Sprawozdanie z projektu grupowego "Klasyfikacja rodzajów szkła"

Zespół 17: Maria Mierzejewska, Karol Cieślik, Jakub Matłacz

1 Wstęp

Zbiór danych, który analizowano w trakcie projektu oraz, na którym trenowano modele, można znaleźć pod poniższym linkiem:

https://www.kaggle.com/uciml/glass

Celem ćwiczenia była analiza zbioru oryginalnego, właściwe rozpoznanie wyzwań z nim związanych, rozwiązanie problemów stojących na drodze do zbudowania prawidłowych, działających modeli oraz wybranie takiego z nich, który pozwoli z największym prawdopodobieństwem prawidłowo klasyfikować próbki szkła jako należące do jednego z możliwych jego rodzajów (należało zastosować właściwe miary do określenia jakości modelu).

1 Obiekty i klasy w zbiorze oryginalnym

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
0	1.52101	13.64	4.49	1.10	71.78	0.06	8.75	0.00	0.0	1
1	1.51761	13.89	3.60	1.36	72.73	0.48	7.83	0.00	0.0	1
2	1.51618	13.53	3.55	1.54	72.99	0.39	7.78	0.00	0.0	1
3	1.51766	13.21	3.69	1.29	72.61	0.57	8.22	0.00	0.0	1
4	1.51742	13.27	3.62	1.24	73.08	0.55	8.07	0.00	0.0	1
...	...	...	...	...	...	...	...	...	...	...
209	1.51623	14.14	0.00	2.88	72.61	0.08	9.18	1.06	0.0	7
210	1.51685	14.92	0.00	1.99	73.06	0.00	8.40	1.59	0.0	7
211	1.52065	14.36	0.00	2.02	73.42	0.00	8.44	1.64	0.0	7
212	1.51651	14.38	0.00	1.94	73.61	0.00	8.48	1.57	0.0	7
213	1.51711	14.23	0.00	2.08	73.36	0.00	8.62	1.67	0.0	7

W zbiorze oryginalnym jest obecnych 214 obiektów (próbek szkła z przypisaną prawidłową klasą). Każdy obiekt posiada 9 atrybutów opisujących oraz 1 atrybut decyzyjny (tak zwaną klasę). Twórcy zbioru wyróżnili 7 możliwych klas obiektów. Jednak jedna z nich nie posiada instancji (klasa 4) przez co w rzeczywistości klas jest 6.

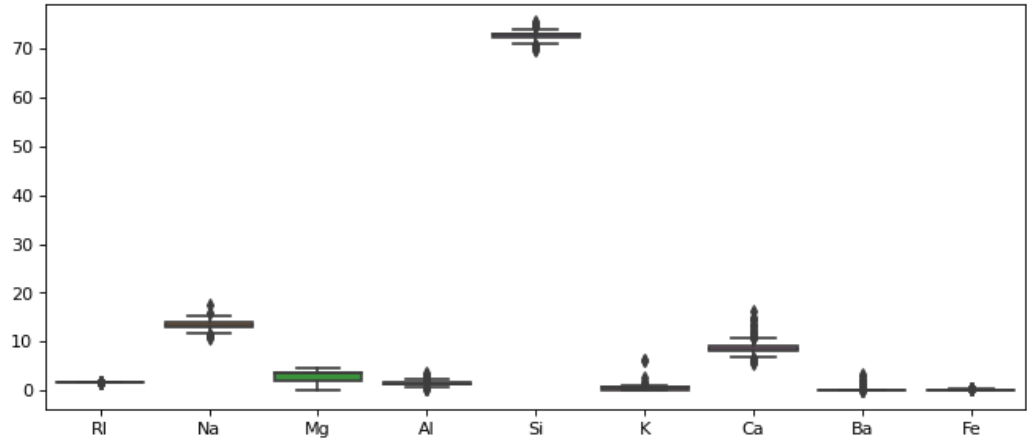
Oznaczenia klas:
-- 1 building windows float processed
-- 2 building windows non float processed
-- 3 vehicle windows float processed
-- 4 vehicle windows non float processed (none in this database)
-- 5 containers
-- 6 tableware
-- 7 headlamps

2 Braki danych w zbiorze oryginalnym

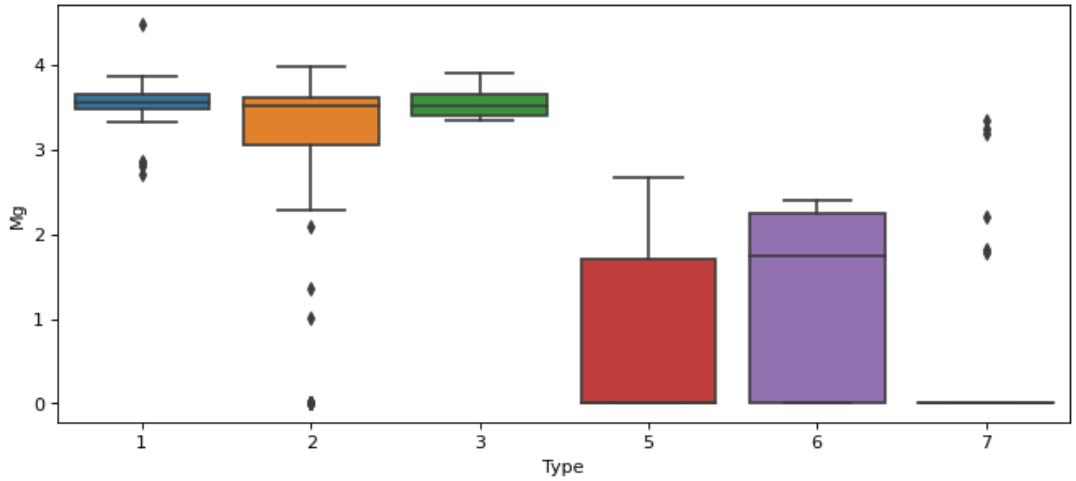
Zbiór oryginalny nie posiada żadnych braków w danych. Ilość niepustych wartości pod każdym z atrybutów jest równa ilości obiektów w zbiorze.

