeli 25.

Tabela 25: Opis zmiennych w zbiorze danych Glass

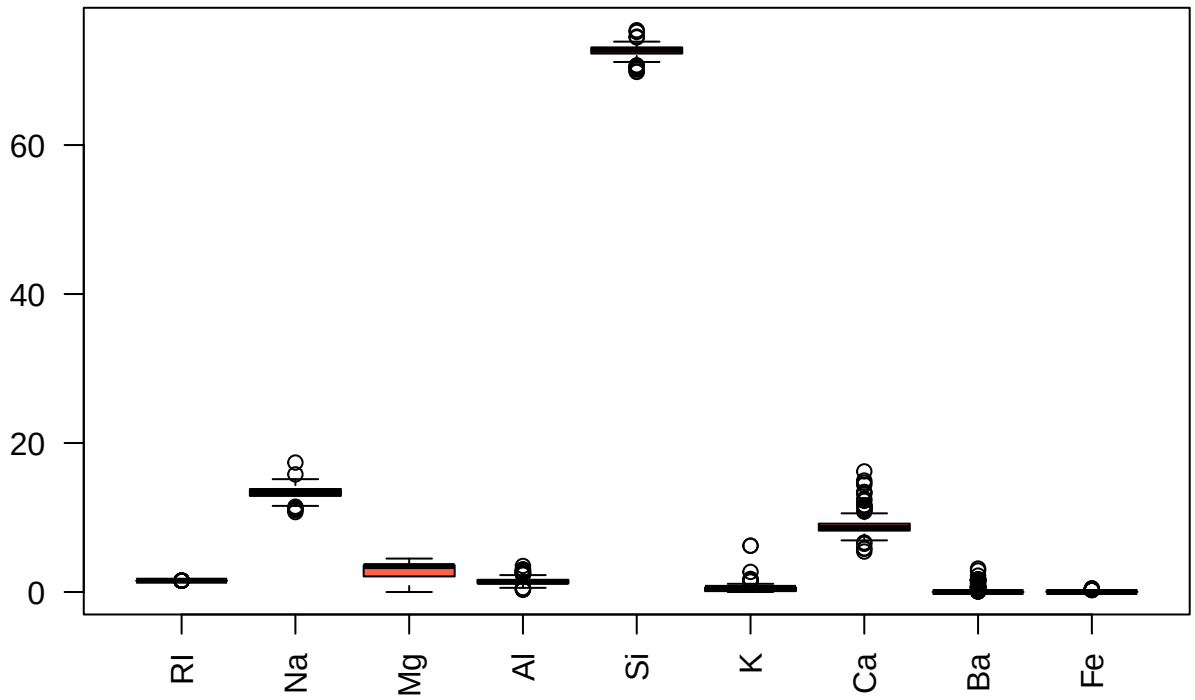
Zmienna	Typ	Opis
RI	numeric	Współczynnik załamania światła
Na	numeric	Procent masowy azotu (%)
Mg	numeric	Procent masowy magnezu (%)
Al	numeric	Procent masowy glinu (%)
Si	numeric	Procent masowy krzemu (%)
K	numeric	Procent masowy potasu (%)
Ca	numeric	Procent masowy wapnia (%)
Ba	numeric	Procent masowy baru (%)
Fe	numeric	Procent masowy żelaza (%)
Type	factor	Typ szkła

W poszczególnych przypadkach analizowanych próbek szkła sumy procentów masowych znajdują się w zakresie od 99.02 do 100.1. Zaobserwowany nadmiar może wynikać z błędów zaokrągleń dokonanych przez autorów pomiarów. Natomiast niedomiar może być spowodowany zarówno obecnością innych pierwiastków chemicznych, nieuwzględnionych w zbiorze danych – tzw. pierwiastków śladowych – jak i niedokładnościami pomiarowymi lub zaokrągleniami dokonanymi przez autorów pomiarów.



Wykres 2: Liczność poszczególnych typów szkła

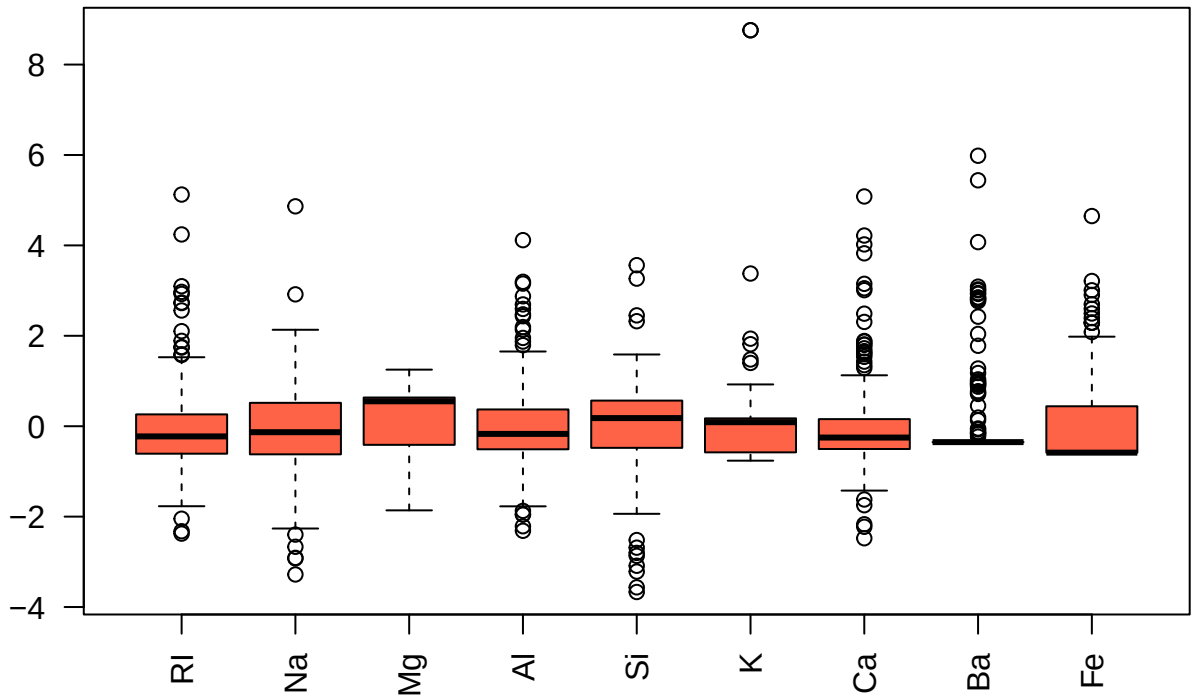
Z Wykresu 2. można odczytać, że liczba obserwacji poszczególnych klas w zbiorze danych jest bardzo zróżnicowana. Siedemdziesiąt obserwacji lub więcej posiadają typy 1. oraz 2. (co stanowi 68.22% danych), reszta już po mniej niż 30 obserwacji.



Wykres 3: Wykres pudełkowy, zmienne bez standaryzacji

Na Wykresie 3. można zaobserwować wyraźną potrzebę standaryzacji danych przed zastosowaniem modelu.

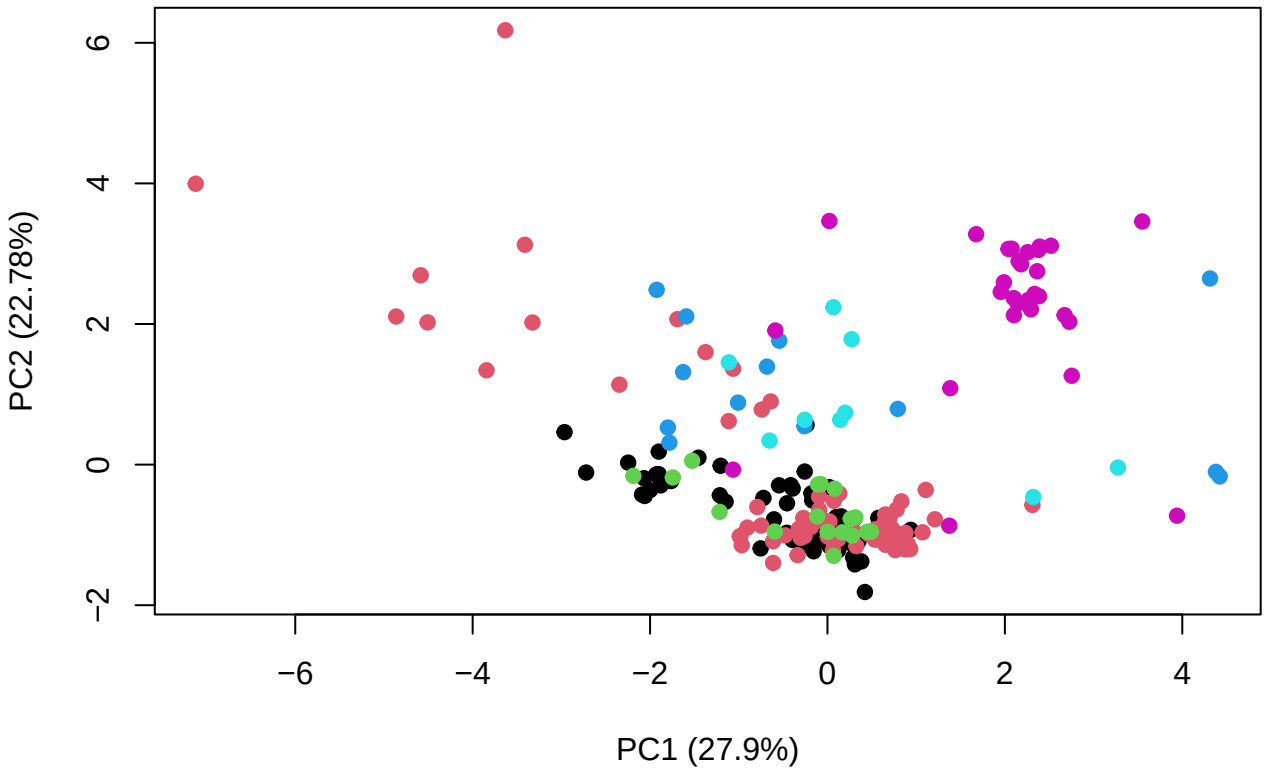
Dodatkowo, na wykresie nie zauważono punktów odpowiadających błędnie przypisanym wartościom brakującym, które wyróżniałyby się jako odstające obserwacje. Świadczy to o odpowiednim przygotowaniu danych oraz skutecznym radzeniu sobie modelu z ewentualnymi brakami w zbiorze.



Wykres 4: Wykres pudełkowe, po standaryzacji

Na Wykresie 4., przedstawiającym dane po standaryzacji, można zauważyć wyraźne odstępstwa, które wskazują na obecność wartości odstających. Przykładowo, jedna z próbek cechuje się wyjątkowo wysokim poziomem potasu, co może sugerować błąd pomiarowy lub anomalię w procesie produkcji szkła.

Ponadto, większość próbek szkła nie zawiera baru, co skutkuje zerowymi lub bardzo niskimi wartościami tej cechy w większości obserwacji. W konsekwencji, próbki zawierające bar stają się wartościami odstającymi względem reszty danych, co jest dobrze widoczne na wykresie.



Wykres 5: Wizualizacja danych, PCA

Na Wykresie 5. przedstawiono wizualizację danych po zastosowaniu redukcji wymiarowości za pomocą analizy głównych składowych (PCA). Choć pierwsze dwie składowe, PC1 i PC2, wyjaśniają łącznie około 50,7% zmienności w danych, nie przekłada się to na wyraźne rozdzielenie klas.

Wieloklasowe punkty nadal skupiają się blisko siebie, co utrudnia ich rozróżnienie w przestrzeni dwuwymiarowej.

