Klasyfikacja na bazie modelu regresji liniowej oraz porównanie metod klasyfikacji na podstawie danych iris, Glass

Eksploracja danych

Tomasz Warzecha, album 282261

2025-05-28

Spis treści

1 Klasyfikacja na bazie modelu regresji liniowej

==========

W tym zadaniu pracujemy na danych iris (R-pakiet datasets). Zbiór danych zawiera wyniki pomiarów uzyskanych dla trzech gatunków irysów (tj. setosa, versicolor i virginica) i został

udostępniony przez Ronalda Fishera w roku 1936.Pomiary dotyczą długości oraz szerokości dwóch różnych części kwiatu – działki kielicha (ang. sepal) oraz płatka (ang. petal).

1.1 Analizowane dane

Tabela 1: Struktura danych

	x
Sepal.Length	numeric
Sepal.Width	numeric
Petal.Length	numeric
Petal.Width	numeric
Species	factor

Nasze dane mają 150 przypadków i 5 cech. W powyższej tabeli możemy zobaczyć wszystkie cechy oraz ich typy. Widzimy, że wszystkie zmienne zostały poprawnie rozpoznane. Nasze dane mają 4 cechy numeryczne oraz jedną jakościową. W żadnej z kolumn nie mamy braku wartości.

1.2 Podzieł danych na zbiór uczący i testowy

W tym kroku wykonujemy podział naszych danych na zbiór uczący oraz testowy. Zbiór uczący zawiera ok. 2/3 przypadków, a testowy 1/3. Dane zostały przydzielone losowo.

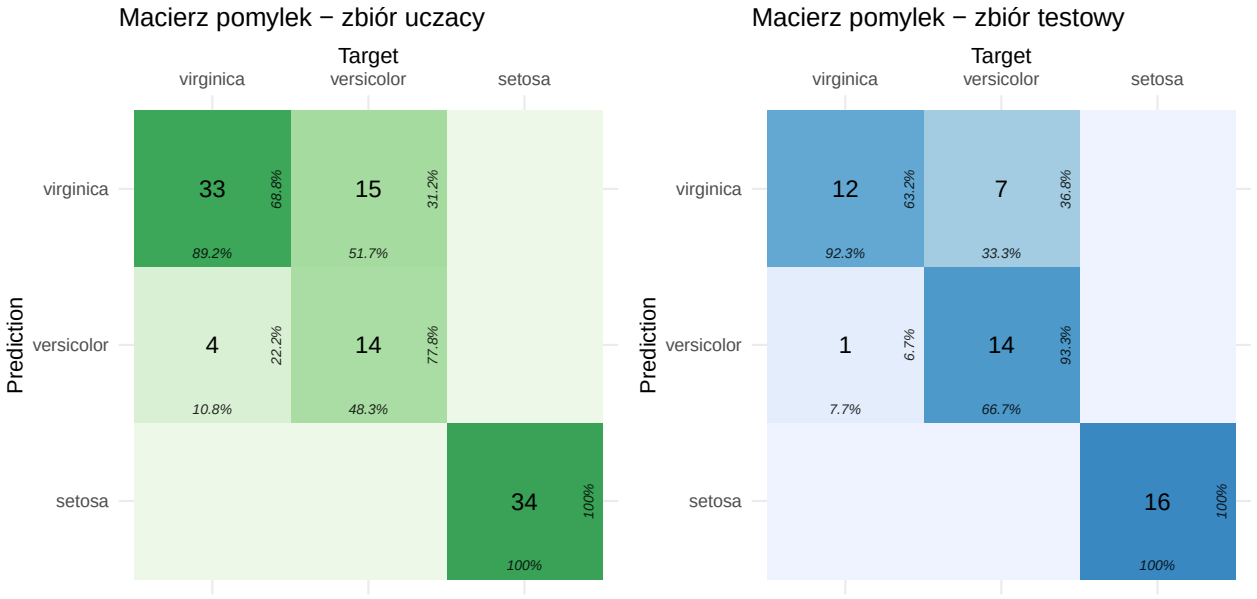
Na powyższych wykresach widzimy rozkład gatunków w obu zbiorach. Widzimy że dane został wylosowane w odpowiedni sposób, liczebności klas zostały dobrze rozdysponowane.

1.3 Konstrukcja klasyfikatora i wyznaczenie prognoz

W tym kroku zastosujemy model regresji liniowej do konstrukcji klasyfikatora na podstawie danych uczących. Wyznaczymy K niezależnych modeli (jednowymiarowych) dla poszczególnych klas, wykorzystamy w tym celu funkcje $lm()$. Następnie wyznaczymy prognozowane etykietki dla danych ze zbioru uczącego i testowego.

1.4 Ocena jakości modelu

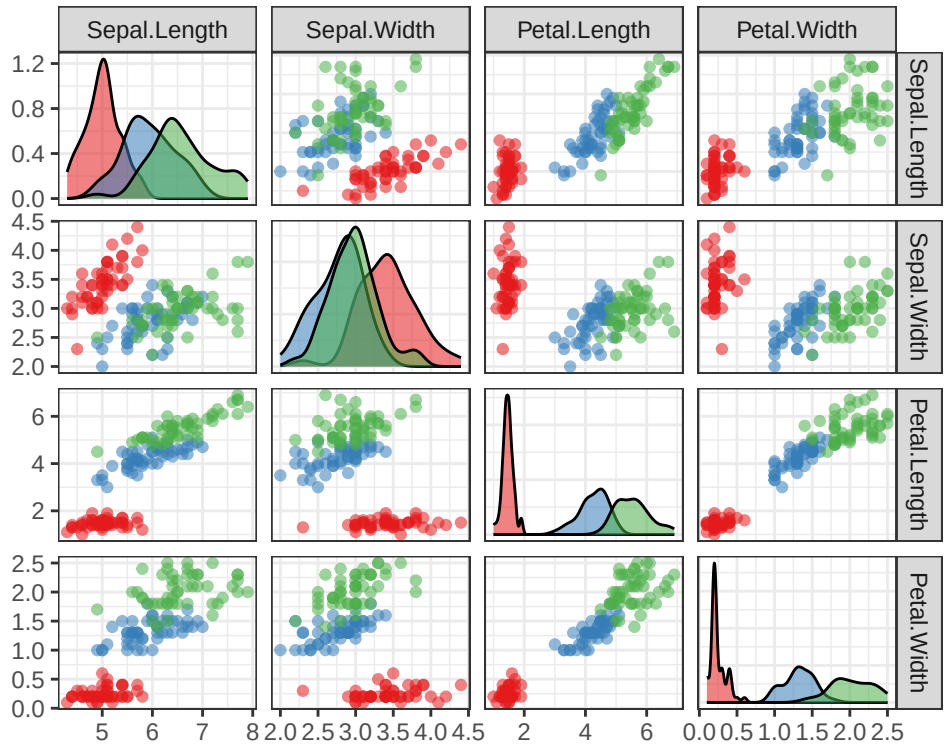
Teraz wyznaczymy macierz pomyłek oraz błąd klasyfikacji na zbiorze uczącym oraz zbiorze testowym.



Na przedstawionych macierzach pomyłek widzimy rezultaty klasyfikacji dla dwóch zbiorów danych: testowego i uczącego. Możemy stwierdzić, że model bardzo dobrze radzi sobie z rozpoznawaniem gatunku setosa, osiągając 100% trafności zarówno w zbiorze uczącym, jak i testowym. x Największe trudności pojawiają się w rozróżnianiu gatunków versicolor i virginica. W szczególności model często myli versicolor z virginica, co widać zwłaszcza w zbiorze uczacym, gdzie aż połowa przypadków versicolor została błędnie sklasyfikowana.

Błąd klasyfikatora na zbiorze uczącym wynosi: 0.81, natomiast na zbiorze testowym: 0.84. Wyniki w zbiorze testowym są porównywalne do tych uzyskanych na zbiorze uczącym, co sugeruje, że model nie uległ przeuczeniu.

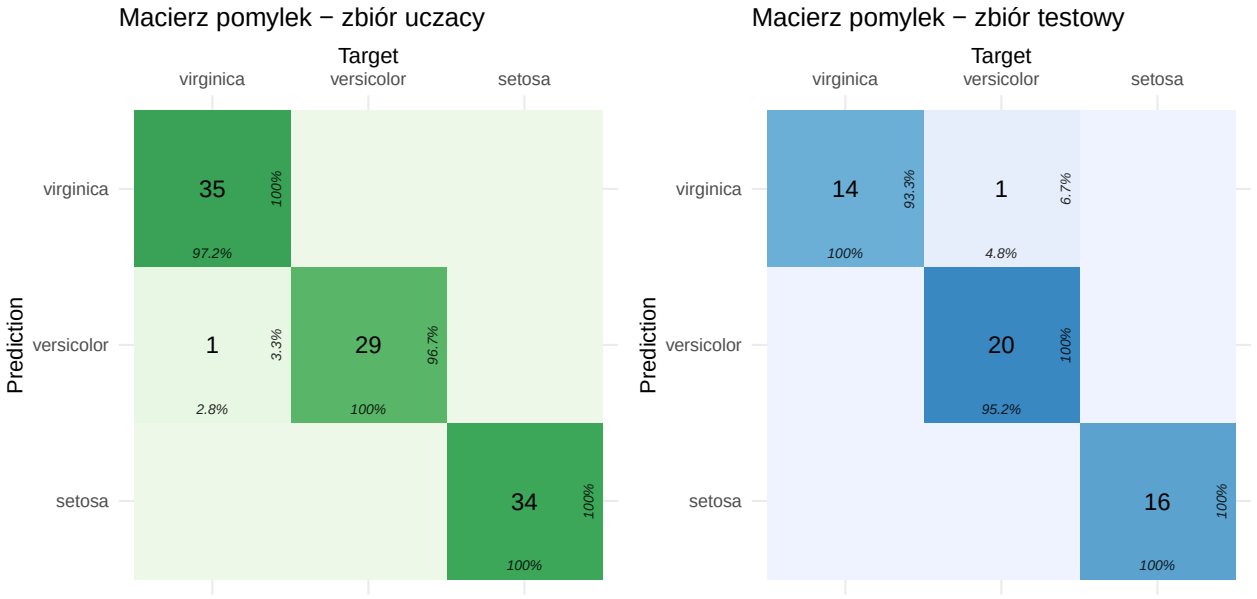
Może tu występować zjawisko maskowania klas, w którym jedna klasa (w tym przypadku virginica) “przykrywa” drugą (versicolor), przez co model częściej przypisuje dane obserwacje do silniej reprezentowanej lub lepiej dopasowanej klasy. Spróbujmy się temu przyjrzeć poprzez zbadanie rozkładów w przestrzeni cech.