#	Column	Non-Null Count
0	RI	214 non-null
1	Na	214 non-null
2	Mg	214 non-null
3	Al	214 non-null
4	Si	214 non-null
5	K	214 non-null
6	Ca	214 non-null
7	Ba	214 non-null
8	Fe	214 non-null
9	Type	214 non-null

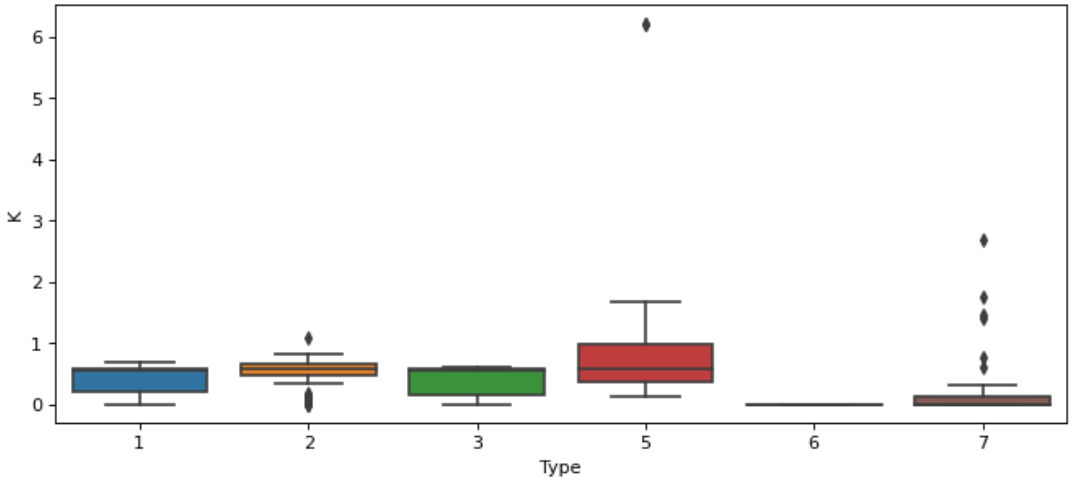
3 Wykresy pudełkowe w zbiorze oryginalnym



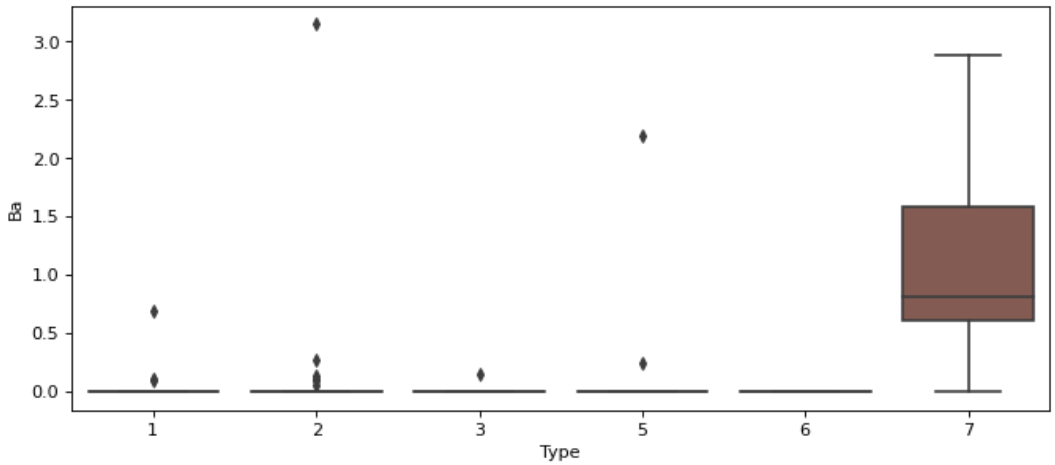
Wartości atrybutu Si (wykres powyżej bez podziału na klasy) są znacznie większe od wartości pozostałych atrybutów.