2.2 Wyniki grupowania

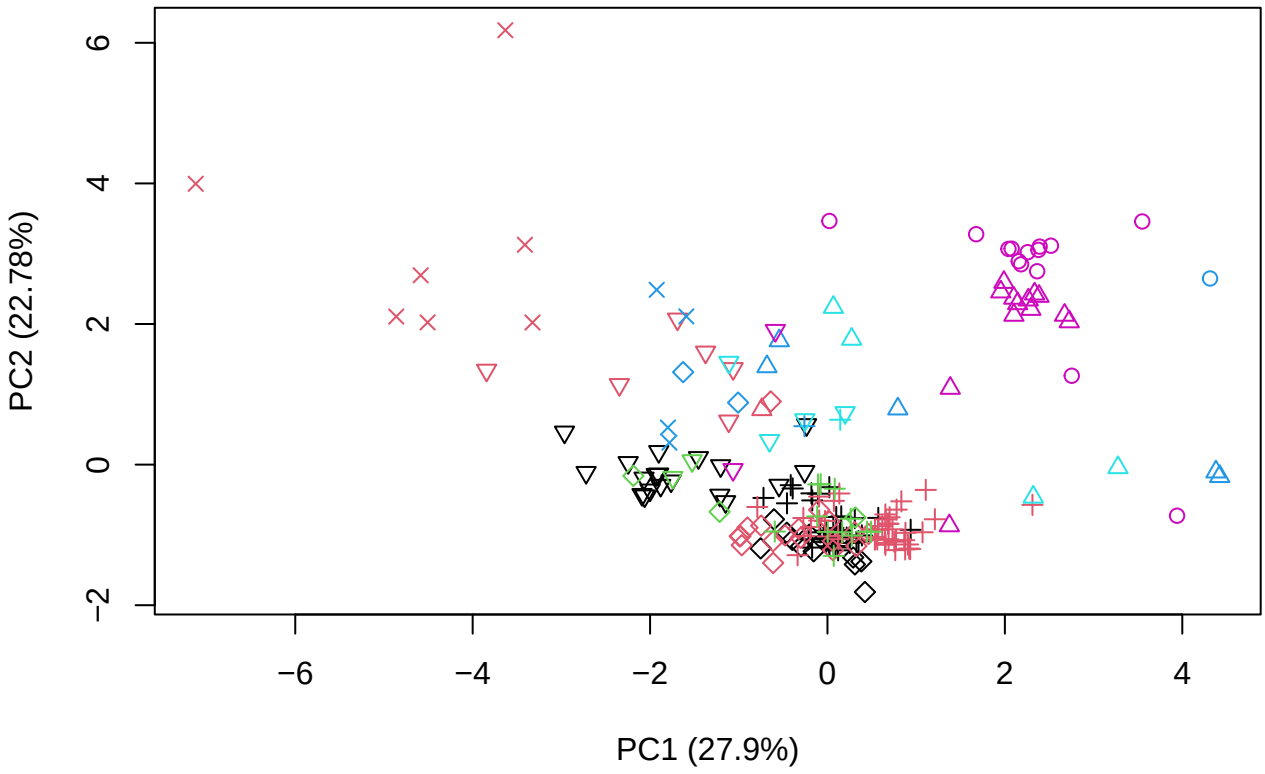
W tej części przeprowadzono grupowanie danych z zastosowaniem czterech metod: k-średnich (k-means), PAM (Partitioning Around Medoids), AGNES (Agglomerative Nesting) oraz DIANA (Divisive Analysis).

Dla zapewnienia spójności porównania, liczba grup została przyjęta zgodnie z rzeczywistą liczbą klas w zbiorze danych, która wynosi **6**. Pozwoli to ocenić, w jakim stopniu poszczególne algorytmy są w stanie odwzorować rzeczywistą strukturę klas obecnych w zbiorze **glass**.

2.2.1 k-średnie

W tej części zastosowano algorytm **k-średnich**, który należy do metod niehierarchicznych, opartych na minimalizacji wewnątrzgrupowej sumy kwadratów odległości. Algorytm iteracyjnie przypisuje obserwacje do jednego z 6 klastrów na podstawie ich podobieństwa, a następnie aktualizuje położenie centroidów, aż do osiągnięcia stabilności układu.

Grupowanie przeprowadzono na danych uprzednio poddanych standaryzacji, co zapewnia równoważny wpływ wszystkich zmiennych na wynik końcowy. Wyniki grupowania zostaną porównane z rzeczywistymi etykietami klas w celu oceny skuteczności metody.

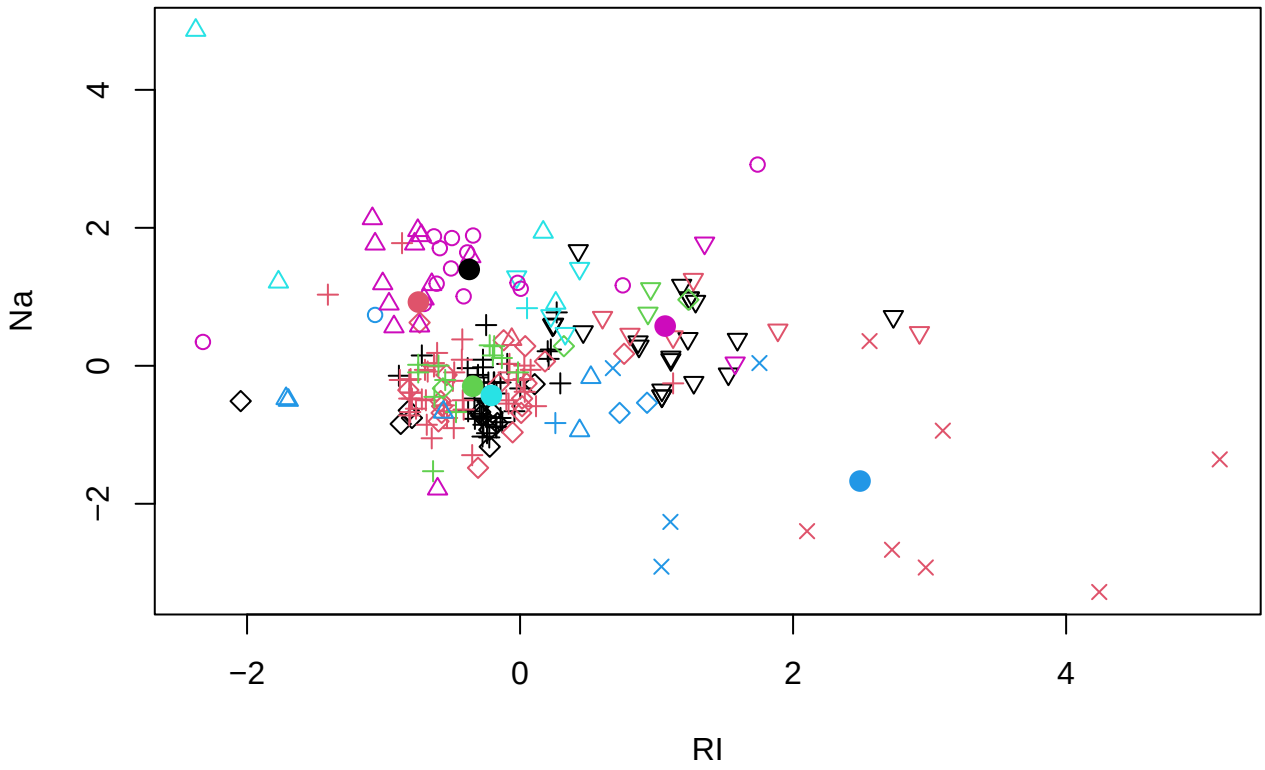


Wykres 6: Wizualizacja danych po grupowania metodą k-średnich - PCA

Na Wykresie 6. zaprezentowano rezultaty grupowania uzyskane metodą k-średnich. Kolory odpowiadają rzeczywistym klasom szkła, natomiast kształty punktów reprezentują przypisanie obserwacji do klastrów uzyskanych w wyniku działania algorytmu.

Już na pierwszy rzut oka można zauważyć, że przypisania do grup są w dużym stopniu niespójne z rzeczywistymi etykietami - klasy zostały znacząco pomieszane. Świadczy to o ograniczonej skuteczności metody k-średnich w odtwarzaniu rzeczywistej struktury danych dla zbioru *glass*.

Aby dokładniej zilustrować charakter uzyskanych grupowań, na kolejnym wykresie przedstawiono zależność współczynnika załamania światła (*Refractive Index*, RI) od procentowej zawartości sodu (Na). Wizualizacja ta pozwala lepiej ocenić, w jakim stopniu podział na klastry odpowiada naturalnym różnicom chemicznym między próbkami szkła.



Wykres 7: Zależnośc współczynnika załamania światła od procentu masowego sodu z kolorami i kształtami klas - zaznaczone centroidy z metody k-średnich

Na Wykresie 7. przedstawiono rozmieszczenie centrów skupień wyznaczonych przez algorytm k-średnich. Pomimo przeprowadzenia 1000 iteracji w celu osiągnięcia stabilnej konwergencji, położenie centroidów wydaje się dość przypadkowe. W ich bezpośrednim otoczeniu znajdują się obserwacje należące do różnych, rzeczywistych klas, co wskazuje na brak wyraźnego dopasowania modelu do rzeczywistej struktury danych.

Taki efekt może być częściowo wynikiem ograniczeń związanych z wizualizacją — przedstawione punkty i centra skupień znajdują się w przestrzeni wielowymiarowej, natomiast wykres stanowi jedynie jej uproszczoną projekcję na dwie zmienne. Aby dokładniej ocenić skuteczność przypisania obserwacji do klastrów, przyjrzyjmy się **macierzy pomyłek**, która zestawia rzeczywiste klasy z przypisaniami wygenerowanymi przez model. Pozwoli to w sposób obiektywny ocenić trafność grupowania.

Tabela 26: Macierz pomyłek - k-średnie

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	20	6	2	0	4	2
2	39	44	12	1	1	0
3	11	18	3	2	0	0
5	0	7	0	4	0	0
6	0	1	0	5	4	13
7	0	0	0	1	0	14

Do oceny dokładności grupowania zastosowano metodę **matchClasses** z parametrem “exact”, która pozwala

