Naive Bayes

Maciej Urbaniak 200842

September 6, 2019

1 Wstęp

Zadanie polegało na implementacji naiwnego klasyfikatora bayesowskiego i zbadanie jego działania na trzech wybranych zbiorach: glass, wine oraz pima indians diabetes.

Naiwny klasyfikator bayesowski zakłada że wartości atrybutów są niezależne od siebie, przez co każda z tych wartości może zwiększać prawdopodobieństwo przynależności do określonej klasy. W praktyce takie zjawisko jest rzadkie, dlatego w nazwie występuje określenie naiwny. Klasyfikator ten przyporządkowuje nowy przypadek do jednej ze zdefiniowanych wcześniej klas poprzez wykorzystanie prawdopodobieństwa przynależności pozyskanych na podstawie danych uczących. Dodatkowo prawdobodobieństwa te wymnażamy, oznacza to że jeśli jakaś klasa ma zerową szansę na wystąpienie danej wartości to przynależność do niej jest niemożliwa, nawet dla obiektu najbardziej zbliżonego pozostałymi atrybutami.

2 Zbiory danych

2.1 Glass

Z powodu niewielkiej ilości danych dla klas 3, 5 i 6 przy losowaniu zbioru uczącego często występuje sytuacja w której jest ich zbyt mało lub wcale, by prawidłowo wytrenować klasyfikator. Dlatego też przy użyciu KFold można zauważyć znaczne wachania celności predykcji w szczególności dla przypadku w którym tworzymy wiele zbiorów. Natomiast przy podziale z wykorzystaniem kroswalidacji stratyfikowanej należy uwzględnić liczność najmniejszego zbioru. Dzięki czemu zaobserwowano znaczną poprawę wyników z wykorzystaniem tej metody.
F1-score:
Kfold10: 0.3363 Standard deviation:0.2483
Stratified KFold9: 0.4417 Standard deviation 0.1149

Class	Precision	Recall	f1-score	support
1	0.00	0.00	0.00	33
2	0.34	0.88	0.49	26
3	0.00	0.00	0.00	8
5	1.00	0.25	0.40	4
6	0.00	0.00	0.00	2
7	0.69	0.85	0.76	13

Table 1: Glass table

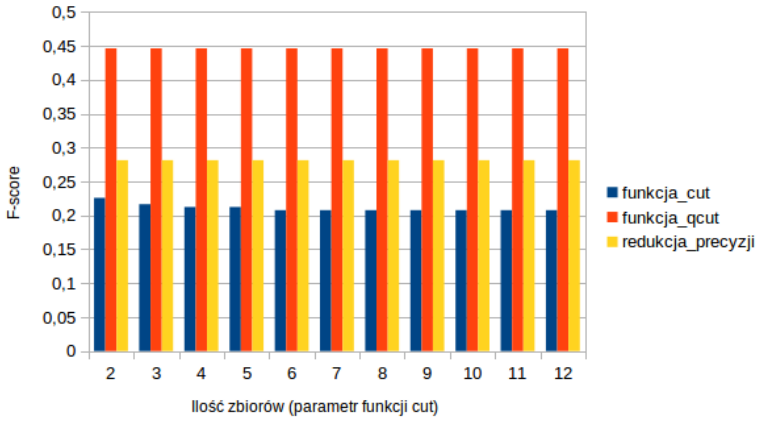

Figure 1: Dyskretyzacja danych glass

Powyżej przedstawiono wpływ dyskretyzacji na uzyskiwane wyniki oś Y oznacza uzyskaną średnią wartość F1-score. Oś X na ile zbiorów podzielono dane wykorzystując funkcję "pandas.cut". Funkcja ta dzieli wartości na przedziały o stałej szerokości. Natomiast dla funkcji qcut wybrano podział na kwartyle. Redukcja precyzji została ustanowiona do części setnych. Najlepszy wynik uzyskano dla podziału na kwartyle (0.4462) natomiast najgorszym sposobem dyskretyzacji powyższych danych jest użycie funkcji cut do podziału na zbiory (0.2077).