Na powyższych wykresach widzimy, że setosa (czerwony) jest dobrze rozróżnialna, szczególnie dla zmiennych dotyczacych Petal, natomiast rozkłady pozostałych dwóch gatunktów nakładają się, szczególnie zmienne `Sepal.Length` oraz `Sepal.Width` w której rozłady versicolor oraz virginica są niemal identyczne, co wpływa na maskowanie klas. Ta cecha ma niską moc klasyfikacyjną i może być źródłem błędów klasyfikatora.

1.5 Budowa modelu liniowego dla rozszerzonej przestrzeni cech

Spróbujmy rozszerzyć nasz model o składniki wielomianowe stopnia 2 (tzn.: PL^2, PW^2, SL^2, SW^2, $PL * PW$, $PL * SW$, $PL * SL$, $PW * SL$, $PW * SW$, $SL * SW$). Znowu wyznaczymy model dla zbioru uczącego (2/3 przypadków), następnie przetestujemy go na zbiorze testowym i uczącym. Przypadki zostaną wybrane losowo korzystając z tego samego ziarna co w poprzednim modelu aby zachować wiarygodność wyników.



Na wykresach powyżej widzimy macierze pomyłek dla zbioru testowego i uczącego. Jak widzimy znacząco poprawiliśmy jakość naszego modelu, jedynie dwa przypadki zostały błędnie sklasyfikowane przez nasz model. W zbiorze uczącym jeden przypadek został sklasyfikowany jako gatunek versicolor zamiast viginica, oraz odwrotnie w zbiorze testowym, jeden przypadek został sklasyfikowany jako virginica, a w rzeczywistości był to versicolor. Oznacza to, że nasze cechy stopnia 2 są na tyle charakterystyczne, że model klasyfikuje gatunki niemal idealnie, z błędem na poziomie: 0.989899, 0.9803922, odpowiednio w zbiorach uczącym i testowym. Wyniki w zbiorze testowym są porównywalne do tych uzyskanych na zbiorze uczącym, co sugeruje, że model nie uległ przeuczeniu.

Pozbyliśmy się również problemu maskowania klas. Zmienne które dodaliśmy, skutecznie rozróżniały nasze gatunki. W szczególności poprawiliśmy skuteczność rozrózniania gatunku virginica oraz versicolor.

1.6 Podsumowanie

Nasz początkowy model oparty wyłącznie na cechach gatunków radził sobie dobrze jedynie z gatunkiem setosa, w pozostałych przypadkach jakość pozostawiała wiele do życzenia. Nasze cechy nie rozróżniały wystarczająco dobrze dwóch pozostałych gatunków, dlatego zdecydowaliśmy się wprowadzić wielomiany stopnia 2 naszych cech, tak aby spróbować poprawić skuteczność modelu. Przyniosło to bardzo dobre efekty, skuteczność naszego modelu znacząco wzrosła i jednynie w dwóch przypadkach model błędnie ocenił gatunek.

2 Porównanie metod klasyfikacji

2.1 Wybór i zapoznanie się z danymi

W tym zadaniu pracujemy na danych Glass (R-pakiet mlbench). Zbiór danych Glass charakteryzuje się złożoną strukturą klas oraz wyraźnymi różnicami w rozkładach cech chemicznych.

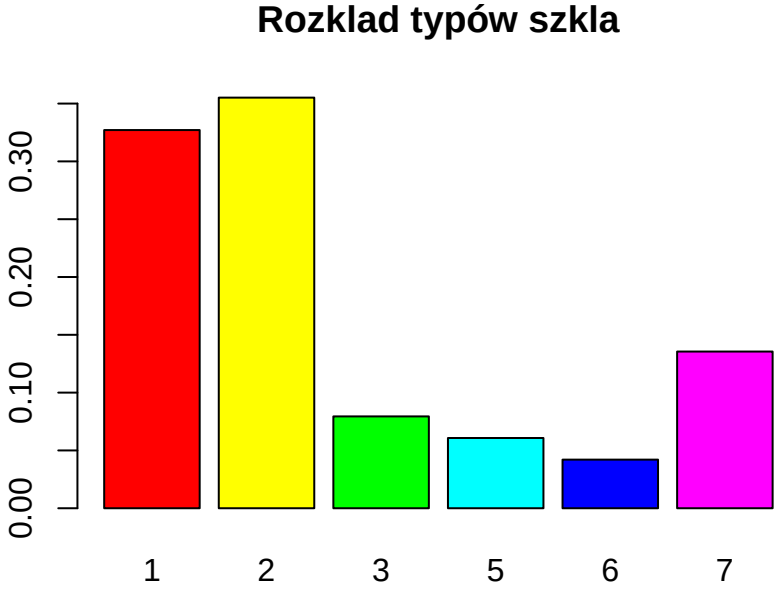
W przeciwieństwie do prostych zbiorów jak iris, tutaj efekt maskowania klas jest silniejszy, a nierównowaga klas wymaga specjalnego podejścia.

RI Na Mg Al Si K Ca Ba Fe Type 0 0 0 0 0 0 0 0 0 0 Zbiór danych Glass zawiera 214 przypadków opisujących różne rodzaje szkła na podstawie ich składu chemicznego. Każdy przypadek charakteryzuje się 9 cechami numerycznymi, w tym zawartością pierwiastków takich jak sód (Na), magnez (Mg), glin (Al), krzem (Si), potas (K), wapń (Ca), bar (Ba), żelazo (Fe) oraz współczynnikiem załamania światła (RI). Klasyfikacja odbywa się na podstawie zmiennej Type, która określa typ szkła i przyjmuje 6 różnych wartości (od 1 do 6).

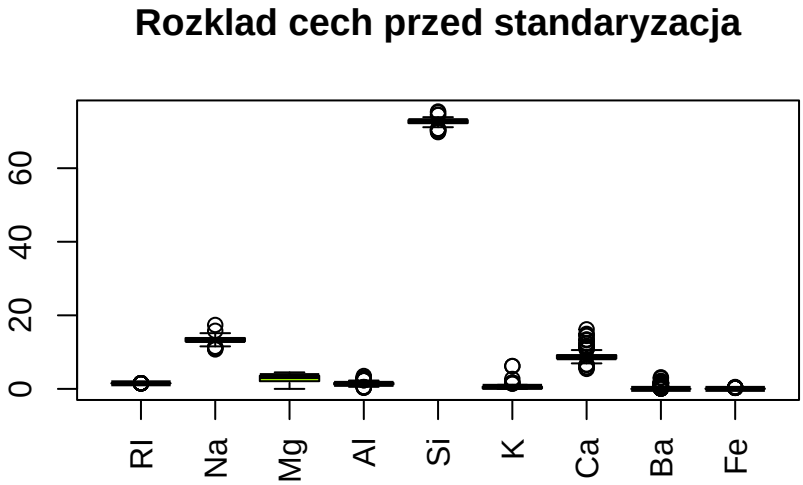
Nasze dane nie posiadają żadnych braków danych, brak występowania danego pierwiastka w danym rodzaju szkła oznaczany jest jako 0.0 zatem nie przeszkadza nam to w dalszej analizie.

Wszystkie typy zmiennych są numeryczne, z wyjątkiem zmiennej Type zawierającej etykietki naszych klas, która jest zmienną typu factor