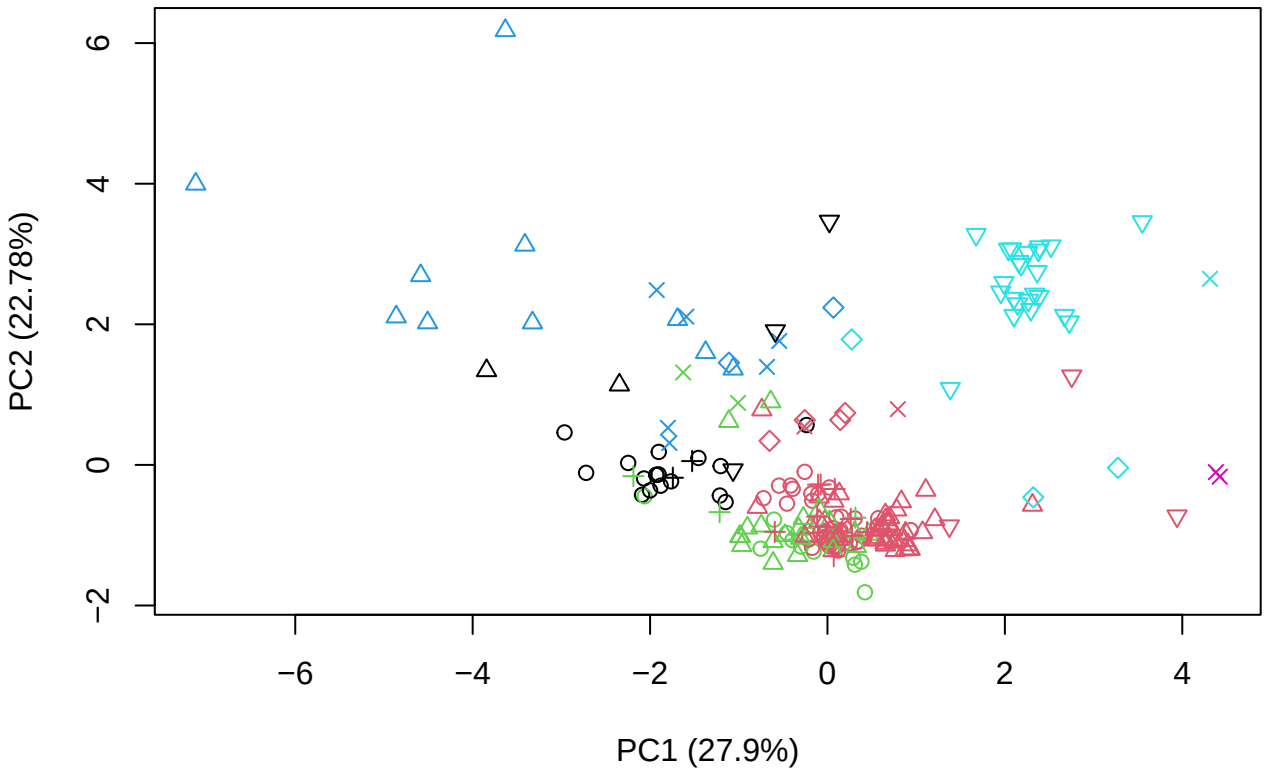
na precyzyjne dopasowanie przypisań klastrów do rzeczywistych etykiet klas.

Na podstawie tej metody wyznaczono, że dokładność klasyfikacji wynosi 41.59%. Wynik ten stanowi miarodajną miarę skuteczności zastosowanego algorytmu grupowania w odzwierciedlaniu rzeczywistej struktury danych.

2.2.2 Partitioning Around Medoids (PAM)

W tej części zastosowano algorytm **Partitioning Around Medoids (PAM)**, który podobnie jak k-średnie należy do metod grupujących, jednak zamiast **centroidów** wykorzystuje rzeczywiste obiekty zwane **medoidami** jako reprezentanty klastrów.

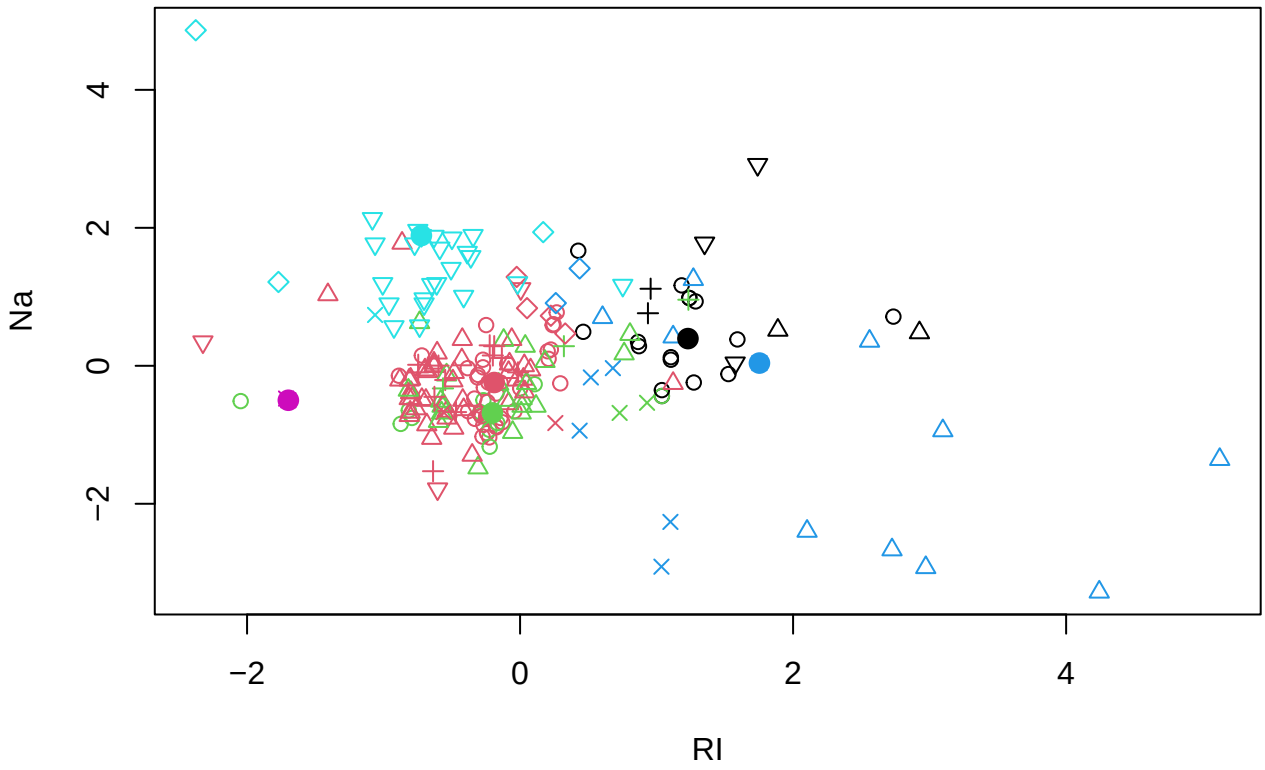
Metoda PAM jest bardziej odporna na wartości odstające oraz szum w danych, co czyni ją szczególnie przydatną w analizie zbiorów, gdzie występują takie anomalie. Grupowanie przeprowadzono dla liczby klastrów równej rzeczywistej liczbie klas w zbiorze danych **glass**, a wyniki zostaną poddane ocenie pod kątem zgodności z rzeczywistymi etykietami.



Wykres 8: Wizualizacja danych po grupowania metodą PAM - PCA

Z Wykresu 8. można zauważyć, że algorytm PAM doprowadził do lepszego rozdzielenia klas w porównaniu z metodą k-średnich — obserwacje wydają się być przypisane do klastrów w sposób bardziej uporządkowany i częściowo zgodny z rzeczywistymi etykietami.

Niemniej jednak, ocena „na oko” może być myląca i niewystarczająca do rzetelnej analizy jakości grupowania. Subiektywne wrażenie nie zastąpi obiektywnych miar skuteczności. W związku z tym konieczne jest przeprowadzenie dokładniejszej oceny, na przykład za pomocą **macierzy pomyłek** oraz miary dokładności dopasowania, aby jednoznacznie stwierdzić, na ile grupowanie rzeczywiście odwzorowuje strukturę danych.



Wykres 9: Zależnośc współczynnika załamania światła od procentu masowego sodu z kolorami i kształtami klas - zaznaczone medoidy z metody PAM

Z Wykresu 9. wynika, że poza lepszym odwzorowaniem rzeczywistych klas, metoda PAM skuteczniej lokalizuje reprezentatywne punkty – **medoidy** – w przestrzeni danych. W przeciwieństwie do metody k-średnich, gdzie centra skupień mogą przyjmować pozycje oderwane od rzeczywistych obserwacji, medoidy znajdują się bezpośrednio wśród danych i w wielu przypadkach są położone blisko środka odpowiadającej im grupy.

Taka właściwość przekłada się na większą interpretowalność i odporność algorytmu na obserwacje odstające. Szczegółowe informacje dotyczące wybranych medoidów oraz odpowiadających im klas przedstawiono w Tabeli 27..

Tabela 27: Dane medoidów

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type	PAM
44	1.52210	13.73	3.84	0.72	71.76	0.17	9.74	0.00	0.00	1	1
43	1.51779	13.21	3.39	1.33	72.76	0.59	8.59	0.00	0.00	1	2
33	1.51775	12.85	3.48	1.23	72.97	0.61	8.56	0.09	0.22	1	3
171	1.52369	13.44	0.00	1.58	72.22	0.32	12.24	0.00	0.00	5	4
205	1.51617	14.95	0.00	2.27	73.30	0.00	8.71	0.67	0.00	7	5
173	1.51321	13.00	0.00	3.02	70.70	6.21	6.93	0.00	0.00	5	6

Tabela 28: Macierz pomyłek - PAM

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	17	2	2	0	0	3
2	40	43	12	2	4	3
3	13	21	3	2	0	0
5	0	10	0	6	2	0
6	0	0	0	2	0	0
7	0	0	0	1	3	23

Dokładność klasyfikacji uzyskana metodą PAM wynosi 42.99%. Choć rezultat ten przewyższa wynik osiągnięty przez algorytm k-średnich, nadal nie jest on satysfakcjonujący z punktu widzenia jakości odwzorowania rzeczywistej struktury danych.

Oznacza to, że mimo poprawy, metoda PAM nie zapewnia jeszcze precyzyjnego rozróżnienia klas w zbiorze *glass* i wymaga dalszej analizy porównawczej z innymi podejściami, takimi jak algorytmy hierarchiczne.

2.2.3 Agglomerative Nesting (AGNES)

W tej części zastosowano algorytm **AGNES (Agglomerative Nesting)**, który reprezentuje podejście aglomeracyjne w hierarchicznej analizie skupień. Metoda ta rozpoczyna proces grupowania od traktowania każdej obserwacji jako osobnego klastra, a następnie iteracyjnie łączy najbliższe pary klastrów, aż do momentu, gdy wszystkie obserwacje znajdą się w jednej nadrzędnej strukturze. Efektem końcowym jest **dendrogram** — drzewo hierarchiczne ilustrujące kolejność i poziom łączenia poszczególnych elementów.

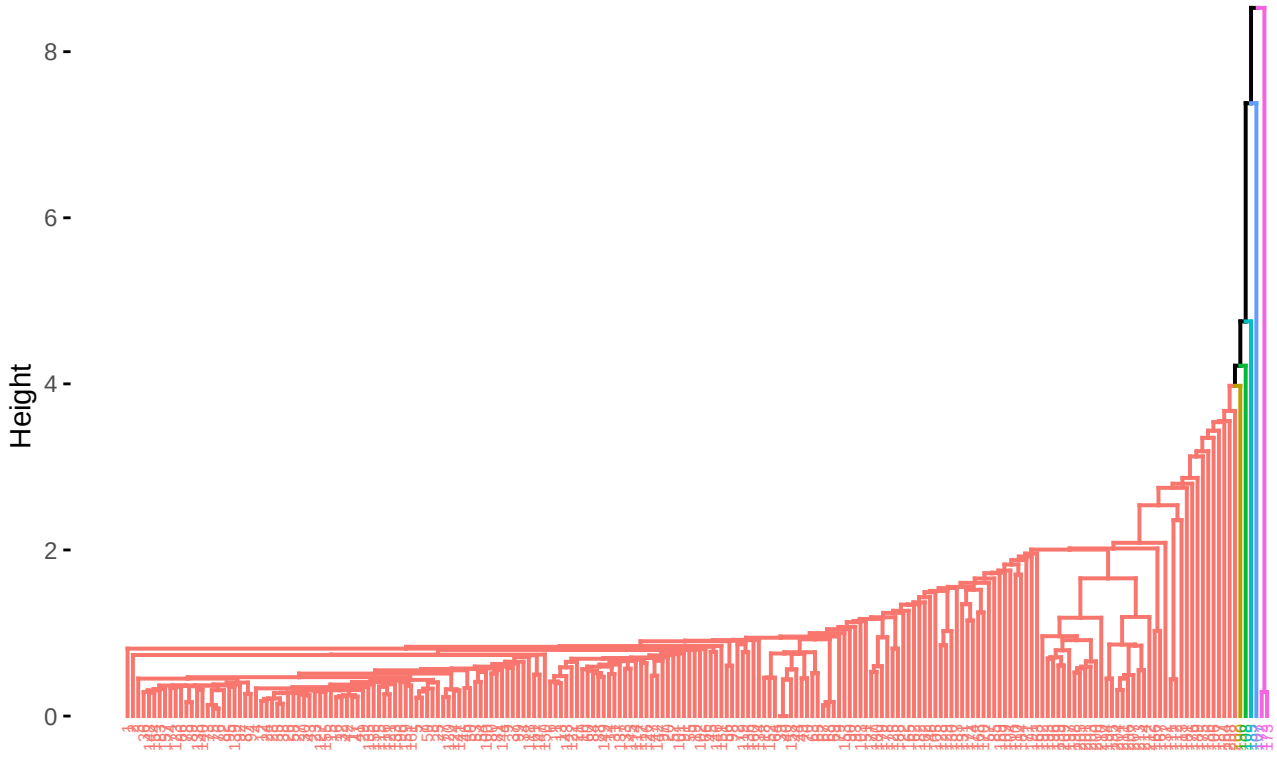
Kluczowym aspektem działania AGNES jest sposób definiowania odległości między klastrami. W niniejszej analizie rozważano trzy popularne metody:

- **Najbliższy sąsiad (single linkage)** – odległość między klastrami jest definiowana jako najmniejsza odległość między dowolną parą punktów z różnych klastrów. Metoda ta może prowadzić do wydłużonych, łańcuchowatych klastrów i jest wrażliwa na szum.
- **Najdalszy sąsiad (complete linkage)** – uwzględnia największą możliwą odległość między parami punktów z różnych klastrów, co skutkuje bardziej zwartymi i wyraźnie odseparowanymi grupami.
- **Średnia odległość (average linkage)** – opiera się na średniej odległości pomiędzy wszystkimi punktami w dwóch klastrach. Stanowi kompromis między dwoma powyższymi podejściami, oferując stabilne i zrównoważone grupowanie.

Dla każdej z metod dokonano cięcia dendrogramu na **6** klastrów — odpowiadających rzeczywistej liczbie klas w zbiorze danych *glass* — co umożliwia porównanie wyników z etykietami klas i ocenę jakości odwzorowania struktury danych przez algorytm AGNES.

2.2.3.1 Najbliższy sąsiad (*Single Linkage*)

W tej części zastosowano metodę **najbliższego sąsiada** w ramach algorytmu AGNES.



Wykres 10: AGNES - najbliższy sąsiad

Na Wykresie 10. przedstawiono dendrogram otrzymany w wyniku działania algorytmu AGNES z zastosowaniem metody najbliższego sąsiada

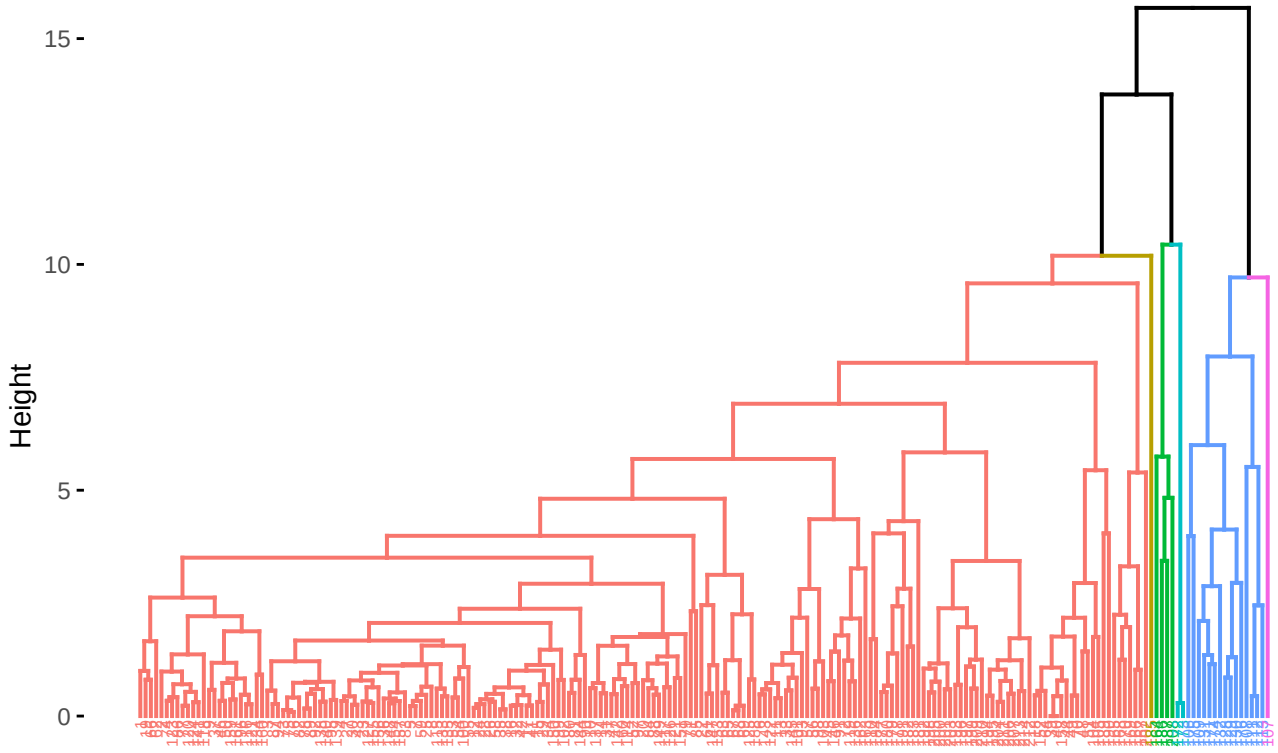
Tabela 29: Macierz pomyłek - AGNES - najbliższy sąsiad

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	70	74	17	11	8	28
2	0	1	0	0	0	0
3	0	1	0	0	0	0
5	0	0	0	2	0	0
6	0	0	0	0	1	0
7	0	0	0	0	0	1

Dokładność klasyfikacji uzyskana metodą AGNES z wykorzystaniem najbliższego sąsiada wynosi 36.45%.

Wynik ten należy ocenić jako **bardzo niski**, co znajduje odzwierciedlenie na Wykresie 10. – większość obserwacji została przypisana do jednej, dominującej grupy, a pozostałe klastry są nieliczne lub niemal puste. Taki efekt jest typowy dla metody najbliższego sąsiada, która ma tendencję do łączenia punktów w wydłużone struktury, często prowadzące do *efektu łańcucha* i zaburzenia rzeczywistej struktury klas.

2.2.3.2 Najdalszy sąsiad (*Complete Linkage*) W tej części zastosowano metodę najdalszego sąsiada w ramach algorytmu AGNES.



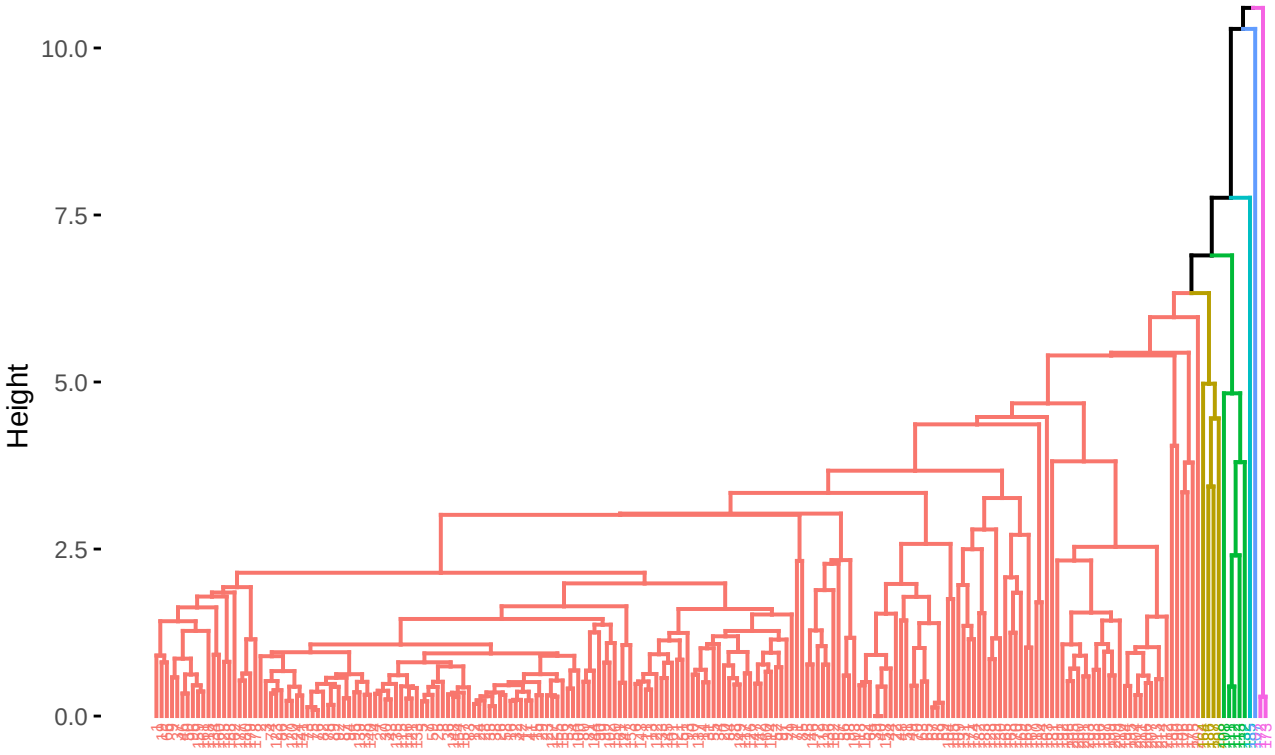
Wykres 11: AGNES - najdalszy sąsiad

Tabela 30: Macierz pomyłek - AGNES - najdalszy sąsiad

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	70	64	17	6	8	26
2	0	11	0	4	0	0
3	0	1	0	0	0	0
5	0	0	0	1	0	3
6	0	0	0	2	0	0
7	0	0	0	0	1	0

Dokładność klasyfikacji uzyskana metodą AGNES z wykorzystaniem najdalszego sąsiada wynosi 40.65%. Wynik jest lepszy niż z zastosowaniem najbliższego sąsiada.

2.2.3.3 Średnia odległość (*Average Linkage*) W tej części zastosowano metodę **średniej odległości** w ramach algorytmu AGNES.



Wykres 12: AGNES - średnia odległość

Tabela 31: Macierz pomyłek - AGNES - średnia odległość

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	70	70	17	10	8	26
2	0	1	0	0	0	0
3	0	5	0	0	0	0
5	0	0	0	1	0	3
6	0	0	0	2	0	0
7	0	0	0	0	1	0

Dokładność klasyfikacji uzyskana metodą AGNES z wykorzystaniem średniej odległości wynosi **37.85%**, co również stanowi wynik niski.

Jak pokazują uzyskane rezultaty, wszystkie zastosowane warianty algorytmu AGNES charakteryzują się niską skutecznością w odzwierciedlaniu rzeczywistej struktury klas w zbiorze *glass*. Oznacza to, że metody hierarchiczne, w tym przypadku, nie sprawdziły się w analizie tego konkretnego zbioru danych.

2.2.4 Divisive Clustering (DIANA)

Metoda **DIANA (Divisive Analysis)** jest podejściem hierarchicznego grupowania typu dzielącego (divisive), które działa odwrotnie do metod aglomeracyjnych. Zamiast łączyć pojedyncze obserwacje w klastry, DIANA rozpoczyna od traktowania całego zbioru danych jako jednej grupy i iteracyjnie dzieli ją na mniejsze klastry, aż do uzyskania oczekiwanej liczby podgrup lub pojedynczych elementów.