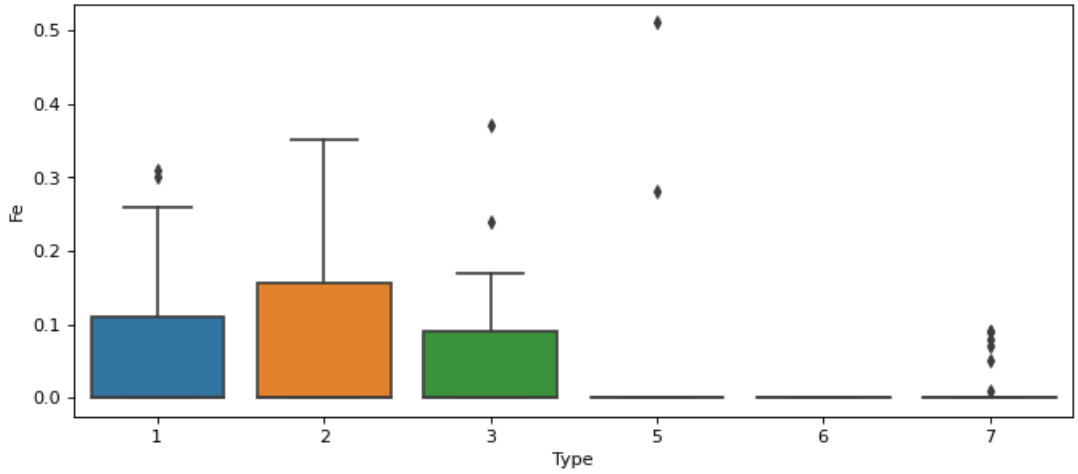
Wartość atrybutu Mg dość dobrze oddziela klasy 1,2,3 od 5,6. Wartość Mg w klasie 7 jest prawie zawsze zerowa.



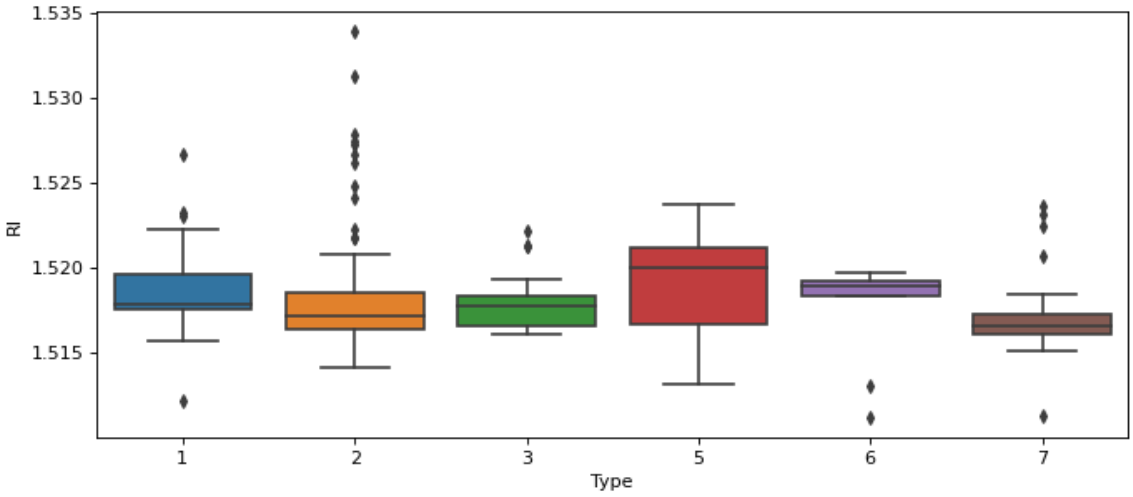
Wartość K w klasie 6 jest prawie zawsze zerowa.

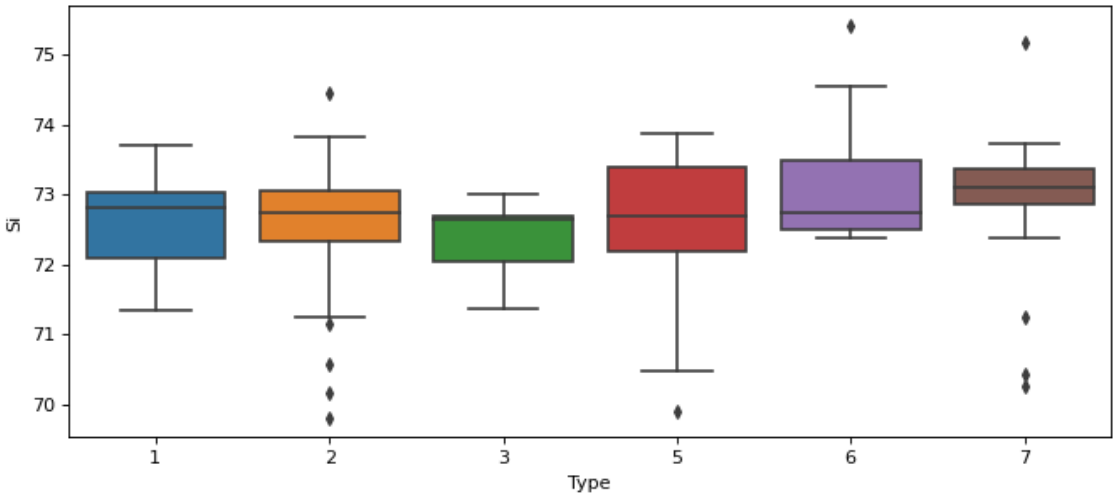


Wartość Ba świetnie wyznacza klasę 7 (inne klasy mają jej wartość prawie zawsze zerową).