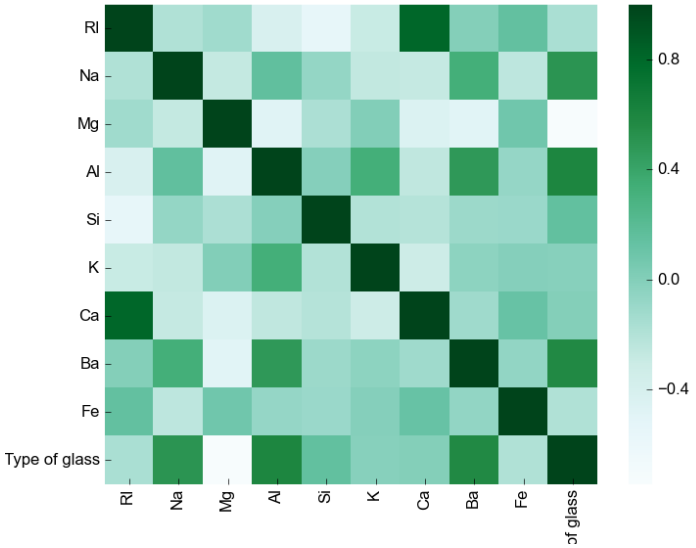

Figure 2: Macierz korelacji

0	33	0	0	0	0
0	23	0	0	1	2
0	8	0	0	0	0
0	1	0	1	0	2
0	1	0	0	0	1
1	1	0	0	0	11

Table 2: Glass confusion matrix

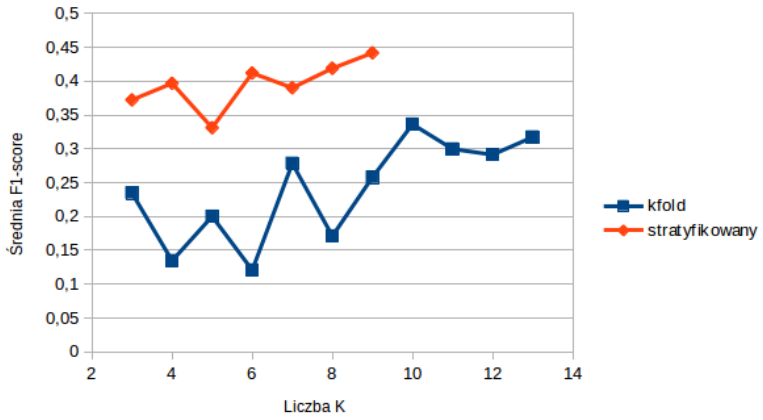

Figure 3: Kroswalidacja

Najmniej liczna jest klasa 6 posiada ona 9 obiektów, dlatego też przy kroswalidacji stratyfikowanej większej od 9 zwracany jest błąd o niedostatecznym zasobie danych aby równo podzielić zbiory. Mimo to program wykonywany jest dalej. Przy kroswalidacji stratyfikowanej można zauważyć znaczny wzrost celności predykcji, a wiąże się to z tym że najmniej liczna klasa 6 często nie jest wylosowywana do zbioru uczącego z wykorzystaniem zwykłego sposobu podziału jakim jest KFold.

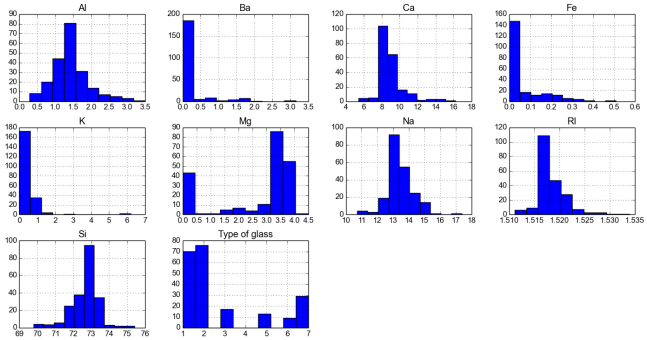

Figure 4: Rozkład parametrów

2.2 Wine

Dzięki podzieleniu obiektów na zbilansowane pod względem ilości klasy wyniki klasyfikacji są dokładne. Przy użyciu 10Fold można zauważyć że dzięki temu wybiór zbiorów w znikomym stopniu wpływa na zmianę wyników klasyfikacji. Podobnie przy wykorzystaniu kroswalidacji stratyfikowanej można zauważyć nieznaczną poprawę.
F1-score:
Kfold10: 0.9611 Standard deviation:0.0558
Stratified Kfold10: 0.9616 Standard deviation: 0.0424

Class	Precision	Recall	f1-score	support
1	0.96	0.86	0.91	28
2	0.80	0.89	0.84	27
3	0.76	0.76	0.76	17