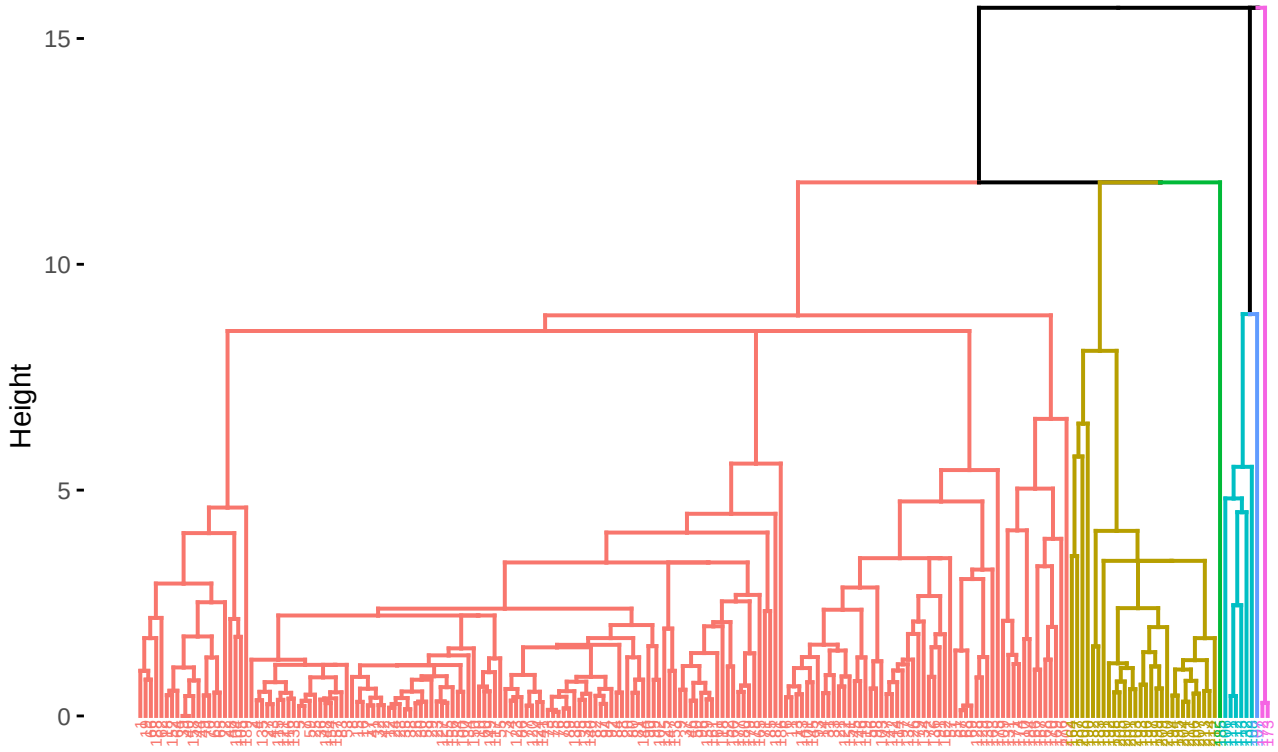
W trakcie procesu podziału DIANA wybiera klaster o największej heterogeniczności, a następnie wyodrębnia z niego obserwacje najbardziej odległe od pozostałych, tworząc nowe skupienia.

Do mierzenia odległości między punktami wykorzystywane są najczęściej metryki:

- **Euklidesowa** — klasyczna odległość geometryczna w przestrzeni n-wymiarowej, która mierzy „prosto" liniową odległość między punktami,
- **Manhattan (lub taksówkowa)** — suma wartości bezwzględnych różnic między współrzędnymi, przypominająca dystans pokonywany wzdłuż osi prostopadłych (jak ulice w siatce miejskiej).

Wybór metryki wpływa na kształt i strukturę powstałych klastrów, a także na wrażliwość algorytmu na specyfikę danych. DIANA z wykorzystaniem obu tych metryk pozwoliła na szczegółową analizę podziału zbioru *glass* i porównanie efektów z innymi metodami grupowania.

2.2.4.1 Odległość euklidesowa W tej części zastosowano metrykę **odległości euklidesowej**, która jest najczęściej wykorzystywaną miarą dystansu w analizie skupień.



Wykres 13: DIANA - odległość euklidesowa

Tabela 32: Macierz pomyłek - DIANA - odległość euklidesowa

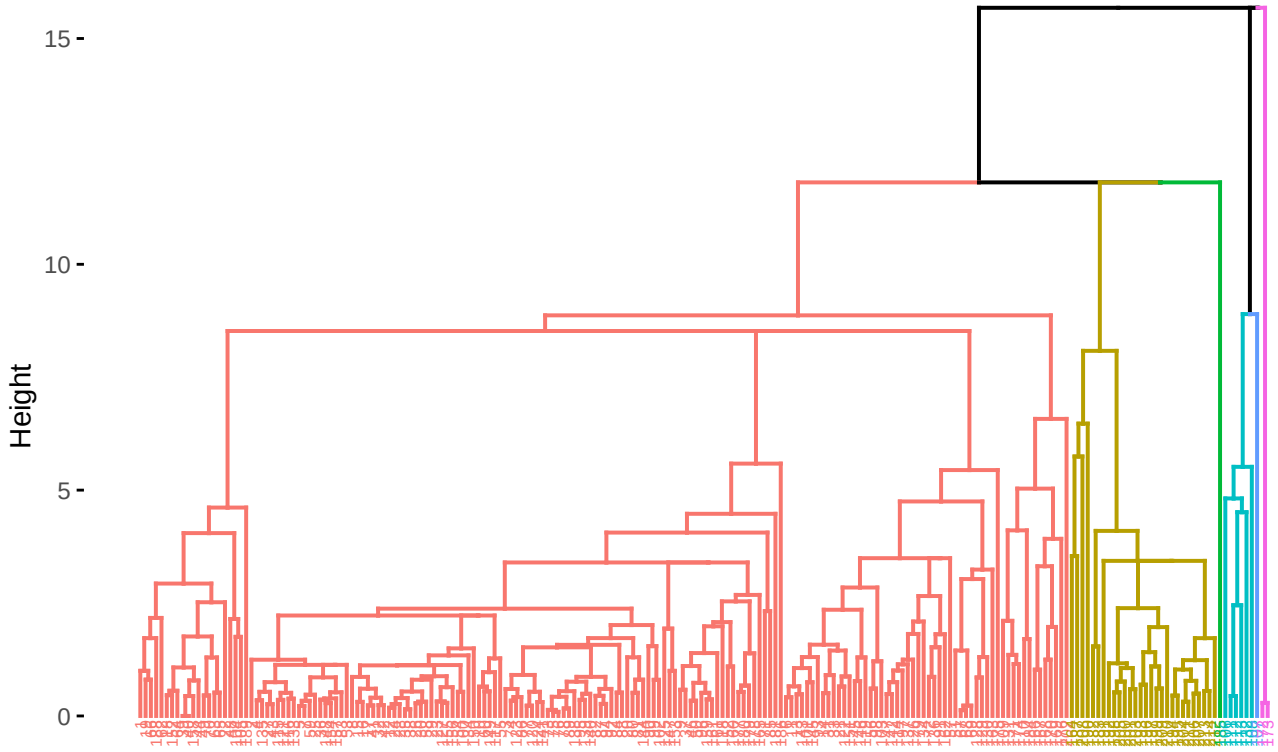
Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	70	69	17	10	6	4
2	0	6	0	0	0	0
3	0	1	0	0	0	0
5	0	0	0	1	2	25

Tabela 32: Macierz pomyłek - DIANA - odległość euklidesowa (kontynuacja)

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
6	0	0	0	2	0	0
7	0	0	0	0	1	0

Dokładność klasyfikacji uzyskana metodą DIANA z wykorzystaniem odległości euklidesowej wynosi 48.6%. Wynik lepszy niż w przypadku poprzednich metod.

2.2.4.2 Odległość Manhattan (taksówkowa) W tej części zastosowano metrykę odległości Manhattan (znaną także jako odległość taksówkową), która mierzy dystans między punktami jako sumę wartości bezwzględnych różnic ich współrzędnych.



Wykres 14: DIANA - odległość manhattańska

Tabela 33: Macierz pomyłek - DIANA - odległość manhattańska

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
1	70	69	17	10	6	4
2	0	6	0	0	0	0

Tabela 33: Macierz pomyłek - DIANA - odległość manhattańska (kontynuacja)

Przewidywana klasa	Prawdziwa klasa					
	1	2	3	5	6	7
3	0	1	0	0	0	0
5	0	0	0	1	2	25
6	0	0	0	2	0	0
7	0	0	0	0	1	0

Dokładność klasyfikacji uzyskana metodą DIANA z wykorzystaniem odległości manhattańskiej wynosi 48.6%.

Jak widać, niezależnie od zastosowanej metryki — zarówno euklidesowej, jak i Manhattan — wyniki klasyfikacji uzyskane za pomocą algorytmu DIANA pozostają niezmienne.

2.3 Ocena jakości grupowania

W celu kompleksowej oceny jakości uzyskanych grupowań porównamy przypisania obserwacji do klastrów z rzeczywistymi klasami dostępnymi w zbiorze danych. Pozwoli to ocenić, na ile efekty grupowania odpowiadają faktycznej strukturze danych.

Dodatkowo dokonamy doboru optymalnej liczby skupień, wykorzystując trzy wskaźniki wewnętrzne:

- **Connectivity** — mierzy spójność klastrów, uwzględniając bliskość obserwacji należących do tej samej grupy; **niższe wartości** oznaczają lepszą jakość skupień,
- **Wskaźnik Dunna** — stosunek minimalnej odległości międzyklastrowej do maksymalnego rozmiaru klastra; **wyższe wartości** wskazują na dobrze odseparowane i zwarte klastry,
- **Indeks Silhouette** — ocenia stopień dopasowania obserwacji do własnego klastra w porównaniu z sąsiednimi grupami; **wartości bliskie 1** świadczą o wyraźnym podziale i dobrze zdefiniowanych skupiskach.

Analiza tych miar pozwoli na wybór liczby klastrów, która najlepiej odzwierciedla strukturę danych, niezależnie od rzeczywistej liczby i rzeczywistych etykiet klas.

2.3.1 Ocena za pomocą wskaźników zewnetrznych

Do oceny jakości uzyskanych grupowań w niniejszej analizie wykorzystano wskaźnik **dokładności klasyfikacji**.

Tabela 34: Porównanie dokładności różnych metod klasteryzacji

Metoda	Dokladnosc
DIANA - odległość euklidesowa	48.60
DIANA - odległość manhattańska	48.60
PAM	42.99
k-średnich	41.59
AGNES - najdalszy sąsiad	40.65
AGNES - średnia odległość	37.85
AGNES - najbliższy sąsiad	36.45

Na podstawie danych przedstawionych w Tabeli 34. można wywnioskować, że najlepsze wyniki klasyfikacji osiąga algorytm **DIANA** wykorzystujący metrykę **euklidesową**.

Wskaźnik dokładności dla tej konfiguracji jest najwyższy spośród analizowanych metod, co wskazuje na najskuteczniejsze odwzorowanie rzeczywistej struktury klas w zbiorze danych.