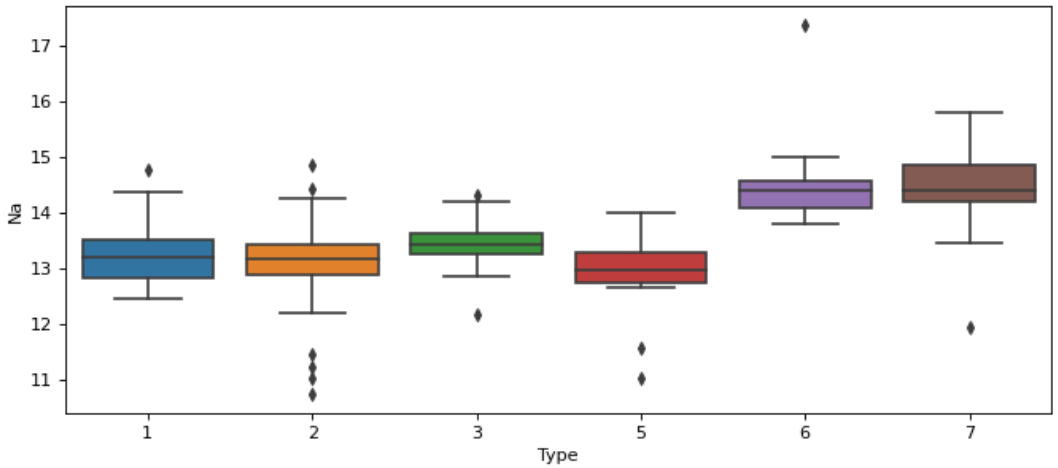


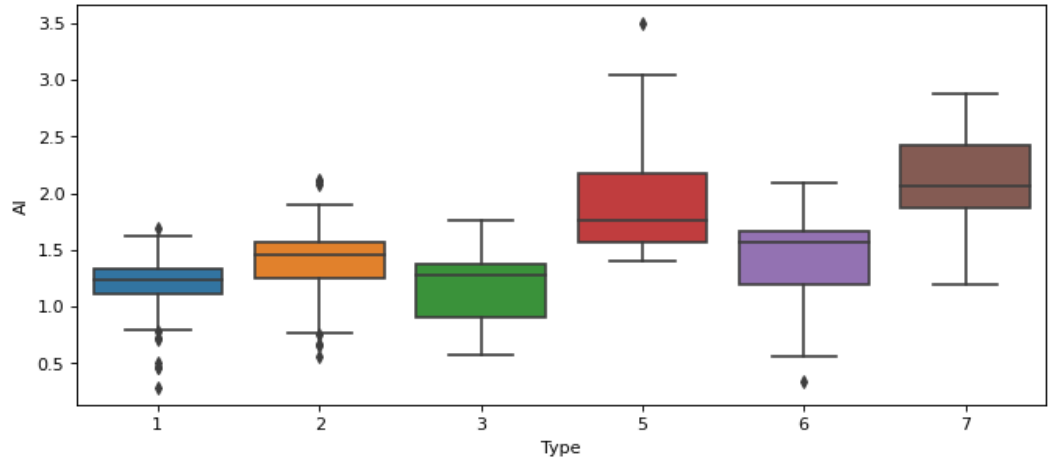
Wartość Fe jest prawie zawsze zerowa w klasach 5,6,7, więc świetnie oddziela je od klas 1,2,3.

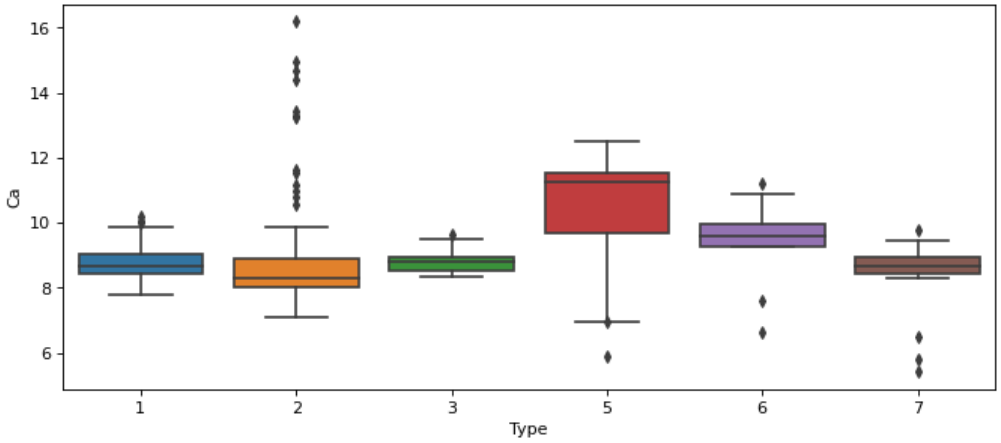




Przedział wartości atrybutu RI (Refractive Index) jest we wszystkich klasach podobny. Podobnie jak wartość atrybutu Si.



