Uniwersytet Warszawski
Wydział Matematyki, Informatyki i Mechaniki

Zuzanna Branicka
Nr albumu: 214711

Metody konstrukcji oraz symulacyjne badanie właściwości jednorodnych i niejednorodnych komitetów klasyfikatorów

Praca magisterska
na kierunku MATEMATYKA
w zakresie MATEMATYKI STOSOWANEJ

Praca wykonana pod kierunkiem
dr. inż. Przemysława Biecka
Instytut Matematyki Stosowanej i Mechaniki
Zakład Statystyki Matematycznej

Czerwiec 2011

Oświadczenie kierującego pracą

Potwierdzam, że niniejsza praca została przygotowana pod moim kierunkiem i kwalifikuje się do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data Podpis kierującego pracą

Oświadczenie autora (autorów) pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data Podpis autora (autorów) pracy

Streszczenie

W pracy podjęto temat metod klasyfikacji opartych na głosowaniu. Zaprezentowano krótki przegląd najpopularniejszych algorytmów budowy jednorodnych komitetów klasyfikatorów (takich jak bagging, boosting i lasy losowe) oraz wyniki eksperymentów przeprowadzonych na lasach losowych o wybranych cechach poddanych modyfikacjom. Ponadto w oparciu o przeanalizowane metody zaproponowano własny algorytm budowy komitetu niejednorodnego, który następnie wszechstronnie przetestowano w porównaniu z różnymi wariantami komitetów jednorodnych.

Słowa kluczowe

niejednorodny komitet klasyfikatorów, ważony komitet klasyfikatorów, lasy losowe, bagging, boosting

Dziedzina pracy (kody wg programu Socrates-Erasmus)

11.2 Statystyka

Klasyfikacja tematyczna

62H30 Classification and discrimination; cluster analysis

Tytuł pracy w języku angielskim

Construction methods and simulation testing of properties of homogeneous and heterogeneous ensemble classifiers

Spis treści

Wstęp

Metody oparte na głosowaniu to dynamicznie rozwijające się od dwudziestu lat podejście w dziedzinie klasyfikacji. Charakterystyczną cechą tych metod jest wykorzystywanie nie pojedynczych klasyfikatorów, ale całych ich rodzin (komitetów). Klasyfikacje dokonane przez poszczególne klasyfikatory traktowane są jak głosy oddane na daną klasę, które następnie przy zastosowaniu wybranej reguły głosowania składają się na decyzję całego komitetu. Komitety klasyfikatorów osiągają bardzo dobrą jakość klasyfikacji, często dając błąd klasyfikacji wielokrotnie mniejszy niż ich składowe. Najpopularniejsze wśród nich, boosting i lasy losowe, są praktycznie od razu gotowe do użycia, nie wymagają wstępnej analizy danych ani skomplikowanej kalibracji. Są to jednak komitety jednorodne, a więc składające się z klasyfikatorów tego samego typu. Nasuwa się pytanie, czy wprowadzenie do komitetu klasyfikatorów różnych rodzajów nie pozwoliłoby dodatkowo poprawić ich jakości klasyfikacji. Jednoczesne wykorzystanie kilku typów klasyfikatorów może pomóc w lepszym dopasowaniu komitetu do struktury klas w badanym zbiorze, a zarazem jest oszczędniejsze obliczeniowo od budowy kilku komitetów jednorodnych i wyboru najlepszego spośród nich. Ponadto klasyfikatory różnych typów są od siebie słabiej zależne niż klasyfikatory jednego rodzaju, co również może pozytywnie wpłynąć na jakość klasyfikacji komitetu.

W niniejszej pracy podjęto próbę skonstruowania algorytmu budowy komitetu niejednorodnego, który osiągałby możliwie najlepszą jakość klasyfikacji (w porównaniu z komitetami jednorodnymi) bez względu na badany zbiór. Jednocześnie na wzór lasów losowych i boostingu nie powinien on wymagać wstępnych testów czy analizy danych oraz charakteryzować się niewielką liczbą parametrów. Aby przygotować się do tego zadania, w rozdziale pierwszym przeanalizowano najważniejsze algorytmy konstrukcji komitetów klasyfikatorów (bagging, lasy losowe, boosting), natomiast w rozdziale drugim dokładniej zbadano mechanizmy prowadzące do redukcji błędu komitetu, testując różne modyfikacje lasów losowych. Ostatecznie w rozdziale trzecim przedstawiono kolejne etapy budowania i testowania zaproponowanego komitetu niejednorodnego.

Rozdział 1

Komitety klasyfikatorów - wprowadzenie

Rozważając zagadnienie klasyfikacji, dobrze jest przyjąć następującą postać **zbioru danych** (**próby uczącej**): $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, gdzie każda z **obserwacji** x_i, $i = 1, \ldots, N$, jest wektorem losowym o wartościach w M-wymiarowej przestrzeni $\mathcal{X}$, natomiast y_i, $i = 1, \ldots, N$, to **etykieta klasy** (populacji, grupy), do której należy obserwacja x_i. Jeśli obserwacje pochodzą z C różnych klas, to można dla uproszczenia przyjąć, że $y_i \in \mathcal{C} = \{1, 2, \ldots, C\}$. Współrzędne wektora obserwacji x_i będą zaś określane mianem **zmiennych** bądź **atrybutów**.

Przez **regułę klasyfikacyjną** (**klasyfikator**) rozumie się odwzorowanie przypisujące dowolnej obserwacji $x \in \mathcal{X}$ przynależność do klasy ze zbioru klas $\mathcal{C}$, czyli funkcję $d : \mathcal{X} \to \mathcal{C}$. Klasyfikator dzieli więc przestrzeń $\mathcal{X}$ na C rozłącznych podprzestrzeni. Konstruuje się go na podstawie próby uczącej Z[1].

Prawdopodobieństwo błędu klasyfikacji klasyfikatora d to inaczej prawdopodobieństwo zaklasyfikowania przez regułę d losowej obserwacji do niewłaściwej klasy, czyli:

$$\mathbb{P}(d(x) \neq y) = \mathbb{E}_{xy} I(d(x) \neq y), \tag{1.1}$$

gdzie x – obserwacja z przestrzeni $\mathcal{X}$, a y – jej etykieta o wartościach w C. Prawdopodobieństwo błędu można więc szacować przez ułamek błędnych klasyfikacji na próbie testowej $Z^t = \{(x_1^t, y_1^t), \ldots, (x_J^t, y_J^t)\}$:

$$err = \frac{1}{J} \sum_{j=1}^{J} I(d(x_j^t) \neq y_j^t) \tag{1.2}$$

Dla uproszczenia powyższe oszacowanie nazywa się w niniejszej pracy **błędem klasyfikacji** klasyfikatora d.

Załóżmy teraz, że dysponujemy K niezależnymi klasyfikatorami, które na pewnym zbiorze danych o liczbie klas $C = 2$ charakteryzują się prawdopodobieństwem błędu równym $p = \frac{1}{2} - \epsilon$, $\epsilon > 0$. Wtedy prawdopodobieństwo prawidłowej klasyfikacji przez pojedynczy klasyfikator wynosi $1 - p = \frac{1}{2} + \epsilon$. Oczekiwana proporcja prawidłowych klasyfikacji wśród wszystkich K klasyfikatorów to $1 - p$, natomiast wariancja i odchylenie standardowe tej proporcji będą równe odpowiednio: $\frac{p(1-p)}{K}$ i $\sqrt{\frac{p(1-p)}{K}}$. Dla wystarczająco dużych K zachodzi:

$$1 - p - \frac{1}{2} = \epsilon > \sqrt{\frac{p(1-p)}{K}}, \tag{1.3}$$

[1] J. Koronacki, J. Ćwik, *Statystyczne systemy uczące się*, s. 15-17.

a więc można być prawie pewnym, że większość klasyfikatorów podejmie poprawną decyzję[2]. Tę obserwację wykorzystano do konstruowania komitetów klasyfikatorów, choć w praktyce klasyfikatory oparte na tej samej próbie uczącej będą od siebie zależne. Zależność klasyfikatorów nie zmienia wartości oczekiwanej proporcji poprawnych klasyfikacji, natomiast powoduje wzrost jej odchylenia standardowego, a tym samym spadek prawdopodobieństwa, że większość klasyfikatorów podejmie prawidłową decyzję. W przypadku rodziny klasyfikatorów o błędzie klasyfikacji p oraz korelacji ρ między każdymi dwoma klasyfikatorami wariancja proporcji prawidłowych klasyfikacji wyniesie:

$$\rho(1-p)p + \frac{(1-\rho)(1-p)p}{K}. \tag{1.4}$$

W takiej sytuacji zwiększanie K pozwala dowolnie zmniejszać drugi składnik wariancji, ale nie może wpłynąć na pierwszy[3]. Zależność klasyfikatorów znacznie ogranicza więc korzyści z uśredniania ich decyzji. Dlatego w budowie komitetu redukcja korelacji klasyfikatorów jest często istotniejsza od troski o niskie prawdopodobieństwo błędu pojedynczego klasyfikatora. Właśnie na tym aspekcie skupił się Leo Breiman, twórca dwóch pierwszych metod klasyfikacji opartych na głosowaniu opisanych w niniejszym rozdziale.

1.1. Bagging

Pierwszym podejściem do ograniczania zależności klasyfikatorów jest trenowanie ich na próbach bootstrapowych zamiast na próbie oryginalnej. Klasyfikacji przy pomocy komitetu stworzonego z takich klasyfikatorów dokonuje się, przydzielając obserwację do tej klasy, do której zakwalifikowała ją większość członków komitetu. Tę metodę agregacji klasyfikatorów nazywa się baggingiem. Bardziej formalnie, zdefiniujmy oryginalną próbę jako $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, gdzie x_i – wektor obserwacji, $y_i \in \{1, \ldots, C\}$ – etykieta klasy. Niech $G(x) = \arg\max_c \hat{f}(x)$ będzie predykcją etykiety klasy dla obserwacji x dokonywaną przez klasyfikator wytrenowany na Z na podstawie funkcji wektorowej $f(x) = (f_1, f_2, \ldots, f_C)$, przy czym $f_c \in \{0, 1\}$, $\sum_{c=1}^{C} f_c = 1$. Teraz przez Z^k oznaczmy próbę bootstrapową na Z, $k = 1, 2, \ldots, K$, natomiast przez $f^k(x)$ – wartość funkcji indykującej klasę dla obserwacji x klasyfikatora wytrenowanego na pseudopróbie Z^k. Wtedy zagregowany klasyfikator G_{bag} można sformułować następująco[4]:

$$f_{bag}(x) = \frac{1}{K}\sum_{k=1}^{K} f^k(x) \tag{1.5}$$

$$G_{bag}(x) = \arg\max_{c\in\{1,\ldots,C\}} f_{bag}(x) \tag{1.6}$$

Zagregowana funkcja indykująca klasę $f_{bag}(x)$ ma więc postać wektora $(p_1, p_2, \ldots, p_C)$, gdzie p_c jest równe stosunkowi liczby klasyfikatorów przewidujących c-tą klasę do liczby wszystkich klasyfikatorów.

Dokładny algorytm baggingu ma zaś następującą postać (krok 1 obejmuje budowę komitetu, zaś kroki $2-4$ predykcję przy jego pomocy):

[2]Ibidem, s. 148-149.
[3]T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning*, s. 588
[4]Ibidem, s. 283.

Algorytm 1. Bagging

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$,
K – liczba klasyfikatorów, x – obserwacja wymagająca klasyfikacji.

1. Dla każdego $k = 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem, tworząc pseudopróbę Z^k.
   (b) Wytrenuj klasyfikator na pseudopróbie Z^k.
2. Dokonaj predykcji klasy dla x przy pomocy wszystkich klasyfikatorów $f^k(x)$, $k = 1, \ldots, K$.
3. Oblicz średnią $f_{bag}(x) = \frac{1}{K} \sum_{k=1}^{K} f^k(x)$.
4. Podaj $G_{bag}(x) = \arg\max_{c \in \{1,\ldots,C\}} f_{bag}(x)$.

1.2. Lasy losowe

Lasy losowe to druga z metod zaproponowanych przez Breimana, w dużym stopniu oparta na baggingu. Jednak podczas gdy klasyfikatory stosowane w baggingu mogą być dowolnego typu, las losowy musi składać się z drzew decyzyjnych. Jest to spowodowane wprowadzeniem dodatkowego sposobu redukcji zależności między klasyfikatorami w komitecie. Każde drzewo trenowane jest na próbie bootstrapowej, ale dodatkowo w każdym węźle niezależnie losuje się podzbiór zmiennych, spośród których wybierany jest atrybut wykorzystywany do podziału podpróby w tym węźle. Liczba losowanych zmiennych jest stała i znacznie mniejsza od całkowitej liczby atrybutów w próbie. Ten zabieg nie tylko ogranicza zależność między drzewami w lesie, ale też umożliwia klasyfikację na zbiorach o bardzo dużej liczbie zmiennych. Należy jeszcze dodać, że drzewa w lesie losowym nie są przycinane, powinny osiągać maksymalny możliwy rozmiar (liście mają zawierać albo obserwacje z tej samej klasy, albo pojedyncze obserwacje).

Formalnie las losowy definiuje się jako zbiór klasyfikatorów o strukturze drzew decyzyjnych $\{T(x, \Theta_k)\}_1^K$, gdzie Θ_k są losowymi wektorami i.i.d, natomiast funkcja indykująca klasę oraz predyktor dla obserwacji x mają postać[5]:

$$T_{rf}(x) = \frac{1}{K} \sum_{k=1}^{K} T(x, \Theta_k) \tag{1.7}$$

$$G_{rf}(x) = \arg \max_{c \in \{1,\ldots,C\}} T_{rf}(x), \tag{1.8}$$

przy oznaczeniach analogicznych do poprzedniego podrozdziału. Wektor losowy Θ_k zawiera pełną charakterystykę k-tego drzewa, a więc informacje o wybranej w każdym węźle zmiennej służącej do podziału podpróby wraz z jej wartością dzielącą oraz o etykietach liści.

Algorytm konstruowania i predykcji przy pomocy lasu losowego można zaś sformułować następująco (krok 1 opisuje budowę komitetu, a kroki $2-4$ – predykcję):

[5] Ibidem, s. 589, L. Breiman, *Random Forests*, s. 6.

Algorytm 2. Las losowy

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$, K – liczba drzew, M – liczba zmiennych w próbie Z, x – obserwacja wymagająca klasyfikacji.

1. Dla każdego $k = 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem, tworząc pseudopróbę Z^k.
   (b) Wytrenuj drzewo decyzyjne T^k na pseudopróbie Z^k, dla każdego węzła wykonując następujące czynności, dopóki liczba obserwacji w węźle nie będzie równa 1 lub wszystkie obserwacje w węźle nie będą miały jednakowych etykiet:
      i. Spośród M zmiennych wylosuj $m << M$ zmiennych bez zwracania.
      ii. Spośród m wylosowanych zmiennych wybierz najlepszy podział.
      iii. Podziel węzeł na dwa.
2. Dokonaj predykcji klasy dla x przy pomocy wszystkich drzew $T^k(x),\ k = 1, \ldots, K$.
3. Oblicz średnią $T_{rf}(x) = \frac{1}{K} \sum_{k=1}^{K} T^k(x)$.
4. Podaj $G_{rf}(x) = \arg\max_{c \in \{1,\ldots,C\}} T_{rf}(x)$.

Wartość m zalecana przez Breimana to w przypadku klasyfikacji $\sqrt{M}$.

Zarówno w baggingu, jak i w lasach losowych poszczególne klasyfikatory trenowane są na próbach bootstrapowych, z których każda składa się średnio z około dwóch trzecich spośród obserwacji oryginalnej próby. Obserwacje spoza pseudopróby uczącej (tak zwane obserwacje *out-of-bag*, OOB) można więc wykorzystać do szacowania różnych wielkości przydatnych tak w trakcie budowy komitetu, jak i na etapie analizy danych i predykcji. Na podstawie obserwacji OOB można obliczyć między innymi oszacowanie błędu klasyfikacji poszczególnych klasyfikatorów, błędu klasyfikacji komitetu (w takim wypadku każda z obserwacji oryginalnej próby jest klasyfikowana tylko przez te klasyfikatory, w których trenowaniu nie brała udziału), miarę istotności zmiennych oraz bliskości obserwacji, którą można wykorzystać m.in. do identyfikowania obserwacji odstających (ostatnia miara jest możliwa do obliczenia tylko w przypadku zastosowania drzew jako klasyfikatorów)[6].

1.3. Boosting

Twórcom boostingu przyświecała ta sama idea, co autorowi baggingu i lasów losowych: ich celem było skonstruowanie takiej kombinacji słabych klasyfikatorów (o błędzie klasyfikacji niewiele mniejszym od $\frac{1}{2}$), której jakość klasyfikacji byłaby dowolnie dobra[7]. Jednak zamiast starać się ograniczyć zależność klasyfikatorów (co było podstawą konstrukcji baggingu i lasów losowych), skupili się na redukcji ich błędu klasyfikacji. W przeciwieństwie do wcześniej opisanych metod, w których kolejność trenowania klasyfikatorów nie miała znaczenia, w boostingu komitet powstaje sekwencyjnie. Skład zbioru uczącego każdego kolejnego klasyfikatora zależy od zachowania poprzednich klasyfikatorów. Począwszy od drugiego klasyfikatora obserwacje do pseudopróby nie są już bowiem losowane zgodnie z rozkładem jednostajnym, lecz prawdopodobieństwo wylosowania danej obserwacji do pseudopróby rośnie, jeśli została źle zaklasyfikowana przez poprzedni klasyfikator. Można powiedzieć, że kolejne klasyfikatory są trenowane

[6] L. Breiman, *Random forests*, s. 11, T. Hastie, R. Tibshirani, J. Friedman, *The Elements...*, s. 592-596,
[7] Y. Freund, R. E. Schapire, *Game Theory, On-Line Prediction and Boosting*, s. 5.

na tych obserwacjach, które sprawiły najwięcej trudności ich poprzednikom. Następnie głosy poszczególnych klasyfikatorów w komitecie są ważone w zależności od popełnianego przez nie błędu klasyfikacji. Im mniejszy błąd, tym większa waga przypisana klasyfikatorowi, a więc większy wpływ jego głosu na decyzję komitetu. Innymi słowy, funkcją agregującą klasyfikatory w boostingu jest średnia ważona (w przypadku baggingu i lasów losowych była to zwykła średnia arytmetyczna).

Klasyfikator boostingowy G_{boost} opiera się więc na ciągu klasyfikatorów $\{G^k\}_1^K$ (przy czym $G^k(x) = \arg\max_c f^k(x)$) wytrenowanych na ciągu pseudoprób $\{Z^k\}_1^K$ takich, że rozkład Z^k zależy od klasyfikacji dokonanej na Z przez G^{k-1} (oznaczenia tak jak w poprzednich podrozdziałach). Klasyfikator G_{boost} może być sformułowany następująco:

$$f_{boost}(x) = \frac{1}{\sum_{k=1}^K \gamma_k} \sum_{k=1}^K \gamma_k f^k(x) \tag{1.9}$$

$$G_{boost}(x) = \arg \max_{c \in \{1,\ldots,C\}} f_{boost}(x), \tag{1.10}$$

Zagregowana funkcja indykująca klasę $f_{boost}(x)$ to podobnie jak w przypadku baggingu i lasu losowego wektor $(p_1, p_2, \ldots, p_C)$, jednak tym razem p_c jest równe ważonej średniej głosów oddanych na c-tą klasę wśród K klasyfikatorów. Szczegóły konstruowania pseudoprób treningowych Z^k oraz obliczania wag γ_k prezentuje poniższy algorytm (kroki 1-2 opisują konstrukcję komitetu, a kroki 3-5 – predykcję)[8]:

Algorytm 3. Boosting (AdaBoost.M1)

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$,
K – liczba klasyfikatorów, x – obserwacja wymagająca klasyfikacji.

1. Przyjmij wagi $w_i = \frac{1}{N}, \quad i = 1, \ldots, N$.
2. Dla każdego $k = 1, \ldots, K$:
   (a) Oblicz $p_i = \frac{w_i}{\sum_{i=1}^N w_i}, \quad i = 1, \ldots, N$.
   (b) Zgodnie z rozkładem p wylosuj ze zwracaniem N obserwacji z próby Z, tworząc pseudopróbę Z^k.
   (c) Wytrenuj klasyfikator na pseudopróbie Z^k i dokonaj predykcji $G^k(x_i) = \arg\max_c f^k(x_i)$ na wszystkich obserwacjach z próby Z.
   (d) Oblicz błąd klasyfikatora $err_k = \sum_{i=1}^N p_i I(y_i \neq G^k(x_i))$ oraz $\gamma_k = \log\left(\frac{1-err_k}{err_k}\right)$.
   (e) Zaktualizuj wagi $w_i \leftarrow w_i \exp(\gamma_k I(y_i \neq G^k(x_i)))$.
3. Dokonaj predykcji klasy dla obserwacji x przy pomocy wszystkich klasyfikatorów $f^k(x),\ k = 1, \ldots, K$.
4. Oblicz średnią $f_{boost}(x) = \frac{1}{\sum_{k=1}^K \gamma_k} \sum_{k=1}^K \gamma_k f^k(x)$.
5. Podaj $G_{boost}(x) = \arg\max_{c \in \{1,\ldots,C\}} f_{boost}(x)$.

[8] Y. Freund, R. E. Schapire, *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*, s. 131, T. Hastie, R. Tibshirani, J. Friedman, *The Elements...*, s. 339.

Ponieważ głównym celem niniejszej pracy jest skonstruowanie skutecznego niejednorodnego komitetu klasyfikatorów, warto przeanalizować dokładniej skutki rozwiązań ograniczających błąd klasyfikacji zastosowanych w baggingu, lasach losowych i boostingu. Przede wszystkim należy zbadać kwestię losowania zmiennych oraz wykorzystania średniej ważonej do agregacji klasyfikatorów. Rezultaty eksperymentów z różnymi cechami komitetów klasyfikatorów na podstawie lasów losowych prezentuje kolejny rozdział pracy.

Rozdział 2

Modyfikacje lasów losowych

W budowie lasów losowych charakterystyczne są trzy cechy:

1. Zmienne, spośród których wybierana jest jedna dająca najlepszy podział, losowane są niezależnie dla każdego węzła w każdym drzewie;

2. Głosy poszczególnych drzew są nieważone;

3. Drzewa nie są przycinane, powinny osiągać maksymalny możliwy rozmiar.

Sprawdźmy, w jaki sposób uchylenie każdego z tych założeń wpływa na jakość klasyfikacji zmodyfikowanego w ten sposób lasu (rozważane komitety będą w ogólniejszym sensie nazywane lasami, ponieważ poza wskazanymi modyfikacjami zachowują pozostałe cechy lasów losowych, w tym zastosowanie drzew decyzyjnych jako klasyfikatorów).

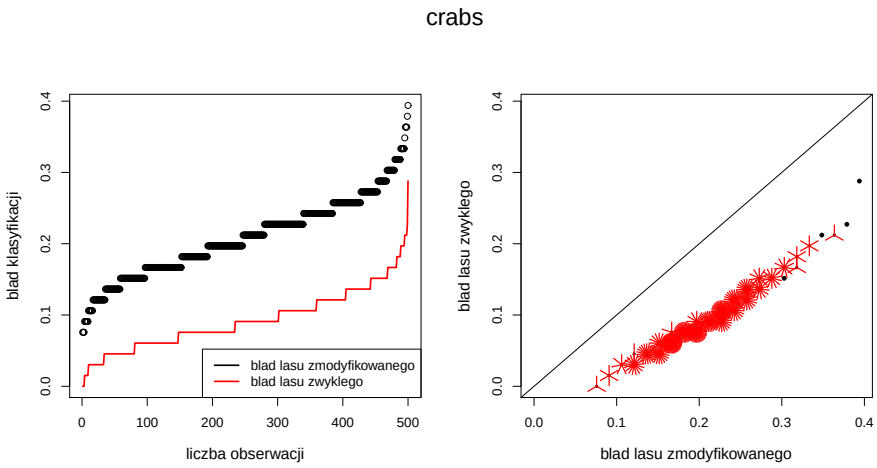
2.1. Lasy losowe z grupą zmiennych losowanych jednorazowo dla każdego drzewa

W pierwszej kolejności prześledzono zachowanie lasów losowych, w których grupa zmiennych losowana jest na nowo nie dla każdego węzła, a jedynie dla każdego drzewa. Zatem dla danego drzewa wybór najlepszego podziału dokonuje się w każdym z węzłów w obrębie tego samego zbioru zmiennych, aczkolwiek zdecydowanie mniej licznego niż wyjściowy zestaw zmiennych w badanej próbie. Algorytm budowania takiego lasu wygląda więc następująco:

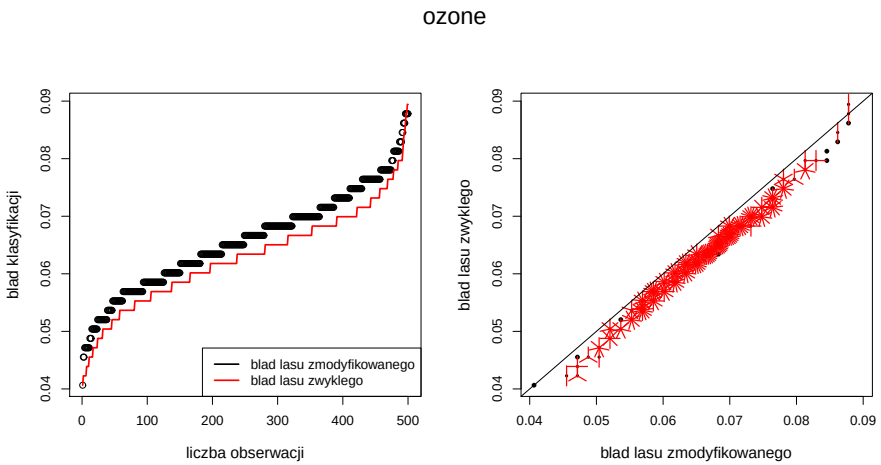
Algorytm 4. Las losowy z grupą zmiennych losowanych jednorazowo dla każdego drzewa

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$, K – liczba drzew,
M – liczba zmiennych w próbie Z, x – obserwacja wymagająca klasyfikacji.