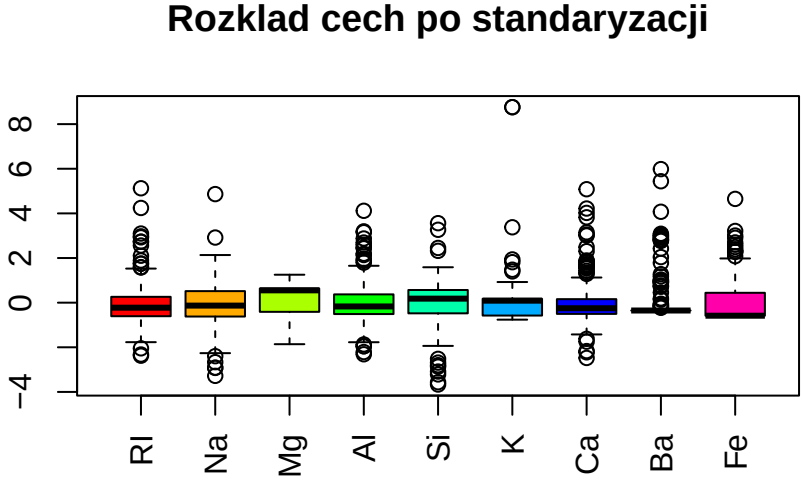
2.2 Wstępna analiza danych



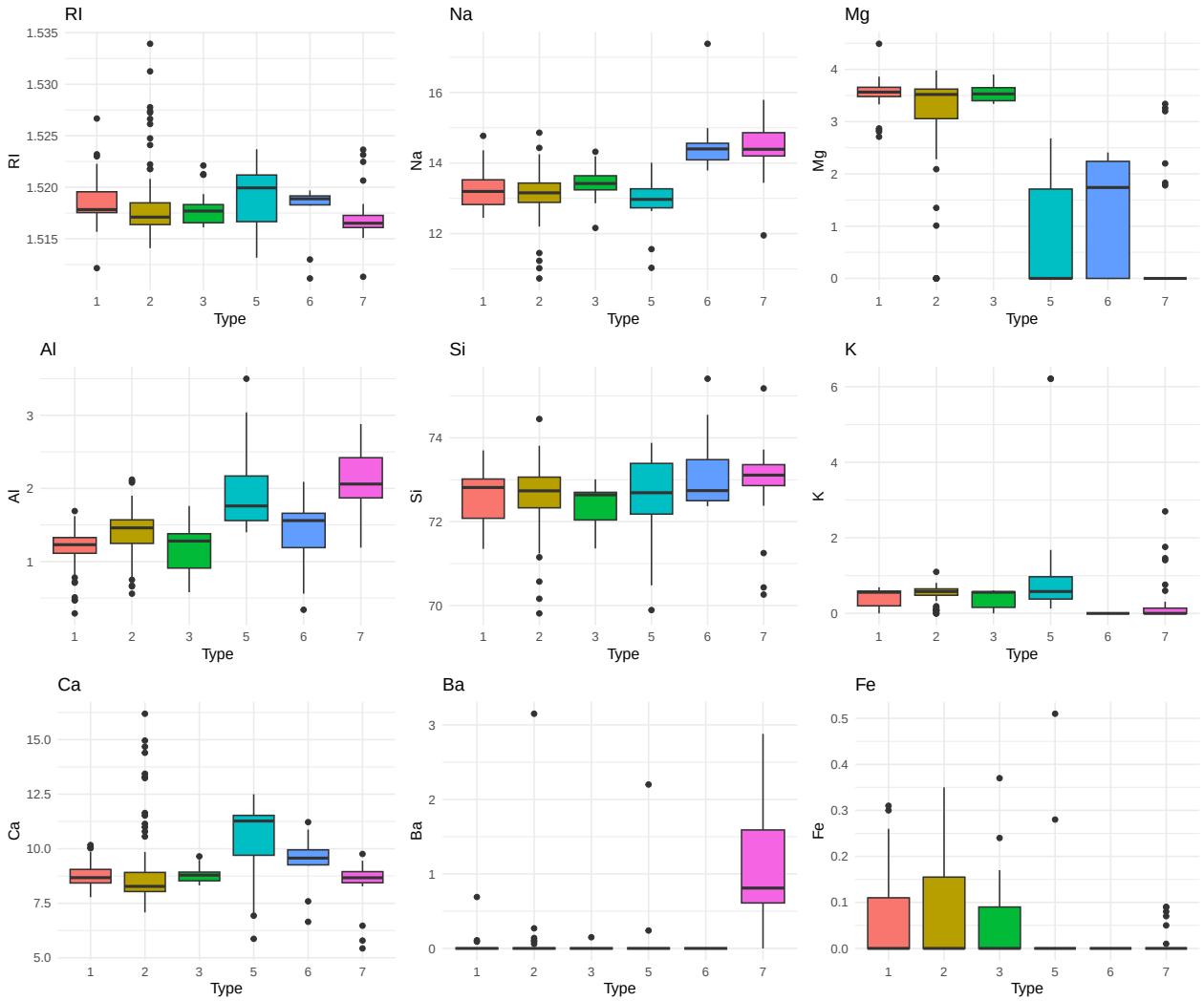
Widzimy na powyższym, że rozkład klas jest silnie niezrównoważony. Klasa 2 dominuje z ok. 35% przypadków, podczas gdy klasa 6 stanowi zaledwie 4.2% zbioru. Przypisując wszystkie obiekty do najczęstszej klasy, otrzymalibyśmy błąd klasyfikacji na poziomie 64%, co wskazuje na znaczną dysproporcję w danych.



Wykres pokazuje wyraźne różnice w skalach i rozproszeniu poszczególnych cech. Tak duże różnice w wariancji sugerują konieczność standaryzacji przed zastosowaniem metod wrażliwych na skalę, takich jak k-NN.

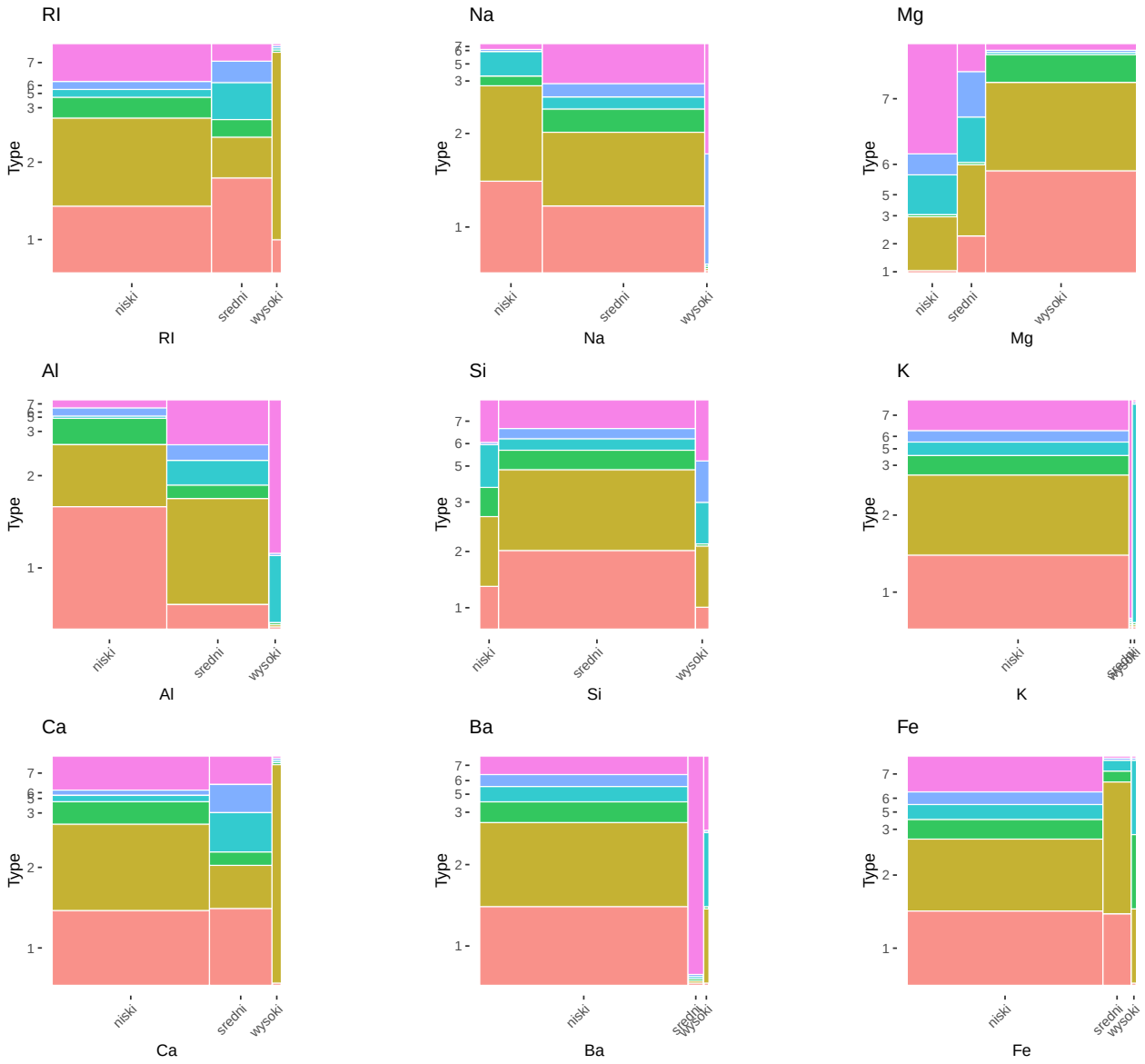


Na powyższym wykresie widać, że po standaryzacji wszystkie cechy mają podobny zakres wartości, co ułatwi porównywanie ich wpływu na klasyfikację.



Na powyższym rysunku widzimy wykresy pudełkowe dla wszystkich zmiennych. Możemy zauważyć, że Mg dobrze rozróżnia klasy 5,6,7 względem 1,2,3. Również Al ładnie dzieli nasze zmienne. We współczynniku RI również mamy zróżnicowane rozkłady zmiennych. Warto także popatrzeć na zmienną Ba, która bardzo dobrze wyróżnia klasę 7. Fe podobnie dzieli nasz zbiór jak Mg, warto jeszcze popatrzeć na Ca, które szczególnie różnicuje klasę 5. Pierwiastki Si, K, Na mają dosyć małe zróżnicowanie.

Spróbójmy jeszcze popatrzeć na wykresy mozaikowe.



Tutaj również widzimy najlepsze zróżnicowanie u podobnych cech. Wyróżniają się przede wszystkim Mg i RI, ale także Al czy Ca. Słabo radzą sobie szczególnie K czy Si.

Podsumowując, największe zdolności dyskryminacyjne mają cechy: Mg, RI, Ba, Al, Ca, natomiast gorzej radzą sobie pierwiastki: K, Si, Na, Fe

3 Budowa modeli klasyfikacyjnych

3.1 Podział na zbiór testowy i uczący

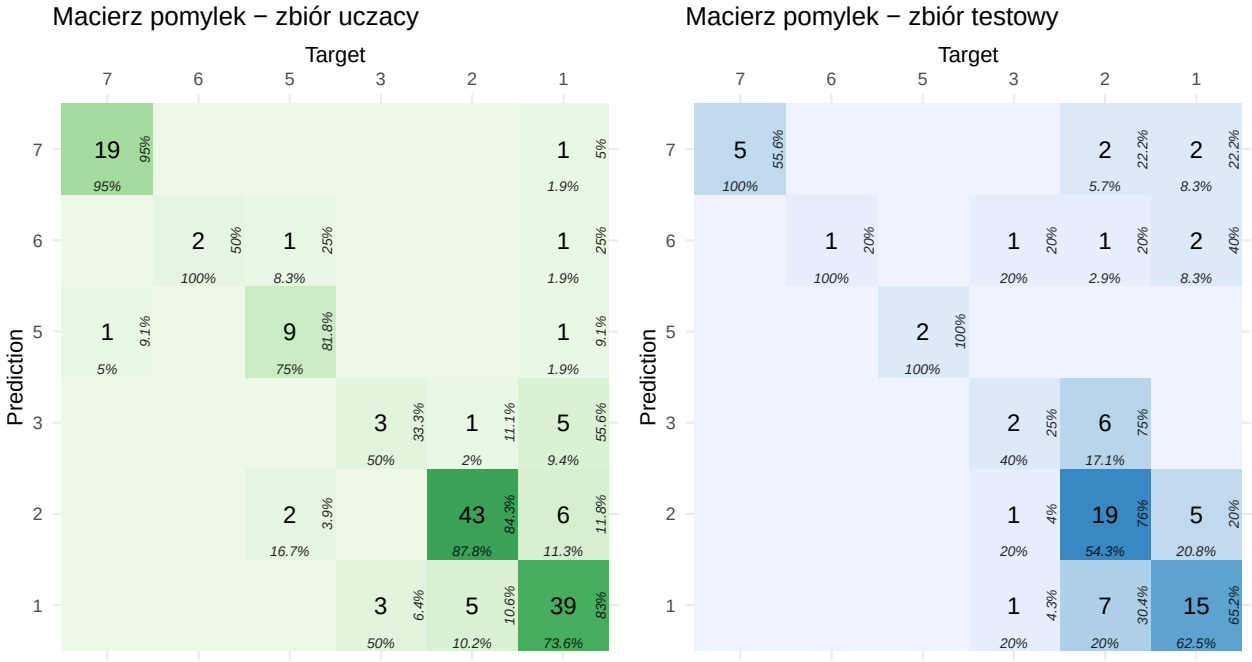
Nasze modele będziemy trenować na losowo wybranym zbiorze uczącym, a następnie testować na nim oraz na zbiorze testowym. Zbiory zostały wybrane losowo w oparciu o ustalone ziarno "123", w proporcjach 2/3, 1/3 odpowienio dla zbioru uczącego i testowego.