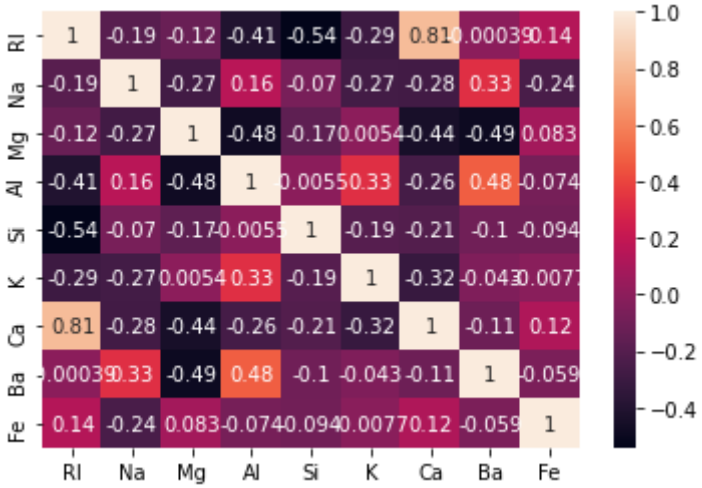




Wartość atrybutu Na oddziela klasy 1,2,3,5 od 6,7 tylko w niewielkim stopniu, ponieważ istnieje spory przedział wartości wspólny dla obu grup klas. W bardzo analogiczny sposób wartość Al oddziela 1,2,3,6 od 5,7 oraz Ca oddziela 1,2,3,7 od 5,6 (oba w bardzo niewielkim stopniu). Jednak nawet jeśli pojedynczy atrybut słabo oddziela grupy klas to w połączeniu z innymi (nawet słabo oddzielającymi atrybutami) mogą być bardzo pomocne dla budowania modelu, gdyż każdy z nich zawiera jakąś częściową informację, a jeśli będą działać razem to całościowa informacja może okazać się wystarczająca, aby z zadowalającym prawdopodobieństwem prawidłowo klasyfikować obiekty (na co należy liczyć przy każdym zbiorze uczącym – że całościowo posiada w sobie informacje, które to umożliwią).

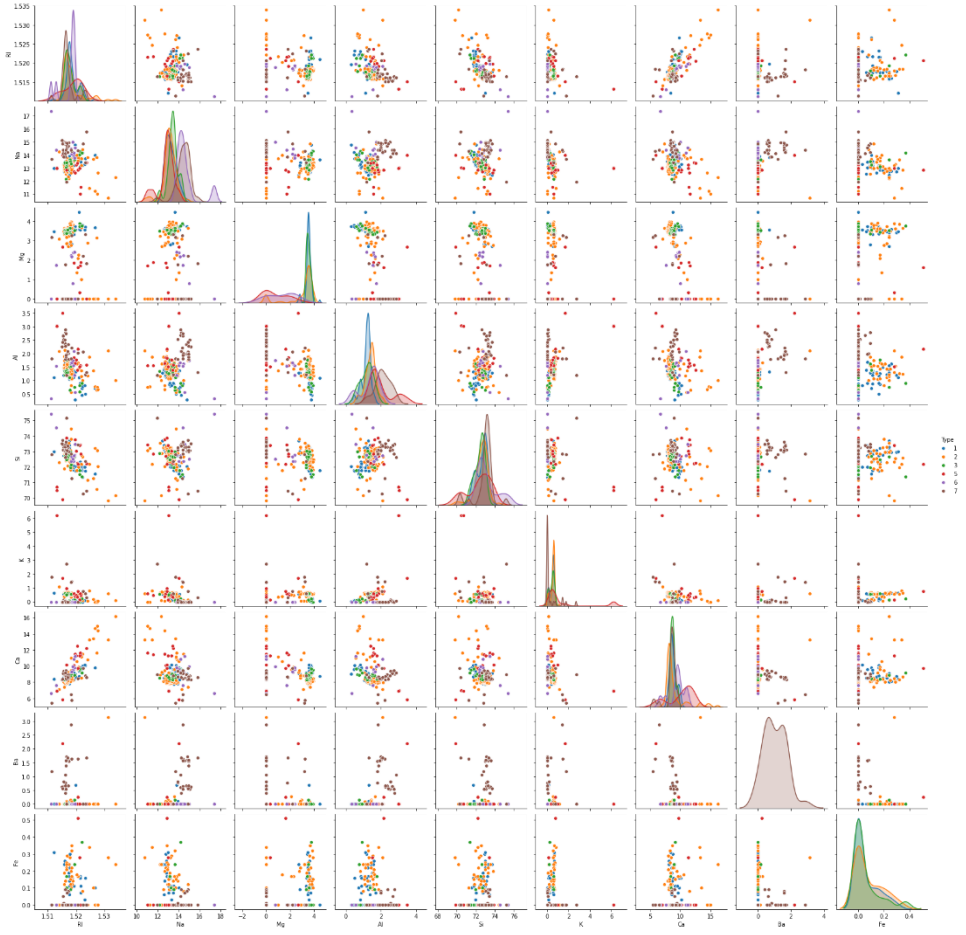
4 Macierz korelacji zbioru oryginalnego

Macierz korelacji Pearsona w celu ewentualnej eliminacji zbędnych atrybutów. Widać, że cecha Ca jest mocno skorelowana z cechą RI jednak nie wystarczająco, aby pominąć jedną z nich. Drugi najwyższy współczynnik korelacji dla Al oraz Ba jednak wciąż mały.