Table 3: Wine table

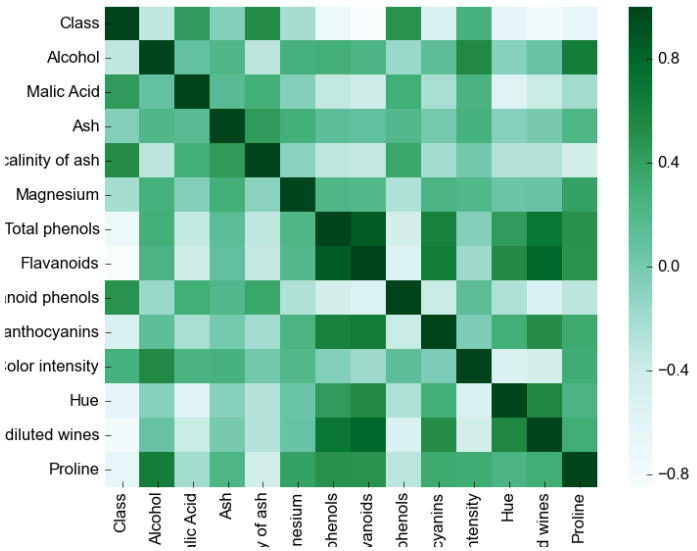

Figure 5: Macierz korelacji

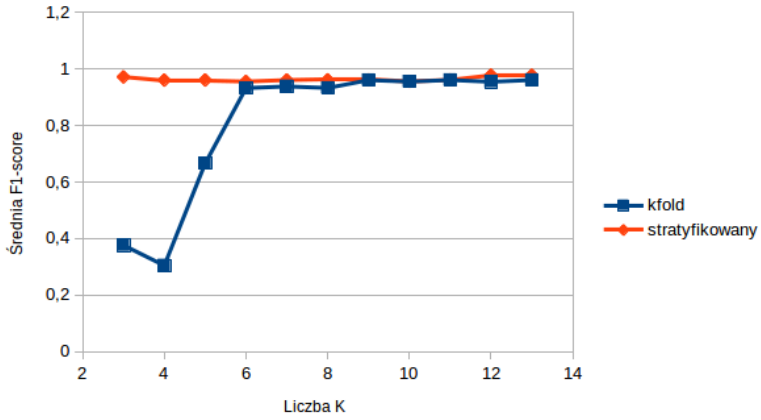

Figure 6: Kroswalidacja

Dzięki podziałowi danych na w miarę równe ilości obiektów przypadających na klasę, rozmiar kroswalidacji nie wpływa znacznie na celność predykcji. Wyjątkiem jest podział bez zastosowania stratyfikacji parametru K mniejszego od 6, z powodu dużego prawdopodobieństwa braku wystarczającej liczby obiektów do wytrenowania klasyfikatora.

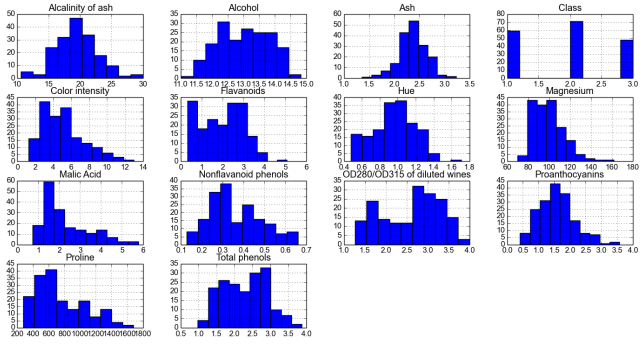

Figure 7: Rozkład parametrów

24	2	2
1	24	2
0	4	13

Table 4: Wine confusion matrix

2.3 Diabetes

Zbiór poniższych danych zawiera wartości zerowe dla przykładowo BMI czy ciśnienia krwi co wskazuje na niepełność tego zbioru informacji. Zmiana KFold nie wpływa w dużej mierze na wyniki. Największą korelację z wystąpieniem cukrzycy ma wysokość insuliny. Dodatkowo w tym miesiącu dane zostały usunięte, komunikat na stronie informuje: "Thank you for your interest in the Pima Indians Diabetes dataset. The dataset is no longer available due to permission restrictions.".
F1-score:
Kfold10: 0.7551 Standard deviation:0.0427
Stratified Kfold10: 0.7422 Standard deviation: 0.0471

Class	Precision	Recall	f1-score	support
0	0.66	0.63	0.65	199
1	0.38	0.40	0.39	109