3.2 Klasyfikacja - k-najbliższych sąsiadów

Do tej metody, dane zostały zestandaryzowane, ponieważ opiera się ona na odelgłościach, gdzie zmienne o większych zmiennościach dominowałyby. Aby uniknąć takiej sytuacji zastosowana została standaryzacja.

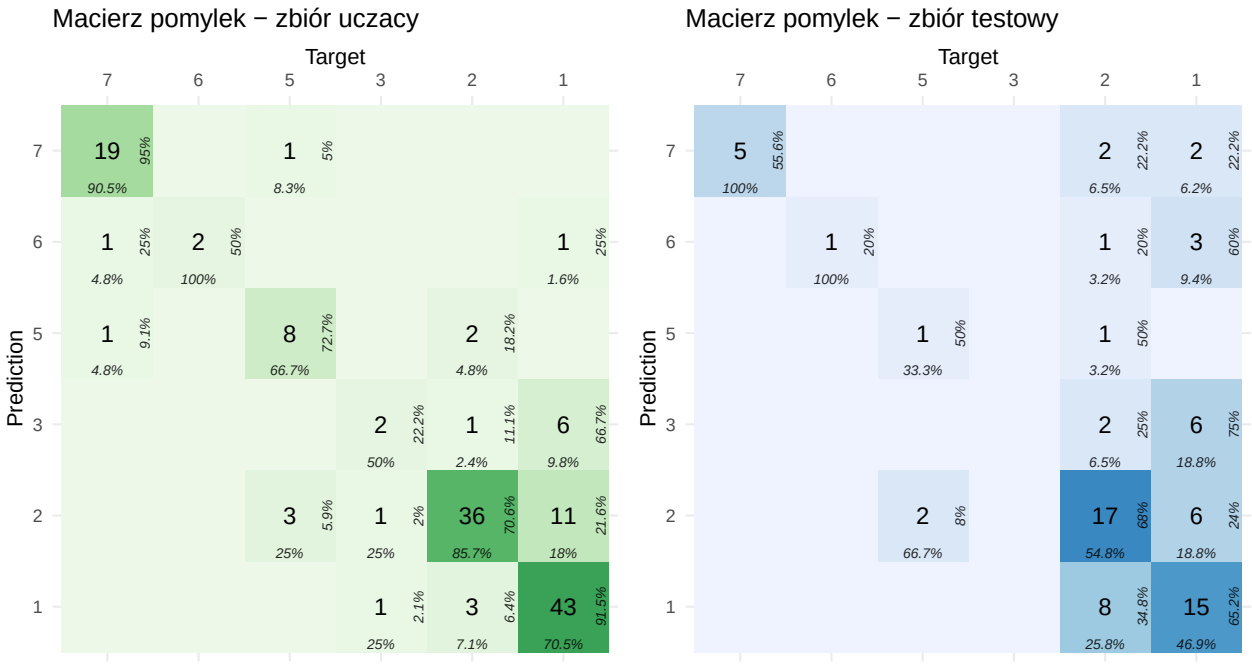
3.2.1 Różny dobór parametrów

Na początku zbadajmy dokładność metody dla 2 najbliższych sąsiadów.



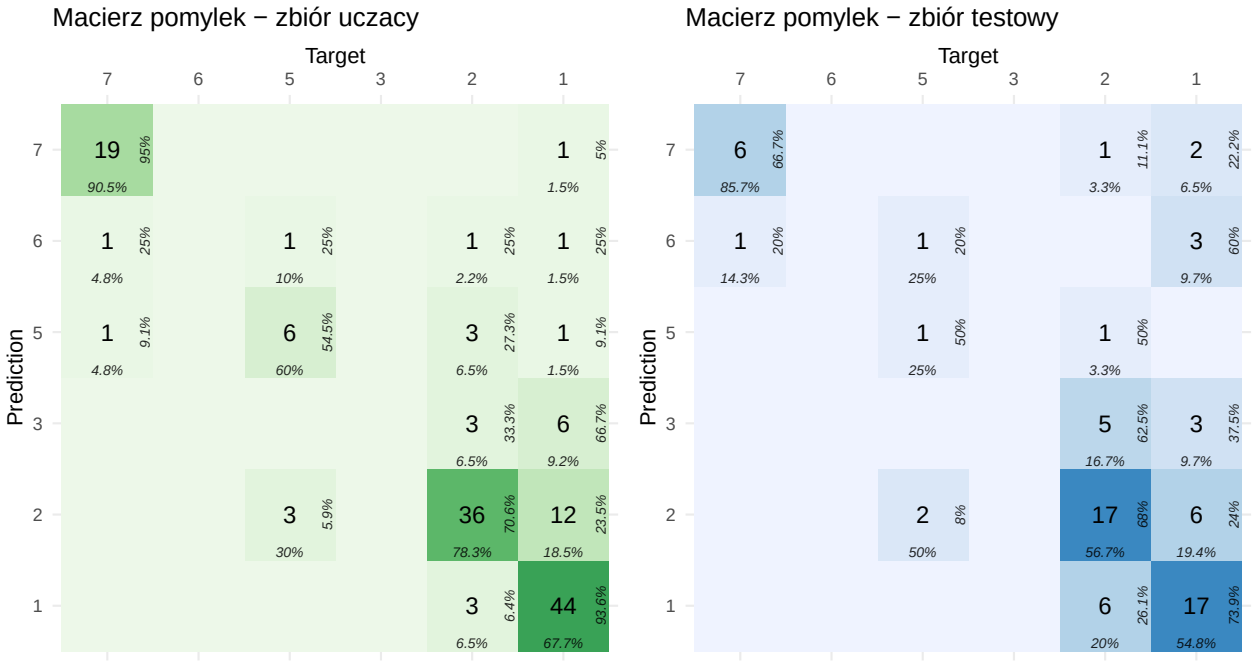
Jak widzimy na macierzach pomyłek, metoda nie przyniosła najlepszych rezultatów, szczególnie dla zbioru testowego. Błędy w tym przypadku wynoszą: 0.19, 0.39, odpowiednio dla zbioru uczącego i testowego.

Zobaczmy teraz wersję dla 5 najbliższych sąsiadów



Jak widzimy na macierzach pomyłek, metoda nie przyniosła poprawy, a nawet się pogorszyła. Błędy w tym przypadku wynoszą: 0.23, 0.46, odpowiednio dla zbioru uczącego i testowego.

Zobaczmy jeszcze dla 7 sąsiadów

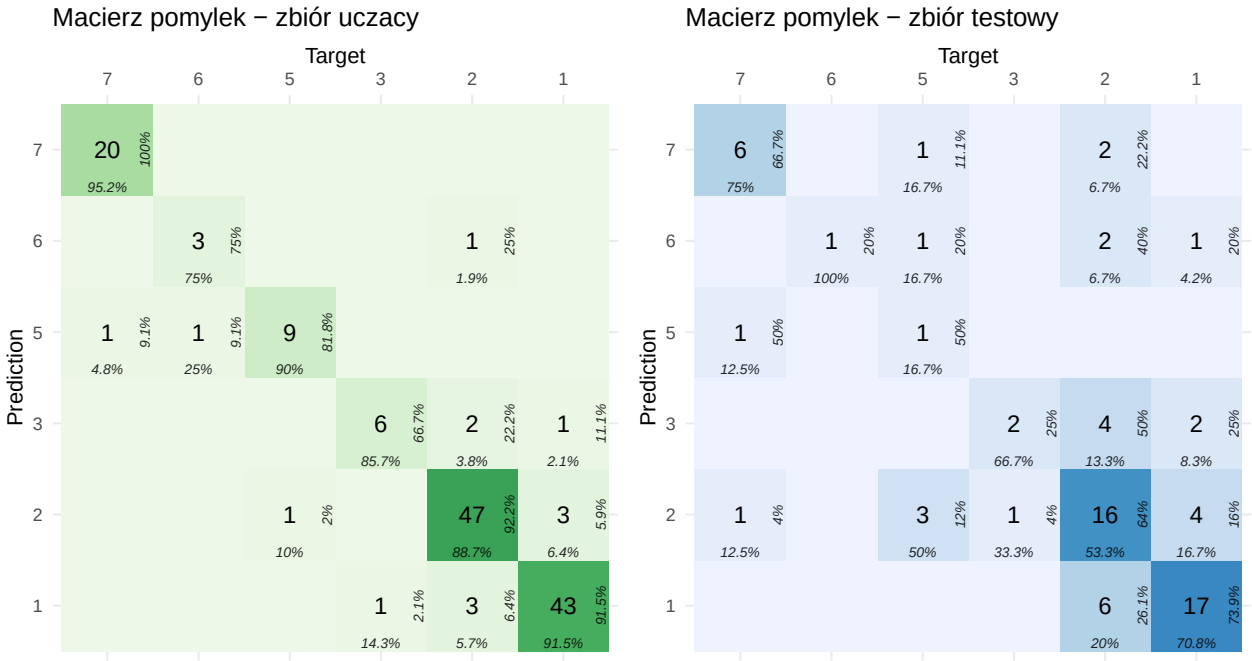


Jak widzimy na macierzach pomyłek, metoda jeszcze pogorszyła rezultaty. Błędy w tym przypadku wynoszą: 0.26, 0.43, odpowiednio dla zbioru uczącego i testowego.

3.2.2 Różny dobór zmiennych

Przetestujmy teraz jak wpływają różne kombinacje (podzbiory) zmiennych wykorzystanych do konstrukcji klasyfikatorów, w szczególności wszystkie zmienne oraz wybrany podzbiór zmiennych o najlepszej zdolności dyskryminacyjnej.

W naszym przypadku będziemy brali pod uwagę metodę 2 najbliższych sąsiadów i przetestujemy ją na zmiennych o najlepszej zdolności dyskryminacyjnej tj. Mg, RI, Ba, Al. Następnie porównamy otrzymany wynik z błędami dla wszystkich danych.



Jak widzimy na macierzach pomyłek, znacząco nie poprawiliśmy rezultatów, jedynie dla zbioru uczącego poprawa była widoczna, natomiast dla zbioru testowego dokładność pozostała bez zmian. Błędy w tym przypadku wynoszą: 0.1, 0.4, odpowiednio dla zbioru uczącego i testowego.

Dokładność została sprawdzona również dla mniejszejych podzbiorów cech, jednak tylko pogorszyły one rezultaty.