2.3.2 Ocena za pomocą wskaźników wewnętrznych

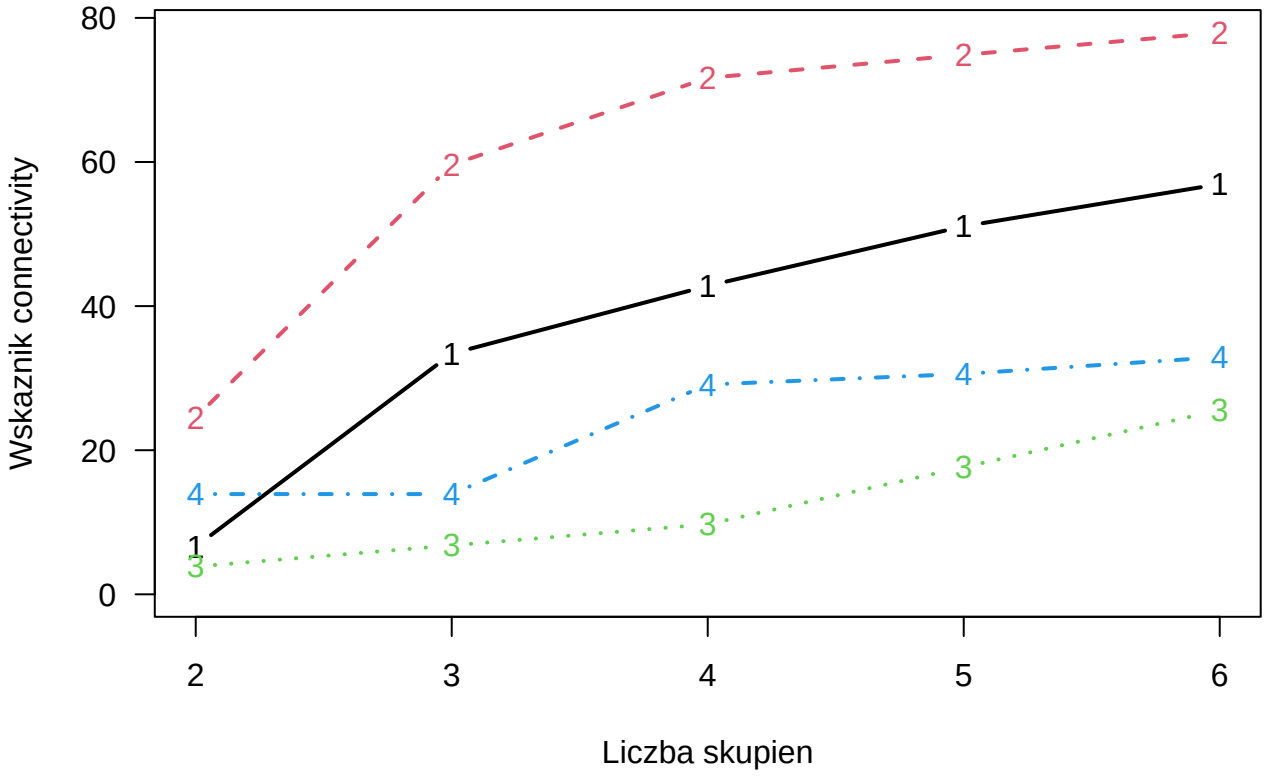
Do oceny jakości grupowania bez odwołania do rzeczywistych etykiet zastosowano trzy wskaźniki wewnętrzne:

- Connectivity
- Wskaźnik Dunna
- Indeks Silhouette

Ze względu na niewielki wpływ wyboru metody łączenia na strukturę wynikowego dendrogramu w algorytmach AGNES i DIANA, w dalszej analizie przyjęto ich podstawowe ustawienia.

We wszystkich kolejnych wykresach zastosowana będzie następująca legenda:

- 1 — k-średnie,
- 2 — PAM,
- 3 — AGNES,
- 4 — DIANA.

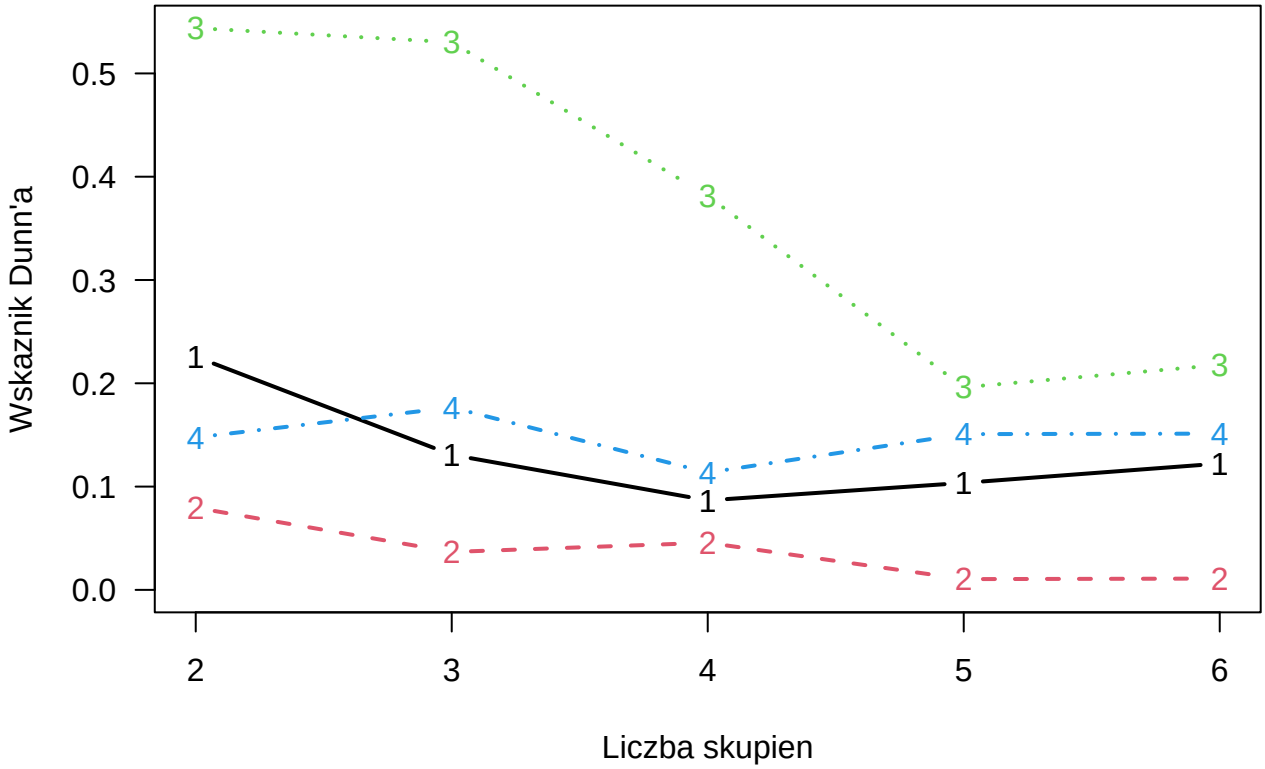


Wykres 15: Connectivity dla metod klasteryzacji nienadzorowanej

Wyniki wskaźnika **Connectivity** wskazują, że najlepszą jakość skupień osiągnięto przy następującej liczbie klastrów:

- **kmeans** – 2 klastry,
- **PAM** – 2 klastry,
- **AGNES** – 2 klastry (najlepszy wynik spośród wszystkich metod),
- **DIANA** – 3 klastry.

Ponieważ dla Connectivity niższe wartości oznaczają lepszą spójność i zwartość skupień, wynik ten świadczy, że algorytm AGNES przy dwóch klastrach generuje najbardziej zwarte i jednorodne grupy spośród analizowanych metod.

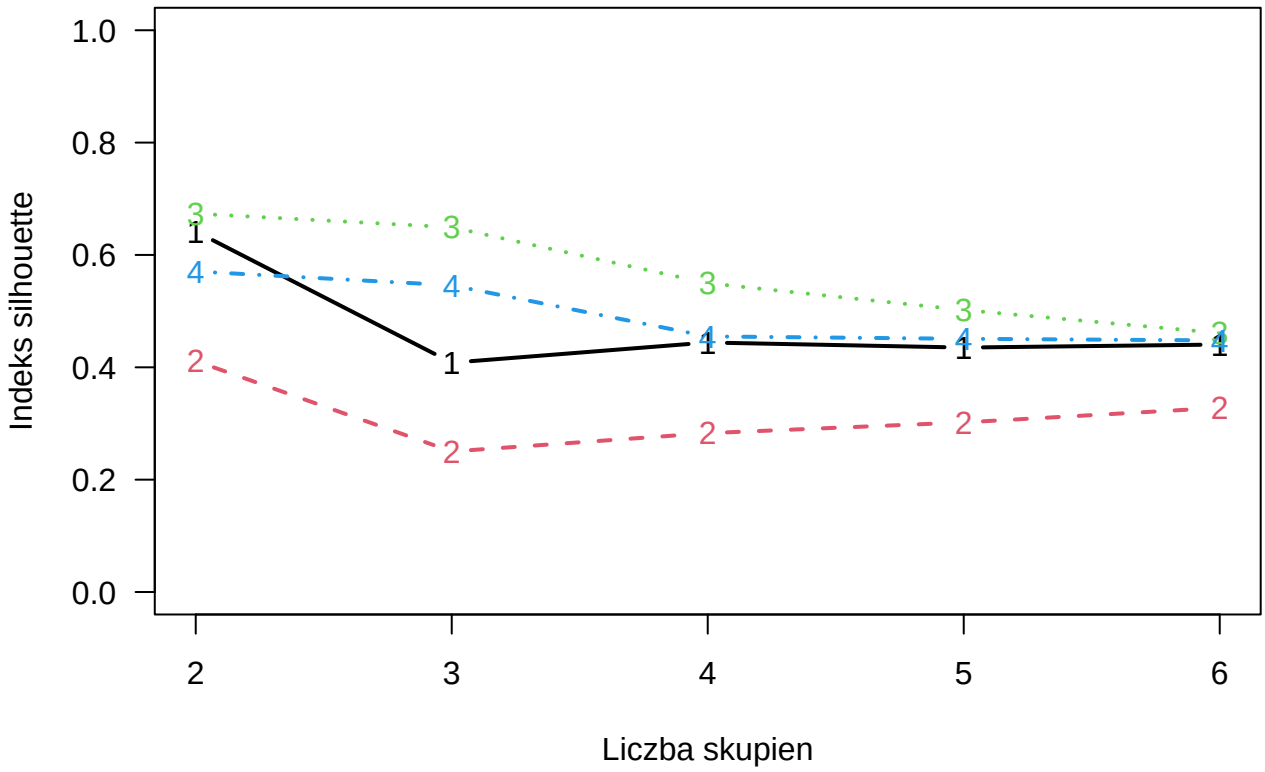


Wykres 16: Wskaźnik Dunn dla metod klasteryzacji nienadzorowanej

Wyniki wskaźnika **Dunn** dla poszczególnych metod przedstawiają się następująco:

- **kmeans** – 2 klastry,
- **PAM** – 2 klastry (najlepszy wynik),
- **AGNES** – 2 klastry,
- **DIANA** – 3 klastry.

Ponieważ wskaźnik Dunn rośnie wraz z poprawą separacji i zwartości klastrów, oznacza to, że metoda **PAM** przy dwóch klastrach zapewnia najbardziej zwarte i wyraźnie rozdzielone skupiska spośród analizowanych algorytmów.



Wykres 17: Indeks Silhouette dla metod klasteryzacji nienadzorowanej

Wyniki wskaźnika **Silhouette** dla poszczególnych metod przedstawiają się następująco:

- **kmeans** – 2 klastry,
- **PAM** – 2 klastry (najlepszy wynik),
- **AGNES** – 2 klastry,
- **DIANA** – 2 klastry.

