AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

DZIEDZINA NAUK INŻYNIERYJNO-TECHNICZNYCH

DYSCYPLINA INŻYNIERIA MECHANICZNA

ROZPRAWA DOKTORSKA

Akustyczna detekcja schorzeń krtani i zaburzeń neurorozwojowych z zastosowaniem x-wektorów

Autor: Katarzyna Aleksandra Kotarba

Promotor rozprawy: prof. dr hab. inż. Piotr Kleczkowski

Praca wykonana w: Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie, Wydziale Inżynierii Mechanicznej i Robotyki, Katedrze Mechaniki i Wibroakustyki

Kraków, 2023

AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY

FIELD OF SCIENCE ENGINEERING AND TECHNOLOGY

SCIENTIFIC DISCIPLINE MECHANICAL ENGINEERING

DOCTORAL THESIS

Acoustical detection of otolaryngeal diseases and neurodevelopmental disorders using x-vectors approach

Author: Katarzyna Aleksandra Kotarba

First supervisor: prof. dr hab. inż. Piotr Kleczkowski

Completed in: AGH University of Science and Technology, Faculty of Mechanical Engineering and Robotics, Department of Mechanics and Vibroacoustics

Krakow, 2023

Uprzedzona o odpowiedzialności karnej na podstawie art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.): „Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystycznego wykonania albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, wideogram lub nadanie.", a także uprzedzona o odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (t.j. Dz. U. z 2012 r. poz. 572, z późn. zm.): „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności studenta student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej «sądem koleżeńskim».", oświadczam, że niniejszą pracę dyplomową wykonałam osobiście i samodzielnie i że nie korzystałam ze źródeł innych niż wymienione w pracy.

Badania prowadzone w ramach prac opisanych w niniejszej rozprawie współfinansowano ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój 2014-2020 Oś III Szkolnictwo wyższe dla gospodarki i rozwoju, Dizłanie 3.5 Kompleksowe programy szkół wyższych Zintegrowany Program Rozwoju Akademii Górniczo-Hutniczej w Krakowie nr POWR.03.05.00-00-Z307/17-00

Składam serdeczne podziękowania mojemu promotorowi,
prof. dr hab. inż. Piotrowi Kleczkowskiemu,
za nieocenione wsparcie, motywację i życzliwość
udzielone mi podczas prowadzenia badań i pracy nad tą rozprawą.
Pragnę również podziękować moim współpracownikom
z firmy StatSoft Polska, Annie Wilk i dr Januszowi Wątrobie,
za cenne uwagi i porady dotyczące opisanych w pracy analiz.

Spis treści

Streszczenie

W niniejszej rozprawie przedstawiono oryginalną metodę automatycznej diagnostyki wybranych schorzeń otolaryngologicznych oraz zaburzeń neurorozwojowych na podstawie sygnału mowy z zastosowaniem x-wektorów – biometrycznych wektorów wykorzystywanych do identyfikacji mówcy. Zaproponowano modyfikacje architektury sieci neuronowej służącej do ekstrakcji x-wektorów oraz procesu jej uczenia, które miały na celu zwiększenie zdolności generalizacyjnych oraz dokładności klasyfikacji zastosowanych modeli klasyfikacyjnych dokonujących detekcji wspomnianych schorzeń i zaburzeń. Wartości metryk sukcesu wyznaczone podczas ewaluacji proponowanych klasyfikatorów uczonych na poszczególnych wariantach x-wektorów poddano analizie statystycznej mającej na celu zbadanie istotności statystycznej różnic pomiędzy wartościami uzyskanymi przez poszczególne modele oraz wyłonienie rozwiązania zapewniającego najlepsze rezultaty. Przedstawione w pracy analizy wykazały, że proponowane rozwiązanie umożliwia detekcję wybranych schorzeń krtani oraz zaburzeń neurorozwojowych z dokładnością przewyższającą wyniki prezentowane w literaturze. Co ważne, zaproponowana metoda jest metodą uniwersalną, prowadzącą do uzyskania satysfakcjonujących rezultatów bez względu na wypowiadaną treść, płeć i wiek mówcy, a także język, którym się on posługuje.