1. Dla każdego $k = 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m << M$ zmiennych bez zwracania. Z wartości m wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^k.
   (c) Wytrenuj drzewo decyzyjne T^k na pseudopróbie Z^k (bez przycinania).
2. Dokonaj predykcji klasy dla x przy pomocy wszystkich drzew $T^k(x),\ k = 1, \ldots, K$.
3. Oblicz średnią $T_{mrf}(x) = \frac{1}{K} \sum_{k=1}^{K} T^k(x)$.
4. Podaj $G_{mrf}(x) = \arg\max_{c \in \{1,\ldots,C\}} T_{mrf}(x)$.



Rysunek 2.1: Porównanie rozkładów błędu klasyfikacji lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa oraz zwykłego lasu losowego na zbiorze `crabs`. Rysunek po lewej prezentuje odwrotne empiryczne dystrybuanty błędów klasyfikacji (bez normalizacji argumentów) dla obu algorytmów. Rysunek po prawej to rodzaj wykresów kwantylowych: na osi poziomej zaznaczono posortowane wartości błędu dla lasu zmodyfikowanego, a na osi pionowej - posortowane wartości błędu dla zwykłego lasu losowego. Liczba ramion w gwiazdach odzwierciedla liczbę obserwacji znajdujących się w danym punkcie. Obserwacja odpowiada pojedynczej symulacji lasu.



Rysunek 2.2: Porównanie rozkładów błędu klasyfikacji na zbiorze ozone dla lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa oraz zwykłego lasu losowego.

Działanie tak zmodyfikowanego lasu przeanalizowano, dokonując przy jego pomocy klasyfikacji na pięciu zbiorach danych[1]. Każdy zbiór dzielono losowo na zbiór uczący, na którym trenowano zmodyfikowany las, oraz zbiór testowy, służący do obliczenia błędu klasyfikacji (rozumianego jako ułamek błędnych klasyfikacji). Przyjęto liczbę drzew w lesie $K = 50$ oraz wartość $m = \lfloor \sqrt{M} \rfloor$, a więc równą wartości tego parametru stosowanej standardowo w zwykłych lasach losowych. Dla każdego zbioru całą procedurę powtórzono 500 razy, uzyskując w ten sposób 500-elementową próbę z populacji błędu. Następnie dokonano analogicznych symulacji dla zwykłego lasu losowego. Na koniec dla każdego zbioru danych porównano błędy

[1]Wszystkie zbiory wykorzystane do symulacji, zarówno rzeczywiste, jak i syntetyczne, zostały opisane w dodatku A na końcu niniejszej pracy.

klasyfikacji uzyskane przy obu algorytmach. Różnicę w błędach średnich przetestowano przy pomocy testu Wilcoxona, którego wyniki prezentuje tabela 2.1.

Zbiór	Błąd średni		Statystyka W	p-wartość
	Las zmodyfikowany	Las zwykły		
iris	0,06	0,05	137759	0,0042
crabs	0,21	0,09	237925	$< 2,2 \cdot 10^{-17}$
pima	0,27	0,23	206845	$< 2,2 \cdot 10^{-16}$
ozone	0,07	0,06	148434	$2,7 \cdot 10^{-7}$
ctg	0,09	0,06	247413	$< 2,2 \cdot 10^{-16}$

Tabela 2.1: Średnie błędy klasyfikacji dla lasu zmodyfikowanego i zwykłego lasu losowego oraz wyniki testu Wilcoxona.

W przypadku wszystkich zbiorów przewagę mają zwykłe lasy losowe. Dla części zbiorów (`crabs`, `pima`) różnica między błędem klasyfikacji zwykłego lasu i lasu zmodyfikowanego jest bardzo znacząca, dla pozostałych (`iris`, `ozone`) nieco mniejsza. Zawsze jednak klasyfikacja zwykłego lasu jest dokładniejsza niż lasu, w którym grupę zmiennych losuje się jednorazowo dla całego drzewa. Dotyczy to nie tylko wartości średniej błędu, ale całego rozkładu, co pokazują rysunki 2.1 i 2.2.

2.1.1. Analiza różnic między lasem zwykłym i zmodyfikowanym

Różnice w jakości klasyfikacji przy pomocy lasu zwykłego i lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa warto przeanalizować głębiej. Przede wszystkim należałoby odpowiedzieć na trzy pytania:

1. Z jakich czynników wynika przewaga lasu zwykłego nad zmodyfikowanym przy wybranych wartościach parametrów lasu?

2. W jaki sposób jakość klasyfikacji obu typów lasu oraz relacja między nimi reaguje na zmianę podstawowych parametrów lasu (liczba drzew K, liczba losowanych zmiennych m)?

3. Dlaczego różnica w jakości klasyfikacji przez oba typy lasu ma różne rozmiary w zależności od zbioru danych? Innymi słowy, w jaki sposób różne cechy zbioru danych wpływają na relację między błędami klasyfikacji obu typów lasu?

Przy poszukiwaniu odpowiedzi na powyższe pytania posłużono się symulacjami na syntetycznych zbiorach danych, co pozwoliło na kontrolę ich wybranych cech. Przetestowano również różne wartości parametrów lasu. Ponieważ na jakość klasyfikacji komitetu składają się przede wszystkim stopień niezależności klasyfikatorów w komitecie oraz jakość klasyfikacji poszczególnych klasyfikatorów, badano nie tylko błąd klasyfikacji całego lasu, ale też średnią korelację drzew w lesie oraz średni błąd klasyfikacji pojedynczego drzewa.

Wpływ różnic w strukturze lasu na składowe jakości klasyfikacji lasu zwykłego i zmodyfikowanego

W lesie zmodyfikowanym grupa zmiennych wykorzystywanych do budowy drzewa jest losowana jednorazowo dla każdego klasyfikatora w komitecie, a zatem w każdym węźle danego drzewa zmienna służąca do podziału będzie wybierana z tego samego zestawu, ale losowanego oddzielnie dla różnych drzew. Można się więc spodziewać, że drzewa w takim lesie będą

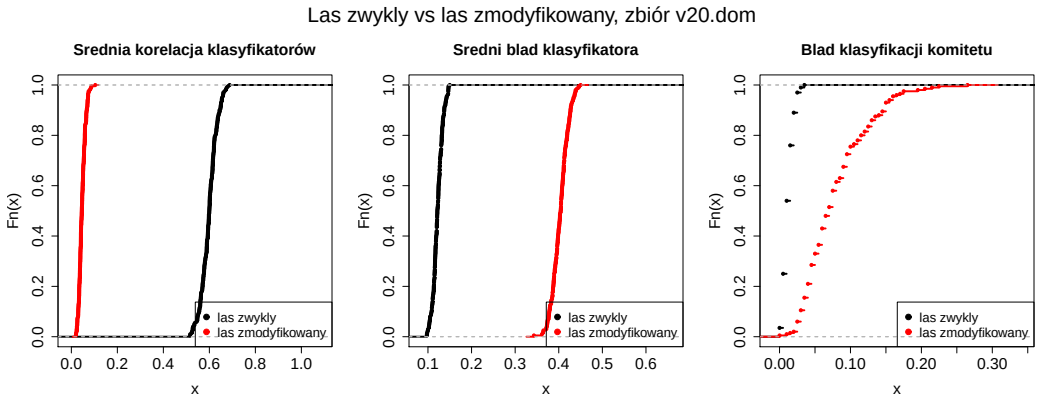
od siebie mniej zależne niż w zwykłym lasie losowym, gdzie zestaw zmiennych jest losowany na nowo dla każdego węzła i tym samym szansa wyboru tych samych zmiennych do budowy dwóch różnych drzew jest większa. Z drugiej strony wybór zmiennych do podziału jest w przypadku lasu zmodyfikowanego bardziej ograniczony, a tym samym szansa wyboru zmiennych najlepiej różnicujących zbiór danych mniejsza. Należałoby zatem oczekiwać, że pojedyncze drzewo w lesie zmodyfikowanym będzie przeciętnie słabszym klasyfikatorem niż drzewo w zwykłym lesie losowym.

W celu weryfikacji tych dwóch hipotez przeprowadzono procedurę 500-krotnej klasyfikacji przy pomocy obu typów lasu na wybranych zbiorach syntetycznych (analogiczną do tej opisanej w poprzednim podrozdziale dla zbiorów rzeczywistych), rejestrując za każdym razem przeciętną korelację drzew w lesie (obliczaną jako średnia z korelacji między wynikami klasyfikacji każdej pary drzew w lesie na zbiorze testowym), przeciętny błąd klasyfikacji pojedynczego drzewa (ułamek błędnych klasyfikacji drzewa na zbiorze testowym, uśredniony po wszystkich drzewach) oraz błąd klasyfikacji lasu. Przyjęto $m = \lfloor \sqrt{M} \rfloor$ oraz liczbę drzew $K = 100$. Wyniki symulacji potwierdzają wcześniejsze przypuszczenia: bez względu na charakter badanego zbioru las zmodyfikowany wykazuje niższą przeciętną korelację drzew oraz wyższy błąd klasyfikacji pojedynczego drzewa. Przy tej wartości m efekt wyższego błędu indywidualnego przeważa pozytywny wpływ niższej korelacji, wskutek czego jakość klasyfikacji całego lasu jest słabsza niż w przypadku lasu zwykłego. Rysunki 2.3-2.4 ilustrują uzyskane rezultaty w postaci wykresów dystrybuant empirycznych badanych wielkości (przeciętnej korelacji drzew, przeciętnego indywidualnego błędu klasyfikacji oraz błędu klasyfikacji lasu). Istotność różnic w średnich tych wielkości przetestowano dodatkowo przy pomocy testu Wilcoxona, którego wyniki zaprezentowano w tabeli 2.2.

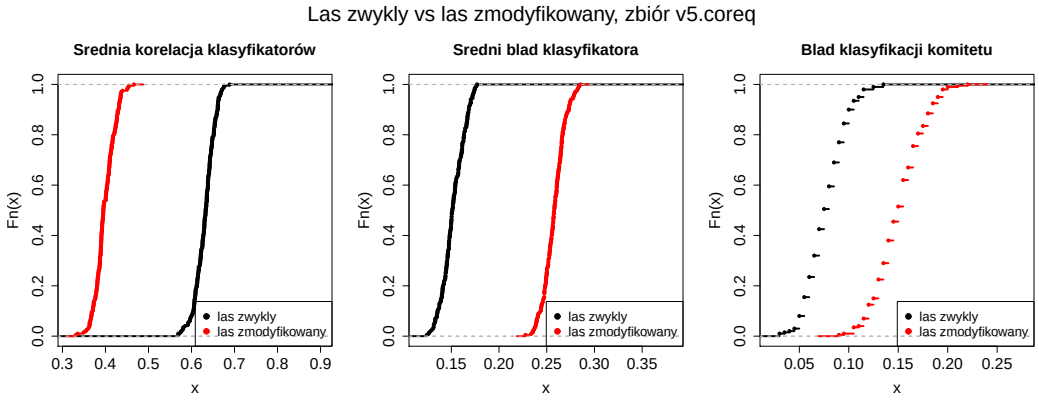
Zbiór	Wielkość	Średnia		Statystyka W	p-wartość
		Las zwykły	Las zmodyfikowany		
v20.dom	błąd klasyfikacji lasu	0,01	0,08	580,5	$< 2,2 \cdot 10^{-16}$
	korelacja drzew	0,59	0,05	40000	$< 2,2 \cdot 10^{-16}$
	błąd klasyfikacji drzewa	0,12	0,40	0	$< 2,2 \cdot 10^{-16}$
v5.coreq	błąd klasyfikacji lasu	0,08	0,15	330,5	$< 2,2 \cdot 10^{-16}$
	korelacja drzew	0,63	0,39	40000	$< 2,2 \cdot 10^{-16}$
	błąd klasyfikacji drzewa	0,15	0,26	0	$< 2,2 \cdot 10^{-16}$

Tabela 2.2: Wyniki testu Wilcoxona dla błędu klasyfikacji lasu, przeciętnej korelacji drzew i przeciętnego błędu klasyfikacji drzewa w lesie zwykłym i zmodyfikowanym dla dwóch zbiorów: `v20.dom` oraz `v5.coreq`.

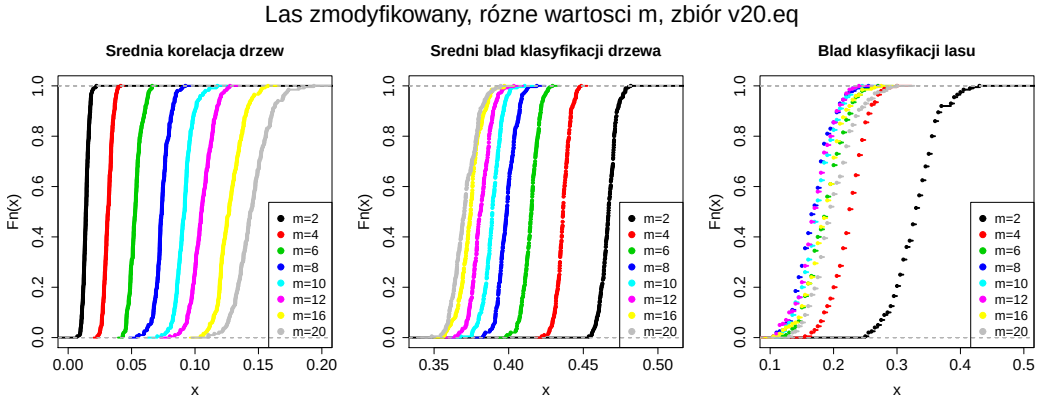
Trzeba jednak zaznaczyć, że w symulacjach użyto wartości m uznanej za optymalną dla zwykłego lasu losowego, jednak niekoniecznie najlepszej także dla lasu zmodyfikowanego. Parametr m wydaje się mieć kluczowy wpływ zarówno na korelację drzew w lesie, jak i na jakość klasyfikacji pojedynczego drzewa, dlatego konieczne jest dokładne zbadanie jego wpływu na zachowanie obu typów lasu losowego.



Rysunek 2.3: Wykresy dystrybuant empirycznych przeciętnej korelacji drzew, przeciętnego błędu klasyfikacji drzewa oraz błędu klasyfikacji lasu zwykłego oraz lasu zmodyfikowanego przy $m = 4$ (wartość domyślna) dla zbioru 20 nieskorelowanych zmiennych, z których jedna ma dominujący wpływ na klasyfikację (zbiór `v20.dom`).



Rysunek 2.4: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zwykłego oraz lasu zmodyfikowanego przy $m = 2$ (wartość domyślna) dla zbioru 5 skorelowanych zmiennych porównywalnie istotnych dla klasyfikacji (zbiór `v5.coreq`).



Rysunek 2.5: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego przy różnych wartościach parametru m dla zbioru 20 nieskorelowanych zmiennych porównywalnie istotnych dla klasyfikacji (`v20.eq`).

Wpływ podstawowych parametrów lasu na jakość klasyfikacji lasu zwykłego i zmodyfikowanego

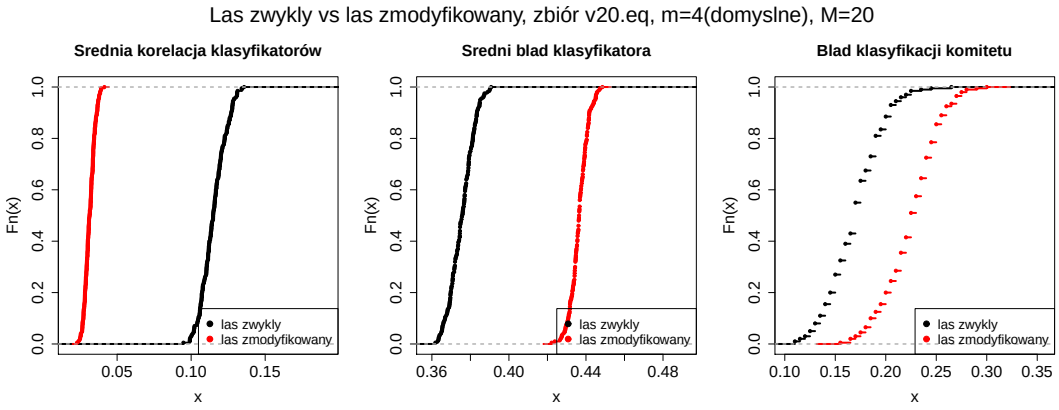
W celu sprawdzenia, jaka wartość m minimalizuje błąd klasyfikacji lasu zmodyfikowanego, przeprowadzono po 200 powtórzeń klasyfikacji przy pomocy tego lasu dla wybranych wartości m z przedziału $[1, M]$. Tę procedurę powtórzono na trzech syntetycznych zbiorach o zróżnicowanych cechach (dla dwóch z nich, `v20.eq` i `v20.dom`, $M = 20$, dla trzeciego,`v5.coreq`, $M = 5$). Dla porównania przeprowadzono analogiczne obliczenia dla zwykłego lasu losowego. Rysunek 2.5 prezentuje wykresy dystrybuant empirycznych korelacji drzew, indywidualnego błędu klasyfikacji oraz błędu klasyfikacji lasu zmodyfikowanego dla wybranych wartości parametru m, dla klasyfikacji przeprowadzonej na zbiorze o 20 nieskorelowanych zmiennych porównywalnie wpływających na przyporządkowanie do klasy (`v20.eq`). Zgodnie z intuicją, przeciętna korelacja drzew rośnie wraz ze wzrostem parametru m, natomiast przeciętny błąd indywidualny przeciwnie: maleje. Na błąd klasyfikacji lasu wpływają jednocześnie oba czynniki, co w analizowanym przypadku prowadzi do osiągnięcia przez niego minimum w okolicach $m = 8$. Jest to wartość dwukrotnie większa niż ta uznawana za najkorzystniejszą dla zwykłego lasu losowego ($\lfloor\sqrt{20}\rfloor = 4$).

Można teraz porównać własności klasyfikacji przy pomocy lasu zmodyfikowanego i zwykłego dla różnych wartości m (rys. 2.6-2.8). Przy $m = 4$ wyraźnie widać przewagę lasu zwykłego pod względem jakości klasyfikacji, ale już dla $m = 8$ oraz $m = 12$ lepszy okazuje się las zmodyfikowany, choć charakter relacji przeciętnych błędów indywidualnych oraz przeciętnych korelacji obu lasów pozostał bez zmian (las zmodyfikowany nadal wykazuje niższą korelację drzew, ale wyższy błąd indywidualny niż las zwykły). Dla $m = M = 20$ badane wielkości wykazują praktycznie jednakowe rozkłady dla obu lasów, co również jest zgodne z oczekiwaniami: w takim przypadku las zmodyfikowany niczym nie różni się od zwykłego, ponieważ w obu typach lasu w każdym węźle dowolnego drzewa wybór zmiennej do podziału dokonywany jest spośród wszystkich zmiennych zbioru danych.

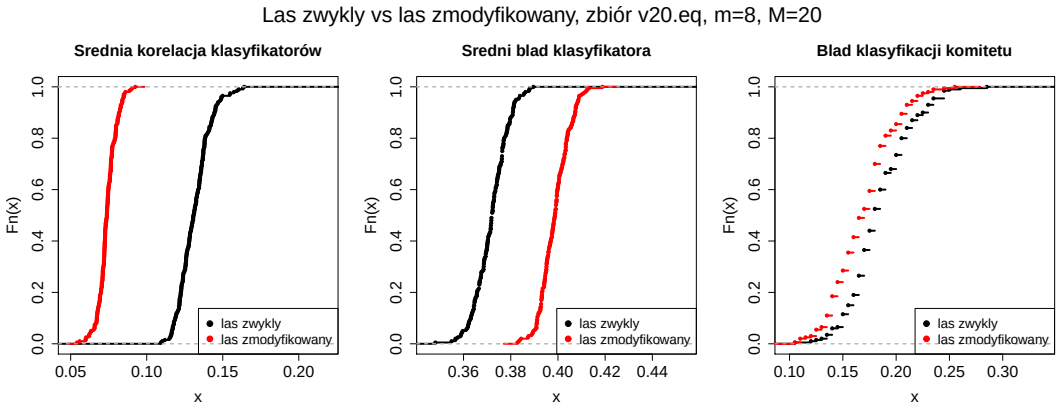
Analogiczne obliczenia przeprowadzono dla zbioru o 20 zmiennych nieskorelowanych, z których jedna ma dominujący wpływ na klasyfikację (`v20.dom`), a także dla zbioru 5 jednakowo istotnych skorelowanych zmiennych (`v5.coreq`). Rezultaty w postaci wykresów zależności uśrednionych wartości przeciętnej korelacji, przeciętnego błędu indywidualnego oraz błędu klasyfikacji lasu od wartości parametru m prezentują rysunki 2.9-2.11. Dla zbioru `v20.dom` różnice w błędzie klasyfikacji zwykłego lasu losowego dla różnych wartości m są minimalne. Podobnie jest dla lasu zmodyfikowanego, ale dopiero dla $m \geqslant 8$. W przypadku lasu zwykłego wybór m nie ma więc dużego znaczenia, podczas gdy dla lasu zmodyfikowanego zmiana m z 4 na 8 spowoduje ponad 5-krotny spadek błędu klasyfikacji. Ponadto dla $m > 8$ błędy klasyfikacji obu lasów nie różnią się istotnie (tabela 2.3).

m	Błąd średni		Statystyka W	p-wartość
	Las zwykły	Las zmodyfikowany		
4	0,013	0,080	581	$2,2 \cdot 10^{-16}$
8	0,012	0,015	16482	0,0018
12	0,012	0,012	19987	0,9908
16	0,012	0,012	19639	0,7495
20	0,014	0,012	22156	0,0567

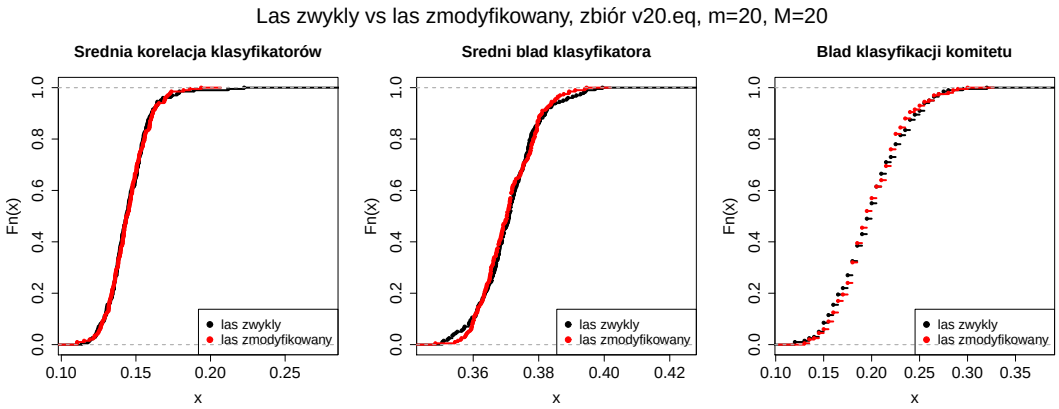
Tabela 2.3: Średnie błędy klasyfikacji lasu zwykłego i zmodyfikowanego oraz wyniki testu Wilcoxona dla różnych wartości parametru m (zbiór `v20.dom`).



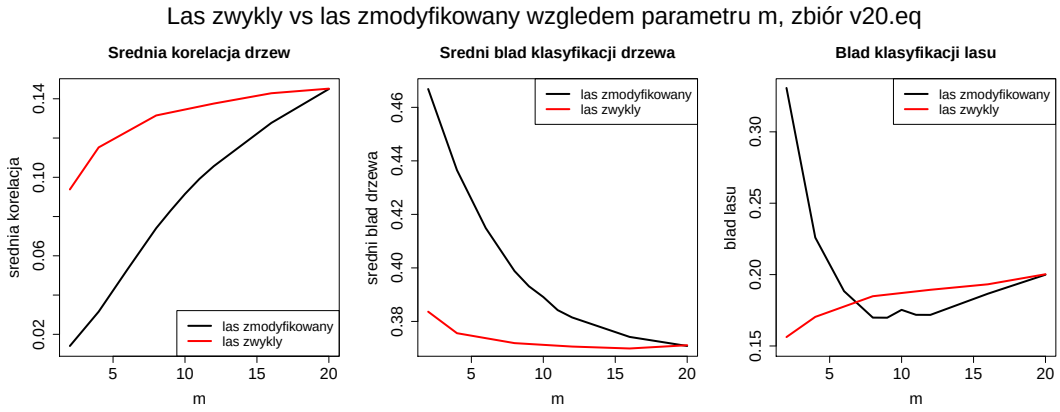
Rysunek 2.6: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego w porównaniu ze zwykłym lasem losowym, przy $m = 4$ dla zbioru `v20.eq`.



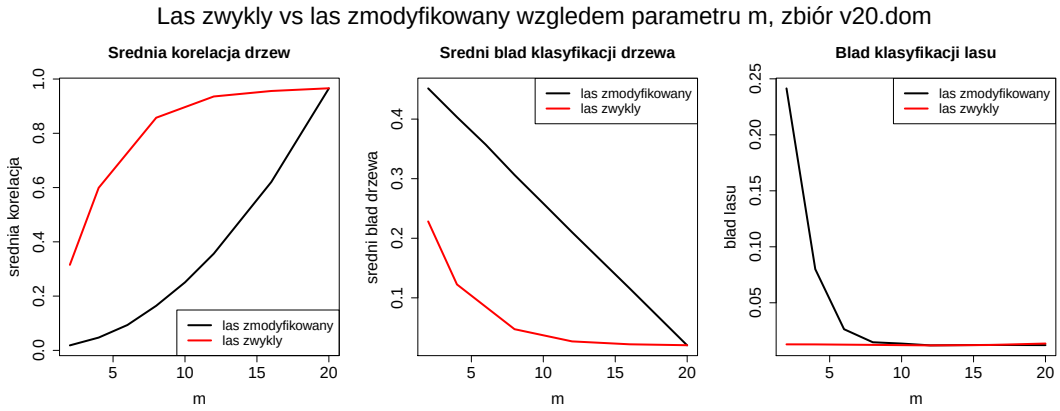
Rysunek 2.7: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego w porównaniu ze zwykłym lasem losowym, przy $m = 8$ dla zbioru `v20.eq`.