Table 5: Diabetes table

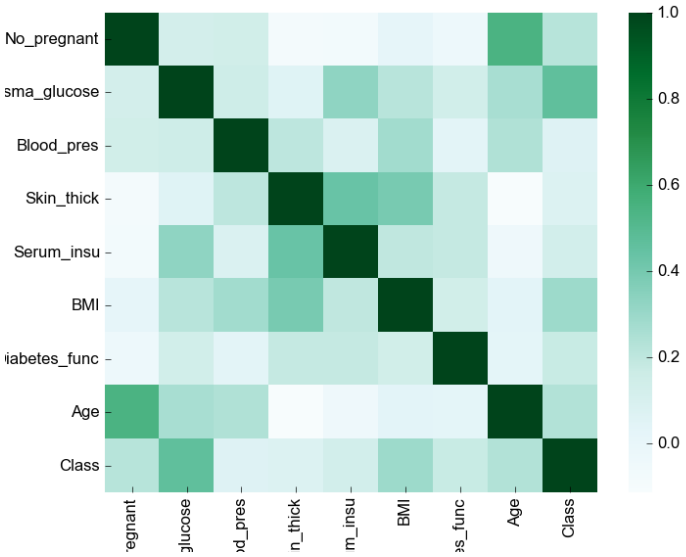

Figure 8: Macierz korelacji

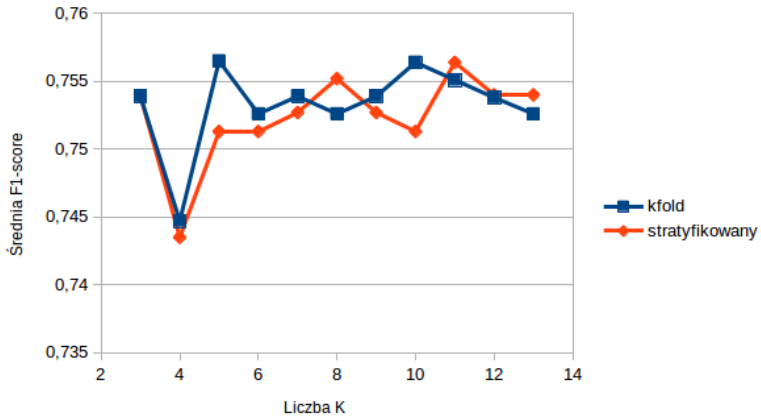

Figure 9: Kroswalidacja

Tak jak w przypadku zbioru wine rozmiar kroswalidacji ma niewielki wpływ na celność predykcji z powodu dużej liczności zbiorów z każdej klasy.

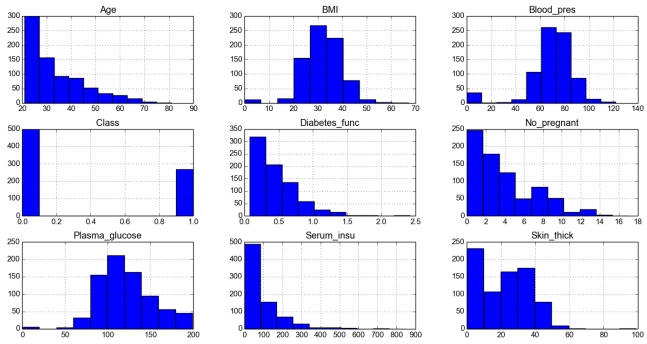

Figure 10: Rozkład parametrów

126	73
65	44

Table 6: Diabetes confusion matrix

3 Wnioski

O ile znaczenie parametru accuracy to liczba prawidłowo sklasyfikowanych do wszystkich obiektów, tak inne miary dostarczają nam ważnych informacji na temat sposobu klasyfikacji. Dla wysokiej wartości parametru precision liczba błędnych przyporządkowań do klasy będzie znacznie mniejsza, czyli zwiększy się tzw. "celność", ale wysoka wartość będzie powodować odrzucanie przypadków o niskiej pewności przyporządkowania do danych klas. Natomiast recall daje pojęcie o liczbie pominiętych obiektów które powinny były zostać zaklasyfikowane do danej klasy. Przy czym jeśli precision będzie niska będzie ona przyporządkowywać obiekty podobne w małym stopniu do danej klasy. F1-score to średnia harmoniczna dwóch poprzednich parametrów jej zastosowanie przydatne jest w szczególności tam gdzie nieprawidłowa klasyfikacja ma znacznie wyższy koszt niż prawidłowa.
Istotnym czynnikiem wpływającym na celność przyporządkowania obiektu do danej klasy była liczba danych uczących. Widać to w szczególności na zbiorze glass, gdzie istniało duże prawdopodobieństwo wylosowania do zbioru uczącego obiektów z tylko kilku klas, a przez to niemożliwość klasyfikacji nowych przypadków.