Ponieważ wartość wskaźnika Silhouette rośnie wraz z poprawą zwartości i separacji klastrów, oznacza to, że metoda **PAM** przy dwóch klastrach zapewnia najbardziej wyraźnie zdefiniowane i dobrze odseparowane skupiska spośród analizowanych algorytmów.

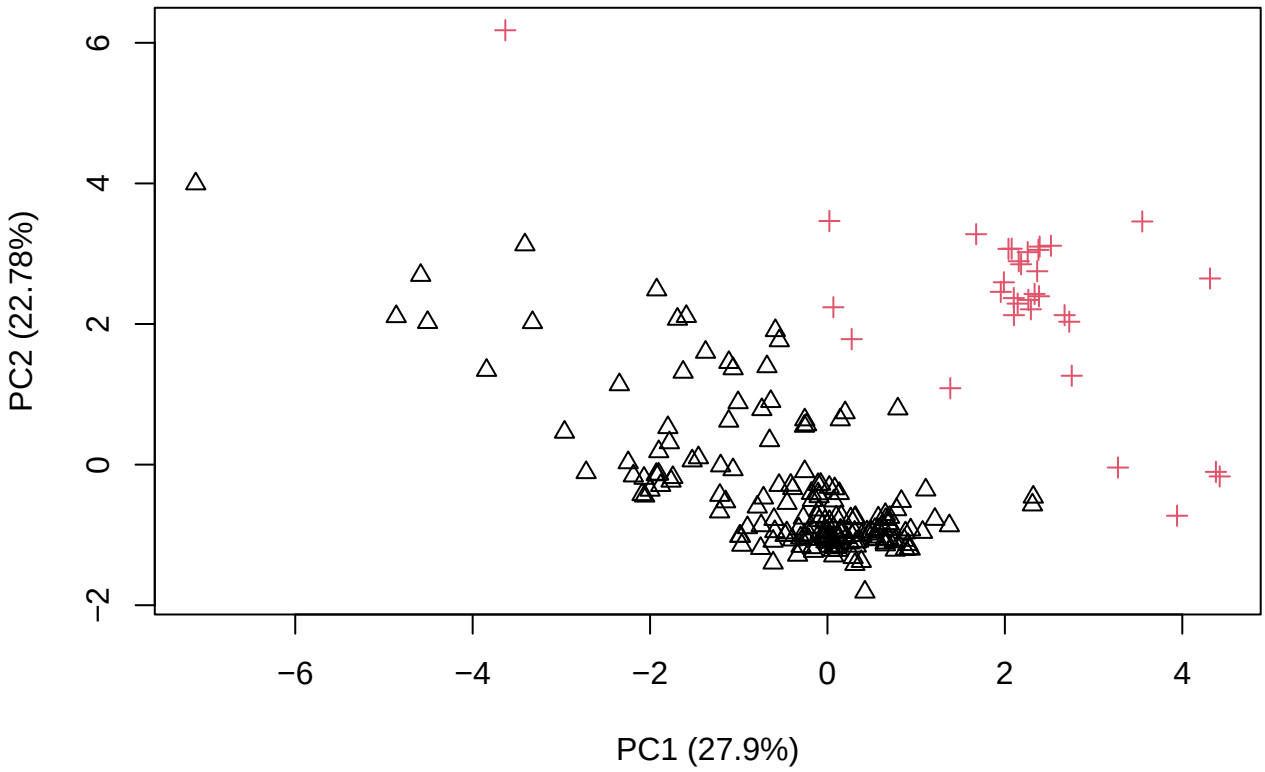
Na podstawie otrzymanych wyników uznaliśmy, że optymalną liczbą skupień jest **2**.

2.3.3 Porównanie metod dla optymalnej liczby skupień

W tej części dokonamy porównania skuteczności zastosowanych metod grupowania przy założeniu optymalnej liczby skupień równej **2**.

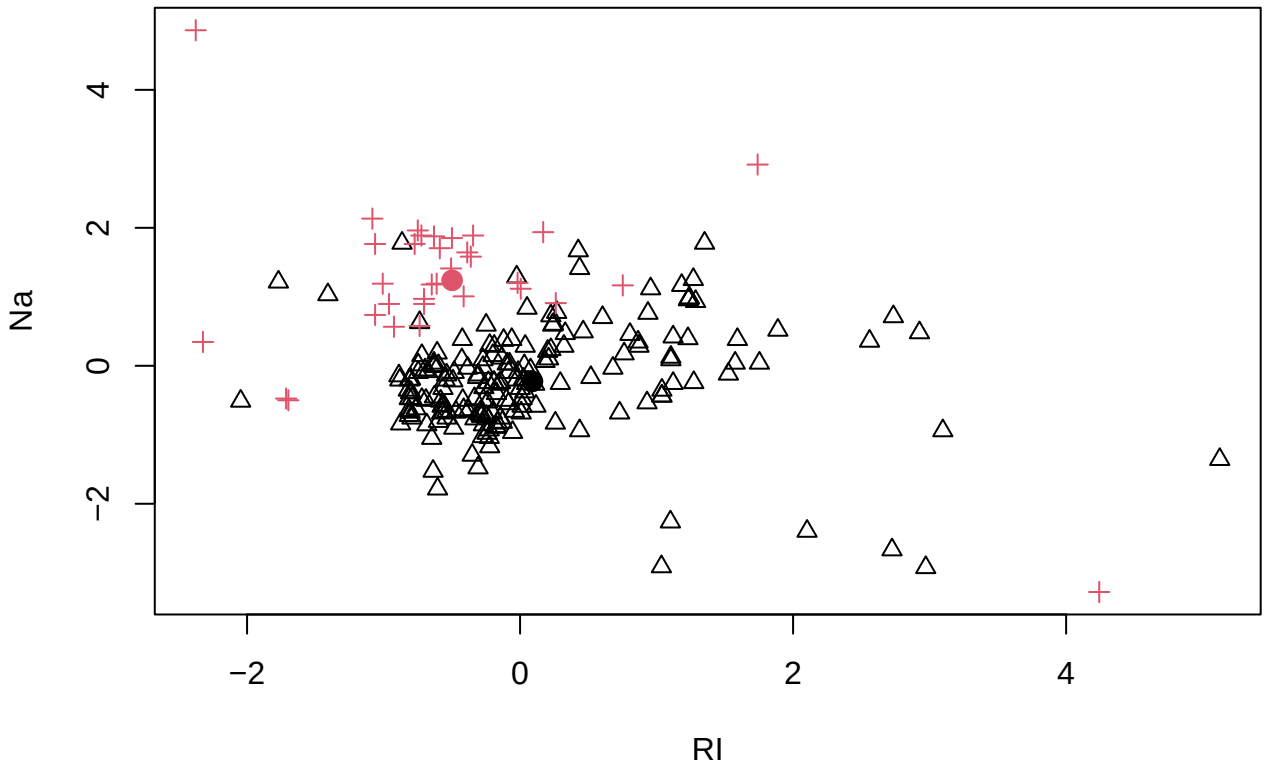
Celem jest wskazanie metody, która najlepiej odwzorowuje rzeczywistą strukturę danych oraz zapewnia zwarte i dobrze rozdzielone klastry.

2.3.3.1 k-średnie Sprawdzamy, jak metoda **k-średnich** radzi sobie przy założeniu optymalnej liczby klastrów równej **2**.



Wykres 18: Wizualizacja danych po grupowaniu metodą k-średnich z optymalną liczbą klastrów - PCA

Z Wykresu 18. można odczytać że metodzie k-średnich dobrze poszło z podziałem na dwa klastry.

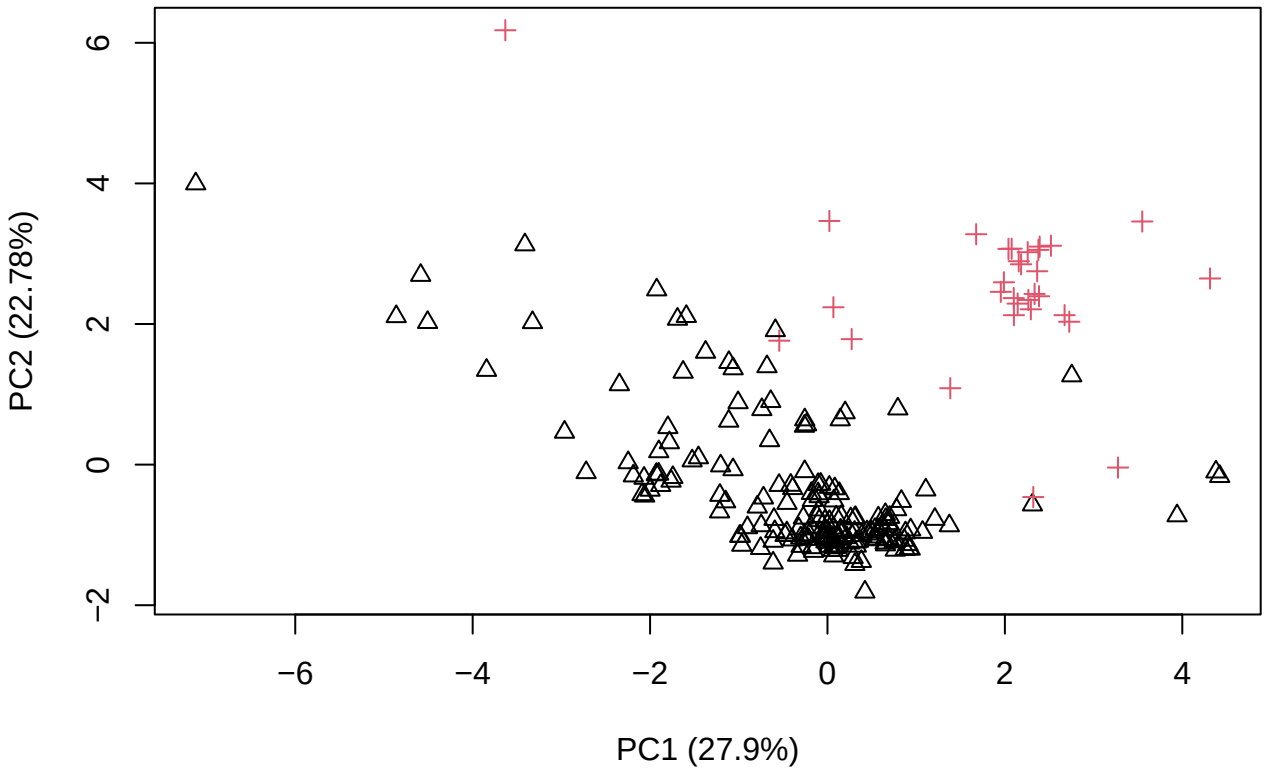


Wykres 19: Zależnośc współczynnika załamania światła od procentu masowego sodu z kolorami i kształtami klas dla optymalnej liczby klastrów - zaznaczone centroidy z metody k-średnich

Na Wykresie 19. widać że centra znajdują się w dorbych miejsach.