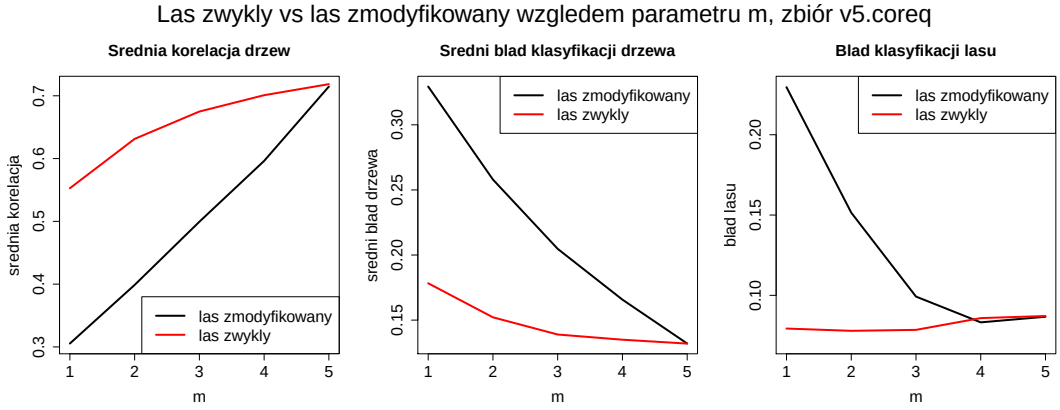
Rysunek 2.8: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego w porównaniu ze zwykłym lasem losowym, przy $m = 20$ dla zbioru `v20.eq`.



Rysunek 2.9: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego w zależności od m, dla zbioru 20 nieskorelowanych zmiennych porównywalnie istotnych dla klasyfikacji.



Rysunek 2.10: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego w zależności od m, dla zbioru `v20.dom`.



Rysunek 2.11: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego w zależności od m, dla zbioru v5.coreq.

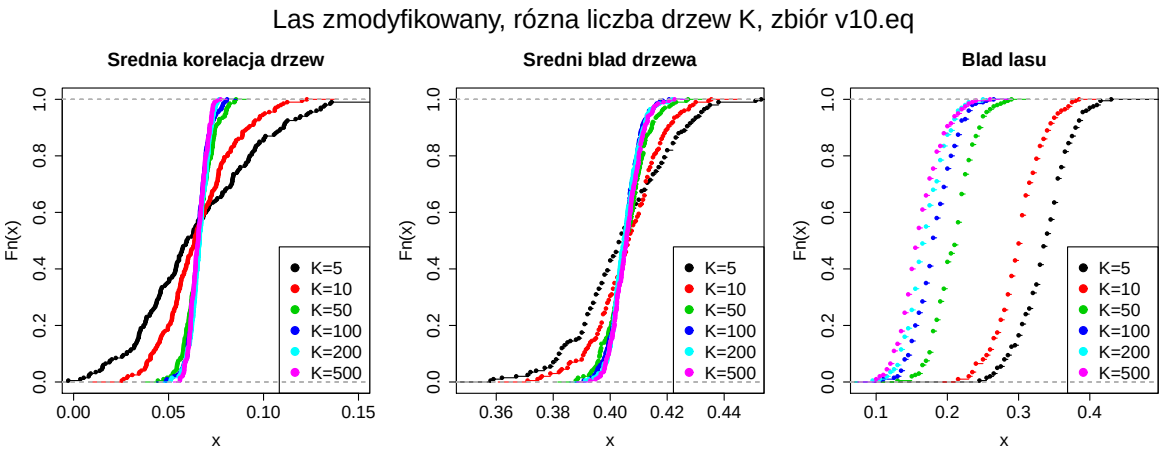
Również dla zbioru `v5.coreq` wyraźnie widać, że m uważane za optymalne dla zwykłego lasu losowego nie minimalizuje błędu klasyfikacji lasu zmodyfikowanego. W tym przypadku las zmodyfikowany osiąga najlepszą jakość klasyfikacji dla $m = 4$, a więc ponownie dla wartości dwukrotnie większej niż ta stosowana dla zwykłych lasów losowych ($\lfloor\sqrt{5}\rfloor = 2$). Wydaje się więc, że dla lasu losowego z grupą zmiennych losowanych jednorazowo dla każdego drzewa powinno się stosować wartość parametru m przynajmniej dwukrotnie większą od tej przyjmowanej dla zwykłego lasu losowego. Dokładne ustalenie przeciętnie najkorzystniejszej wartości m wymagałoby dalszych obliczeniochłonnych symulacji z użyciem kolejnych różnorodnych zbiorów danych.

Warto jednak zauważyć, że minimalny (względem m) błąd klasyfikacji lasu zmodyfikowanego jest dla wszystkich zbadanych zbiorów większy bądź równy minimalnemu błędowi zwykłego lasu losowego.

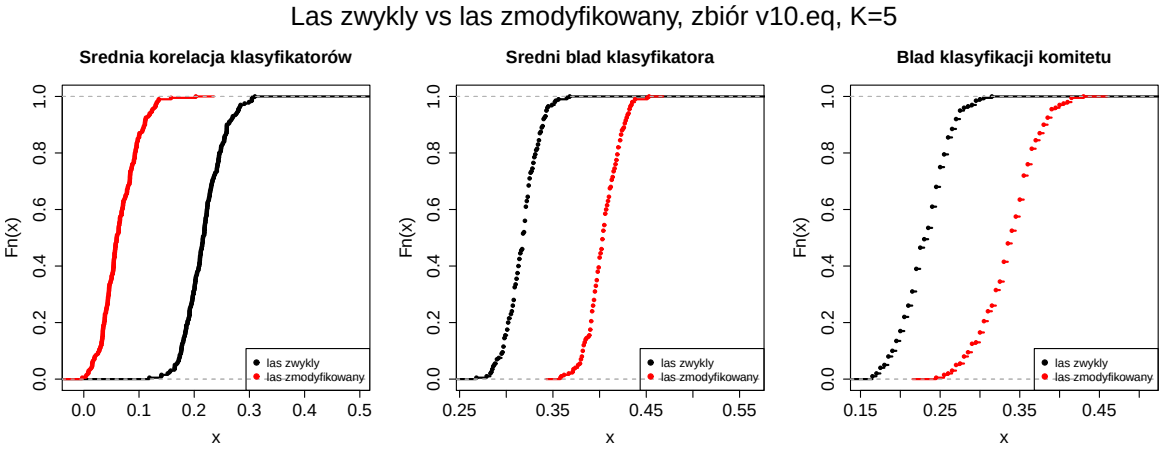
Drugim podstawowym parametrem lasu losowego jest liczba drzew. Jego wpływ na jakość klasyfikacji i jej składowe została przetestowana na syntetycznym zbiorze `v10.eq` o 10 nieskorelowanych zmiennych mających porównywalny wpływ na przyporządkowanie do klas (przyjęto m standardowo stosowane dla lasu zwykłego). Reakcja obu rodzajów lasu na wzrost liczby drzew jest bardzo podobna. Średnie wartości przeciętnej korelacji drzew oraz przeciętnego indywidualnego błędu klasyfikacji pozostają zbliżone, lecz maleje ich wariancja, podczas gdy błąd klasyfikacji całego lasu systematycznie spada (przypadek lasu zmodyfikowanego prezentuje rysunek 2.12). Porównując zachowania lasu zwykłego i zmodyfikowanego dla różnej liczby drzew można natomiast zauważyć, że różnica w jakości klasyfikacji pomiędzy lasami maleje wraz ze wzrostem wartości badanego parametru (rys. 2.13-2.14). Można przypuszczać, że błąd każdego z lasów zbiega do swojej granicy i dla liczby drzew równej 500 jest jej bardzo bliski (spadek błędu dla każdego z lasów przy zmianie liczebności komitetu z 200 do 500 jest już nieznaczny), stąd prawdopodobnie minimalny błąd dla lasu zmodyfikowanego jest mimo wszystko wyższy niż dla lasu zwykłego (przy m ustalonym na poziomie optymalnym dla lasu zwykłego).

Wpływ wybranych własności zbioru danych na jakość klasyfikacji lasu zwykłego i zmodyfikowanego

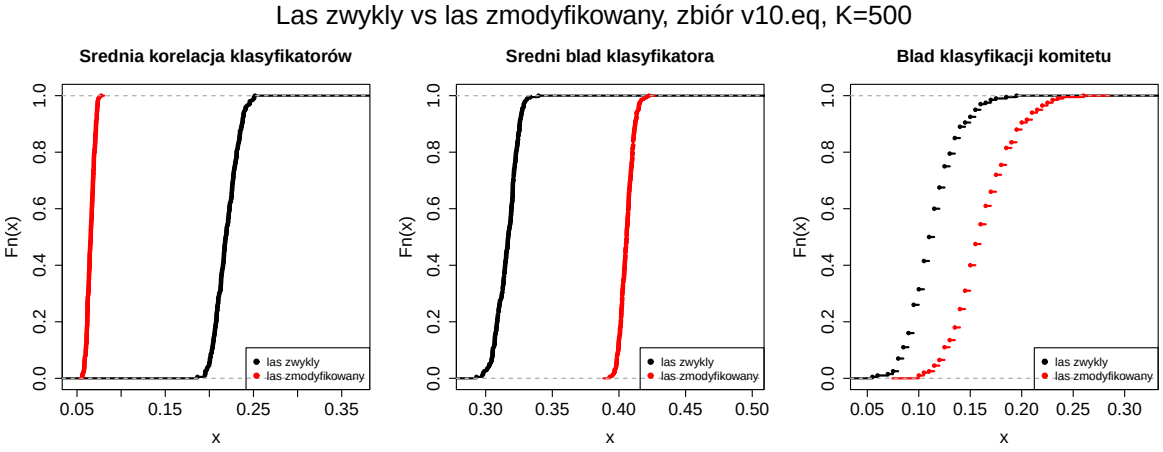
Przy badaniu oddziaływania własności zbioru danych na jakość klasyfikacji postanowiono skupić się na dwóch cechach, które wydają się wywierać znaczący wpływ na składowe błędu klasyfikacji lasu, a mianowicie na korelacji zmiennych oraz na ich istotności przy przyporządkowywaniu obserwacji do klas. Korelacja zmiennych w zbiorze danych może powodować wzrost korelacji drzew w lesie, ponieważ w takiej sytuacji drzewa powinny dawać podobne wyniki bez względu na wybrane zmienne. Jednocześnie prawdopodobny jest spadek przeciętnego błędu drzewa (w porównaniu ze zbiorem zmiennych nieskorelowanych): nawet jeśli tylko niektóre zmienne mają rzeczywisty wpływ na przyporządkowanie do klas, to pozostałe mogą stanowić podstawę do ich oszacowania i służyć porównywalnie dobrze jako wielkości różnicujące zbiór danych. Sama istotność zmiennych dla klasyfikacji również może wpływać na indywidualny błąd drzewa. Jeżeli jedna ze zmiennych ma dominujący wpływ na przyporządkowanie do klas, to wylosowanie jej w którymkolwiek węźle znacznie zwiększa szanse na prawidłową klasyfikację. W związku z tym można się spodziewać, że drzewa w takim lesie będą miały przeciętnie mniejszy błąd klasyfikacji, ale jednocześnie będą ze sobą bardziej skorelowane niż w przypadku lasu wytrenowanego na zbiorze o wszystkich zmiennych umiarkowanie istotnych dla klasyfikacji.



Rysunek 2.12: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego dla różnej liczby drzew w lesie, dla zbioru 10 nieskorelowanych zmiennych porównywalnie istotnych dla klasyfikacji (`v10.eq`).



Rysunek 2.13: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego przy liczbie drzew równej 5, dla zbioru `v10.eq`.



Rysunek 2.14: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego przy liczbie drzew równej 500, dla zbioru `v10.eq`.

Symulacje służące przetestowaniu tych hipotez oraz zbadaniu wzajemnej relacji lasu zwykłego i zmodyfikowanego przeprowadzono na czterech zbiorach danych: zbiorze o zmiennych nieskorelowanych, porównywalnie umiarkowanie istotnych (`v4.eq`), zbiorze o zmiennych nieskorelowanych, z których jedna ma dominujący wpływ na przyporządkowanie do klasy (`v4.dom`) oraz dwóch analogicznych zbiorach zmiennych skorelowanych (`v4.coreq` i `v4.cordom`). Rezultaty potwierdzają wcześniejsze przypuszczenia: dominacja jednej zmiennej oraz korelacja zmiennych prowadzi do wyższej korelacji drzew. W przypadku zbiorów zmiennych nieskorelowanych błędy indywidualne także zachowują się zgodnie z przewidywaniami (rys. 2.15-2.16), jednak ich porównywanie jest nieco bardziej ryzykowne, ponieważ zależą one również od separowalności klas, która może się różnić między zbiorami. Można jednocześnie zaobserwować, że rozmiary różnicy między jakością klasyfikacji lasu zwykłego i zmodyfikowanego zależą od badanych własności zbioru. Dominacja jednej ze zmiennych sprawia, że różnica między lasami nie jest tak znacząca jak przy klasyfikacji na zbiorze o wszystkich zmiennych jednakowo istotnych (por. rys. 2.17 z 2.18 oraz rys. 2.19 z 2.20). Korelacja zmiennych również skutkuje zbliżeniem wyników dla lasu zwykłego i zmodyfikowanego, choć dla zbiorów z jedną zmienną dominującą jest to słabo widoczne (por. rys. 2.17 z 2.19 oraz rys. 2.18 z 2.20).

2.2. Ważone lasy losowe

Kolejnym krokiem w analizie modyfikacji lasu losowego jest zbadanie skutków ważenia głosów komitetu. Ważenie powinno służyć wzrostowi znaczenia dobrych klasyfikatorów w ostatecznym werdykcie lasu, dlatego zdecydowano się wprowadzić wagi zależne od błędu klasyfikacji popełnianego przez poszczególne drzewa. Przyjęto ich następującą postać:

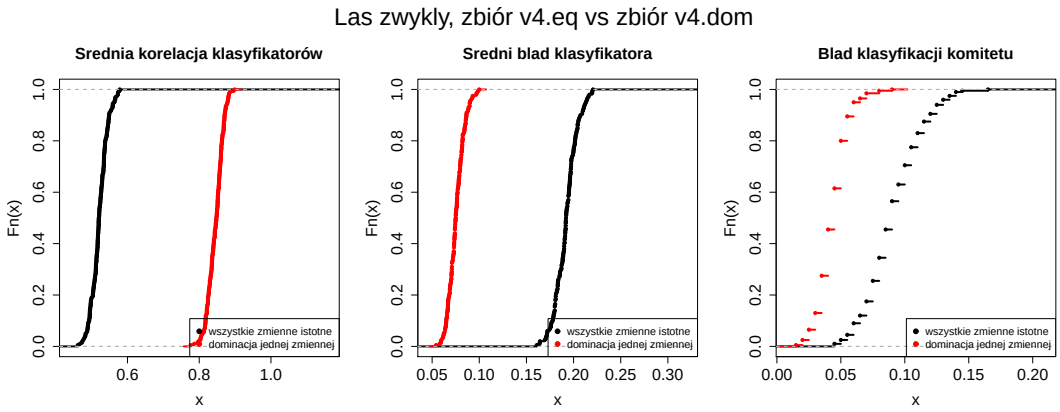
$$w_k^{boost} = \ln\left(\frac{1 - err_k}{err_k}\right), \quad k = 1, \ldots, K, \tag{2.1}$$

gdzie err_i – błąd klasyfikacji popełniany przez i-te drzewo, K –liczba drzew w lesie. Jest ona analogiczna do postaci wag γ_k stosowanej w boostingu do ważenia klasyfikatorów. Do szacowania błędów klasyfikacji można używać zbioru obserwacji OOB, o których była mowa w podrozdziale 1.2, dzięki temu obliczanie wag może być wbudowane w proces tworzenia lasu i nie prowadzi do zmniejszania rozmiarów zbioru treningowego. Algorytm budowania lasu analizowanego w tym podrozdziale ma więc postać:

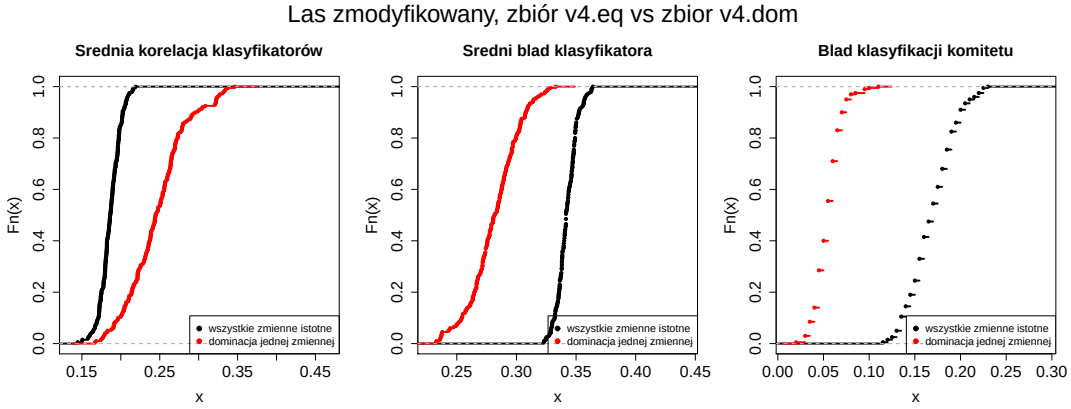
Algorytm 5. Ważony las losowy z grupą zmiennych losowanych jednorazowo dla każdego drzewa

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$, K – liczba drzew,
M – liczba zmiennych w próbie Z, x – obserwacja wymagająca klasyfikacji.

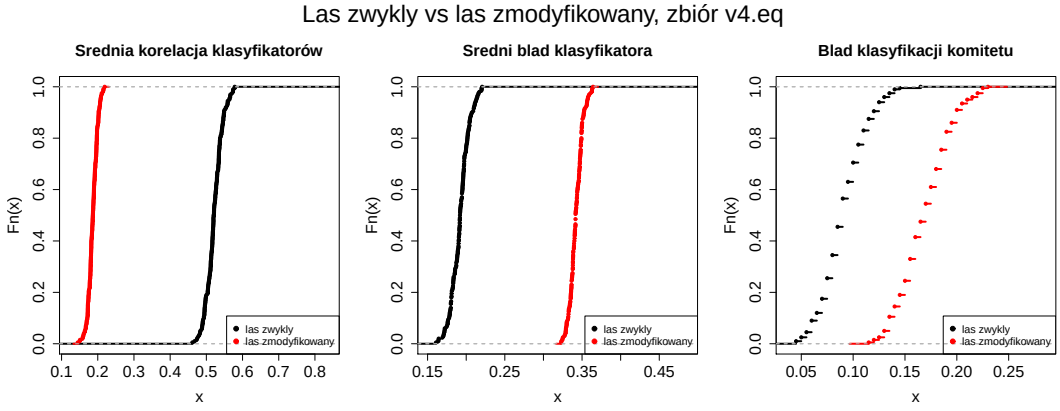
1. Dla każdego $k = 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m << M$ zmiennych bez zwracania. Z wartości m wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^k.
   (c) Wytrenuj drzewo decyzyjne T^k na pseudopróbie Z^k (bez przycinania).
   (d) Dokonaj predykcji klasy dla obserwacji OOB $G^k(x_i^{oob})$, $i = 1, \ldots, N^{oob}$.
   (e) Oblicz błąd $err_k = \frac{1}{N^{oob}} \sum_{i=1}^{N^{oob}} I(G^k(x_i^{oob}) \neq y_i^{oob})$ oraz wagę $w_k = \log\left((1 - err_k)/err_k\right)$.
2. Dokonaj predykcji klasy dla x przy pomocy wszystkich drzew $T^k(x)$, $k = 1, \ldots, K$.
3. Oblicz średnią $T_{mrfw}(x) = \frac{1}{\sum_{k=1}^{K} w_k} \sum_{k=1}^{K} w_k T^k(x)$.
4. Podaj $G_{mrfw}(x) = \arg\max_{c \in \{1,\ldots,C\}} T_{mrfw}(x)$.



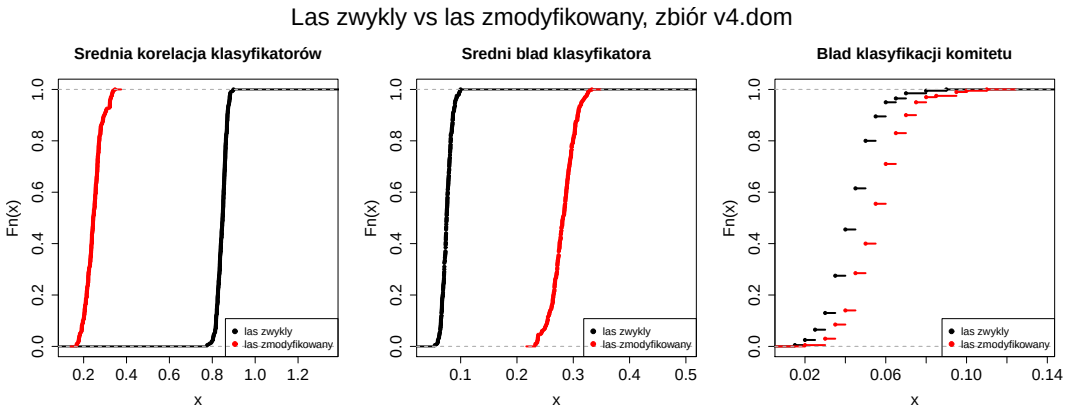
Rysunek 2.15: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji zwykłego lasu losowego przy $m = \lfloor \sqrt{M} \rfloor$, porównanie wyników dla zbiorów `v4.eq` i `v4.dom`.



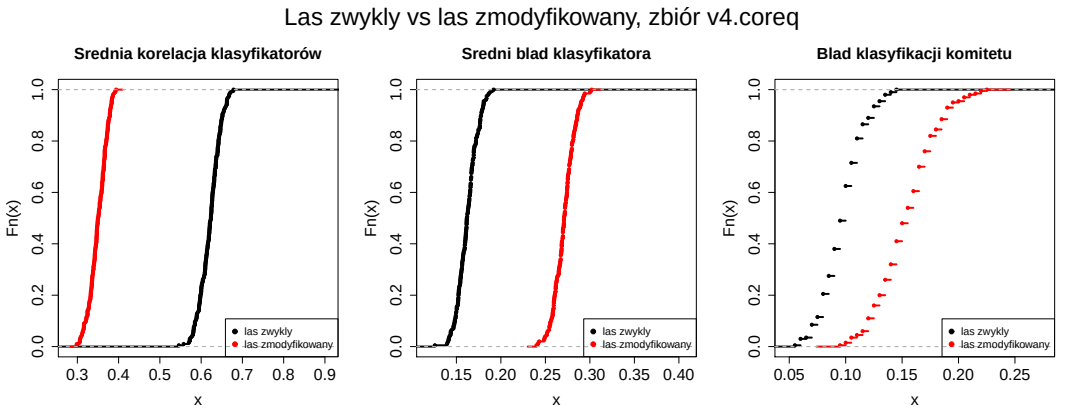
Rysunek 2.16: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji zmodyfikowanego lasu losowego przy $m = \lfloor \sqrt{M} \rfloor$, porównanie wyników dla zbioru `v4.eq` i `v4.dom`.



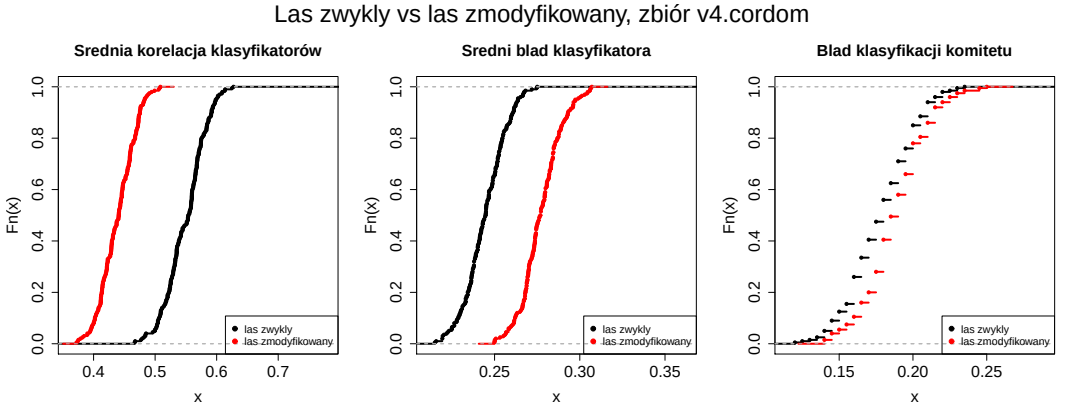
Rysunek 2.17: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego przy $m = \lfloor \sqrt{M} \rfloor$ dla zbioru `v4.eq`.



Rysunek 2.18: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego przy $m = \lfloor \sqrt{M} \rfloor$ dla zbioru `v4.dom`.



Rysunek 2.19: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego przy $m = \lfloor \sqrt{M} \rfloor$ dla zbioru `v4.coreq`.



Rysunek 2.20: Przeciętna korelacja drzew, przeciętny błąd klasyfikacji drzewa oraz błąd klasyfikacji lasu zmodyfikowanego i zwykłego lasu losowego przy $m = \lfloor \sqrt{M} \rfloor$ dla zbioru `v4.cordom`.

Lasem podstawowym w tym algorytmie jest więc nadal las z grupą zmiennych losowanych jednorazowo dla każdego drzewa. Podobnie jak w przypadku lasów nieważonych, działanie lasów ważonych przeanalizowano na pięciu zbiorach danych, dla każdego z nich powtarzając proces uczenia i klasyfikacji 500 razy. Ponownie zastosowano $T = 50$ oraz $m = \lfloor\sqrt{M}\rfloor$, a więc wartość przeciętnie najkorzystniejszą dla zwykłego lasu losowego. Wyniki porównano w pierwszej kolejności z próbami błędu klasyfikacji uzyskanymi dla zwykłego lasu losowego. Istotność różnicy w średnich błędach zweryfikowano przy pomocy testu Wilcoxona, którego wyniki prezentuje tabela 2.4.