Abstract

In the following thesis a method of automatic detection of selected otolaryngeal diseases and neurodevelopmental disorders based on speaker embeddings extracted from speech signal, called x-vectors, is presented. Modifications of the architecture of a neural network used for x-vectors extraction and its training process are proposed. These modifications were performed in order to increase generalization abilities and accuracy of classification models used for the detection of the aforementioned medical conditions. Statistical analysis of evaluation metrics obtained by models trained on proposed x-vectors variants was performed in order to examine statistical significance of differences in values obtained by these models and to select the solution leading to the best results. Presented analysis proved that the proposed approach allows for the detection of selected otolaryngeal diseases and neurodevelopmental disorders with higher accuracy than methods described in literature. Importantly, the proposed method is text-independent and universal, i.e. leads to satisfactory results irrespective of speaker's age, sex and language they speak.

Spis tabel

Spis rysunków

Spis skrótów i oznaczeń

ADASYN - ang. *adaptive synthetic algorithm*, adaptacyjny algorytm generowania obserwacji syntetycznych
ASD - ang. *autism spectrum disorder*, zaburzenia ze spektrum autyzmu
AUC PR - ang. *area under curve precision-recall*, pole pod krzywą precyzja-czułość
AUC ROC - ang. *area under receiver operating characteristic*, pole pod krzywą ROC
DNN - ang. *deep neural network*, głęboka sieć neuronowa
EER - ang. *equal error rate*, metryka wykorzystywana z rozpoznawaniu mowy, oznacza wartość błędu przy jednakowej częstotliwości popełnianai błędu I i II rodzaju
GFCC - ang. *gammatone freauency cepstral coefficients*, współczynniki cepstralne wyznaczane przy użyciu banku filtrów gammatone
ICA - ang. *independent components analysis*, analiza składowych niezależnych
KernelPCA - ang. *kernel principal component analysis*, jądrowa analiza składowych głównych
KNN - ang. *k nearest neighbours*, metoda k najbliższych sąsiadów
LDA - ang. *linear discriminant analysis*, liniowa analiza dyskryminacyjna
LR - ang. *logistic regression*, regresja logistyczna
MFCC - ang. *mel-frequency cepstral coefficients*, melowo-częstotliwościowe współczynniki cepstralne
MLP - ang. *multi-layer perceptron*, perceptron wielowarstwowy
PCA - ang. *principal component analysis*, analiza składowych głównych
QDA - ang. *quadratic distriminant analysis*, kwadratowa analiza dyskryminacyjna
RF - ang. *random forest*, metoda lasów losowych
RNN - ang. *radius nearest neighbours*, wariant metody najbliższych sąsiadów
SGD - ang. *stochastic gradient descent*, stochastyczne zejście gradientowe
SLI - ang. *specific language impairment*, specyficzne zaburzenia językowe
SMOTE - ang. *synthetic minority over-sampling technique*
SVM - ang. *support vector machine*, maszyna wektorów nośnych
TruncatedSVD - ang. *truncated singular value decomposition*, algorytm rozkładu według wartości osobliwych

1. Wstęp

Mowa towarzyszy człowiekowi w życiu codziennym na każdym etapie życia – od etapu życia płodowego aż do śmierci. Służy przede wszystkim do komunikacji, jest też jednak źródłem innych informacji analizowanych przez człowieka – pozwala ocenić wiek, płeć i stan emocjonalny rozmówcy oraz wyrazić własny, a mózg człowieka wykształcił odpowiednie mechanizmy umożliwiające możliwie sprawne i skuteczne przetwarzanie sygnału mowy i jego interpretację.