3.2.3 Podsumowanie

Dane w użytej metodzie zostały podzielone na zbiór uczący i testowy, a jako miary oceny użyto macierzy pomyłek oraz błędów klasyfikacji.Na podstawie przeprowadzonych eksperymentów z metodą k-najbliższych sąsiadów (k-NN) można stwierdzić, że skuteczność klasyfikacji zależy zarówno od liczby sąsiadów (k), jak i od doboru zmiennych, jednak nie zmienia to w naszym przypadku za dużo, szczególnie dla zbioru testowego.

Wyniki pokazały, że wielkość błędu zależy od prawidłowego doboru liczby sąsiadów. W naszych danych najlepszym parametrem k z liczb 2,5 i 7, okazała się wartość k = 2. Wtedy osiągnięto łącznie najmniejszy błąd dla zbioru testowego - 19%, jak i uczącego - 39%.

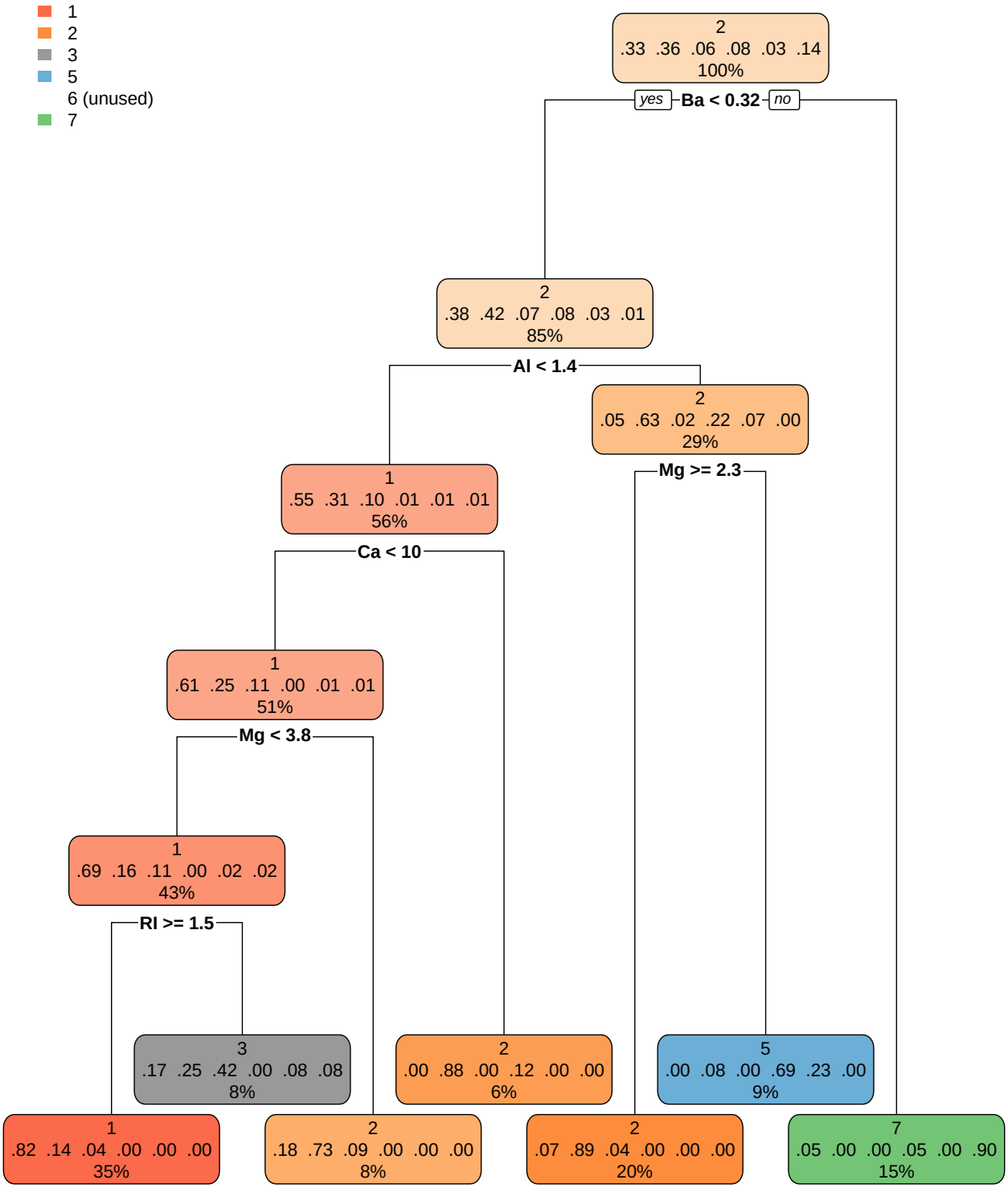
Dodatkowo sprawdzono skuteczność klasyfikacji przy użyciu tylko czterech najbardziej istotnych zmiennych (Mg, RI, Ba, Al). Wyniki były najlepsze dla zbioru testowego, gdy pod uwagę wzięto cały zbiór (39%), natomiast błąd dla zbioru uczącego wyszedł najmniejszy dla zbioru czterech zmiennych - 10%.

Można się zastanawiać, czy model nie został w tym przypadku przeuczony, natomiast większe liczby sąsiadów dawały coraz gorsze wyniki zarówno dla zbioru testowego jak i uczącego.

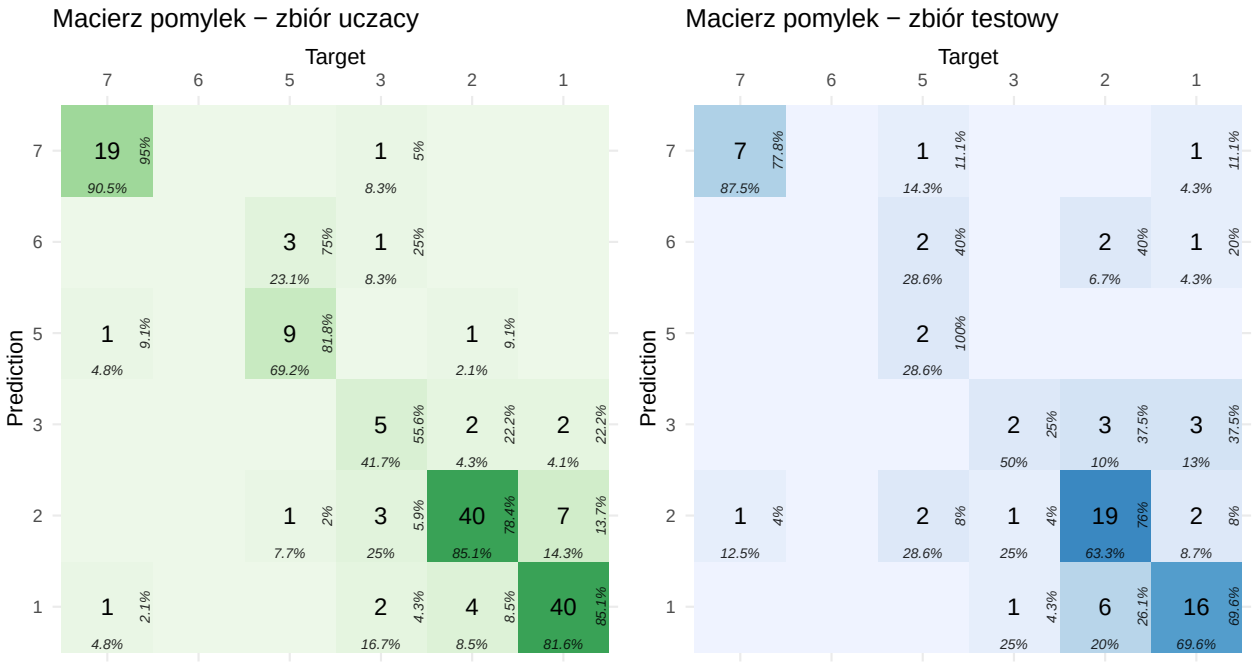
3.3 Metoda drzew klasyfikacyjnych

W tej metodzie skorzystamy z surowych danych z naszego zbioru, bez standaryzacji.