Zbiór	Błąd średni		Statystyka W	p-wartość
	Las ważony	Las zwykły		
iris	0,05	0,05	131155	0,1667
crabs	0,11	0,09	145540	$6,1 \cdot 10^{-6}$
pima	0,26	0,23	177299	$< 2,2 \cdot 10^{-16}$
ozone	0,07	0,06	147132	$1,2 \cdot 10^{-6}$
ctg	0,09	0,06	244972	$< 2,2 \cdot 10^{-16}$

Tabela 2.4: Średnie błędy klasyfikacji dla lasu zmodyfikowanego ważonego i zwykłego lasu losowego oraz wyniki testu Wilcoxona.

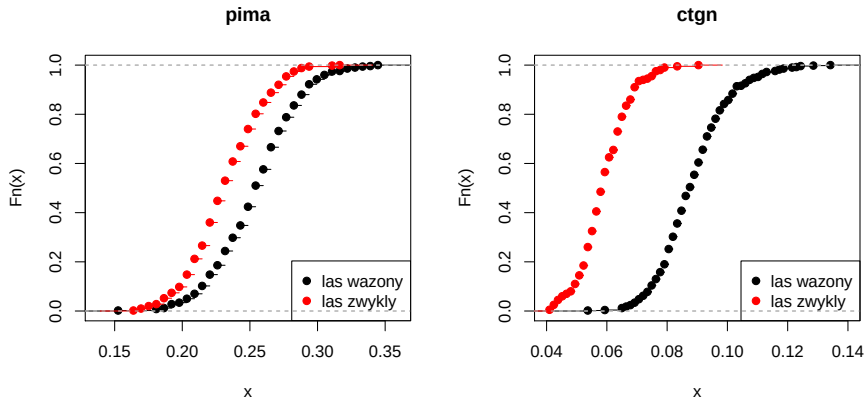
Ponownie więc różnica między średnimi błędami lasu ważonego i zwykłego jest znacząca dla wszystkich zbiorów poza zbiorem `iris`, nadal też przewagę ma zwykły las losowy. Potwierdzają to także wykresy na rysunku 2.21. Choć według testu Wilcoxona różnica między błędem klasyfikacji lasu zwykłego i lasu ważonego dla zbioru `iris` nie jest istotna, to jednak rysunek 2.22 pokazuje, że błąd zwykłego lasu jest systematycznie mniejszy bądź równy błędowi lasu ważonego, a zatem także w tym przypadku zwykły las wykazuje pewną przewagę. Źródłem takiej relacji pomiędzy badanymi lasami jest prawdopodobnie znów wybór m korzystnego dla zwykłego lasu losowego, a nie dla lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa.

Porównano także jakość klasyfikacji zmodyfikowanego lasu ważonego ze zmodyfikowanym lasem nieważonym. Wyniki testu Wilcoxona wykazują wyraźny spadek błędu klasyfikacji po wprowadzeniu wag w trzech z pięciu przypadków (tabela 2.5).

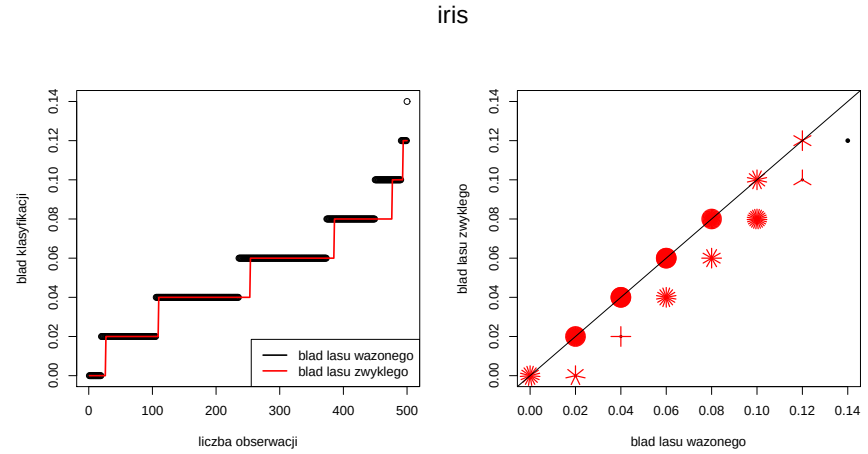
Zbiór	Błąd średni		Statystyka W	p-wartość
	Las ważony	Las nieważony		
iris	0,053	0,056	118630	0,1534
crabs	0,106	0,209	20821	$< 2,2 \cdot 10^{-16}$
pima	0,256	0,273	88862	$2,3 \cdot 10^{-15}$
ozone	0,066	0,066	123519	0,7451
ctg	0,089	0,094	91556	$2,3 \cdot 10^{-13}$

Tabela 2.5: Średnie błędy klasyfikacji dla lasu zmodyfikowanego ważonego i nieważonego oraz wyniki testu Wilcoxona.

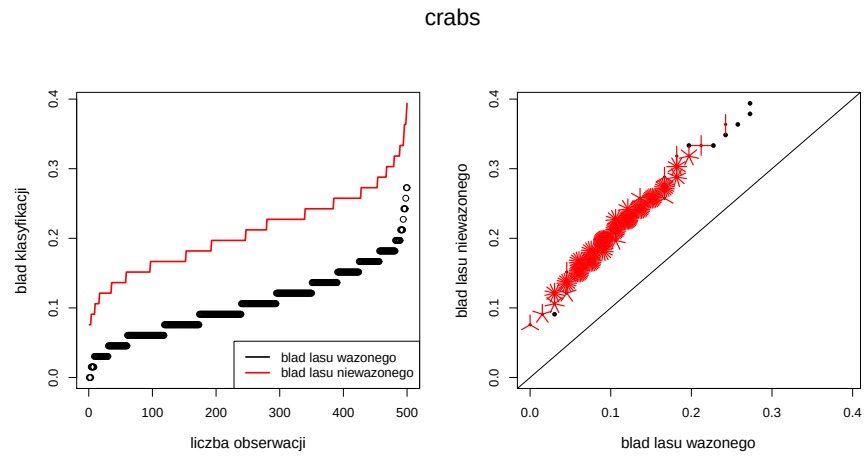
Ten rezultat częściowo potwierdzają także wykresy kwantylowe widoczne na przykładowych rysunkach 2.23 i 2.24. W przypadku zbioru `crabs` las ważony osiąga zdecydowanie mniejszy błąd klasyfikacji, natomiast dla zbioru `ozone` nie ma widocznej różnicy w jakości klasyfikacji. Rysunek 2.25 pokazuje z kolei, że dla zbioru `iris` błąd klasyfikacji lasu nieważonego jest systematycznie większy lub równy od błędu lasu ważonego, a więc także w tym przypadku dostrzegalna jest pewna poprawa jakości klasyfikacji, choć różnica błędów średnich nie jest istotna.



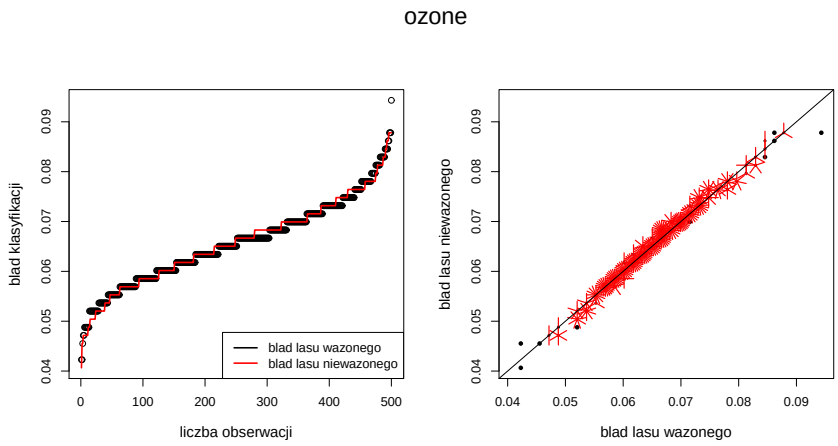
Rysunek 2.21: Dystrybuanty błędu klasyfikacji ważonego lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa oraz zwykłego lasu losowego na zbiorach `pima` oraz `ctgn`.



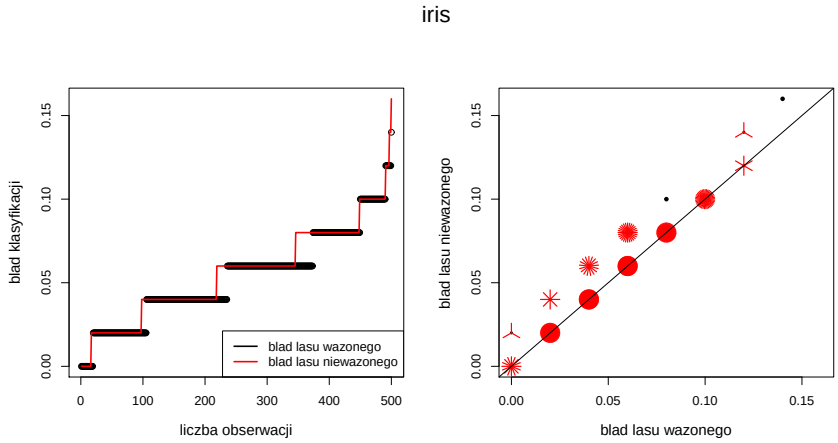
Rysunek 2.22: Porównanie rozkładów błędu klasyfikacji na zbiorze iris dla ważonego lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa oraz zwykłego lasu losowego (błąd lasu zwykłego zaznaczono kolorem czerwonym).



Rysunek 2.23: Porównanie rozkładów błędu klasyfikacji na zbiorze `crabs` dla ważonego oraz nieważonego lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa (błąd lasu nieważonego zaznaczono kolorem czerwonym).



Rysunek 2.24: Porównanie rozkładów błędu klasyfikacji na zbiorze `ozone` dla ważonego oraz nieważonego lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa (błąd lasu nieważonego zaznaczono kolorem czerwonym).



Rysunek 2.25: Porównanie rozkładów błędu klasyfikacji na zbiorze `iris` dla ważonego oraz nieważonego lasu z grupą zmiennych losowanych jednorazowo dla każdego drzewa (błąd lasu nieważonego zaznaczono kolorem czerwonym).

2.3. Przycinane lasy losowe

Następnym etapem badania własności lasów losowych jest sprawdzenie, w jaki sposób na ich jakość klasyfikacji wpływa przycinanie drzew w lesie. Przycinanie polega na zadaniu pewnych ograniczeń na minimalną liczbę obserwacji w węźle lub na minimalny zysk z podziału węzła, które prowadzą do wcześniejszego zatrzymania procesu budowy drzewa, a tym samym zmniejszenia jego rozmiarów. W oryginalnych lasach losowych drzewa są natomiast budowane aż do momentu, w którym nie jest możliwy dalszy podział żadnego z węzłów, a więc uzyskano węzły zawierające obserwacje tylko z jednej klasy lub węzły zawierające tylko jedną obserwację. W przypadku pojedynczych drzew taki sposób ich budowy prowadzi do nadmiernego dopasowania do danych treningowych i w rezultacie pogorszenia ogólnej jakości klasyfikacji. Dlatego warto się przekonać, czy lasy złożone z drzew przycinanych nie osiągnęłyby wyników lepszych niż zwykłe lasy losowe. Algorytm budowy lasu analizowanego w niniejszym podrozdziale przyjmie więc postać:

Algorytm 6. Przycinany las losowy z grupą zmiennych losowanych jednorazowo dla każdego drzewa

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$, K – liczba drzew,
M – liczba zmiennych w próbie Z, x – obserwacja wymagająca klasyfikacji.

1. Dla każdego $k = 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m << M$ zmiennych bez zwracania. Z wartości m wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^k.
   (c) Wytrenuj drzewo decyzyjne T^k na pseudopróbie Z^k, budując je do momentu osiągnięcia narzuconych ograniczeń.
2. Dokonaj predykcji klasy dla x przy pomocy wszystkich drzew $T^k(x),\ k = 1, \ldots, K$.
3. Oblicz średnią $T_{mrf}(x) = \frac{1}{K} \sum_{k=1}^{K} T^k(x)$.
4. Podaj $G_{mrf}(x) = \arg\max_{c \in \{1,\ldots,C\}} T_{mrf}(x)$.

Las przycinany został też przetestowany w wersji ważonej. Tak jak w poprzednich podrozdziałach, proces uczenia i klasyfikacji przy pomocy tak zmodyfikowanego lasu został powtórzony 500 razy dla każdego zbioru danych, a wyniki zostały porównane z uzyskanymi wcześniej próbami błędu dla zwykłego lasu losowego. Ponownie okazało się, że oryginalny las losowy daje zdecydowanie mniejszy błąd klasyfikacji, co prezentuje tabela 2.6.

Zbiór	Błąd średni		Statystyka W	p-wartość
	Las przycinany	Las zwykły		
iris	0,054	0,050	134285	0,0367
crabs	0,197	0,093	145540	$< 2,2 \cdot 10^{-16}$
pima	0,244	0,234	147443	$8,5 \cdot 10^{-7}$
ozone	0,067	0,063	159947	$1,7 \cdot 10^{-14}$
ctg	0,122	0,061	249976	$< 2,2 \cdot 10^{-16}$

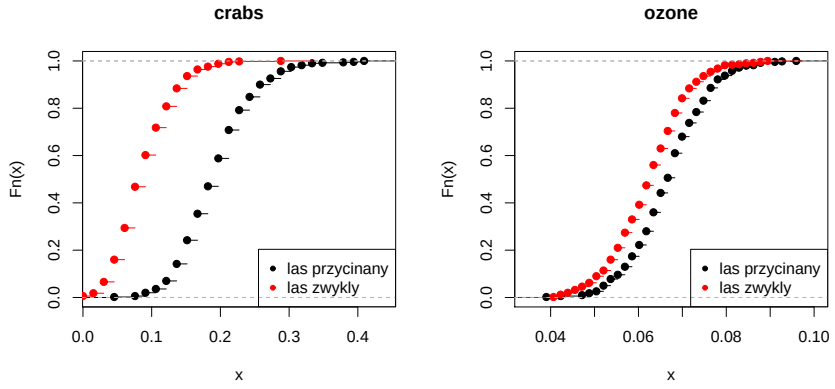
Tabela 2.6: Średnie błędy klasyfikacji dla lasu zmodyfikowanego ważonego i zwykłego lasu losowego oraz wyniki testu Wilcoxona.

Potwierdzają to również przykładowe wykresy widoczne na rysunku 2.26. W obu przypadkach błąd klasyfikacji zwykłego lasu losowego jest systematycznie mniejszy od błędu lasu przycinanego, zwłaszcza dla zbioru `crabs` różnica jest bardzo wyraźna.

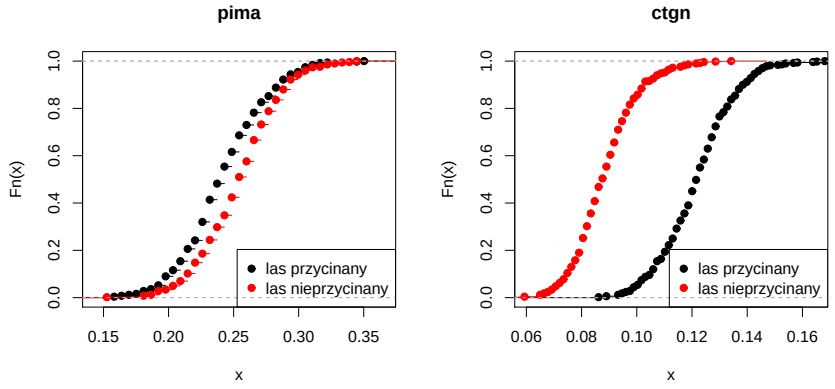
Warto też porównać zachowanie ważonego lasu przycinanego z ważonym lasem nieprzycinanym z grupą zmiennych losowanych jednorazowo dla każdego drzewa (rozważanego w podrozdziale 2.2). Wyniki testu Wilcoxona (tabela 2.7) również w tym wariancie wskazują na przewagę lasów nieprzycinanych, wyjątkiem jest jedynie zbiór `pima`, dla którego niższy średni błąd osiągnął ważony las przycinany. Podobne wnioski można wyciągnąć na podstawie wykresów zaprezentowanych na rysunku 2.27.

Zbiór	Błąd średni		Statystyka W	p-wartość
	Las przycinany	Las nieprzycinany		
iris	0,054	0,053	127862	0,5204
crabs	0,197	0,106	225465	$< 2,2 \cdot 10^{-16}$
pima	0,244	0,256	96062	$2,2 \cdot 10^{-10}$
ozone	0,067	0,066	111948	0,0042
ctg	0,122	0,089	241997	$< 2,2 \cdot 10^{-16}$

Tabela 2.7: Średnie błędy klasyfikacji dla ważonego lasu przycinanego i ważonego lasu nieprzycinanego z grupą zmiennych losowanych jednorazowo dla każdego drzewa oraz wyniki testu Wilcoxona.



Rysunek 2.26: Dystrybuanty błędu klasyfikacji ważonego lasu przycinanego z grupą zmiennych losowanych jednorazowo dla każdego drzewa oraz zwykłego lasu losowego na zbiorach `crabs` oraz `ozone`.



Rysunek 2.27: Dystrybuanty błędu klasyfikacji ważonego lasu przycinanego oraz ważonego lasu nieprzycinanego z grupą zmiennych losowanych jednorazowo dla każdego drzewa na zbiorach `pima` oraz `ctgn`.

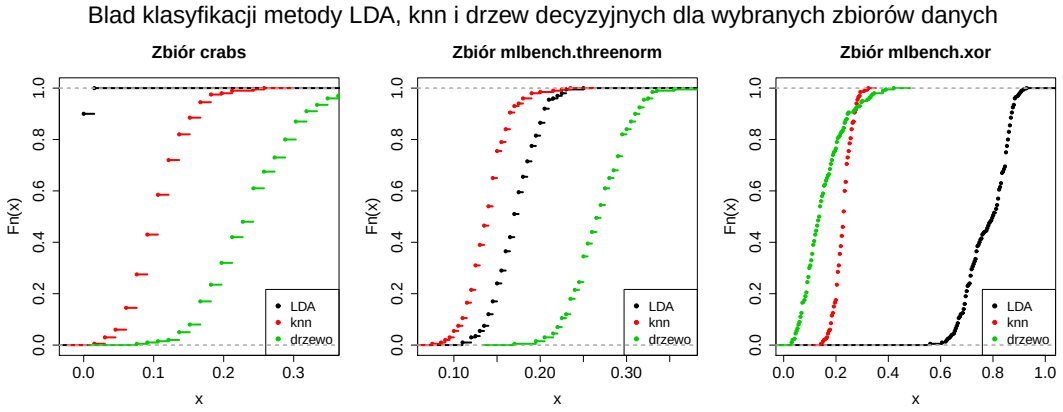
2.4. Podsumowanie

Eksperymenty z lasami losowymi wykazały, że ich cechy takie jak niezależne losowanie zmiennych dla każdego węzła w drzewie oraz nieprzycinanie drzew istotnie podnoszą jakość klasyfikacji komitetu. Włączając drzewa w skład komitetu niejednorodnego, nie należy więc z nich rezygnować, co więcej, można spróbować w analogiczny sposób potraktować klasyfikatory innych typów. Redukując zależność klasyfikatorów przez losowanie zmiennych do zbioru treningowego, warto zwrócić uwagę na liczbę losowanych zmiennych m. Wyniki symulacji z niniejszego rozdziału pokazują, że ten parametr bardzo silnie wpływa na błąd klasyfikacji komitetu, zwłaszcza w przypadku losowania jednej grupy zmiennych dla całego klasyfikatora. Do obniżenia błędu komitetu można też wykorzystać ważenie głosów klasyfikatorów. Przykład lasu losowego z jedną grupą zmiennych dla całego drzewa z podrozdziału 2.2 przekonuje, że zastosowanie wag może skutecznie poprawić jakość klasyfikacji komitetu. Budowę komitetu niejednorodnego z wykorzystaniem powyższych wniosków opisuje kolejny rozdział pracy.

Rozdział 3

Niejednorodne komitety klasyfikatorów

Przystępując do konstrukcji niejednorodnego komitetu klasyfikatorów, należało zdecydować się na jedną ze struktur: sekwencyjną (zastosowaną w boostingu) lub równoległą (reprezentowaną przez bagging i lasy losowe). Oba podejścia dają szansę uzyskania jednakowo dobrej jakości klasyfikacji (boosting i lasy losowe osiągają porównywalnie niski błąd klasyfikacji), ale sekwencyjna budowa komitetu wymagałaby poza decyzją o proporcji poszczególnych typów klasyfikatorów również znalezienia ich optymalnej kolejności w ciągu. Dlatego wybrano strukturę równoległą, analogiczną do tej stosowanej w baggingu (klasyfikatory będą więc trenowane na niezależnych próbach bootstrapowych, a kolejność ich włączania do komitetu nie będzie mieć znaczenia). Ponadto, na wzór lasów losowych, zdecydowano się zmniejszyć zależność klasyfikatorów poprzez losowanie zmiennych do zbioru treningowego każdego klasyfikatora. Z boostingu zaczerpnięto natomiast pomysł ważenia głosów klasyfikatorów w zależności od popełnianego przez nie błędu klasyfikacji. Szczegóły budowy komitetu, takie jak dobór metod klasyfikacji oraz ich proporcji w komitecie, ustalenie liczby zmiennych losowanych do zbioru treningowego oraz wybór postaci wag przedyskutowano w kolejnych podrozdziałach.



Rysunek 3.1: Porównanie rozkładów błędu klasyfikacji metody LDA, knn oraz drzewa decyzyjnego na wybranych zbiorach danych).

3.1. Wybór typów klasyfikatorów

Poszczególne typy klasyfikatorów produkują obszary decyzyjne o różnych własnościach geometrycznych, dlatego na ich skuteczność wpływa geometryczny rozkład klas w zbiorze danych. Komitet niejednorodny ma stanowić uniwersalną metodę klasyfikacji, która dostosowuje się do własności zbioru i nie wymaga od użytkownika wcześniejszej decyzji o wyborze typu klasyfikatora. Dlatego istotne jest, aby w komitecie znalazły się klasyfikatory o możliwie różnym charakterze obszarów decyzyjnych, a jednocześnie stosunkowo szybkie pod względem obliczeniowym (w komitecie będzie ich bowiem co najmniej kilkadziesiąt). Do bazowego komitetu jednorodnego, za jaki obrano las losowy, zdecydowano się więc dołączyć grupę klasyfikatorów liniowych (LDA) oraz klasyfikatorów knn. Klasyfikatory LDA powinny być skuteczniejsze od drzew w przypadku zbiorów o klasach dobrze separowalnych przez hiperpłaszczyzny, natomiast klasyfikatory knn mogą osiągać lepsze wyniki w przypadku klas separowalnych krzywoliniowo (są jednocześnie mniej czasochłonne od metody SVM).

Hipotezy na temat wybranych typów klasyfikatorów przetestowano na syntetycznych i rzeczywistych zbiorach danych różniących się liczbą i geometrycznym rozkładem klas. Rezultaty przedstawia tabela 3.1 oraz rysunek 3.1. Dla każdej z badanych metod istnieje zbiór danych, na którym jest ona lepsza od pozostałych, zatem komitet stworzony z wybranych typów klasyfikatorów powinien osiągać wyniki zbliżone do najlepszych na wszystkich badanych zbiorach. Jakość klasyfikacji komitetu zależy jednak nie tylko od błędu popełnianego przez pojedyncze klasyfikatory, ale także od korelacji między nimi, dlatego istotne jest również ustalenie wartości parametrów wpływających na stopień zależności między składowymi komitetu. Dopiero potem można będzie ocenić, czy skład komitetu jest właściwy.

Zbiór	Błąd średni		
	LDA	knn	drzewo
v20.eq	**0,079**	0,167	0,365
v20.dom	0,050	0,222	**0,016**
v5.coreq	**0,034**	0,074	0,148
mlbench.threenorm	0,172	**0,139**	0,269
mlbench.simplex	**0,014**	0,018	0,070
mlbench.xor	0,782	0,229	**0,151**
mlbench.smiley	0,250	**0,004**	0,006
iris	**0,029**	0,065	0,061
crabs	**0,002**	0,109	0,239
pima	**0,224**	0,246	0,269
ozone	0,318	**0,239**	0,278
ctg	0,126	0,104	**0,089**

Tabela 3.1: Średnie błędy klasyfikacji dla klasyfikatorów LDA, knn oraz drzew decyzyjnych. Pogrubioną czcionką oznaczono błąd najmniejszy dla każdego zbioru danych.

Wstępne ustalenie składu komitetu pozwala sformułować ogólny algorytm jego budowy i predykcji przy jego pomocy. Algorytm ten można będzie stosować w eksperymentach prowadzących do ustalenia wartości parametrów i innych szczegółów komitetu. Poszukiwane parametry to liczba zmiennych losowana dla poszczególnych typów klasyfikatorów (m_{tr}, m_{lda}, m_{knn}) oraz liczba sąsiadów k dla klasyfikatora knn. Proporcje klasyfikatorów w komitecie ustalono wstępnie na równym poziomie, a więc $K^{tr} = K^{lda} = K^{knn} = \frac{K}{3}$.

Algorytm 7. Komitet niejednorodny drzew, LDA i knn

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$, K^{tr} – liczba drzew,
K^{lda} – liczba klasyfikatorów LDA, K – liczba wszystkich klasyfikatorów,
M – liczba zmiennych w próbie Z, x – obserwacja wymagająca klasyfikacji.