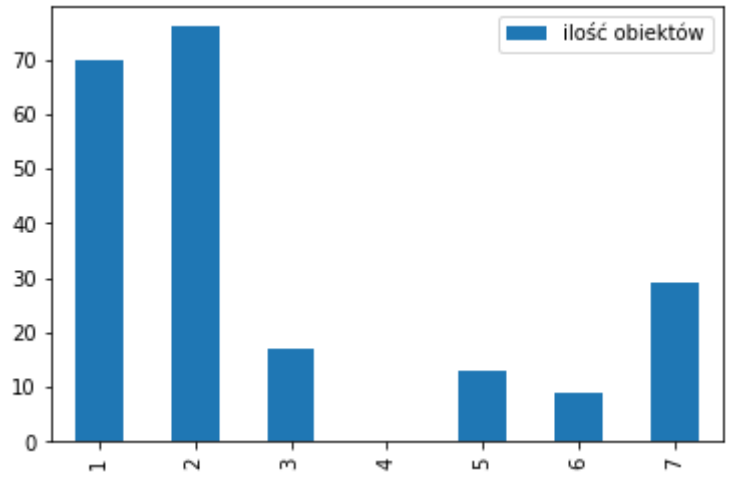


5 Macierz wykresów punktowych zbioru oryginalnego

Macierz wykresów punktowych w celu lepszej wizualizacji zbioru. Widać, że cecha Ba może być dobrym wyznacznikiem przynależności obiektu do klasy 7. Wiele z histogramów przenika się w ogromnym stopniu. Wiele atrybutów nie może samodzielnie dawać dużego pojęcia o przynależności obiektu do klasy gdyż histogramy mają duże części wspólne. Wnioski o cechach rozdzielających klasy łatwiej było wyciągnąć patrząc na wykresy pudełkowe powyżej.



6 Dystrybucja klas w zbiorze oryginalnym



	ilość obiektów
1	70
2	76
3	17
4	0
5	13
6	9
7	29

Zbiór oryginalny jest niezbalansowany, czyli klasy nie posiadają takiej samej ilości instancji. Idealnie zbalansowany zbiór zdarza się oczywiście bardzo rzadko, lecz jeśli jest tak drastycznie niezbalansowany jak w tym przypadku, to z całą pewnością należy zwrócić na to uwagę i problem ten rozwiązać. Nie chcemy przecież, aby nasze modele ucząc się, zignorowały klasę mającą w porównaniu z inną klasą np. 8 razy mniej obiektów. Chcemy być w stanie w zadowalający sposób klasyfikować obiekty do wszystkich klas – także tych o małej liczbie instancji.

7 Przygotowanie danych do uczenia i testowania

Skorzystamy z walidacji krzyżowej StratifiedKFold z liczbą splitów (różnych losowych podziałów zbioru na część uczącą i testową) równą 9 (ogranicza nas ilość obiektów najsłabiej reprezentowanej klasy w zbiorze równa właśnie 9).

W każdym splicie będziemy oversamplować część uczącą. Użyjemy więc metody oversamplingu czyli dogenerowania sztucznych obiektów SMOTE, która rysuje między obiektami danej klasy odcinki i tworzy nowe obiekty na nich leżące. Celem tego zabiegu jest wyrównanie wpływu każdej z klas na wynik uczenia tak, aby klasy najlepiej reprezentowane nie zdominowały modelu co mogłoby skutkować tym, że zdecydowana większość obiektów byłaby klasyfikowana jako klasy 2 lub 1.