3.3.1 Drzewa klasyfikacyjne dla różnych parametrów

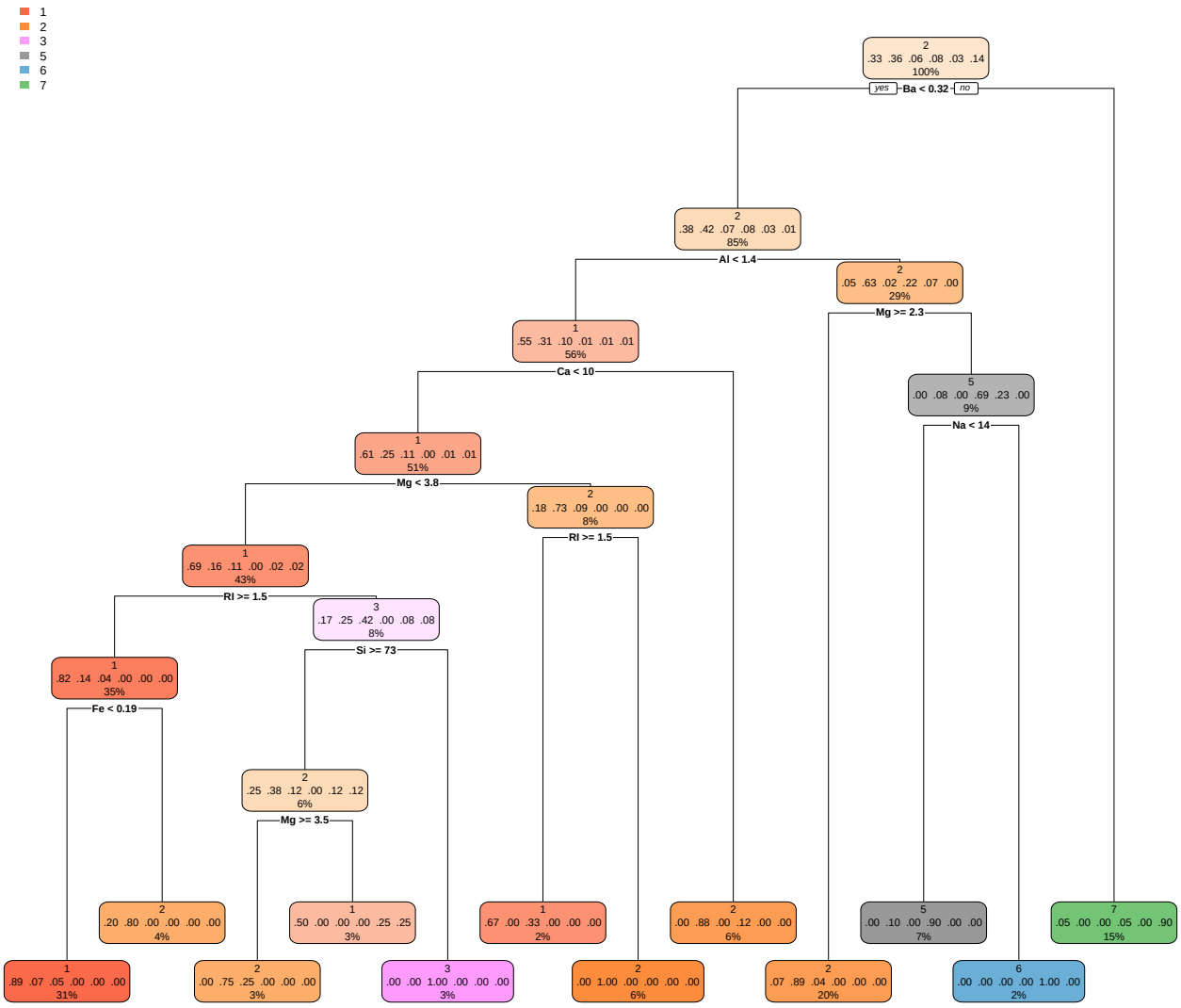


Na powyższym rysunku możemy zobaczyć konstrukcję drzewa klasyfikacyjnego

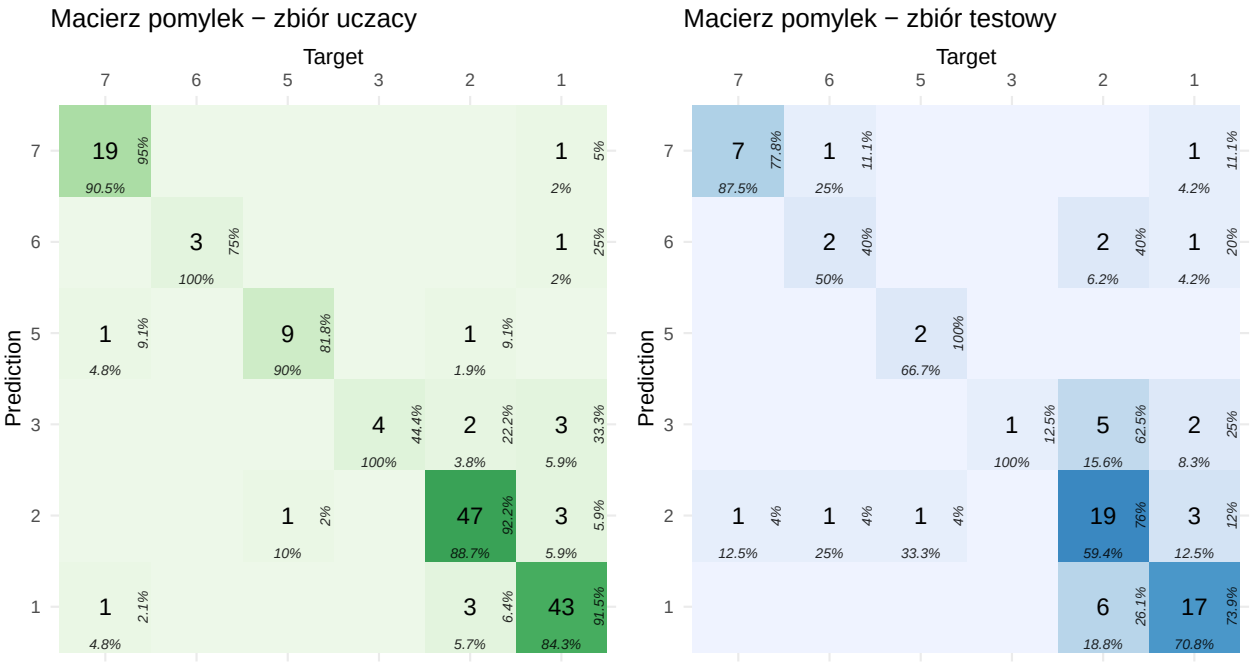


Powyżej na macierzach odmienności widzimy, jak zachowywał się nasz model. Otrzymaliśmy błędy na poziomie: 0.2, 0.36, odpowiednio dla zbioru uczącego i testowego.

Spróbujmy stworzyć teraz drzewo z innymi parametrami niż standardowe:



Powyżej mamy skonstruowane drzewo z parametrami: cp=.02, minsplit=8, maxdepth=7. Są to zoptymalizowane parametry dla których łączna suma błędów na zbiorze uczącym i testowym wynosi odpowiednio: 0.12, 0.33.

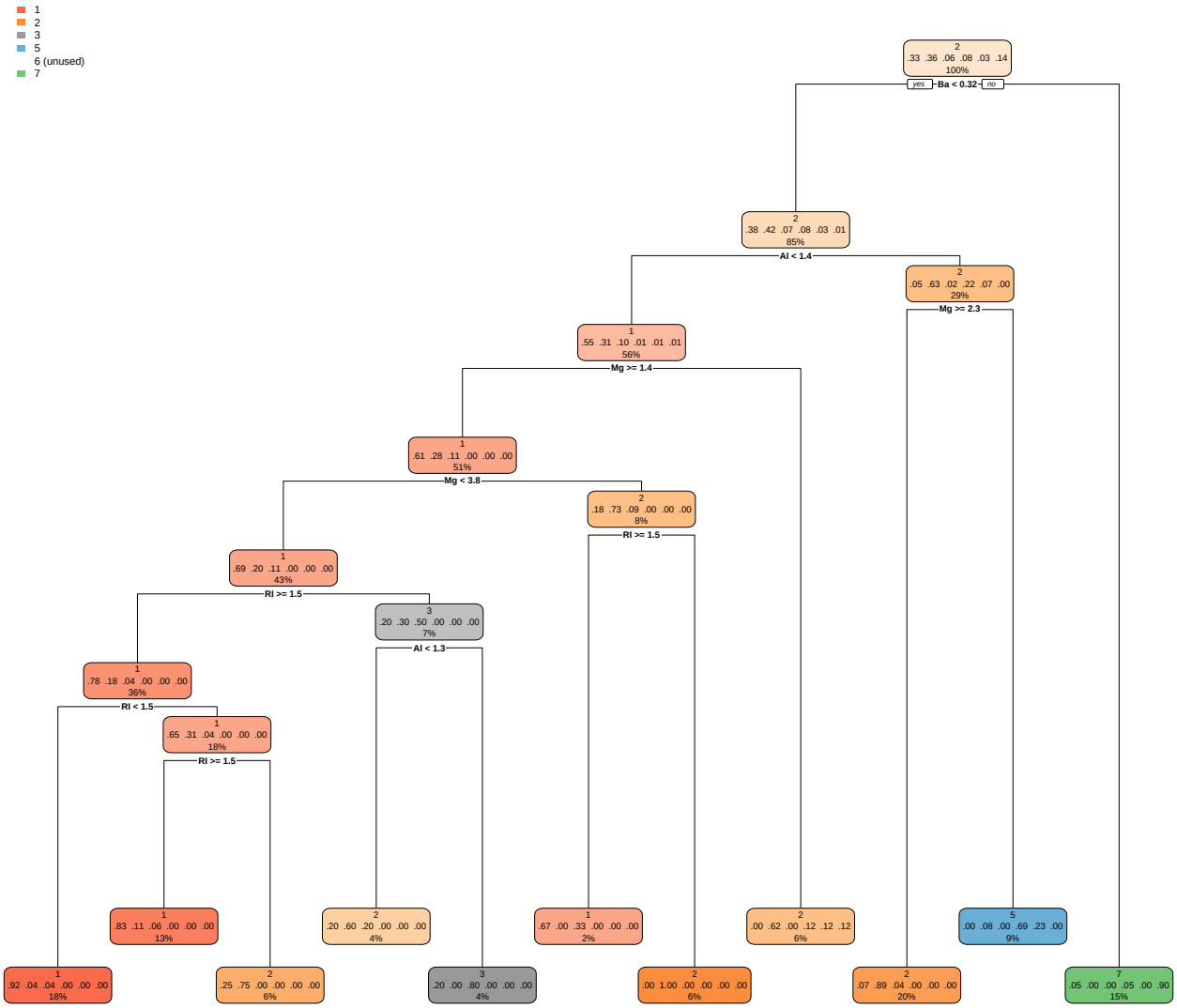


Na powyższych macierzach odmienności widzimy że udało nam się znacząco zmniejszyć błąd klasyfikacyjny.

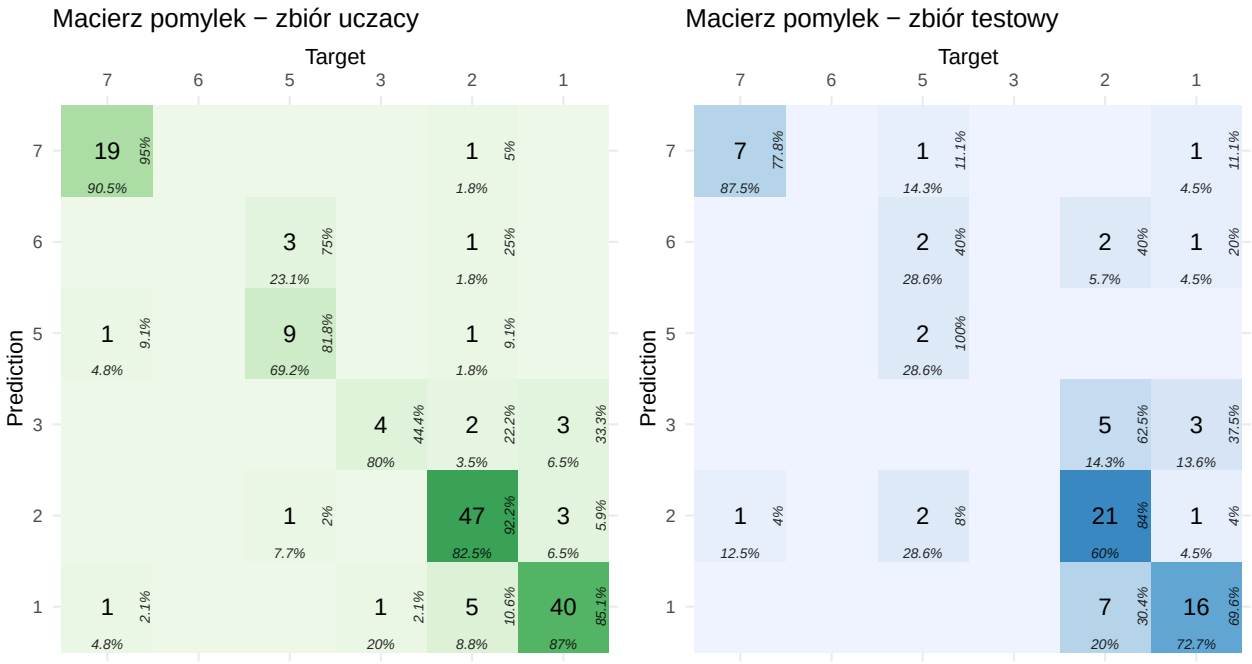
3.3.2 Drzewa klasyfikacyjne dla różnych podzbiorów

W tym podpunkcie wykorzystamy naszą lepszą wersję drzewa, która posiada zoptymalizowane parametry.