1. Dla każdego $j = 1, \ldots, K^{tr}$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem, tworząc pseudopróbę Z^j.
   (b) Wytrenuj drzewo decyzyjne f^j_{tr} na pseudopróbie Z^j, dla każdego węzła wykonując następujące czynności, dopóki liczba obserwacji w węźle nie będzie równa 1 lub wszystkie obserwacje w węźle nie będą miały jednakowych etykiet:
      i. Spośród M zmiennych wylosuj $m_{tr} << M$ zmiennych bez zwracania.
      ii. Spośród m_{tr} wylosowanych zmiennych wybierz najlepszy podział.
      iii. Podziel węzeł na dwa.
2. Dla każdego $j = K^{tr}, \ldots, K^{tr} + K^{lda}$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m_{lda} << M$ zmiennych bez zwracania. Z wartości m_{lda} wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^j.
   (c) Wytrenuj klasyfikator LDA f^j_{lda} na pseudopróbie Z^j.
3. Dla każdego $j = K^{tr} + K^{lda} + 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m_{knn} << M$ zmiennych bez zwracania. Z wartości m_{knn} wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^j.
   (c) Wytrenuj klasyfikator knn f^j_{knn} z liczbą sąsiadów k na pseudopróbie Z^j.
4. Dokonaj predykcji klasy dla x przy pomocy wszystkich klasyfikatorów $f^j(x)$, $j = 1, \ldots, K$, gdzie
$$f^j = \begin{cases} f^j_{tr} & \text{dla} \quad j = 1, \ldots, K^{tr} \\ f^j_{lda} & \text{dla} \quad j = K^{tr} + 1, \ldots, K^{tr} + K^{lda} \\ f^j_{knn} & \text{dla} \quad j = K^{tr} + K^{lda} + 1, \ldots, K \end{cases}$$
5. Oblicz średnią $f_{hf}(x) = \frac{1}{K} \sum_{j=1}^{K} f^j(x)$.
6. Podaj $G_{hf}(x) = \arg\max_{c \in \{1,\ldots,C\}} f_{hf}(x)$.

3.2. Dobór parametrów komitetu

Dobór parametrów dla drzew decyzyjnych oparto o rezultaty badania lasów losowych z poprzedniego rozdziału. Ponieważ najlepsze wyniki osiągały komitety drzew nieprzycinanych, o zmiennych losowanych niezależnie dla każdego węzła, to takie właśnie drzewa zdecydowano się włączyć do komitetu niejednorodnego. Liczbę losowanych zmiennych (parametr m_{tr}) ustalono na poziomie $\lfloor\sqrt{M}\rfloor$, a więc takim samym, jak dla lasów losowych.

Zbiór	M	m_{lda} dające najmniejszy błąd	$2\lfloor\sqrt{M}\rfloor$
v20.eq	20	20	8
v20.dom	20	8	8
v5.coreq	5	5	4
mlbench.threenorm	10	6	6
mlbench.simplex	4	4	4
mlbench.xor	4	2	4

Tabela 3.2: Wartości parametru m_{lda}, dla których komitet niejednorodny drzew i LDA osiąga najmniejszy błąd klasyfikacji na poszczególnych zbiorach testowych, w porównaniu z przyjętą wartością domyślną $m_{lda} = 2\lfloor\sqrt{M}\rfloor$.

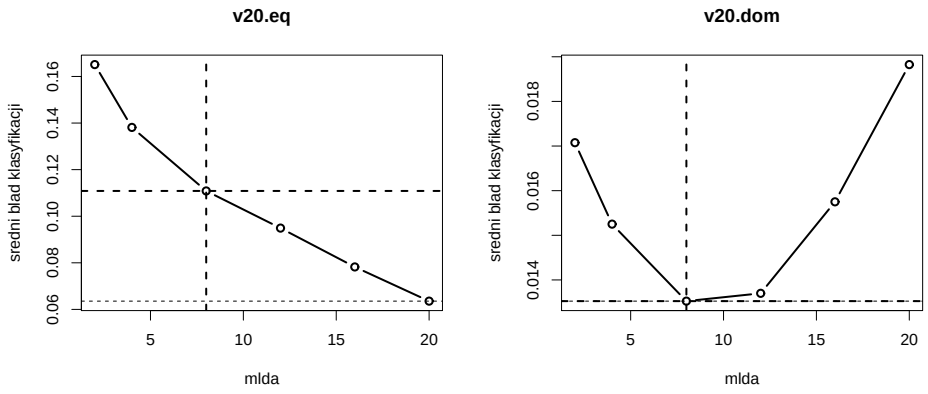
W przypadku klasyfikatorów LDA jedynym parametrem wymagającym wyznaczenia była liczba zmiennych losowanych do zbioru treningowego. Jego wartość ustalono eksperymentalnie, wykorzystując procedurę analogiczną do tej zastosowanej dla lasu zmodyfikowanego w podrozdziale 2.1.1, przy czym wszystkich obliczeń dokonywano dla komitetu składającego się z 50 drzew i 50 klasyfikatorów LDA. Wyniki przedstawione w tabeli 3.2 oraz na rysunkach 3.2-3.4 pokazują, że trudno ustalić regułę wiążącą m_{lda} z liczbą zmiennych M, która dawałaby najmniejszy błąd dla wszystkich badanych zbiorów. Jednak ustalając wartość parametru m_{lda} na poziomie $2\lfloor\sqrt{M}\rfloor$, można w większości przypadków (poza zbiorem `v20.eq` i `mlbench.xor`) otrzymać błąd klasyfikacji o wartości najmniejszej lub bliskiej najmniejszej. Do dalszych eksperymentów zdecydowano się jednak dla każdego zbioru użyć wartości m_{lda} minimalizującej błąd.

Grupa klasyfikatorów knn wymagała natomiast ustalenia dwóch parametrów: liczby zmiennych losowanych do zbioru treningowego m_{knn} oraz liczby sąsiadów k. Eksperymentalnego wyznaczenia wartości m_{knn} dokonano analogicznie do m_{lda}, a więc badając komitety złożone z 50 drzew i 50 klasyfikatorów knn, przy czym przyjęto wartość $k = 5$. Wyniki prezentuje tabela 3.3 oraz rysunki 3.5-3.7. Podobnie jak w przypadku klasyfikatorów LDA, trudno ustalić optymalną wartość m_{knn}, jednak przyjmując $m_{knn} = 2\lfloor\sqrt{M}\rfloor$, we wszystkich przypadkach poza zbiorem `mlbench.xor` uzyskamy błąd klasyfikacji o wartości najmniejszej lub jej bliskiej.

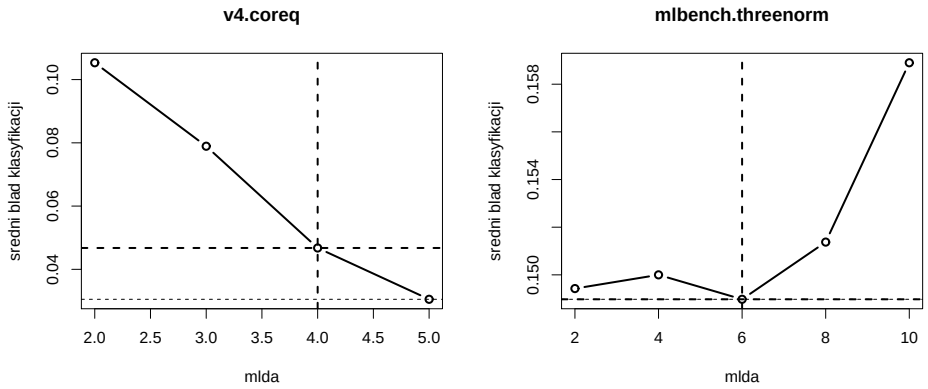
Zbiór	M	m_{knn} dające najmniejszy błąd	$2\lfloor\sqrt{M}\rfloor$
v20.eq	20	12	8
v20.dom	20	4	8
v5.coreq	5	5	4
mlbench.threenorm	10	8	6
mlbench.simplex	4	4	4
mlbench.xor	4	2	4

Tabela 3.3: Wartości parametru m_{knn}, dla których komitet niejednorodny drzew i knn osiąga najmniejszy błąd klasyfikacji na poszczególnych zbiorach testowych, w porównaniu z przyjętą wartością domyślną $m_{knn} = 2\lfloor\sqrt{M}\rfloor$.

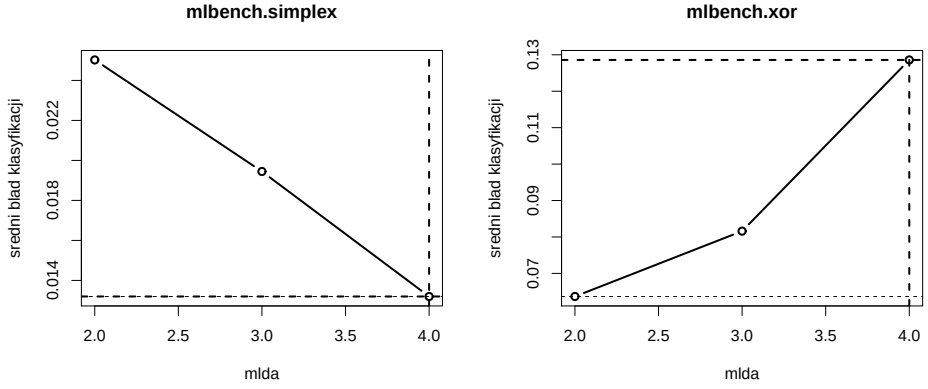
Wyboru wartości parametru k dokonano w podobny sposób, dla każdego zbioru przyjmując m_{knn} minimalizujące błąd. Dla uproszczenia przyjmowano jednakowe k dla wszystkich klasyfikatorów knn w komitecie. Wyznaczenie optymalnej liczby sąsiadów k jest z kilku powodów trudniejsze niż znalezienie najlepszej wartości m. Przede wszystkim k może przyjmować wszystkie całkowite wartości z przedziału $[1, N]$, podczas gdy w przypadku m górną granicę stanowi liczba zmiennych M, dla zdecydowanej większości zbiorów znacznie mniejsza niż liczba obserwacji N. Po drugie, najprawdopodobniej na optymalną wartość k wpływa nie tylko liczba zmiennych, ale również m.in. liczba obserwacji, na co wskazują wykresy błędu klasyfikacji w zależności od k dla zbiorów testowych (3.8-3.9) oraz wyniki zgromadzone w tabeli 3.4. Po trzecie zaś, własności tej zależności nie są znane, zatem znalezienie k lokalnie minimalizującego błąd klasyfikacji nie oznacza jeszcze, że jest to optymalna liczba sąsiadów dla badanego zbioru.



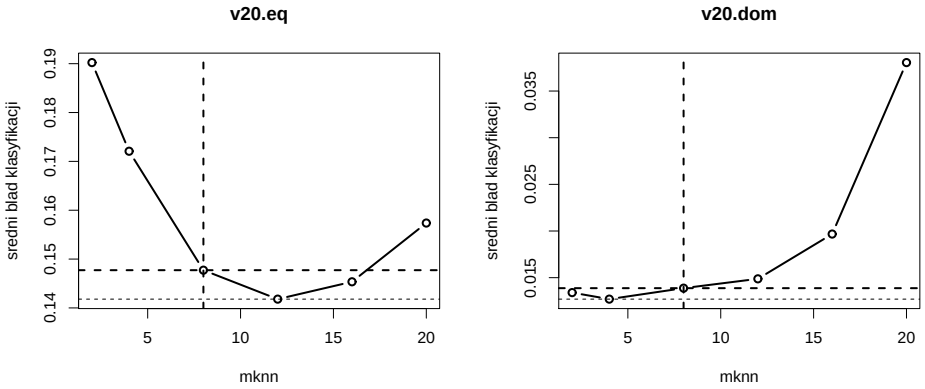
Rysunek 3.2: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru m_{lda} dla zbiorów `v20.eq` i `v20.dom`. Pogrubioną linią przerywaną zaznaczono $m_{lda} = 2\lfloor\sqrt{M}\rfloor$ oraz średni błąd klasyfikacji komitetu przy tej wartości parametru.



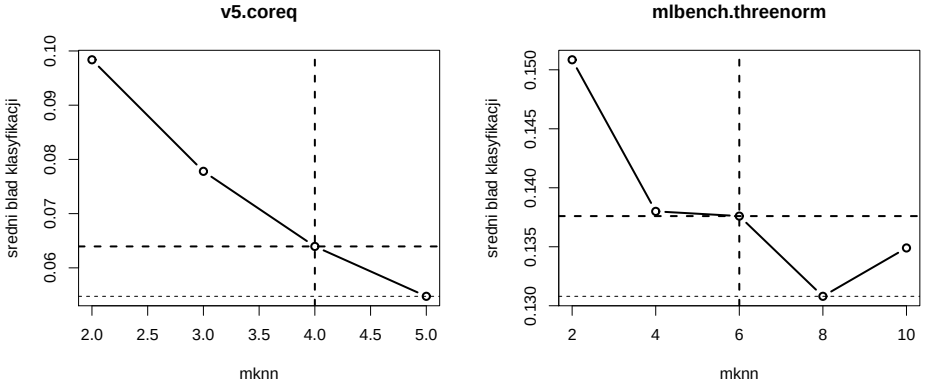
Rysunek 3.3: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru m_{lda} dla zbiorów `v5.coreq` i `mlbench.threenorm`. Pogrubioną linią zaznaczono $m_{lda} = 2\lfloor\sqrt{M}\rfloor$ oraz średni błąd klasyfikacji komitetu przy tej wartości parametru.



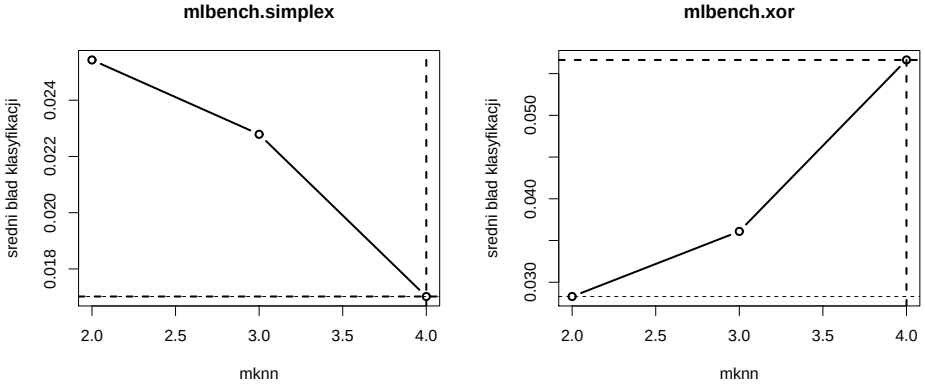
Rysunek 3.4: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru m_{lda} dla zbiorów `mlbench.simplex` i `mlbench.xor`. Pogrubioną linią zaznaczono $m_{lda} = 2\lfloor\sqrt{M}\rfloor$ oraz średni błąd klasyfikacji komitetu przy tej wartości parametru.



Rysunek 3.5: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru m_{knn} dla zbiorów `v20.eq` i `v20.dom`. Pogrubioną linią przerywaną zaznaczono $m_{knn} = 2\lfloor\sqrt{M}\rfloor$ oraz średni błąd klasyfikacji komitetu przy tej wartości parametru.



Rysunek 3.6: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru m_{knn} dla zbiorów `v5.coreq` i `mlbench.threenorm`. Pogrubioną linią przerywaną zaznaczono $m_{knn} = 2\lfloor\sqrt{M}\rfloor$ oraz średni błąd klasyfikacji komitetu przy tej wartości parametru.



Rysunek 3.7: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru m_{knn} dla zbiorów `mlbench.simplex` i `mlbench.xor`. Pogrubioną linią przerywaną zaznaczono $m_{knn} = 2\lfloor\sqrt{M}\rfloor$ oraz średni błąd klasyfikacji komitetu przy tej wartości parametru.

Zbiór	M	N	k dające najmniejszy błąd
v20.eq	20	600	25
v20.dom	20	600	5
v5.coreq	5	600	5
mlbench.threenor	10	600	9
mlbench.simplex	4	625	57
mlbench.xor	4	800	5
mlbench.smiley	2	1500	1
iris	4	150	5
crabsn	6	200	1
pima	7	532	25
ozone.bclass	72	256	1
ctgn	21	2126	1

Tabela 3.4: Wartości parametru k, dla których komitet niejednorodny drzew i knn osiąga najmniejszy błąd klasyfikacji na poszczególnych zbiorach testowych.

Poszukiwanie optymalnego k na całym przedziale $[1, N]$ dla wszystkich badanych zbiorów byłoby jednak bardzo kosztowne obliczeniowo, dlatego ograniczono się do zbadania zakresu $[1, \frac{N}{6}]$. Za najlepszą liczbę sąsiadów dla danego zbioru przyjęto zatem taką wartość k, dla której błąd klasyfikacji był najmniejszy na tym przedziale. Mimo to nie udało się ustalić reguły wiążącej najlepszą liczbę sąsiadów z M oraz N, która byłaby wspólna dla wszystkich zbiorów. Wartości k dla zbiorów o jednakowych bądź bardzo zbliżonych poziomach obu parametrów (np. `v20.eq` i `v20.dom`, `v5.coreq` i `mlbench.simplex`) są skrajnie różne, zatem najwyraźniej optymalna liczba sąsiadów zależy również od innych, bardziej złożonych charakterystyk zbioru, takich jak separowalność klas czy nieregularność granic pomiędzy nimi. Stosując komitet niejednorodny w praktyce trzeba więc będzie dokonywać wyboru k przy pomocy kroswalidacji. Na potrzeby dalszych testów ustalono k na poziomie uznanym za najlepszy indywidualnie dla każdego z wykorzystywanych zbiorów.

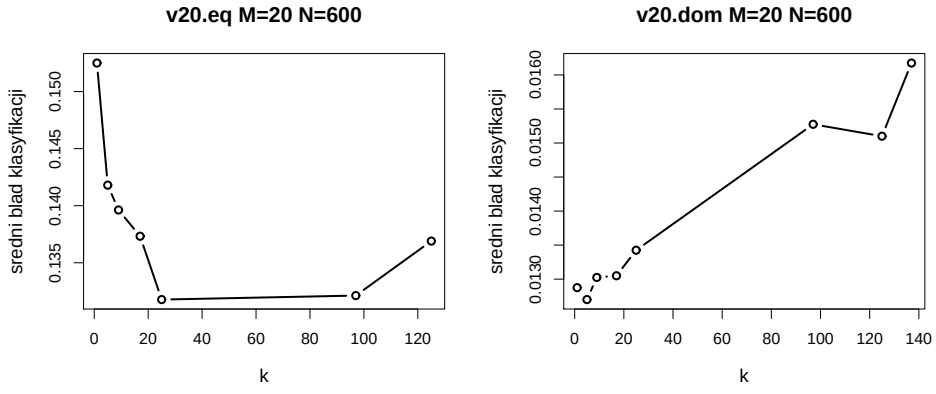
Po ustaleniu wartości parametrów można porównać zachowanie komitetu niejednorodnego (o równych proporcjach poszczególnych typów klasyfikatorów) z wynikami komitetów jednorodnych: lasu losowego, komitetu klasyfikatorów LDA oraz komitetu klasyfikatorów knn. Wyniki ilustruje tabela 3.5 oraz rysunki 3.10-3.13. Na początku warto zauważyć, że przewaga poszczególnych metod na badanych zbiorach zaobserwowana dla pojedynczych klasyfikatorów została też zachowana dla ich jednorodnych komitetów. Zwykły las losowy ma nadal zdecydowaną przewagę nad jednorodnymi komitetami LDA i knn m.in. na zbiorze `mlbench.xor`, komitet LDA jest najlepszy wśród jednorodnych np. na zbiorze `v20.eq`, natomiast na zbiorze `mlbench.threenorm` najmniejszy błąd uzyskuje komitet knn. Przyjrzyjmy się teraz komitetowi niejednorodnemu — we wszystkich przypadkach jest on albo najlepszy, albo drugi pod względem błędu klasyfikacji. Warto przeanalizować dokładniej rezultaty uzyskane na zbiorach `ozone.bclass` i `pima`: mimo, że średni błąd klasyfikatora w komitecie niejednorodnym nie jest w przypadku tych zbiorów najmniejszy (pod tym względem jednorodne komitety LDA oraz knn wypadają od niego lepiej), to jednak jakość klasyfikacji całego komitetu jest najlepsza spośród badanych metod. Obecność drzew w komitecie najwyraźniej podnosi średni indywidualny błąd klasyfikatora, ale jednocześnie znacząco obniża korelację komponentów, dzięki czemu cały komitet klasyfikuje lepiej.

Efekt niższej korelacji nie zawsze jest wystarczająco silny, by przewyższyć spadek jakości wywołany pojawieniem się w komitecie słabszych klasyfikatorów, jednak w większości przypadków skutecznie nadrabia ten ubytek (zbiory `v20.dom`, `v5.coreq`, `mlbench.threenorm`, `mlbench.simplex`, `mlbench.smiley`). Być może jednak przy innych proporcjach poszczególnych metod w komitecie udałoby się zmienić relację tych dwóch przeciwstawnych efektów

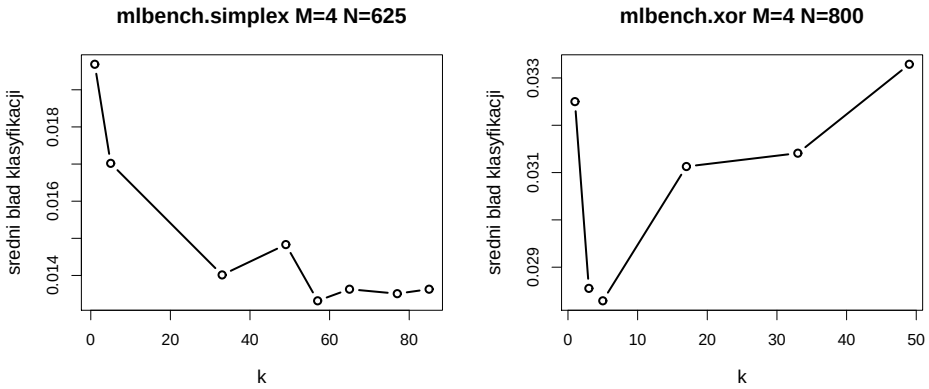
i uzyskać jeszcze lepszą jakość klasyfikacji komitetu. Można na przykład zmienić proporcje typów klasyfikatorów w taki sposób, aby znaczenie każdej z metod w końcowym werdykcie komitetu odpowiadało jakości klasyfikacji tej metody na badanym zbiorze. Tego zagadnienia dotyczy kolejny podrozdział.

Zbiór	Błąd średni			
	Las zwykły	Komitet LDA	Komitet knn	Komitet niejednorodny
v20.eq	0,167	**0,061**	0,124	0,063
v20.dom	**0,013**	0,081	0,124	0,015
v5.coreq	0,078	**0,023**	0,051	**0,033**
mlbench.threenorm	0,147	0,169	**0,129**	0,133
mlbench.simplex	0,032	**0,013**	0,014	**0,013**
mlbench.xor	**0,020**	0,782	0,515	0,138
mlbench.smiley	**0,002**	0,249	0,003	0,003
iris	0,051	**0,022**	0,045	0,039
crabsn	0,138	**0,001**	0,068	0,024
pima	0,228	0,220	0,250	**0,215**
ozone.bclass	0,177	0,173	0,194	**0,163**
ctgn	**0,059**	0,140	0,082	0,086

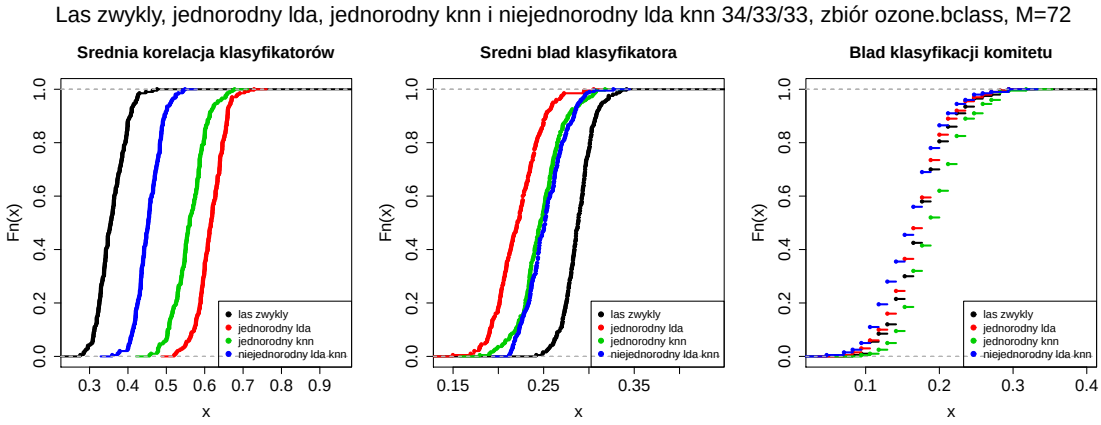
Tabela 3.5: Średnie błędy klasyfikacji komitetów różnych typów dla zbiorów testowych. Pogrubioną czcionką oznaczono błąd najmniejszy dla danego zbioru (może być więcej niż jeden, jeśli różnica między błędami różnych metod nie była istotna statystycznie, co badano testem Wilcoxona).



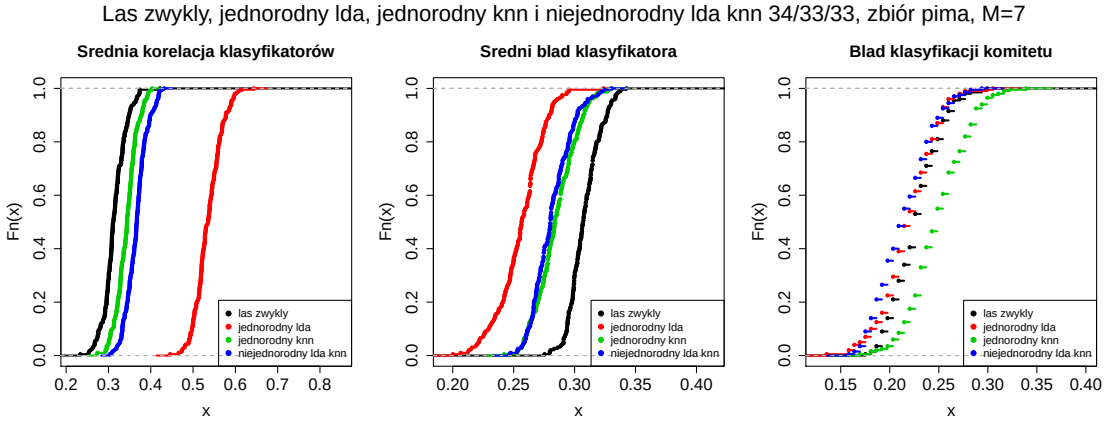
Rysunek 3.8: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru k dla zbiorów `v20.eq` i `v20.dom`.