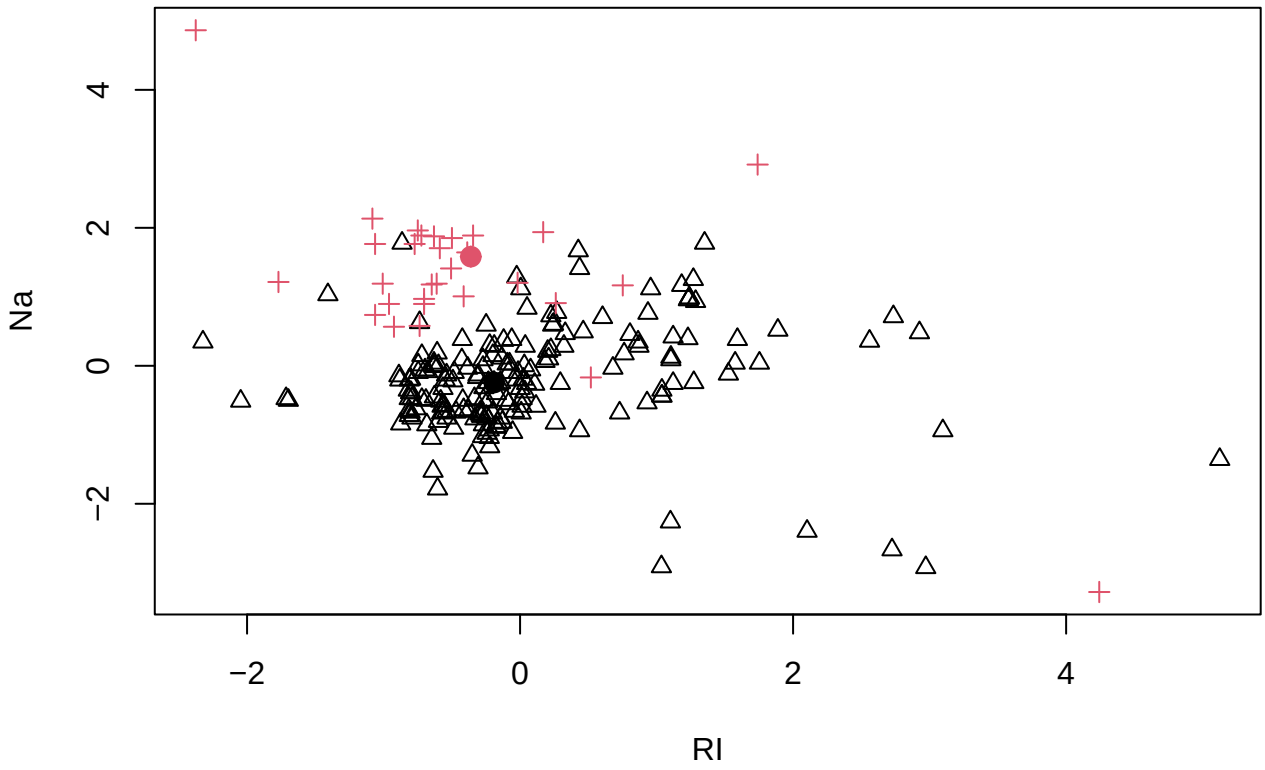
Dokładność klasyfikacji uzyskana metodą k-średnich wynosi 96.19%.

2.3.3.2 PAM Sprawdzamy, jak metoda **PAM** radzi sobie przy założeniu optymalnej liczby klastrów równej **2**.



Wykres 20: Wizualizacja danych po grupowania metodą PAM z optymalną liczbą klastrów - PCA

Na Wykresie 20. również widać dobrą klasyfikację danych.



Wykres 21: Zależnośc współczynnika załamania światła od procentu masowego sodu z kolorami i kształtami klas z optymalną liczbą klastrów - zaznaczone medoidy z metody PAM

Z Wykresu 21. można wywnioskować, że medoidy zostały umiejscowione w odpowiednich punktach przestrzeni, co świadczy o dobrej reprezentatywności tych obiektów dla poszczególnych klastrów.

Pomimo ograniczeń wizualizacji dwuwymiarowej, tło wskazuje, że większość obserwacji została poprawnie zakwalifikowana do odpowiadających im klas, co potwierdza skuteczność zastosowanego algorytmu.

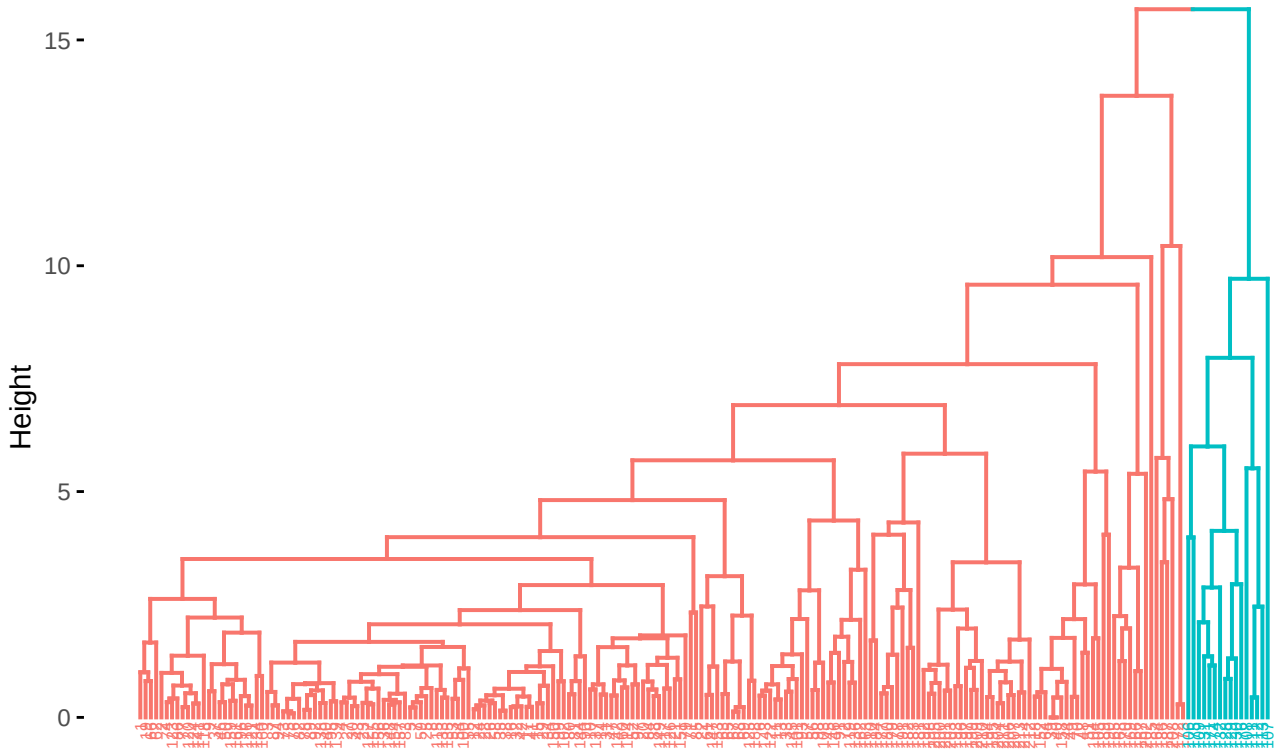
Tabela 35: Dane medoidów dla optymalnej liczby klas

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type	PAM
43	1.51779	13.21	3.39	1.33	72.76	0.59	8.59	0.00	0	1	1
198	1.51727	14.70	0.00	2.34	73.28	0.00	8.95	0.66	0	7	2

Z Tabeli 35. widać co charakteryzuje medoidy. Jest nimi zawartość Magnezu (Mg) i Potasu (K) dla jednej klasy i Baru (Ba) dla drugiej.

Dokładność klasyfikacji uzyskana metodą PAM z optymalną liczbą klastrów wynosi 94.29%.

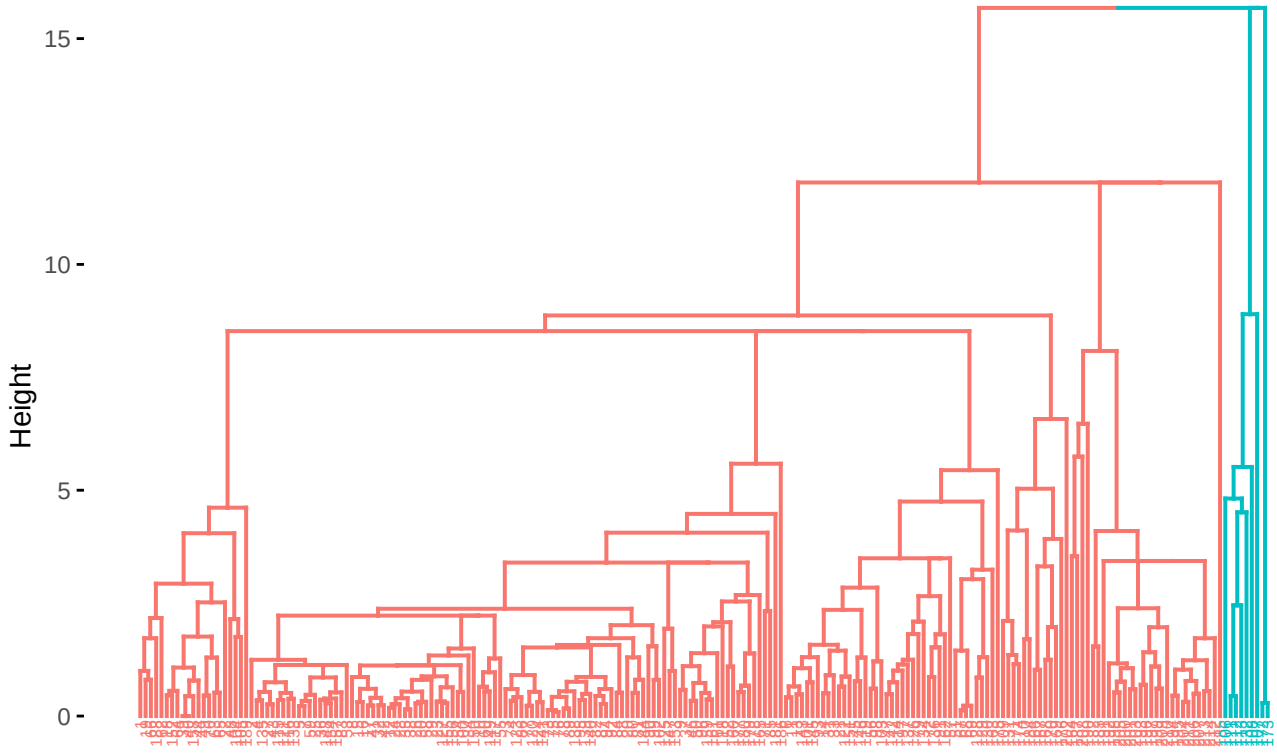
2.3.3.3 AGNES Sprawdzamy, jak metoda **AGNES** radzi sobie przy założeniu optymalnej liczby klastrów równej **2**.



Wykres 22: AGNES - najbliższy sąsiad - optymalna liczba klastrów

Dokładność klasyfikacji uzyskana metodą AGNES z optymalną liczbą klastrów wynosi 87.64%.

2.3.3.4 DIANA Sprawdzamy, jak metoda **DIANA** radzi sobie przy założeniu optymalnej liczby klastrów równej **2**.



Wykres 23: DIANA - odległość euklidesowa - optymalna liczba klastrów

Dokładność klasyfikacji uzyskana metodą k-średnich wynosi 52.74%.

Tabela 36: Porównanie dokładności różnych metod klasteryzacji - optymalna liczba klastrów

Metoda	Dokladnosc
k-średnich	96.19
PAM	94.29
AGNES	87.64
DIANA	52.74

Tabela 36. pokazuje, że najlepszą dokładność osiągnęła metoda **k-średnich** (96,19%), następnie **PAM** (94,29%). Metody hierarchiczne AGNES i DIANA uzyskały niższe wyniki — odpowiednio 87,64% i 52,74%.

2.4 Wnioski

- Najlepszą metodą do klasyfikacji **6 klastrów** okazała się **DIANA**.
- Metody nienadzorowane wskazały optymalną liczbę klastrów na **2**.
- Dla liczby skupień równej **2** najlepszym klasyfikatorem jest metoda **k-średnich**.

PS. Czas wykonywania kodu wynosi 4 minut i 37 sekund.