Ponownie przetestujemy ją na zmiennych o najlepszej zdolności dyskryminacyjnej tj. Mg, RI, Ba, Al. Następnie porównamy otrzymany wynik z błędami dla wszystkich danych.



Powyżej widzimy drzewo dla czterech najważniejszych parametrów.



W tym przypadku zmniejszenie liczby cech nie przyniosło dobrego skutku, przy takich samych parametrach drzewo zwiększyło swój błąd klasyfikacyjny, który wynosi : 0.16, 0.36 dla odpowiednio zbbioru uczącego i testowego. Przy zmianie parametrów można było zmniejszyć błąd, jednak w każdym przypadku był on większy od tego uzyskanego przy wszystkich cechach.

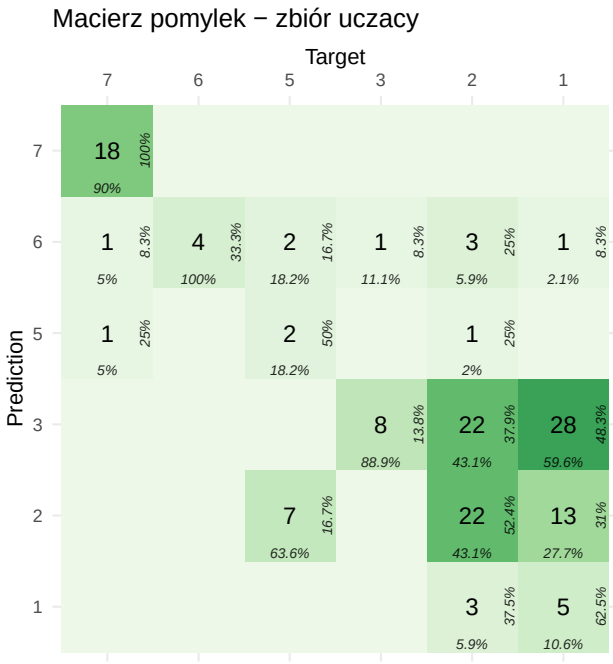
3.3.3 Podsumowanie

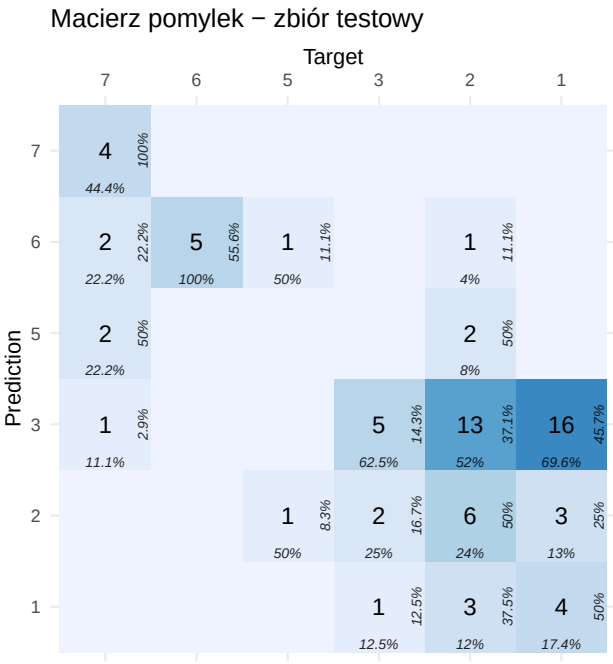
Podsumowując, metoda drzew klasyfikacyjnych okazała się skuteczna po odpowiedniej optymalizacji parametrów. Redukcja liczby cech nie zawsze prowadzi do poprawy jakości modelu – może natomiast zwiększyć błąd klasyfikacyjny, jeśli usunięte zmienne zawierały cenne informacje.

3.4 Naiwny klasyfikator bayesowski

3.4.1 Naiwny klasyfikator bayesowski na różnych parametrach

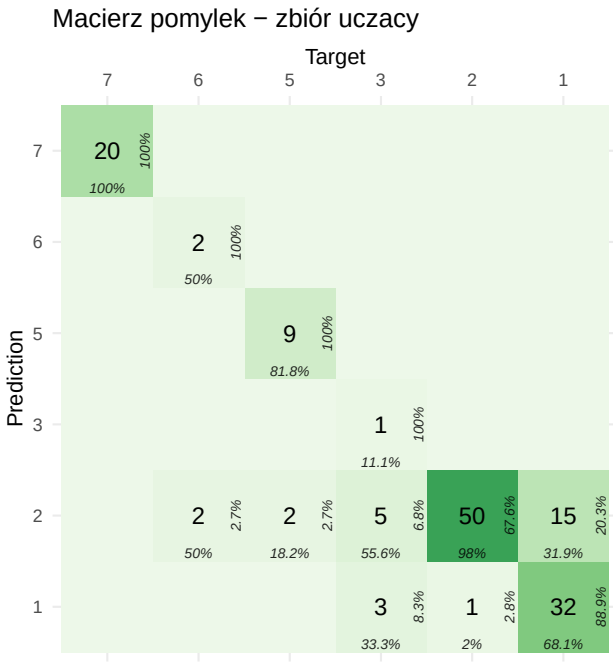
Spróbujemy stworzyć nasz model na podstawie naiwnego klasyfikatora bayesowskiego, najpierw standardową wersję, a następnie z jądrową estymacją gęstości.

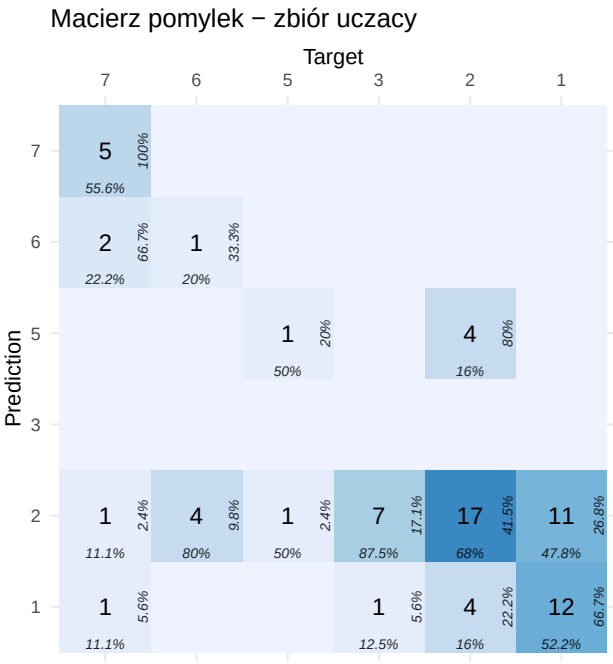




Jak możemy zobaczyć na powyższych macierzach, naiwny klasyfiaktor bayesowski nie poradził sobie dobrze z naszymi danymi. Błąd w zbiorze uczącym wynosi: 0.16, oraz w zbiorze testowym: 0.36.

Spróbujmy teraz zastosować jądrową estymacje gęstości za pomocą funkcji NaiveBayes z pakietu klaR.



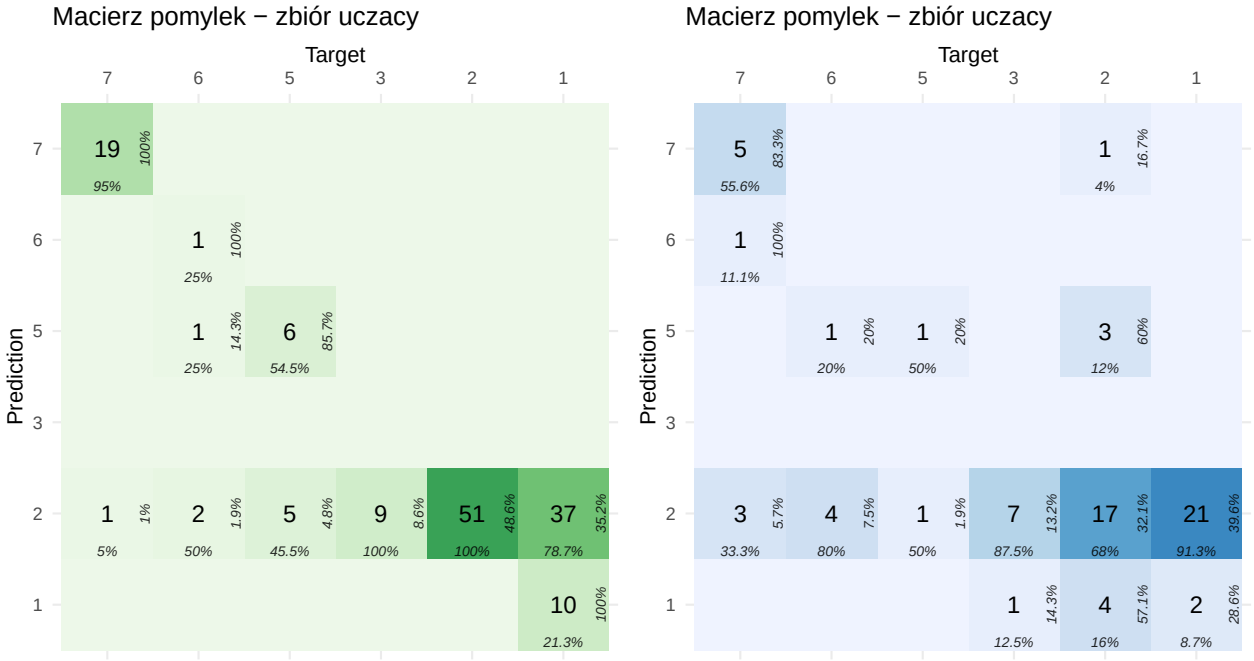


Jak widzimy, jądrowa estymacja gęstości poprawiła nasz błąd i wynosi teraz 0.2 dla zbioru uczącego oraz 0.5 dla zbioru testowego.