8 Strategia oceny jakości modelu

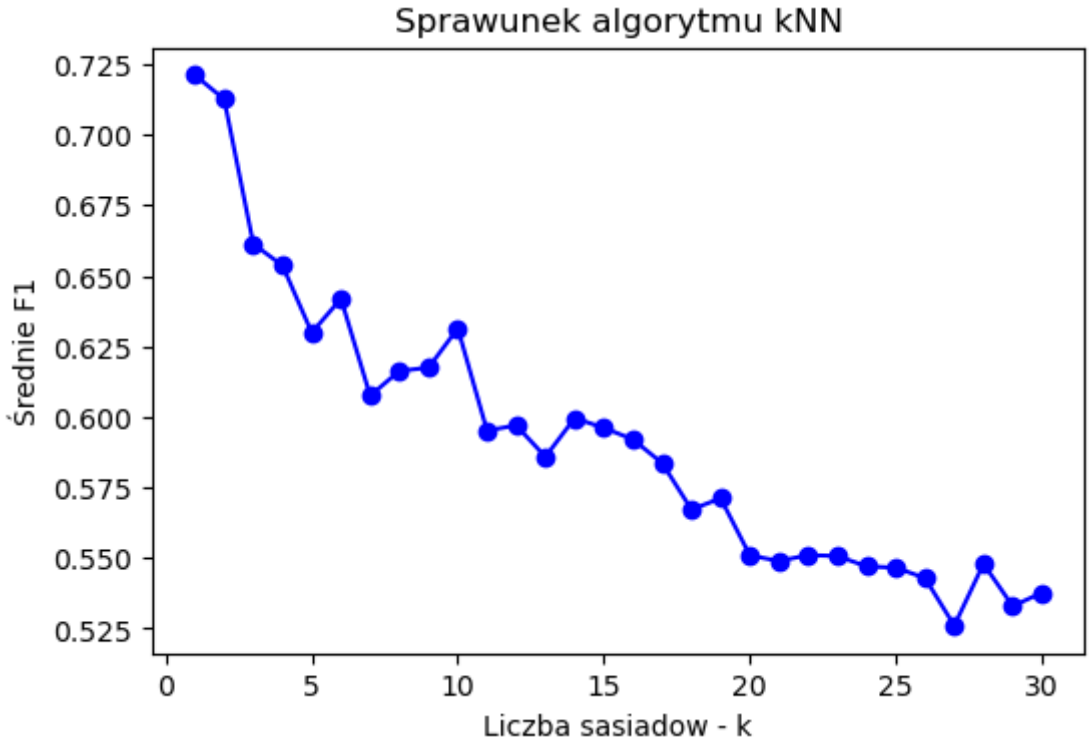
Wyniki każdego modelu zmierzono z pomocą miary f1 (będącej średnią ze średnich f1 w każdym splicie). Nie chciano przywiązywać się do miary accuracy, ponieważ jest ona stosunkiem prawidłowo zaklasyfikowanych obiektów do wszystkich zaklasyfikowanych obiektów, a nasz zbiór testowy jest niezbalansowany czyli np. przypadek błędnej klasyfikacji 100% obiektów klasy mającej tylko 2 instancje w zbiorze testowym, może nie mieć dużego wpływu na wynik accuracy, ponieważ zdominują go prawidłowe klasyfikacje klasy mającej 20 obiektów w zbiorze testowym. Z tego powodu lepiej było użyć miar precision oraz recall, a miara f1 jest właśnie ich średnią harmoniczną.

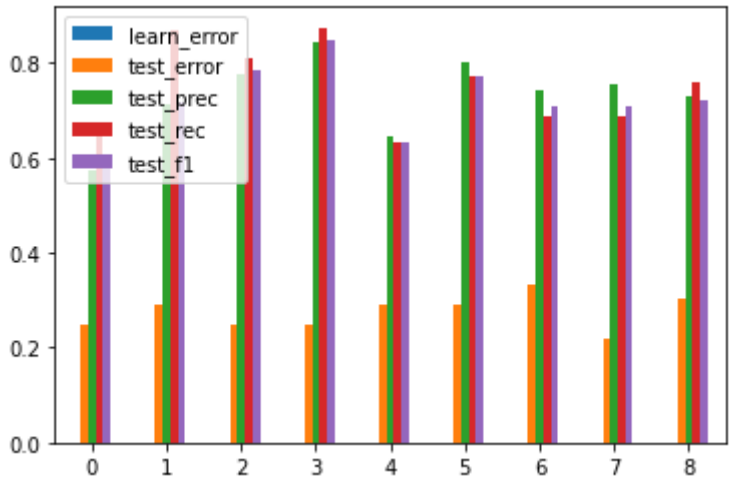
9 Wyniki

Aby porównać modele potrzebna była pojedyncza wartość, którą można było przypisać każdemu z nich i wybrać ten, który miał ją najkorzystniejszą – f1. Dla każdego modelu przedstawiono wykres słupkowy z wynikami w każdym splicie, aby pokazać jak ważna jest walidacja krzyżowa (można trafić na bardzo korzystny lub niekorzystny podział danych - split). Pokazano także wyniki będące średnią wyników w splitach. Poniżej widać modele w kolejności od dającego najlepsze wyniki do dającego najgorsze wyniki.

9.1 KNN

Model ten dał najlepsze rezultaty (najwyższe f1). Wynik modelu KNeighborsClassifier dla wyznaczonego najlepszego k = 1.