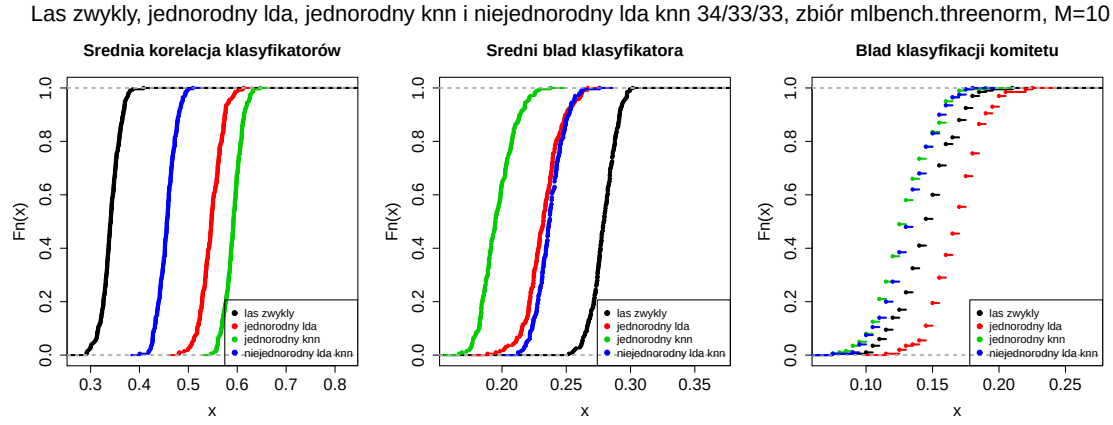
Rysunek 3.9: Wykresy średniego błędu klasyfikacji komitetu niejednorodnego w zależności od parametru k dla zbiorów `mlbench.simplex` i `mlbench.xor`.



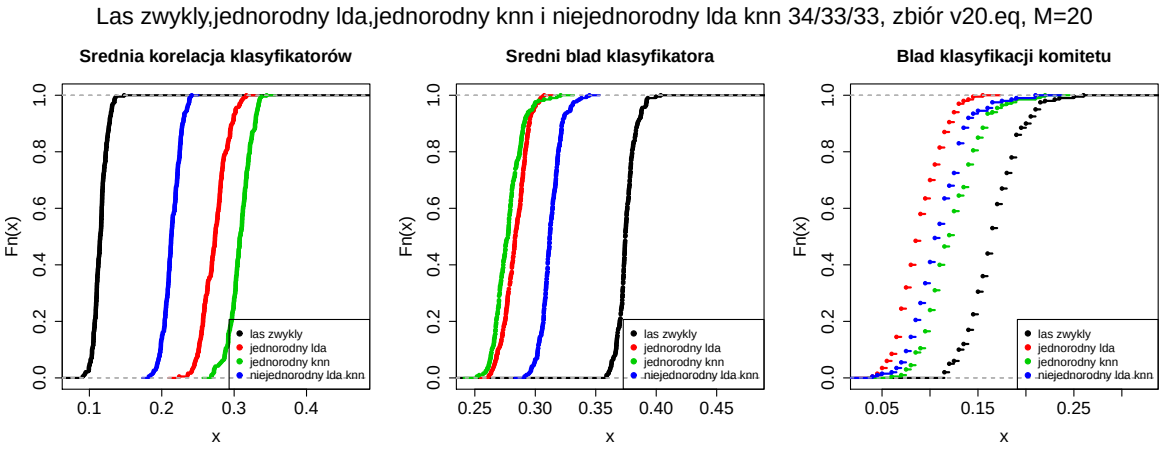
Rysunek 3.10: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetów jednorodnych (drzew, LDA, knn) oraz komitetu niejednorodnego; zbiór `ozone.bclass`



Rysunek 3.11: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetów jednorodnych (drzew, LDA, knn) oraz komitetu niejednorodnego; zbiór `pima`.



Rysunek 3.12: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetów jednorodnych (drzew, LDA, knn) oraz komitetu niejednorodnego; zbiór `mlbench.threenorm`.



Rysunek 3.13: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetów jednorodnych (drzew, LDA, knn) oraz komitetu niejednorodnego; zbiór `v20.eq`.

3.3. Agregacja głosów — ważenie głosów i proporcje klasyfikatorów

W dotychczas badanym wariancie komitetu niejednorodnego wszystkie typy klasyfikatorów miały równe proporcje, a ich głosy nie były ważone. Innymi słowy, zarówno poszczególne metody klasyfikacji, jak i pojedyncze klasyfikatory miały jednakowy wpływ na ostateczny werdykt komitetu, bez względu na popełniany błąd klasyfikacji czy korelację z pozostałymi członkami komitetu. Ponieważ jednak dla wielu zbiorów danych jedna z metod klasyfikacji ma wyraźną przewagę nad pozostałymi, istnieje szansa poprawy jakości klasyfikacji komitetu poprzez zwiększenie udziału klasyfikatorów tego typu w końcowej decyzji komitetu. Można to zrobić, zmieniając proporcje poszczególnych rodzajów klasyfikatorów lub wprowadzając wagi uzależnione od błędu pojedynczych klasyfikatorów. Ważenie głosów wpływa pozytywnie na jakość klasyfikacji nawet w przypadku komitetów jednorodnych (ilustruje to przykład lasu

losowego w podrozdziale 2.2), zatem można się spodziewać jeszcze lepszych efektów dla komitetów niejednorodnych. Analizę sposobów agregacji głosów rozpoczęto więc od komitetów ważonych.

Przyjęto wagi w postaci (2.1), szacując indywidualne błędy klasyfikacji na podstawie obserwacji OOB, tak jak to miało miejsce w przypadku lasu losowego w podrozdziale 2.2. Nadal więc obliczanie wag może być wbudowane w proces konstruowania komitetu i nie zmniejsza rozmiarów zbioru treningowego. Algorytm budowy i predykcji przy pomocy komitetu będzie więc wyglądać następująco:

Algorytm 8. Komitet niejednorodny drzew, LDA i knn

Dane: próba $Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, $y_i \in \{1, \ldots, C\}$, K^{tr} – liczba drzew,
K^{lda} – liczba klasyfikatorów LDA, K – liczba wszystkich klasyfikatorów,
M – liczba zmiennych w próbie Z, x – obserwacja wymagająca klasyfikacji.

1. Dla każdego $j = 1, \ldots, K^{tr}$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem, tworząc pseudopróbę Z^j.
   (b) Wytrenuj drzewo decyzyjne f_{tr}^j na pseudopróbie Z^j, dla każdego węzła wykonując następujące czynności, dopóki liczba obserwacji w węźle nie będzie równa 1 lub wszystkie obserwacje w węźle nie będą miały jednakowych etykiet:
      i. Spośród M zmiennych wylosuj $m_{tr} << M$ zmiennych bez zwracania.
      ii. Spośród m_{tr} wylosowanych zmiennych wybierz najlepszy podział.
      iii. Podziel węzeł na dwa.
   (c) Dokonaj predykcji klasy dla obserwacji OOB $G_{tr}^j(x_i^{oob})$, $i = 1, \ldots, N^{oob}$.
   (d) Oblicz błąd $err_j = \frac{1}{N^{oob}} \sum_{i=1}^{N^{oob}} I(G_{tr}^j(x_i^{oob}) \neq y_i^{oob})$ oraz wagę $w_j = \log\left((1 - err_j)/(err_j)\right)$.
2. Dla każdego $j = K^{tr}, \ldots, K^{tr} + K^{lda}$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m_{lda} << M$ zmiennych bez zwracania. Z wartości m_{lda} wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^j.
   (c) Wytrenuj klasyfikator LDA f_{lda}^j na pseudopróbie Z^j.
   (d) Dokonaj predykcji klasy dla obserwacji OOB $G_{lda}^j(x_i^{oob})$, $i = 1, \ldots, N^{oob}$.
   (e) Oblicz błąd $err_j = \frac{1}{N^{oob}} \sum_{i=1}^{N^{oob}} I(G_{lda}^j(x_i^{oob}) \neq y_i^{oob})$ oraz wagę $w_j = \log\left((1 - err_j)/(err_j)\right)$.
3. Dla każdego $j = K^{tr} + K^{lda} + 1, \ldots, K$:
   (a) Z próby Z wylosuj N obserwacji ze zwracaniem.
   (b) Spośród M zmiennych wylosuj $m_{knn} << M$ zmiennych bez zwracania. Z wartości m_{knn} wylosowanych atrybutów dla N wylosowanych obserwacji stwórz pseudopróbę Z^j.
   (c) Wytrenuj klasyfikator knn f_{knn}^j z liczbą sąsiadów k na pseudopróbie Z^j.
   (d) Dokonaj predykcji klasy dla obserwacji OOB $G_{knn}^j(x_i^{oob})$, $i = 1, \ldots, N^{oob}$.
   (e) Oblicz błąd $err_j = \frac{1}{N^{oob}} \sum_{i=1}^{N^{oob}} I(G_{knn}^j(x_i^{oob}) \neq y_i^{oob})$ oraz wagę $w_j = \log\left((1 - err_j)/(err_j)\right)$.
4. Dokonaj predykcji klasy dla x przy pomocy wszystkich klasyfikatorów $f^j(x)$, $j = 1, \ldots, K$, gdzie

$$f^j = \begin{cases} f_{tr}^j & \text{dla} \quad j = 1, \ldots, K^{tr} \\ f_{lda}^j & \text{dla} \quad j = K^{tr} + 1, \ldots, K^{tr} + K^{lda} \\ f_{knn}^j & \text{dla} \quad j = K^{tr} + K^{lda} + 1, \ldots, K \end{cases}$$

5. Oblicz średnią $f_{hfw}(x) = \frac{1}{\sum_{j=1}^{K}} \sum_{j=1}^{K} w_j f^j(x)$.
6. Podaj $G_{hfw}(x) = \arg\max_{c \in \{1,\ldots,C\}} f_{hfw}(x)$.

Porównanie komitetów niejednorodnych ważonych z nieważonymi oraz z najlepszym spośród komitetów jednorodnych pod kątem błędu klasyfikacji prezentuje tabela 3.6, a wyniki ilustrują rysunki 3.14-3.17. Ważenie głosów rzeczywiście poprawiło jakość klasyfikacji komitetu dla większości zbiorów (9 na 12), jednak spadek błędu jest istotny statystycznie tylko w połowie przypadków. Najbardziej spektakularna poprawa jakości nastąpiła dla zbioru `mlbench.xor`, dla którego błąd klasyfikacji spadł ponad sześciokrotnie. Stało się to dzięki całkowitemu wyeliminowaniu z komitetu klasyfikatorów LDA i knn — ich błąd klasyfikacji przekraczał na tym zbiorze 50%, wskutek czego ich decyzje otrzymały wagi równe 0 (rys. 3.14). Dla dziewięciu zbiorów komitet ważony okazał się również najlepszą metodą klasyfikacji spośród badanych, choć w pięciu przypadkach skutecznością klasyfikacji dorównuje mu jeden z komitetów jednorodnych. Niestety przykład zbiorów `v5.coreq`, `iris`, a zwłaszcza `ctgn` pokazuje, że wprowadzone wagi nie zawsze wystarczająco modyfikują werdykt komitetu (rys. 3.15-3.16). Wydaje się, że za słabo promowane są lepsze metody/klasyfikatory w komitecie.

Zbiór	Błąd średni komitetu				Istotność różnicy	
	Jednorodny		Niejednorodny	Niejednorodny ważony	(p-wartość)	
v20.eq	lda	**0,0613**	**0,0626**	**0,0606**	0,3064	
v20.dom	las	**0,0126**	0,0153	**0,0129**	0,0022	*
v5.coreq	lda	**0,0230**	0,0327	0,0276	$4,4 \cdot 10^{-05}$	*
mlbench.threenorm	knn	**0,1299**	0,1329	**0,1296**	0,1336	
mlbench.simplex	lda	**0,0130**	0,0134	**0,0127**	0,2948	
mlbench.xor	las	**0,0201**	0,1379	**0,0209**	$< 2,2 \cdot 10^{-16}$	*
mlbench.smiley	las	**0,0024**	0,0032	**0,0026**	0,0454	*
iris	lda	**0,0215**	0,0388	0,0351	0,2662	
crabsn	lda	0,0008	0,0239	**0**	$< 2,2 \cdot 10^{-16}$	*
pima	lda	0,2201	**0,2153**	**0,2174**	0,3455	
ozone.bclass	lda	0,1727	**0,1629**	**0,1652**	0,6917	
ctgn	las	**0,0595**	0,0863	0,0821	$1,1 \cdot 10^{-5}$	*

Tabela 3.6: Średnie błędy klasyfikacji komitetu niejednorodnego ważonego, nieważonego i najlepszego komitetu jednorodnego dla zbiorów testowych oraz p-wartość testu Wilcoxona na istotność różnicy między błędem komitetu ważonego i nieważonego (różnice istotne oznaczono gwiazdką). Pogrubioną czcionką oznaczono błąd najmniejszy dla danego zbioru (może być więcej niż jeden, jeśli różnica między błędami różnych komitetów nie była istotna statystycznie).

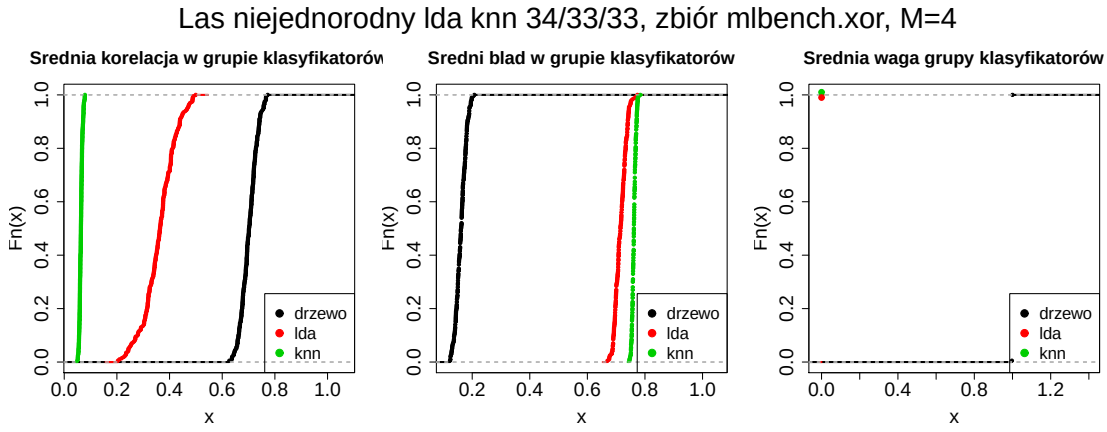
Warto też zauważyć, że dla zbiorów `pima` oraz `ozone.bclass` wprowadzenie wag wręcz pogorszyło jakość klasyfikacji, co prawda nieznacznie (rys. 3.17). Może to mieć związek z faktem, że postać wag uzależniona jest jedynie od indywidualnego błędu klasyfikacji, nie bierze się natomiast pod uwagę korelacji między składnikami komitetu. Tymczasem, po pierwsze, klasyfikatory jednego typu są ze sobą skorelowane silniej niż z klasyfikatorami pozostałych rodzajów, po drugie zaś, metoda o niższym indywidualnym błędzie klasyfikacji będzie się charakteryzować wyższą korelacją w grupie klasyfikatorów. W związku z tym ważenie głosów będzie promować klasyfikatory, które są ze sobą silnie skorelowane, zwłaszcza jeśli jedna z metod jest zdecydowanie lepsza od pozostałych, a potencjalna poprawa jakości klasyfikacji może zostać zniweczona wzrostem zależności między składowymi decyzjami w komitecie. Podsumowując, ważenie głosów może być sposobem na poprawę skuteczności klasyfikacji komitetu niejednorodnego, ale kwestia postaci wag wymaga jeszcze dopracowania.

Spróbujmy teraz wpłynąć na jakość klasyfikacji komitetu poprzez bezpośrednią zmianę proporcji metod klasyfikacji. Trzeba ją oprzeć na oszacowaniu skuteczności klasyfikatorów na analizowanym zbiorze danych. Można do tego celu wykorzystać wagi policzone na podstawie indywidualnych błędów klasyfikatorów — aby otrzymać pożądane udziały poszczególnych metod w komitecie, wystarczy zsumować indywidualne wagi wewnątrz grup klasyfikatorów.

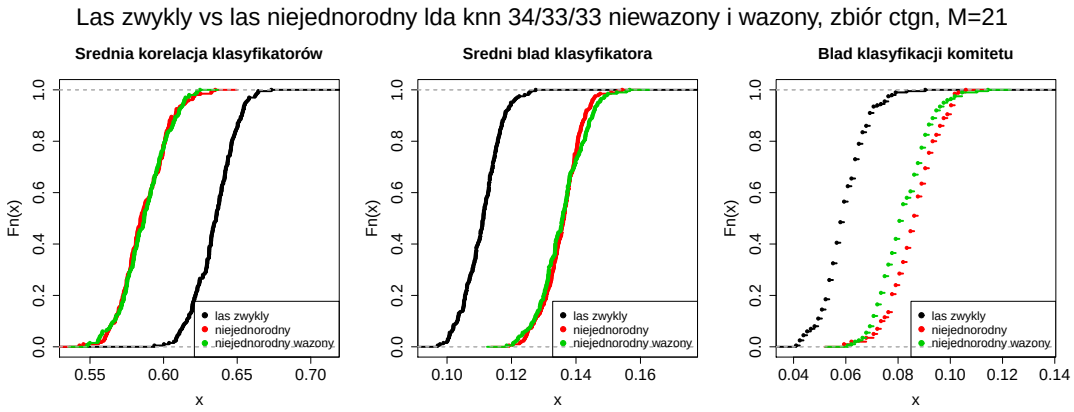
Następnie należy skonstruować komitet o proporcjach pomiędzy metodami klasyfikacji odpowiadających wagom przypisanym poszczególnym grupom klasyfikatorów. Dostosowywanie proporcji komitetu do zbioru danych wymaga więc dwukrotnej budowy komitetu i tym samym wydłuża czas realizacji algorytmu, nadal jednak można dokonać niezbędnych obliczeń bez poświęcania na ten cel części zbioru treningowego, ponieważ opierają się one na błędach OOB. Komitety o zmienionych proporcjach (ale nieważonych głosach) porównano następnie z komitetami ważonymi. Ilustrują to rysunki 3.18-3.20. Różnice między obiema metodami są w większości przypadków nieistotne statystycznie (tabela 3.7).

Zbiór	Błąd średni komitetu		Istotność różnicy	
	zmiana proporcji	ważenie	(p-wartość)	
v20.eq	0,0620	0,0606	0,346	
v20.dom	0,0131	0,0129	0,674	
v5.coreq	0,0280	0,0276	0,77	
mlbench.threenorm	0,1334	0,1296	0,115	
mlbench.simplex	0,0126	0,0127	0,758	
mlbench.xor	0,0201	0,0209	0,269	
mlbench.smiley	0,0031	0,0026	0,059	*
iris	0,0360	0,0351	0,671	
crabsn	0	0	—	
pima	0,2155	0,2174	0,484	
ozone.bclass	0,1599	0,1652	0,309	
ctgn	0,0846	0,0821	0,010	*

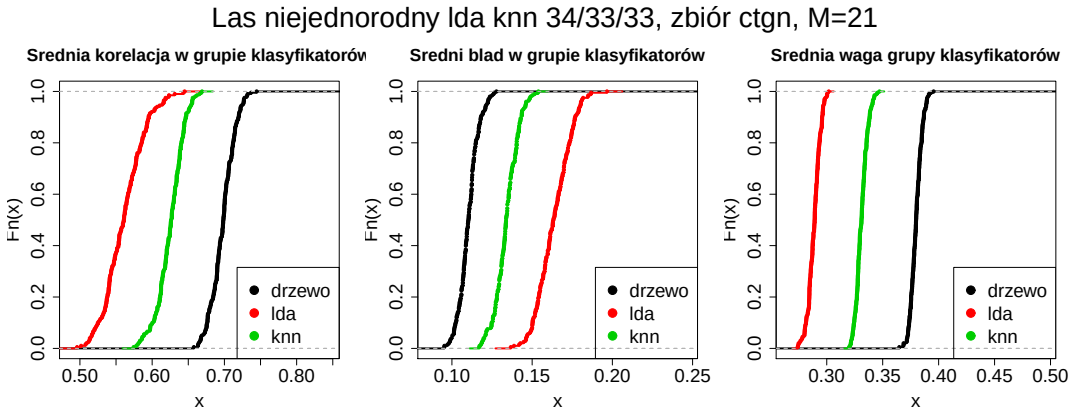
Tabela 3.7: Średnie błędy klasyfikacji komitetu niejednorodnego ważonego i komitetu ze zmienioną proporcją dla zbiorów testowych oraz p-wartość dla testu Wilcoxona na istotność różnicy między nimi (różnice istotne oznaczono gwiazdką).



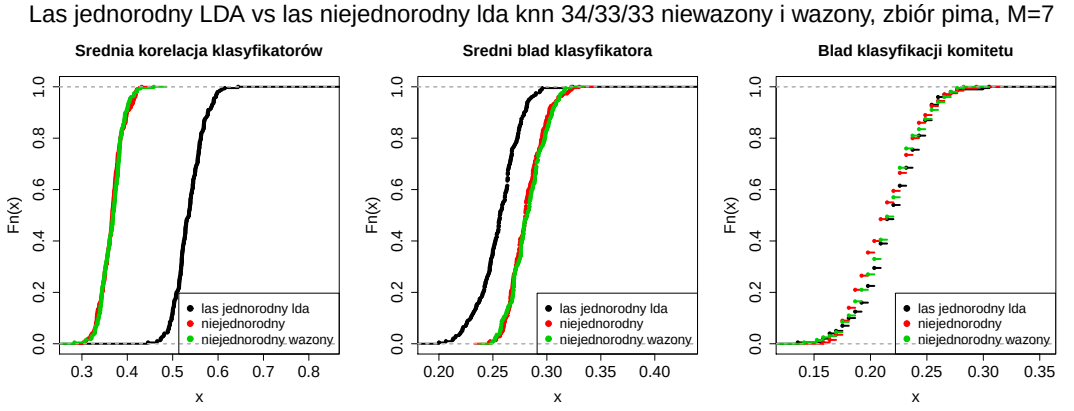
Rysunek 3.14: Dystrybuanty średniej korelacji w grupach klasyfikatorów, średniego błędu indywidualnego w grupie klasyfikatorów oraz średnich wag zastosowanych do grup klasyfikatorów dla komitetu niejednorodnego na zbiorze `mlbench.xor`.



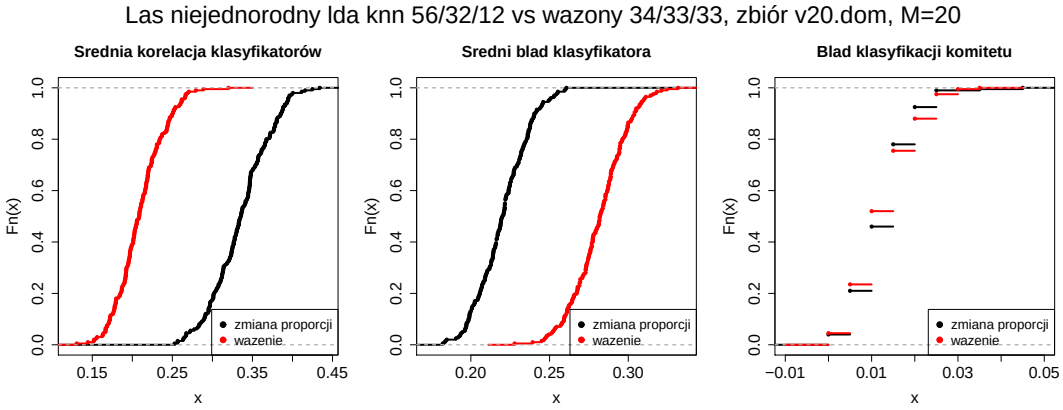
Rysunek 3.15: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla lasu losowego (najlepszego spośród komitetów jednorodnych) oraz komitetu niejednorodnego nieważonego i ważonego; zbiór `ctgn`.



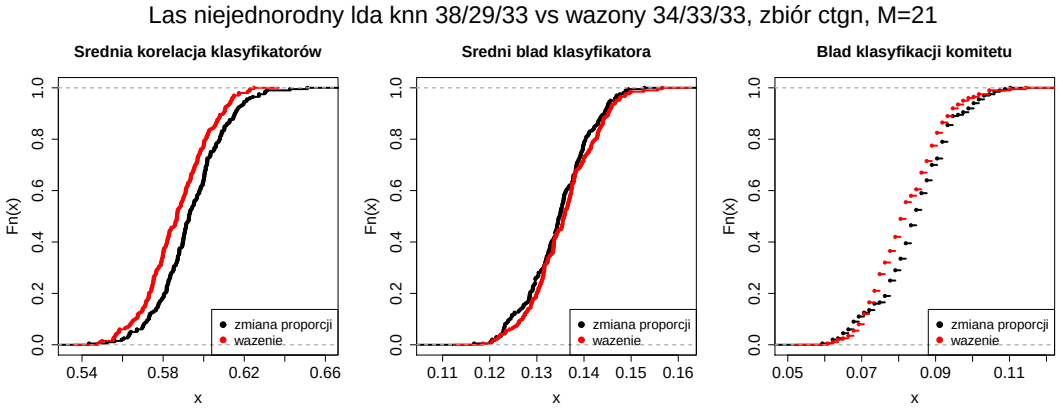
Rysunek 3.16: Dystrybuanty średniej korelacji w grupach klasyfikatorów, średniego błędu indywidualnego w grupie klasyfikatorów oraz średnich wag zastosowanych do grup klasyfikatorów dla komitetu niejednorodnego na zbiorze `ctgn`.