3.4.2 Naiwny klasyfikator bayesowski na różnych danych

Spróbujmy teraz sprawdzić jaki wpływ na wynik będzie miało wybranie podzbioru cech o najlepszych zdolnościach dyskryminacyjnych tj. Mg, RI, Ba, Al. Następnie porównamy otrzymany wynik z błędami dla wszystkich danych.

Będziemy stosować model o lepszej dokładności - czyli z jądrową estymacją gęstości.

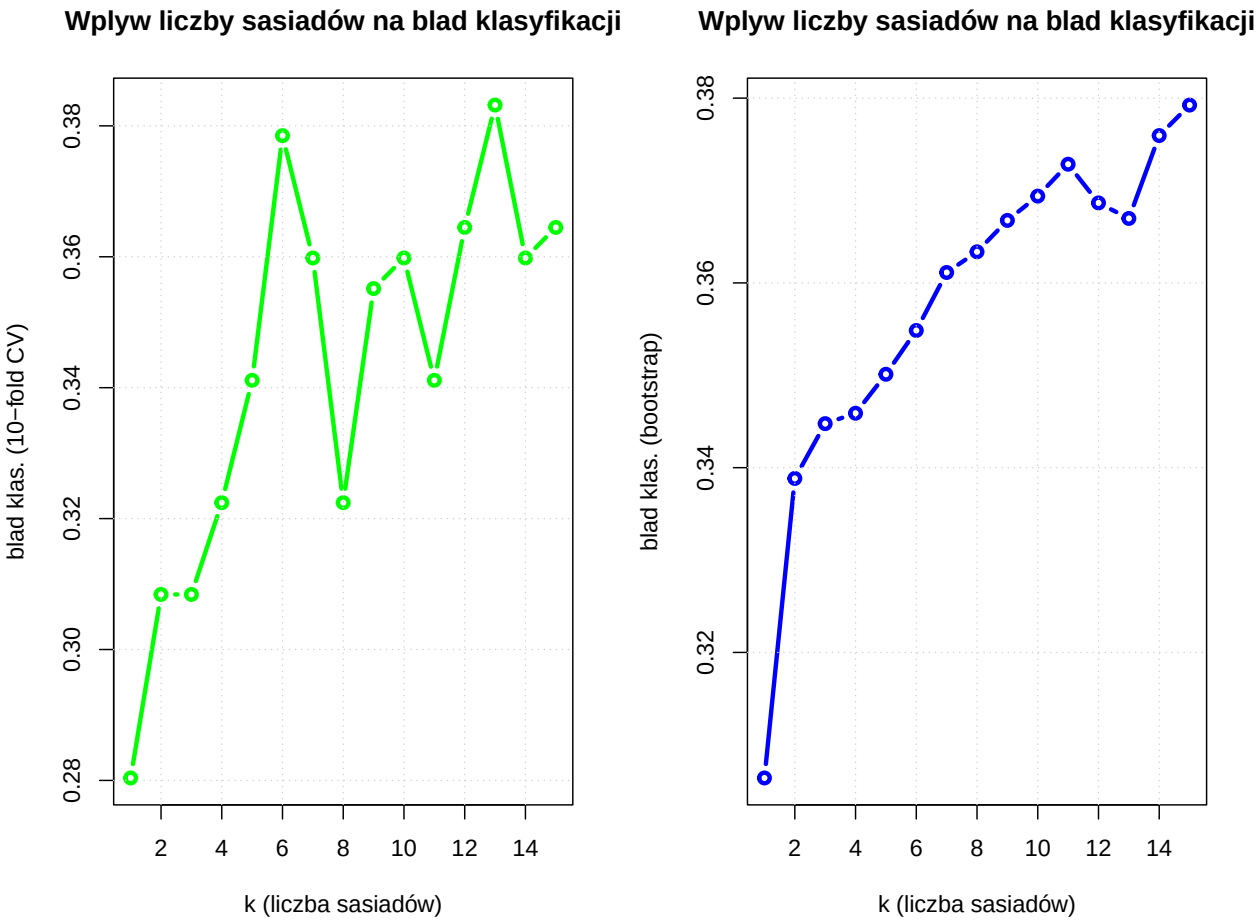


Na powyższych macierzech pomyłek widzimy, że nasz model niektóre klasy zaczął rozróżniać bardzo dobrze, natomiast niektórych nie potrafi kompletnie. Ogólnie jakość dopasowania spadła, głównie za sprawą słabej rozróżnialności klasy 1, co może sugerować, że usunęliśmy jakąś cechę dobrze dyskryminującą klasę 1. Błąd dla zbioru uczącego wynosi 0.39, natomiast dla testowego: 0.65

3.5 Zaawansowane porównanie metod klasyfikacji

Dla naszych modeli będziemy stosować metodę cross-validation oraz schemat typu bootstrap.

3.5.1 Metoda kNN



Jak możemy zobaczyć, wyniki zaawansowanych metod oceny błędu dały podobne wyniki co podział na zbiór uczący i testowy. Najmniejsze błędy dała nam klasyfikacja dla 2 najbliższych sąsiadów. Wyniki błędu klasyfikacyjnego dla metody bootstrap i 10-krotnej walidacji dały podobne błędy jednak nieco wyższe niż liczone podstawowymi metodami. Pokazuje to że nawet proste metody oceny dokładności dają nam czasem dobre rezultaty.

3.5.2 Metoda - Naiwny Klasyfikator Bayesowski

Błąd klasyfikacji dla naiwnego klasyfikatora bayesowskiego według metody 10-fold CV wyniósł: 0.593, natomiast dla metody bootstrap: 0.614. Widzimy że wyniki te są bardzo podobne do naszej podstawowej metody gdzie błąd dla zbioru testowego był trochę wyższy (ok. 0.66).

3.5.3 Metoda - drzewa klasyfikacyjne

Znowu wyniki mamy bardzo podobne do tych które obliczyliśmy podstawowymi metodami. Wtedy otrzymaliśmy błąd wielkości 0.2 dla zbioru uczącego oraz 0.36 dla zbioru testowego. W tym przypadku bliżej nam do metody 10-fold CV, gdzie błąd wyniósł: 0.299, ale za to metoda Bootstrap dała nam lepsze rezultaty i błąd wyniósł ok. 0.355. Dla większych danych takie

różnice przynoszą znaczne skutki, jednak dla nas metody podstawowe dają tutaj również zadowalające wyniki.

3.6 Wnioski końcowe

3.6.1 Dobór zmiennych i parametrów:

3.6.1.1 k-NN (k-najbliższych sąsiadów) Najlepsze wyniki uzyskano dla parametru k = 2, przy wykorzystaniu pełnego zestawu cech. W takim przypadku błąd klasyfikacji wyniósł 0.19 dla zbioru uczącego oraz 0.39 dla zbioru testowego. Próba ograniczenia zmiennych tylko do tych o największej mocy dyskryminacyjnej (Mg, RI, Ba, Al) nie poprawiła znacząco wyników – co więcej, dla zbioru testowego błąd pozostał na podobnym poziomie. Zwiększanie liczby sąsiadów do k = 5 i k = 7 pogarszało rezultaty, co sugeruje, że dla danych Glass mniejsze wartości k lepiej radzą sobie z lokalną strukturą klas.

3.6.1.2 Naiwny klasyfikator Bayesa Najlepsze rezultaty uzyskano przy zastosowaniu **jądrowej estymacji gęstości (NaiveBayes z klaR), z wykorzystaniem pełnego zbioru cech – wtedy błąd klasyfikacji wynosił 0.20 (uczący) i 0.50 (testowy). Przy ograniczeniu do czterech cech (Mg, RI, Ba, Al), skuteczność znacząco spadła – błąd klasyfikacji wzrósł do 0.39 (uczący) i 0.65 (testowy). Wskazuje to, że model Bayesa, który zakłada niezależność cech, traci na dokładności, gdy brakuje mu informacji ukrytej w dodatkowych zmiennych. Wersja klasyczna (bez estymacji gęstości) wypadła jeszcze słabiej (błąd do 0.66).

3.6.1.3 Drzewa klasyfikacyjne Najlepsze wyniki osiągnięto dla pełnego zestawu cech przy zastosowaniu optymalnych parametrów: `cp = 0.02`, `minsplit = 8`, `maxdepth = 7`. W tym ustawieniu model osiągnął błąd 0.12 na zbiorze uczącym oraz 0.33 na testowym. Próba redukcji cech do najbardziej informacyjnych (Mg, RI, Ba, Al) doprowadziła do pogorszenia wyników – błąd wzrósł do 0.16 (uczący) i 0.36 (testowy). Pokazuje to, że nawet pozornie mniej istotne cechy mogły zawierać dodatkową informację strukturalną istotną dla klasyfikacji w drzewach.

3.6.2 Porównanie metod klasyfikacyjnych:

Spośród zastosowanych metod klasyfikacyjnych, najlepiej wypadły drzewa decyzyjne, szczególnie po dostrojeniu parametrów – były one skuteczniejsze niż metoda k-NN czy naiwny klasyfikator bayesowski. Metoda k-NN dawała przyzwoite wyniki przy niskim k, ale skuteczność szybko malała wraz ze wzrostem liczby sąsiadów. Naiwny klasyfikator bayesowski osiągał gorsze rezultaty w wersji standardowej, a dopiero zastosowanie jądrowej estymacji gęstości poprawiło jakość klasyfikacji – choć i tak pozostawała niższa niż w przypadku drzew. Największe trudności we wszystkich metodach sprawiało rozróżnienie niektórych rzadkich klas (np. typu 1 i 6), co wynikało m.in. z niezrównoważenia danych.

3.6.3 Wpływ schematu oceny dokładności:

Wprowadzenie zaawansowanych metod oceny, takich jak 10-fold cross-validation i bootstrap, nie zmieniło zasadniczych wniosków dotyczących skuteczności metod. Choć uzyskane błędy

były nieco niższe niż przy jednokrotnym podziale danych, chociaż nie we wszystkich przypadkach (np. dla k-NN błędy z CV były większe niż przy zwykłym podziale), ogólna kolejność skuteczności metod pozostała taka sama: drzewa > k-NN > naiwny Bayes. Pokazuje to, że prostsze schematy oceny (np. podział 2/3-1/3) były wystarczające do uzyskania wiarygodnych porównań, choć bardziej zaawansowane metody lepiej szacują uogólnialność modeli i pozwalają uzyskać niższe błędy.