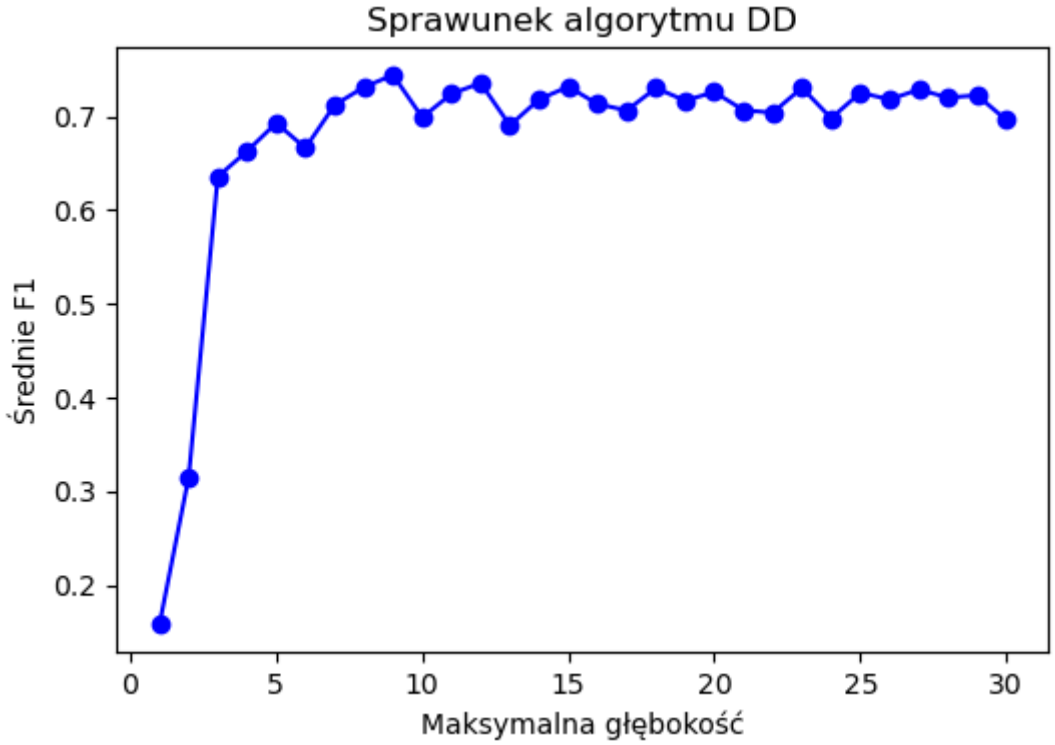


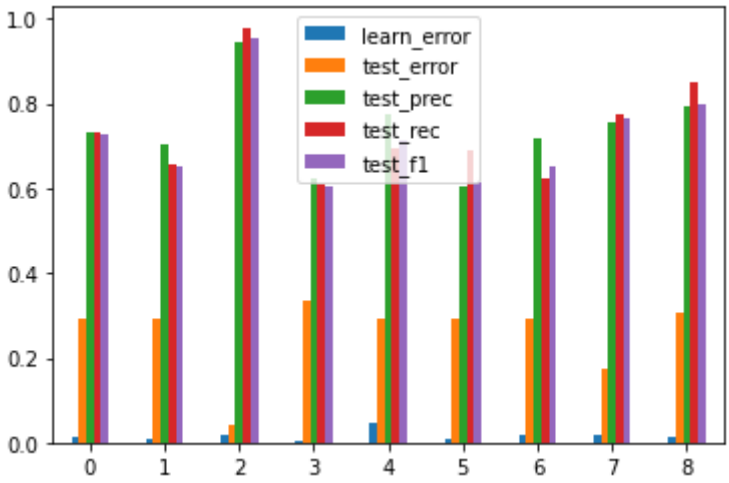
uśrednione wyniki

- ✓ learn_error 0.000000
- ✓ test_error 0.275564
- ✓ test_prec 0.731224
- ✓ test_rec 0.751066
- ✓ test_f1 **0.721107**

9.2 DD

Model DecisionTreeClassifier dla wyznaczonego najlepszego max_depth = 8. Co ciekawe daje on wyniki prawie identyczne z poprzednim modelem (jednak minimalnie gorsze). Można by polemizować czy nie jest jednak korzystniejszy z uwagi na większy, acz niezwykłe mały, błąd uczenia (nieco mniejsze ryzyko przetrenowania).

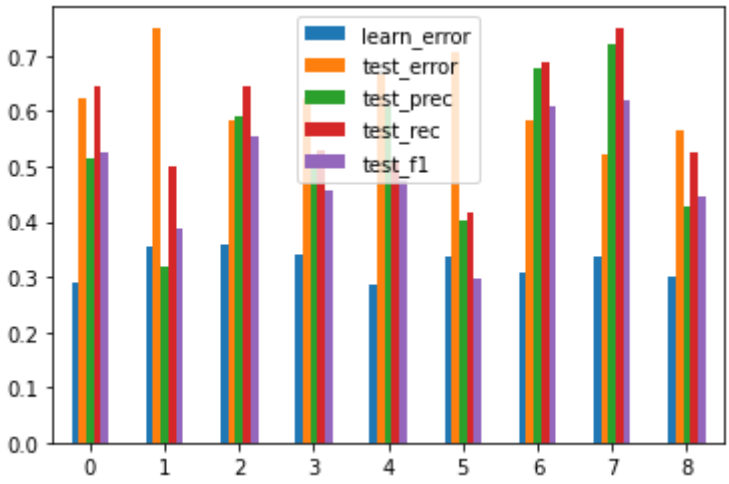




uśrednione wyniki

- ✓ learn_error 0.017254
- ✓ test_error 0.256844
- ✓ test_prec 0.738043
- ✓ test_rec 0.733943
- ✓ test_f1 **0.720090**

9.3 Naiwny Gauss



uśrednione wyniki

- ✓ learn_error 0.323448
- ✓ test_error 0.625403
- ✓ test_prec 0.531085
- ✓ test_rec 0.578300
- ✓ test_f1 **0.484799**

9.4 Najbliższy Centroid

uśrednione wyniki

- ✓ learn_error 0.342942
- ✓ test_error 0.565821
- ✓ test_prec 0.487787
- ✓ test_rec 0.562169
- ✓ test_f1 **0.475761**