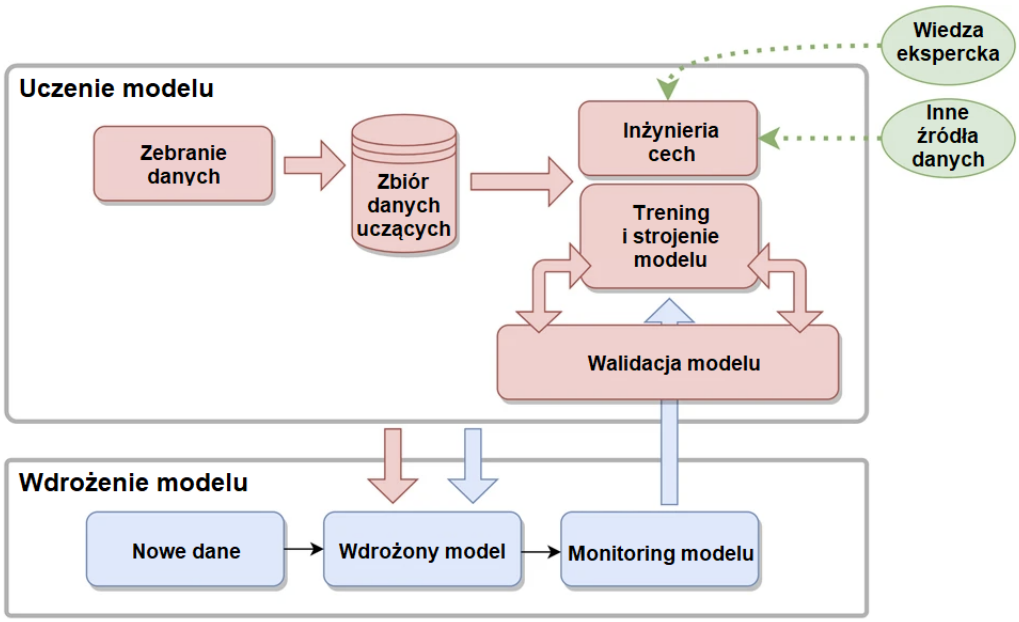
Idea wykorzystania sygnału mowy jako informacji diagnostycznej stosowanej w systemach automatycznej diagnostyki medycznej pojawiła się już w momencie pojawienia się odpowiednich rozwiązań sprzętowych umożliwiających rejestrację dźwięku, automatyczną parametryzację sygnału oraz zastosowanie algorytmów klasyfikacyjnych. Początkowo stosowane były przede wszystkim proste w implementacji algorytmy, niewymagające dużych nakładów obliczeniowych. Wraz z rozwojem technologicznym i wzrostem mocy obliczeniowej komputerów zarówno proces przetwarzania sygnału, jego parametryzacji, jak i następnej klasyfikacji mającej na celu wykrycie różnego rodzaju schorzeń mogły być coraz bardziej skomplikowane i wymagające.

Osiągnięcie odpowiednio wysokiej dokładności klasyfikacji, pozwalającej na skuteczną diagnostykę schorzeń oraz upowszechnienie komputerów i innych urządzeń elektronicznych rozpoczęło erę automatyzacji procesów diagnostycznych i wdrażania tych rozwiązań do praktyki klinicznej. Coraz nowocześniejsze i bardziej zaawansowane metody obrazowania medycznego oraz rejestracji i analizy sygnałów medycznych pozwalają na coraz szybsze i dokładniejsze diagnozowanie problemów zdrowotnych. W przypadku wielu z tych metod problematyczna pozostaje natomiast dostępność aparatury medycznej oraz koszty jej eksploatacji. Sygnały diagnostyczne możliwe do zarejestrowania w gabinecie lekarskim niewyposażonym w kosztowne, zaawansowane aparaty lub wręcz w warunkach domowych stanowią więc obszar zainteresowania wielu badaczy zajmujących się algorytmami stosowanymi w systemach automatycznej diagnostyki medycznej. Znalezienie rozwiązań umożliwiających skuteczną diagnostykę przy użyciu ogólnodostępnych urządzeń elektronicznych stanowiłoby dużą wartość w szczególności w przypadku obszarów słabo zurbanizowanych, znajdujących się w znacznej odległości od dużych placówek medycznych wyposażonych w odpowiedni sprzęt oraz odpowiednio wykształconą kadrę lekarską. Kwestia dostępności usług lekarskich stała się szczególnie istotna w ostatnim czasie, gdy możliwość odbycia wizyty lekarskiej i podjęcia odpowiednich środków terapeutycznych została znacząco ograniczona z powodu pandemii COVID-19. Wydarzenie to spowodowało jeszcze większy wzrost zainteresowania telemedycyną – dziedziną zajmująca się

diagnostyką medyczną prowadzoną na odległość. Standardowe teleporady, polegające na rozmowie lekarza z pacjentem i próbie postawienia diagnozy w oparciu o wywiad lekarski mogą być jednak stosunkowo mało skuteczne. Ich efektywność i skuteczność zależy przede wszystkim od jakości przeprowadzonego wywiadu, umiejętności opisania doświadczanych objawów przez pacjenta oraz jego samoświadomości dotyczącej stanu zdrowia. Jeszcze większym problemem jest sytuacja, w której pacjent ma ograniczone możliwości komunikacyjne z przyczyn wynikających z występujących zaburzeń, wieku, czy też bariery językowej.