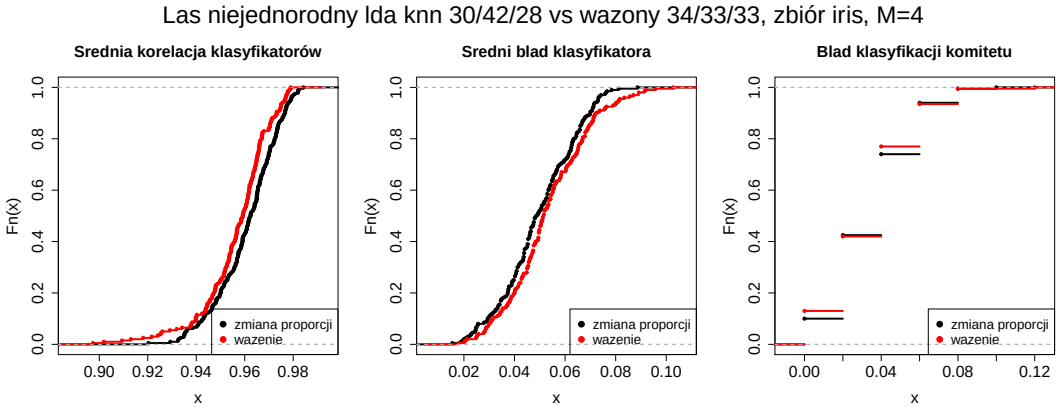
Rysunek 3.17: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetu LDA (najlepszego spośród jednorodnych) oraz komitetu niejednorodnego nieważonego i ważonego; zbiór `pima`.



Rysunek 3.18: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetu niejednorodnego ze zmienioną proporcją klasyfikatorów oraz komitetu niejednorodnego ważonego; zbiór `v20.dom`.



Rysunek 3.19: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetu niejednorodnego ze zmienioną proporcją klasyfikatorów oraz komitetu niejednorodnego ważonego; zbiór `ctgn`.



Rysunek 3.20: Dystrybuanty średniej korelacji klasyfikatorów, średniego błędu indywidualnego klasyfikatora i błędu klasyfikacji komitetu dla komitetu niejednorodnego ze zmienioną proporcją klasyfikatorów oraz komitetu niejednorodnego ważonego; zbiór `iris`.

Ważenie głosów jest więc dobrym ekwiwalentem zmiany proporcji w komitecie niejednorodnym, a jednocześnie nie wymaga dwukrotnej budowy komitetu. Dla niektórych zbiorów (`mlbench.smiley`, `ctgn`) komitety ważone dają nawet nieco lepsze rezultaty, ponieważ ważenie nie tylko zwiększa znaczenie najlepszej metody w werdykcie komitetu, ale też promuje decyzje najlepszych klasyfikatorów bez względu na ich rodzaj. Można jednocześnie zaobserwować, ze we wszystkich przypadkach zmiana proporcji w komitecie spowodowała wzrost korelacji między klasyfikatorami.

Przejdźmy teraz do eksperymentów z wagami w innej postaci niż (2.1). Ponieważ wydaje się, że zastosowana forma za słabo promuje dobre klasyfikatory, zdecydowano się przetestować dwa warianty wag, których wartości będą bardziej zróżnicowane pomiędzy gorszymi i lepszymi klasyfikatorami. Pierwszy z nich to postać najbardziej intuicyjna, oparta na odwrotności błędu klasyfikacji:

$$w_i^{rec} = \max\left(\frac{1}{err_i} - 2, 0\right) = \max\left(\frac{1 - 2\,err_i}{err_i}, 0\right), \quad i = 1, \ldots, T, \tag{3.1}$$

gdzie err_i – błąd klasyfikacji i-tego klasyfikatora, T – liczba klasyfikatorów w komitecie. Stałą równą 2 odjęto, aby w^{rec} jako funkcja błędu była ciągła. Klasyfikatory gorsze niż losowe przypisywanie etykiet ($err_i \geqslant \frac{1}{2}$) powinny zostać całkowicie wyeliminowane z komitetu, dlatego nadaje się im wagi równe 0. Przy ustalonej relacji między błędami dwóch klasyfikatorów, $\frac{err_1}{err_2} = k$, $k > 1$, $err_i < \frac{1}{2}$, stosunek ich wag w postaci 3.1 będzie wynosić:

$$\frac{w_2^{rec}}{w_1^{rec}} = \frac{(1 - 2\,err_2)err_1}{(1 - 2\,err_1)err_2} = \frac{k(1 - 2\,err_2)}{1 - 2k\,err_2}, \tag{3.2}$$

podczas gdy analogiczny stosunek wag postaci (2.1) wynosi:

$$\frac{w_2^{boost}}{w_1^{boost}} = \frac{\ln\left(\frac{1-err_2}{err_2}\right)}{\ln\left(\frac{1-k\,err_2}{k\,err_2}\right)} \tag{3.3}$$

Można pokazać, że dla dowolnego $k > 1$ i dla każdego $err_2 \in (0, \frac{1}{2k})$ zachodzi $w_2^{rec}/w_1^{rec} > w_2^{boost}/w_1^{boost}$.

Drugi wariant wag jest silniejszą wersją (3.1):

$$w_i^{sq} = \max\left(\frac{1 - 2\,err_i}{err_i^2}, 0\right), \quad i = 1, \ldots, T, \tag{3.4}$$

Stosunek wag przy ustalonej relacji błędów pary klasyfikatorów będzie tym razem równy:

$$\frac{w_2^{sq}}{w_1^{sq}} = \frac{k^2(1 - 2\,err_2)}{1 - 2k\,err_2}. \tag{3.5}$$

Widać, że $\forall\, k > 1, \forall\, err_2 \in (0, \frac{1}{2k})$ $w_2^{sq}/w_1^{sq} > w_2^{rec}/w_1^{rec}$, a zatem także $w_2^{sq}/w_1^{sq} > w_2^{boost}/w_1^{boost}$. Postać (3.4) powinna więc doprowadzić do największej rozpiętości wag dobrych i złych klasyfikatorów w komitecie (warto też zauważyć, że zarówno wyrażenie (3.2), jaki i (3.5) jest większe od k, a więc stosunek wag będzie większy niż odwrotność ilorazu błędów klasyfikacji). Wyrażenia (3.2), (3.3) oraz (3.5) zilustrowano na rysunku 3.21 w zależności od err_2, przy ustalonym $k = 2$.

Przetestowano jeszcze jedną modyfikację ważenia klasyfikatorów w komitecie, a mianowicie zastosowanie oddzielnych wag dla każdej klasy. Pozwala to uwzględnić fakt, że dany klasyfikator może być dobry w klasyfikowaniu obserwacji do jednej z klas, ale gorzej radzić

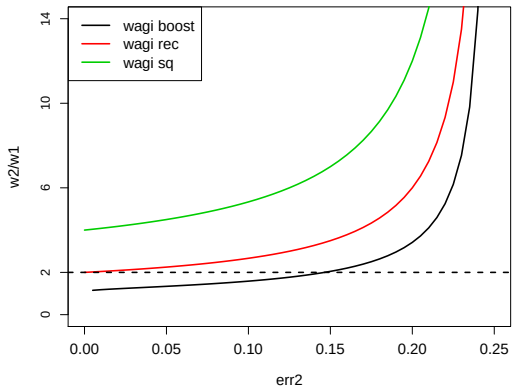
sobie z pozostałymi. Użyto do tego celu zarówno wag w postaci (2.1), jak i (3.4), dla k-tej klasy zastępując zwykły błąd klasyfikacji prawdopodobieństwem błędu pod warunkiem, że obserwacja zaklasyfikowana została do k-tej klasy. Prawdopodobieństwo to oszacowano przez stosunek liczby obserwacji nieprawidłowo zaklasyfikowanych do k-tej klasy do liczby wszystkich obserwacji zaklasyfikowanych do k-tej klasy. Dzięki tak sformułowanym wagom, jeśli pewien klasyfikator popełnia duży błąd klasyfikując obserwacje do k-tej klasy, to jego głos oddawany na k-tą klasę otrzyma niższą wagę i będzie się liczył mniej w werdykcie komitetu. Jednocześnie ten sam klasyfikator może mieć stosunkowo wysoką wagę, jeśli zaklasyfikuje obserwację do innej klasy.

Podsumowując, przetestowano następujące dodatkowe warianty wag:

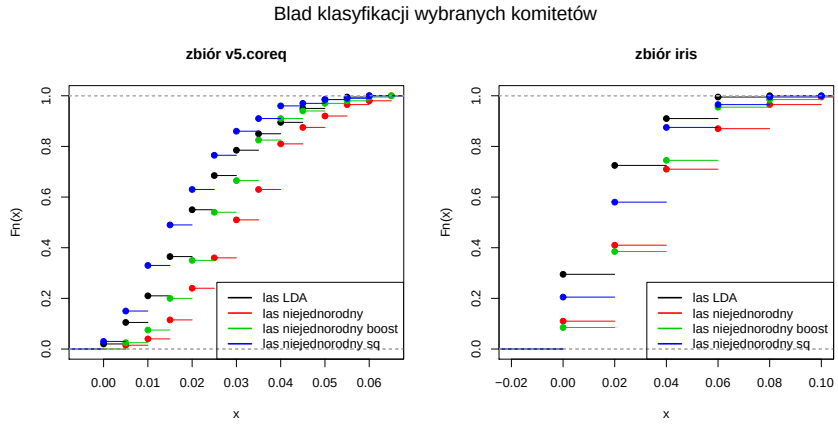
1. Wagi oparte na odwrotności błędu klasyfikacji postaci (3.1), oznaczane dalej jako *rec*;
2. Wagi oparte na odwrotności kwadratu błędu klasyfikacji postaci (3.4), oznaczane dalej jako *sq*;
3. Wagi boostingowe postaci (2.1) oddzielne dla każdej klasy, oznaczane przez *boost lev*;
4. Wagi oparte na odwrotności kwadratu błędu klasyfikacji postaci (3.4) oddzielne dla każdej klasy, oznaczane przez *sq lev*.

Wyniki symulacji z nowymi wariantami wag zawarto w tabeli 3.8. Rysunki 3.22-3.23 ilustrują porównanie między najlepszym komitetem jednorodnym, nieważonym komitetem niejednorodnym oraz najlepszym komitetem ważonym dla wybranych zbiorów. Wprowadzenie nowych metod ważenia nie spowodowało radykalnej zmiany sytuacji, jednak w przypadku kilku zbiorów pozwoliło poprawić dotychczasową jakość klasyfikacji osiąganą przez komitet niejednorodny. Największy spadek błędu klasyfikacji nastąpił na zbiorze `v5.coreq` — dzięki ważeniu przy pomocy wag *sq* udało się zredukować błąd komitetu niejednorodnego o 30% w stosunku do standardowych wag *boost*, otrzymując jednocześnie najniższy błąd klasyfikacji dla tego zbioru. Istotną poprawę można zaobserwować również na zbiorze `ctgn`: komitet niejednorodny z wagami typu *sq lev* daje błąd o ok. 25% mniejszy od komitetu ważonego wagami *boost*. Najmniejszy błąd klasyfikacji na tym zbiorze ma nadal zwykły las losowy, ale różnica między nim a ważonym komitetem niejednorodnym *sq lev* jest już niewielka. Podobnie stało się w przypadku zbioru `iris`: błąd klasyfikacji po zastosowaniu wag *sq* spadł o 25% w stosunku do pierwotnej wersji komitetu ważonego, dzięki czemu różnica między ważonym komitetem niejednorodnym a komitetem LDA jest już nieznaczna, choć nadal istotna statystycznie. Komitety ważone nowymi metodami uzyskały najniższy wśród badanych błąd klasyfikacji także dla zbiorów `ozone.bclass` oraz `v20.dom` (w pierwszym przypadku dzięki użyciu wag *rec*, w drugim zaś wag *boost lev*, jednak uzyskana przewaga okazała się nieistotna statystycznie.

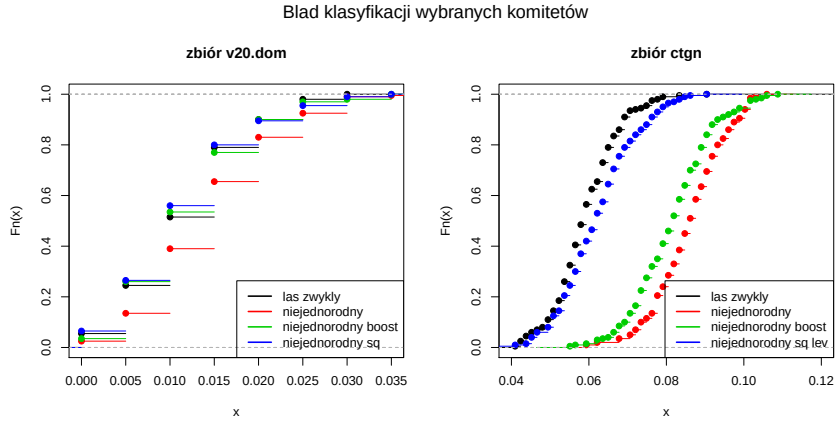
Warto sprawdzić, czy zgodnie z założeniami wagi *rec* oraz *sq* mocniej podkreślają różnicę miedzy słabymi i silnymi klasyfikatorami. Dla wybranych zbiorów wykonano w tym celu rysunki pozwalające porównać wagi przypisane poszczególnym grupom klasyfikatorów (rys. 3.24-3.25). Zarówno w przypadku zbioru `ctgn`, jak i `mlbench.threenorm` wagi zareagowały według przewidywań: rozpiętość wag między najlepszą i najsłabszą grupą klasyfikatorów wzrasta po zastosowaniu wag postaci *rec* i jest największa dla wag *sq*. Jednak wprowadzenie wag *rec* i *sq* daje na obu zbiorach różne efekty: w przypadku zbioru `ctgn` udaje się przy ich pomocy znacznie zredukować błąd klasyfikacji komitetu, natomiast na zbiorze `mlbench.threenorm` nowe wagi (zwłaszcza typu *sq*) prowadzą nawet do niewielkiego wzrostu błędu komitetu. Najwyraźniej wzrost znaczenia silnych klasyfikatorów nie zawsze jest wystarczającym impulsem do obniżenia błędu całego komitetu.



Rysunek 3.21: Zależność stosunku wag klasyfikatorów w_2/w_1 od błędu klasyfikacji err_2 jednego z nich, przy założeniu $err_1/err_2 = 2$.

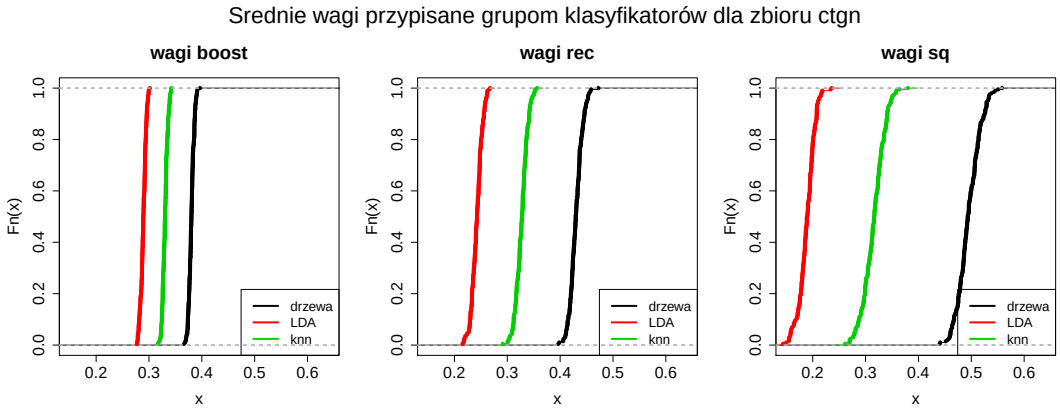


Rysunek 3.22: Dystrybuanty błędu klasyfikacji najlepszego komitetu jednorodnego, nieważonego komitetu niejednorodnego, komitetu ważonego wagami *boost* oraz najlepszego komitetu ważonego dla zbiorów `v5.coreq` oraz `iris`.

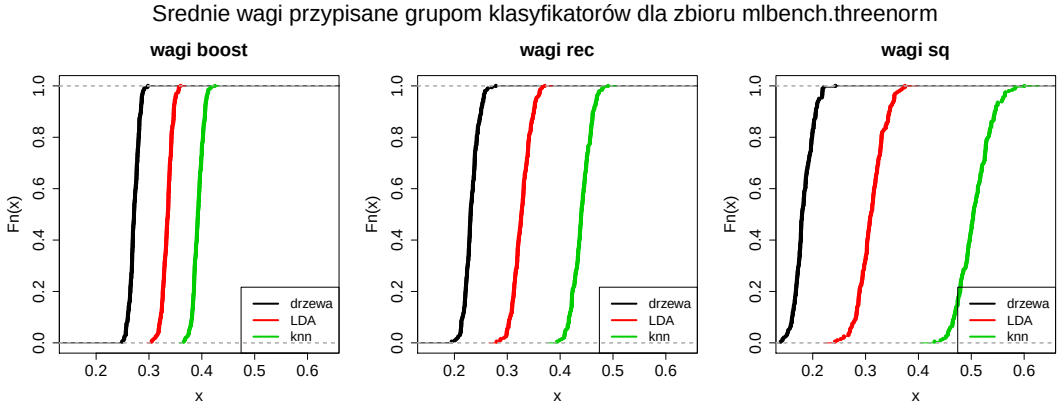


Rysunek 3.23: Dystrybuanty błędu klasyfikacji najlepszego komitetu jednorodnego, nieważonego komitetu niejednorodnego, komitetu ważonego wagami *boost* oraz najlepszego komitetu ważonego dla zbiorów `v20.dom` i `ctgn`.

Wyniki eksperymentów pokazują, że żaden spośród zbadanych typów wag nie ma absolutnej przewagi nad pozostałymi. Dla zdecydowanej większości zbiorów dwa lub więcej komitetów ważonych osiągnęło jednakowo niski błąd klasyfikacji, tylko dla zbiorów `crabsn` oraz `ctgn` jeden z komitetów jest istotnie lepszy od pozostałych (jest to odpowiednio komitet z wagami *boost* oraz komitet z wagami *sq lev*). Chcąc jednak wybrać najbardziej uniwersalny rodzaj wag, należałoby wskazać na typ *rec*: jedynie w przypadku dwóch zbiorów (`crabsn`, `ctgn`) dają one błąd klasyfikacji większy niż najniższy błąd komitetu ważonego. Wagi *boost* również znajdują się w czołówce dla większości zbiorów, lecz w trzech przypadkach prowadzą do znacznego odchylenia od najmniejszego błędu komitetu ważonego (większego niż miało to miejsce przy zastosowaniu wag *rec*). Warto zauważyć, że komitety z wagami *boost*, *rec* oraz *sq* osiągają błędy klasyfikacji tego samego rzędu wielkości, podczas gdy wagi obliczane oddzielnie dla każdej klasy mogą dla pewnych zbiorów pogorszyć jakość klasyfikacji komitetu nawet kilkukrotnie (tak stało się w przypadku zbiorów `v20.eq`, `v5.coreq` oraz `crabsn`). W nielicznych sytuacjach (zbiór `ctgn`) okazują się natomiast bardziej skuteczne od pozostałych metod ważenia. Stosowanie ich bez wstępnej kontroli jakości klasyfikacji (na przykład przy pomocy kroswalidacji) byłoby więc bardzo ryzykowne.



Rysunek 3.24: Dystrybuanty średnich wag przypisanych poszczególnym grupom klasyfikatorów dla zbioru `ctgn` w trzech wariantach: *boost*, *rec* i *sq* (zachowano jednakowe skale na poziomych osiach).



Rysunek 3.25: Dystrybuanty średnich wag przypisanych poszczególnym grupom klasyfikatorów dla zbioru `mlbench.threenorm` w trzech wariantach: *boost*, *rec* i *sq* (zachowano jednakowe skale na poziomych osiach).

Zbiór	Błąd średni komitetu							
	Jednorodny		Niejednorodny					
			nieważony	ważony				
				boost	rec	sq	boost lev	sq lev
v20.eq	lda	**0,061**	**0,063**	**0,061**	**0,061**	**0,061**	0,138	0,149
v20.dom	las	**0,013**	0,015	**0,013**	**0,013**	0,017	**0,012**	0,016
v5.coreq	lda	**0,023**	0,033	0,028	**0,021**	**0,019**	0,052	0,061
mlbench.threenorm	knn	**0,129**	0,133	**0,131**	**0,132**	0,135	**0,131**	0,136
mlbench.simplex	lda	**0,013**	**0,013**	**0,013**	**0,014**	**0,014**	**0,014**	**0,014**
mlbench.xor	las	**0,020**	0,138	**0,021**	**0,022**	0,024	0,023	0,037
mlbench.smiley	las	**0,002**	0,003	0,003	0,004	0,003	0,003	0,003
iris	lda	**0,022**	0,039	0,035	0,029	0,028	0,047	0,048
crabsn	lda	0,001	0,024	**0**	0,001	0,001	0,066	0,051
pima	lda	0,220	**0,215**	**0,219**	**0,215**	**0,219**	0,229	0,233
ozone.bclass	lda	0,173	**0,163**	**0,165**	**0,158**	**0,165**	0,173	0,186
ctgn	las	**0,060**	0,086	0,082	0,077	0,071	0,065	0,063

Tabela 3.8: Średnie błędy klasyfikacji komitetu niejednorodnego ważonego (wraz z nowymi wariantami wag), nieważonego oraz najlepszego komitetu jednorodnego dla zbiorów testowych. Pogrubioną czcionką oznaczono błąd najmniejszy dla danego zbioru (może być więcej niż jeden, jeśli różnica między błędami różnych komitetów nie była istotna statystycznie).

3.4. Podsumowanie

Budowa komitetu niejednorodnego z klasyfikatorów LDA, knn oraz drzew decyzyjnych wydaje się trafnym rozwiązaniem. Na każdym badanym zbiorze jedna z grup klasyfikatorów ma istotną przewagę nad pozostałymi pod względem jakości klasyfikacji, a jednocześnie obecność pozostałych rodzajów klasyfikatorów pozwala utrzymać przeciętną korelację między składowymi komitetu na stosunkowo niskim poziomie. Odpowiedni dobór parametrów komitetu (sugeruje się ustalenie liczby zmiennych losowanych dla klasyfikatorów LDA i knn na poziomie około $2\lfloor\sqrt{M}\rfloor$, choć warto dobrać te parametry eksperymentalnie) pomaga zrównoważyć dążenie do niskiego indywidualnego błędu klasyfikatorów i niskiej korelacji między klasyfikatorami, dzięki czemu nawet nieważone komitety niejednorodne mogą osiągać jakość klasyfikacji istotnie lepszą niż komitety jednorodne (tak się stało w przypadku zbiorów `pima` i `ozone.bclass`). Wprowadzenie wag jest szczególnie istotne w przypadku zbiorów, na których część klasyfikatorów jest zdecydowanie gorsza od pozostałych (np. popełnia błąd klasyfikacji większy niż $\frac{1}{2}$, tak jak klasyfikatory LDA i knn na zbiorze `mlbench.xor`). Ważenie pozwala znacznie zmniejszyć wpływ takich klasyfikatorów na ostateczny werdykt lub wręcz całkowicie wyeliminować je z komitetu. Ważenie głosów zastępuje też dostosowywanie proporcji poszczególnych typów klasyfikatorów w komitecie do ich jakości klasyfikacji na badanym zbiorze.

Po uwzględnieniu wariantów ważonych jedynie w przypadku trzech zbiorów (`iris`, `ctgn` i `mlbench.smiley`) komitety niejednorodne uzyskały błąd klasyfikacji istotnie wyższy od jednorodnych. Również dla trzech zbiorów (`crabsn`, `pima`, `ozone.bclass`) jeden z komitetów niejednorodnych okazał się istotnie lepszy od najlepszego komitetu jednorodnego. Trzeba jednak zaznaczyć, że choć różnica między najmniejszymi błędami komitetów jednorodnych i niejednorodnych jest dla tych sześciu zbiorów istotna statystycznie, to poza jednym przypadkiem (`ozone.bclass`) nie przekracza kilku tysięcznych. Na pozostałych sześciu zbiorach nie ma istotnej różnicy między najlepszym komitetem jednorodnym i niejednorodnym. Można więc stwierdzić, że ważone komitety niejednorodne skutecznie zastępują najlepszy komitet jednorodny.

Zakończenie

Opisane w niniejszej pracy eksperymentalne badanie metod klasyfikacji opartych na głosowaniu miało przede wszystkim doprowadzić do skonstruowania niejednorodnego komitetu klasyfikatorów, ale przyniosło też inne korzyści. Przede wszystkim w toku symulacji można było wielokrotnie przekonać się, że stopień zależności klasyfikatorów jest czynnikiem równie ważnym dla jakości klasyfikacji komitetu, jak indywidualny błąd składowego klasyfikatora[1]. Zarówno wśród zmodyfikowanych lasów losowych w rozdziale 2, jak i komitetów niejednorodnych z rozdziału 3 można znaleźć przykłady komitetów składających się z gorszych, ale słabiej zależnych klasyfikatorów, które popełniają niższy błąd klasyfikacji niż komitety klasyfikatorów lepszych, ale mocno skorelowanych. Jednocześnie trzeba pamiętać, że niska korelacja klasyfikatorów nie jest gwarantem dobrej jakości zbudowanego z nich komitetu, ważna jest relacja między zależnością klasyfikatorów a popełnianym przez nie błędem. Przekonano się, że w lasach losowych relację tę reguluje liczba losowanych zmiennych m — im jest wyższa, tym niższy indywidualny błąd klasyfikacji drzewa, ale jednocześnie większa zależność pomiędzy drzewami w lesie. Liczba zmiennych m minimalizująca błąd komitetu zależy oczywiście od całkowitej liczby zmiennych w badanym zbiorze, ale też od metody losowania zmiennych (jest około dwukrotnie większa w przypadku zmiennych losowanych raz dla całego drzewa niż w przypadku losowania zmiennych dla każdego węzła)[2].

Wpływ liczby losowanych zmiennych na jakość klasyfikacji komitetu jest analogiczny również przy zastosowaniu klasyfikatorów innego typu niż drzewa decyzyjne, co zostało wykorzystane przy budowie komitetu niejednorodnego. Zaproponowano komitet składający się z klasyfikatorów LDA, knn oraz drzew decyzyjnych, przy czym dla każdego typu klasyfikatora eksperymentalnie dobrano liczbę losowanych zmiennych[3]. Następnie poprawiono jakość klasyfikacji komitetu poprzez wprowadzenie wag, dzięki którym klasyfikatory o niskim błędzie zyskały większy wpływ na decyzję komitetu. Dzięki wagom nie ma też potrzeby ręcznego dostosowywania proporcji poszczególnych typów klasyfikatorów w komitecie do ich jakości klasyfikacji na badanym zbiorze[4]. Otrzymany komitet niejednorodny popełnia błąd klasyfikacji porównywalny z najlepszym komitetem jednorodnym (brano pod uwagę las losowy oraz jednorodne komitety klasyfikatorów LDA i knn), a zatem skutecznie zastępuje komitety jednorodne, dając jednocześnie pewne oszczędności obliczeniowe. Eliminuje bowiem etap trenowania poszczególnych komitetów jednorodnych i wyboru najlepszego spośród nich, a tym samym konieczność oceny ich jakości klasyfikacji. Zaproponowany komitet niejednorodny można zatem traktować jako gotowe narzędzie, które znacznie upraszcza proces klasyfikacji, a zarazem charakteryzuje się relatywnie niskim błędem klasyfikacji.

[1] Jakość klasyfikacji, a więc prawdopodobieństwo błędu klasyfikacji, szacowano w niniejszej pracy kroswalidacyjnie.

[2] Zob. podrozdział 2.1.1.

[3] Zob. podrozdział 3.2.

[4] Zob. podrozdział 3.3.

Dodatek A

Opis zbiorów danych

Podstawowe charakterystyki zbiorów wykorzystanych w pracy zgromadzono w tabeli A.1.

Nazwa zbioru	Liczba obserwacji	Liczba zmiennych	Liczba klas	Rozkład obserwacji w klasach
iris	150	4	3	50/50/50
crabsn	200	6	2	100/100
pima	532	7	2	355/177
ozone	1847	72	2	1719/128
ozone.bclass	256	72	2	128/128
ctgn	2126	21	3	1655/295/176
v20.eq	600	20	2	282/318
v10.eq	600	10	2	279/321
v4.eq	600	4	2	274/326
v20.dom	600	20	2	298/302
v4.dom	600	4	2	292/308
v5.coreq	600	5	2	298/302
v4.coreq	600	4	2	252/348
v4.cordom	600	4	2	295/305
mlbench.threenorm	600	10	2	300/300
mlbench.simplex	625	4	5	125/125/125/125/125
mlbench.xor	800	4	8	89/95/123/104/93/108/82/106
mlbench.smiley	1500	2	2	1125/375

Tabela A.1: Podstawowe cechy zbiorów wykorzystanych w pracy.

A.1. Zbiór `iris`

Zbiór `iris` jest dostępny razem z pakietem R. Spośród czterech zmiennych objaśniających trzy (`Sepal.Length, Petal.Length, Petal.Width`) są ze sobą bardzo silnie skorelowane. Klasy są stosunkowo łatwo separowalne, zwłaszcza klasa `setosa`, oznaczona na rysunku A.1 kolorem czarnym.

A.2. Zbiór `crabsn`

Zbiór `crabsn` powstał ze zbioru `crabs` dostępnego razem z pakietem R po usunięciu z niego zmiennej `index` (zmienna ta numerowała obserwacje pochodzące z jednej klasy, dlatego używanie jej w roli predyktora klasy nie miałoby sensu). Wszystkie zmienne w zbiorze są ze sobą bardzo silnie skorelowane. Klasy są łatwo liniowo separowalne, co pokazuje rysunek A.2.

A.3. Zbiór pima

Zbiór pima powstał z połączenia zbiorów Pima.tr i Pima.te dostępnych razem z pakietem R. Wśród zmiennych znajdują się silniej skorelowane pary (bmi-skin, age-npreg), ale pozostałe atrybuty są od siebie umiarkowanie lub słabo zależne. Granice między klasami w dwuwymiarowych przekrojach zbioru są dosyć niewyraźne (rys. A.3).

A.4. Zbiory ozone i ozone.bclass

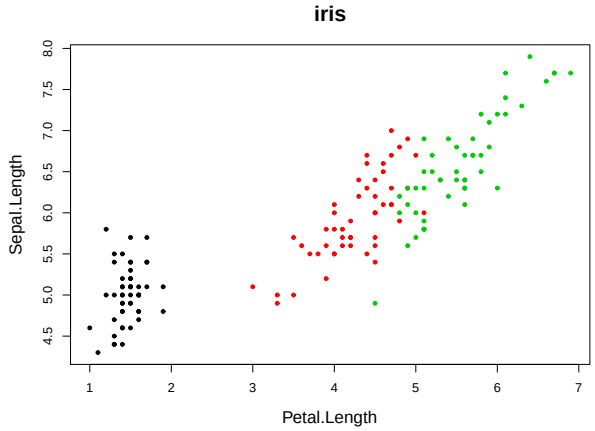
Zbiór ozone.bclass powstał na bazie zbioru danych ozone pochodzącego z repozytorium Uniwersytetu Kalifornijskiego w Irvine (*UCI Machine Learning Repository*). Z oryginalnego zbioru usunięto obserwacje o brakujących wartościach, a następnie wyrównano proporcje klas, losując odpowiednią liczbę obserwacji z większej klasy. Zbiór zawiera bardzo wiele zmiennych, wśród których można wyodrębnić słabo zależne od siebie grupy atrybutów bardzo silnie skorelowanych (zmienne T0-T70, zmienne WSR0-WSR23). Klasy są umiarkowanie dobrze separowalne, co widać na dwuwymiarowym przekroju zbioru na rysunku A.4.

A.5. Zbiór ctgn

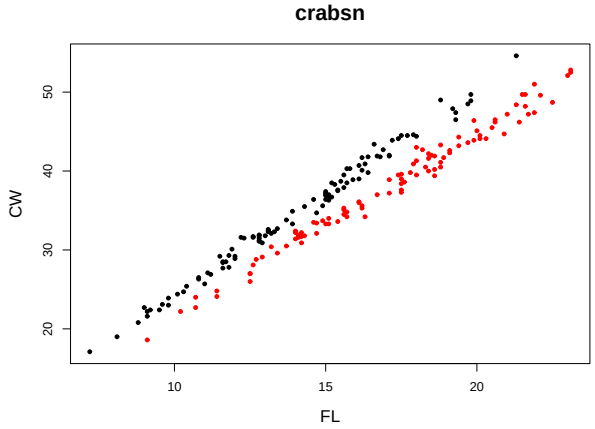
Zbiór ctgn powstał ze zbioru danych ctg dostępnego w repozytorium Uniwersytetu Kalifornijskiego w Irvine, ze zmiennych zawierających pomiary (LBE-Tendency, poza atrybutem DR, który przyjmował wartość 0 dla wszystkich obserwacji) oraz jednej spośród zmiennych zawierających etykiety obserwacji (NSP). Zmienne objaśniające są w większości słabo lub umiarkowanie skorelowane, poza silnie zależną parą LBE-LB oraz grupą Mean-Mode-Median. Zbiór charakteryzuje się dużą dysproporcją w liczebności klas. Granice pomiędzy klasami nie są ostre, ale stosunkowo dobrze widoczne nawet w płaszczyźnie dwóch spośród 21 zmiennych zbioru (rys. A.5).

A.6. Zbiory v20.eq, v10.eq i v4.eq

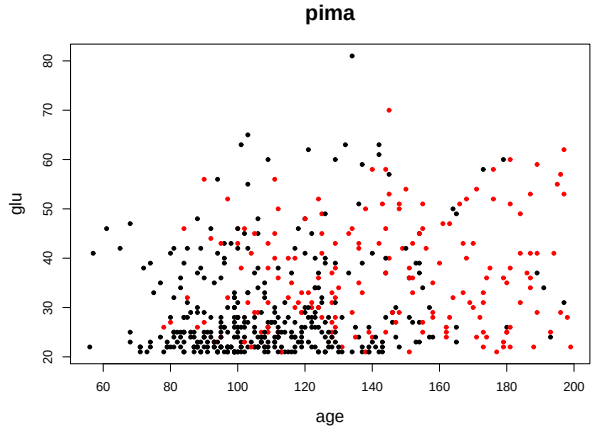
Zbiór v20.eq to zbiór syntetyczny, składający się z 20 zmiennych niezależnie wygenerowanych z rozkładu jednostajnego na odcinku $[0, 1]$ oraz zmiennej class zawierającej informację o klasie obserwacji. Etykiety były przydzielane w zależności od średniej wartości wszystkich zmiennych: średnia poniżej $\frac{1}{2}$ oznaczała przynależność do klasy 0, w przeciwnym wypadku przynależność do klasy 1. Wszystkie zmienne mają zatem jednakowy wpływ na przydział obserwacji do klasy. Do progu $\frac{1}{2}$ dodano zaburzenie losowe o rozkładzie normalnym $N(0, 0.01)$, aby klasy nie miały ostrych granic. Zbiory v10.eq i v4.eq powstały analogicznie, ale z odpowiednio mniejszej liczby zmiennych (dziesięciu i czterech). Ze względu na przyjętą regułę podziału obserwacji między klasy są one dobrze separowalne przez hiperpłaszczyznę, czego niestety nie da się pokazać na przekroju zbioru w płaszczyźnie dwóch zmiennych. Można jednak zilustrować rozkład klas posługując się sumami zmiennych, tak jak na rysunku A.6 (na osi poziomej zaznaczono sumę pierwszych 10 zmiennych, na osi pionowej sumę kolejnych 10 zmiennych zbioru).



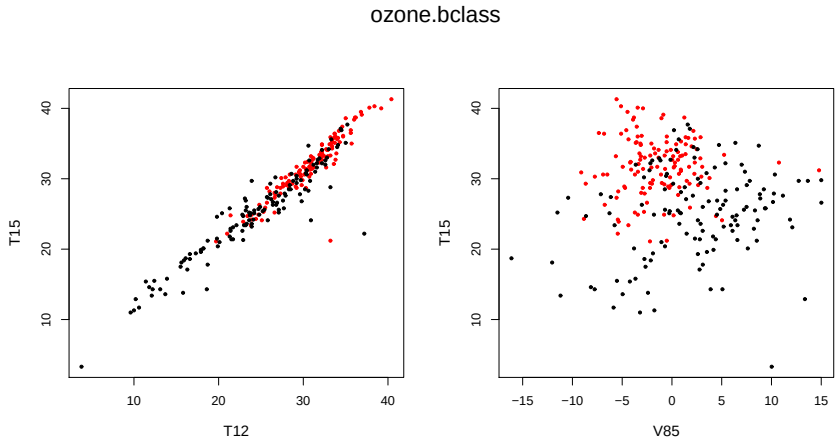
Rysunek A.1: Przekrój zbioru `iris` w płaszczyźnie zmiennych `Petal.Length` – `Sepal.Length`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



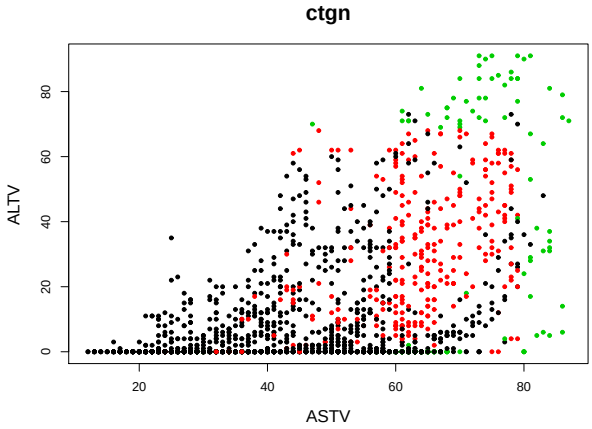
Rysunek A.2: Przekrój zbioru `crabsn` w płaszczyźnie zmiennych `FL` – `CW`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



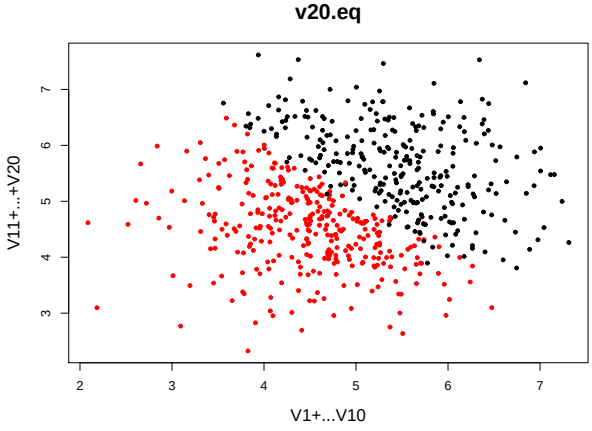
Rysunek A.3: Przekrój zbioru `pima` w płaszczyźnie zmiennych ; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



Rysunek A.4: Przekrój zbioru `ozone.bclass` w płaszczyźnie zmiennych `T12` – `T15`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



Rysunek A.5: Przekrój zbioru `ctgn` w płaszczyźnie zmiennych `ASTV` – `ALTV`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



Rysunek A.6: Przekrój zbioru `v20.eq` w płaszczyźnie `(V1+...+V10)` – `(V11+...+V20)`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.

A.7. Zbiory `v20.dom` i `v4.dom`

Zbiór `v20.dom` to również zbiór syntetyczny, podobnie jak `v20.eq` składa się z 20 zmiennych niezależnie wygenerowanych z rozkładu jednostajnego na odcinku $[0, 1]$, jednak tym razem przydział obserwacji do klasy oparto jedynie na wartości pierwszej zmiennej: jeśli miała ona wartość mniejszą niż $\frac{1}{2} + \epsilon$, obserwacja trafiała do klasy 0, w przeciwnym wypadku do klasy 1, gdzie ϵ był zaburzeniem losowym wygenerowanym z rozkładu normalnego $N(0, 0.01)$. Analogicznie zbudowano zbiór `v4.dom`, ale wykorzystując tylko cztery zmienne. W tym zbiorze tylko jedna zmienna ma wpływ na przydział obserwacji do klasy, co ilustruje rysunek A.7.

A.8. Zbiory `v5.coreq` i `v4.coreq`

Zbiór `v5.coreq` składa się z 5 syntetycznych zmiennych, które są ze sobą znacząco skorelowane (korelacja powyżej 50%). Zmienna $V1$ pochodzi z rozkładu jednostajnego $[0, 1]$, natomiast pozostałe są jej różnymi funkcjami: $V2 = 2V1 + \epsilon_1$, $V3 = 4 - 2V1 + \epsilon_2$, $V4 = V1^2 + \epsilon_3$, $V5 = \ln(V1+1) + \epsilon_4$, gdzie ϵ_1 i ϵ_2 zostały wygenerowane z rozkładu $N(0, 0.5)$, ϵ_3 z rozkładu $N(0, 0.2)$, a ϵ_4 z rozkładu $N(0, 0.1)$. Zbiory `v4.coreq` i `v4.cordom` powstały natomiast z pierwszych czterech zmiennych. Na zbiorach `v5.coreq` i `v4.coreq` przydziału do klas dokonywano na podstawie średniej wszystkich zmiennych dla danej obserwacji, przy czym próg podziału dobrano tak, aby otrzymane klasy miały zbliżoną liczebność. W związku z tym klasy są separowalne przez hiperpłaszczyznę, ale ze względu na mniejszą liczbę zmiennych granice między klasami są dostrzegalne nawet w przekroju zbioru w płaszczyźnie dwóch wybranych zmiennych (rys. A.8). Obserwacje zbioru `v4.cordom` zostały natomiast przydzielone do klas jedynie na podstawie wartości pierwszej zmiennej, tak jak to zrobiono w przypadku zbioru `v20.dom` (przyjęto próg równy $\frac{1}{2} + \epsilon$, gdzie $\epsilon \sim \mathrm{N}(0, 0.2)$).

A.9. Zbiór `mlbench.threenorm`

Zbiór `mlbench.threenorm` powstał przy pomocy funkcji mlbench.threenorm z pakietu mlbench. Funkcja ta generuje punkty z trzech d-wymiarowych rozkładów normalnych o wariancji równej 1 i średnich w punktach $(a, a, \ldots, a)$, $(-a, -a, \ldots, -a)$ i $(a, -a, a, \ldots, a)$, przy czym $a = 2/\sqrt{(d)}$. W przypadku zbioru wykorzystanego w pracy $d = 10$, w związku z tym zbiór zawiera dziesięć zmiennych objaśniających. Obserwacje dzielone są pomiędzy dwie klasy: punkty z dwóch pierwszych rozkładów należą do klasy 0, natomiast punkty z trzeciego rozkładu przydzielane są do klasy 1. Zmienne w zbiorze są ze sobą słabo skorelowane. Granica między klasami nie jest ostra i słabo widoczna w płaszczyznach dwóch zmiennych, ale nieco wyraźniej widać ją na przekroju względem kombinacji liniowej zmiennych, co pokazuje rysunek A.9.

A.10. Zbiór `mlbench.simplex`

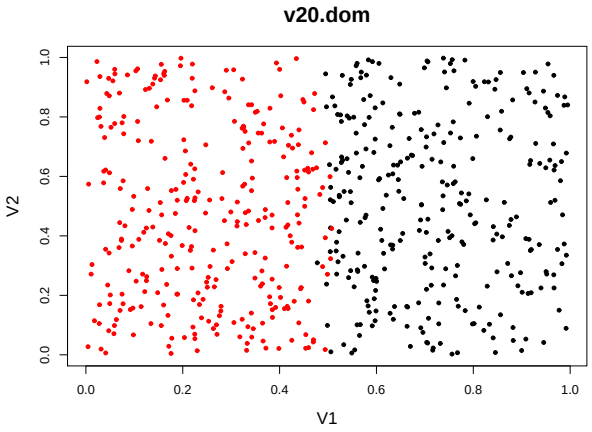
Zbiór `mlbench.simplex` również został wygenerowany przez funkcję o tej samej nazwie pochodzącej z pakietu mlbench. Obserwacje w zbiorze pochodzą z pięciu 4-wymiarowych rozkładów normalnych, których średnie znajdują się w rogach 4-wymiarowego sympleksu, natomiast odchylenie standardowe wynosi $0, 2$. Punkty z każdego z pięciu rozkładów należą do innej klasy. Granice między nimi są nieostre i dość słabo widoczne w przekrojach płaszczyznami dwóch zmiennych, ale odpowiednio dobrane kombinacje zmiennych nieco pomagają w wyodrębnieniu wszystkich klas (rys. A.10).

A.11. Zbiór `mlbench.xor`

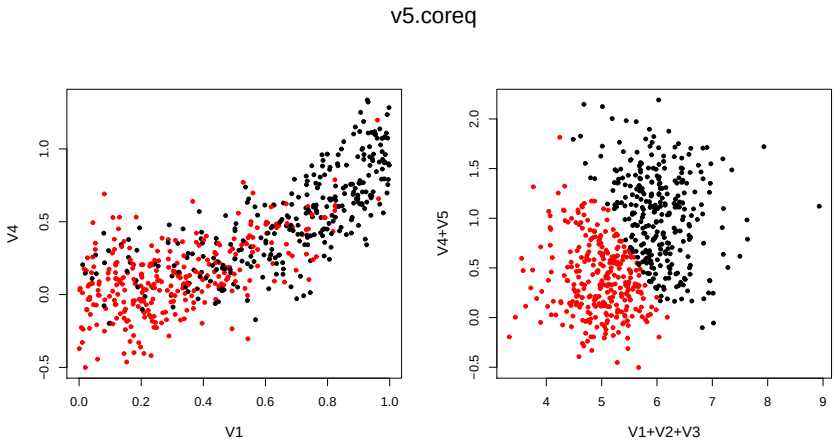
Do wygenerowania zbioru `mlbench.xor` użyto funkcji o tej samej nazwie z pakietu mlbench. Obserwacje są losowane z rozkładu jednostajnego na 4-wymiarowej kostce o boku $[-1, 1]$. Kostka podzielona jest płaszczyznami `Vi`$= 0$, $i = 1, 2, 3, 4$, a punkty znajdujące się w jej przeciwległych rogach należą do tej samej klasy (w przypadku kostki 4-wymiarowej takich klas jest 8). Klasy są więc bardzo dobrze separowalne płaszczyznami `Vi`$= 0$, ale ze względu na łączenie w jedną klasę obserwacji z przeciwległych rogów jest to praktycznie niemożliwe do zilustrowania na dwuwymiarowym rysunku. Rysunek A.11 pokazuje przekrój w płaszczyźnie dwóch wybranych zmiennych oraz próbę wyodrębnienia większej liczby klas poprzez przekrój "po przekątnej" kostki.

A.12. Zbiór `mlbench.smiley`

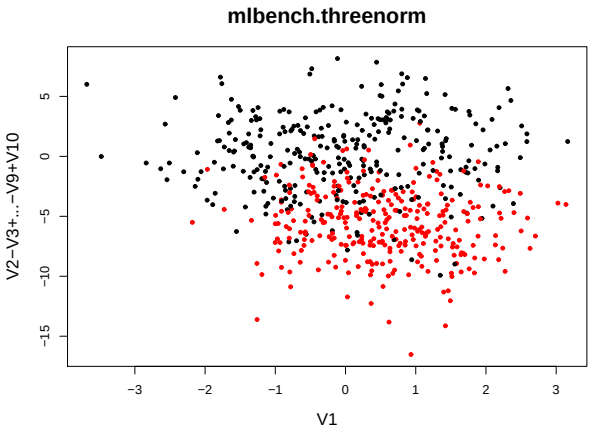
Syntetyczny zbiór `mlbench.smiley` również wygenerowano przy użyciu funkcji o tej samej nazwie z pakietu mlbench. Obserwacje tworzące "oczy" wylosowano z dwóch dwuwymiarowych rozkładów normalnych o odchyleniu standardowym równym $0,3$, "nos" powstał z obserwacji losowanych z rozkładu jednostajnego na trapezie, natomiast "usta" tworzą punkty wygenerowane z rozkładu jednostajnego na paraboli z pionowym gaussowskim zaburzeniem losowym o odchyleniu standardowym równym $0,3$. Punkty wchodzące w skład "oczu" i "ust" połączono w jedną klasę, drugą stanowią obserwacje tworzące "nos". Klasy są dobrze separowalne, co pokazuje rysunek A.12.



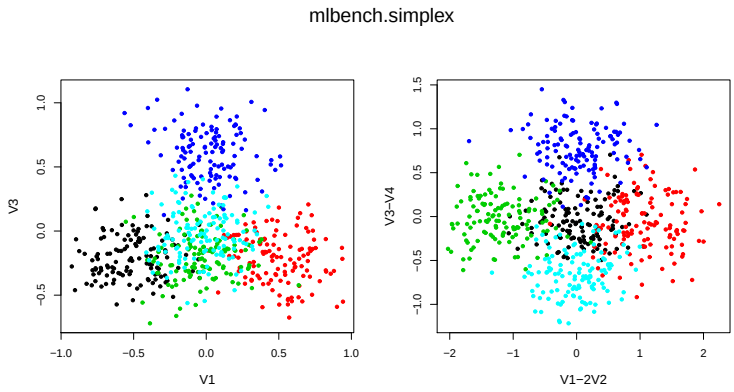
Rysunek A.7: Przekrój zbioru `v20.dom` w płaszczyźnie zmiennych `V1` – `V2`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



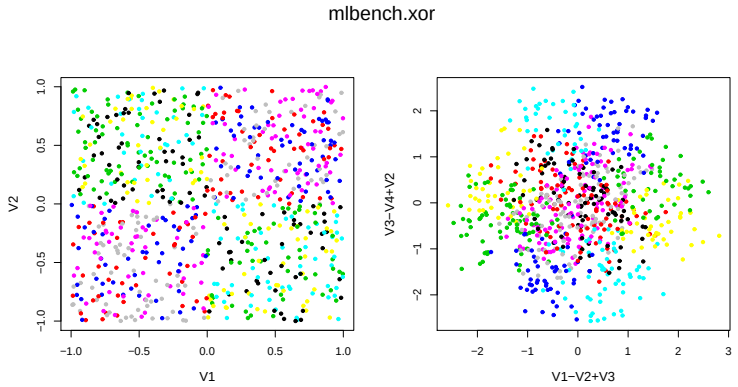
Rysunek A.8: Przekrój zbioru `v5.coreq` w płaszczyźnie zmiennych `V1` – `V4` oraz w płaszczyźnie `(V1+V2+V3)` – `(V4+V5)`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



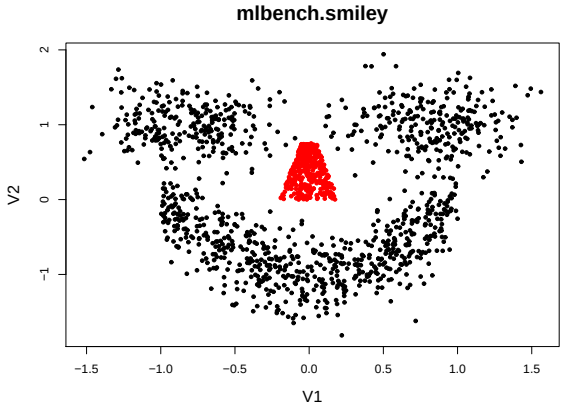
Rysunek A.9: Przekrój zbioru `mlbench.threenorm` w płaszczyźnie `V1` – $\sum_{i=2}^{10}(-1)^i$`Vi`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



Rysunek A.10: Przekrój zbioru `mlbench.simplex` w płaszczyźnie `V1` – `V2` oraz `(V1-2V2)` – `(V3-V4)`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



Rysunek A.11: Przekrój zbioru `mlbench.xor` w płaszczyźnie zmiennych `V1` – `V2` oraz `(V1-V2+V3)` – `(V3-V4+V2)`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.



Rysunek A.12: Zbiór `mlbench.smiley` na płaszczyźnie zmiennych `V1` – `V2`; jednakowym kolorem oznaczono obserwacje z tej samej klasy.

Bibliografia

[1] Leo Breiman, *Random Forests*, „Machine Learning", t. 45 (2001), nr 1, s. 5-32;

[2] Yoav Freund, Robert E. Schapire, *Game Theory, On-line Prediction and Boosting*, „Proceedings of the Ninth Annual Conference on Computational Learning Theory", 1996;

[3] Yoav Freund, Robert E. Schapire, *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*, „Journal of Computer and System Sciences", nr 55 (1997), s. 119-139;

[4] Jacek Koronacki, Jan Ćwik, *Statystyczne systemy uczące się*, Wydawnictwa Naukowo-Techniczne, Warszawa 2005;

[5] Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer, 2008;