Jedynym sygnałem diagnostycznym możliwym do wykorzystania w takich warunkach jest sygnał mowy. Kluczowe staje się więc rozwinięcie metod telemedycznych umożliwiających wsparcie procesu diagnostycznego z użyciem tego sygnału. W tym celu konieczne jest opracowanie nie tylko modeli klasyfikacyjnych dokonujących detekcji schorzeń, ale też odpowiedni wybór cech ekstrahowanych z sygnału mowy oraz algorytmów jest wstępnego przetwarzania. Stosowane techniki powinny zapewnić wysoką wiarygodność zwracanej przez model informacji prognostycznej. Powinny też umożliwić analizę sygnału zarejestrowanego w warunkach niekorzystnych, z użyciem urządzeń dokonujących rejestracji ze stosunkowo niewielką częstotliwością próbkowania, w warunkach wysokiego tła akustycznego. Ponadto istotna jest też kwestia dotycząca samego sygnału mowy – brak ograniczeń dotyczących długości i treści wypowiadanych przez pacjenta może znacząco zwiększyć możliwości wdrożenia opracowanego systemu do praktyki klinicznej. Co więcej, istotna jest możliwość przełożenia wyników osiągniętych z użyciem mowy w jednym języku na inne języki. Nie tylko umożliwiłoby to szybszy rozwój odpowiednich metod, ale też wiązałoby się z mniejszymi kosztami związanymi z ich opracowaniem – jednym z głównych problemów związanych z analizą danych medycznych jest stosunkowo mała liczba dostępnych baz danych, które mogą posłużyć do prac nad rozwojem systemów automatycznej diagnostyki medycznej.

Proces wdrażania rozwiązań wykorzystujących sztuczną inteligencję do praktyki medycznej można podzielić na dwa główne etapy: etap uczenia modelu oraz etap jego wdrożenia (rys. 1.1). W przypadku pierwszego z etapów niezbędny jest dostęp do danych umożliwiających opracowanie skutecznego modelu klasyfikacyjnego. W przypadku diagnostyki opartej o sygnał mowy są to korpusy mowy zawierające mowę osób zdrowych oraz ze zdiagnozowanym konkretnym schorzeniem. Zastosowanie metod niezależnych od wypowiadanej treści i użytego przez mówcę języka umożliwia równoczesne wykorzystanie korpusów mowy w różnych językach, co sprawia, że liczebność danych możliwych do wykorzystania w procesie treningu i ewaluacji modeli jest większa niż w przypadku metod zależnych od treści i języka. Etap wdrożenia modelu do praktyki klinicznej musi być poprzedzony wyczerpującą ewaluacją proponowanego rozwiązania – konieczny jest właściwy dobór metod parametryzacji sygnału, rodzaj modelu klasyfikacyjnego i przeprowadzenie jego strojenia. Nie bez znaczenia jest również odpowiednia technika ewaluacji – dobór metryk sukcesu dających szeroki ogląd możliwości generalizacyjnych modelu, jego stabilności oraz odtwarzalność całego procesu, wpływająca na możliwość uzyskania zbliżonej skuteczności klasyfikacji w przypadku pozyskania nowych danych.



Rys. 1.1. Schemat procesu wdrożenia modelu klasyfikacyjnego do praktyki klinicznej. Źródło: [1].

W niniejszej pracy podjęto się realizacji pierwszego etapu wdrażania systemu automatycznej diagnostyki medycznej mającego posłużyć do diagnostyki schorzeń otolaryngologicznych oraz zaburzeń neurorozwojowych na podstawie sygnału mowy. Skupiono się na budowie systemu przetwarzania i parametryzacji sygnałów z użyciem metod, które pozwolą na prowadzenie klasyfikacji bez względu na wiek i płeć pacjenta, język, w którym mówi oraz schorzenie, na które cierpi. Ponadto przeprowadzono rozbudowaną, wyczerpującą ewaluację proponowanych modeli klasyfikacyjnych w celu zagwarantowania maksymalnej odtwarzalności i wiarygodności przedstawianych rezultatów.

1.1. Cel pracy

Badania przedstawione w niniejszej pracy miały na celu zbadanie możliwości zastosowania x-wektorów – jednej z najnowszych wektorowych reprezentacji mówcy stosowanych w zagadnieniach identyfikacji mówcy – w systemach automatycznej diagnostyki schorzeń otolaryngologicznych oraz zaburzeń neurorozwojowych. Aby zrealizować założony cel przeprowadzono następujące prace badawcze:

– zbadano wpływ cech użytych do ekstrakcji x-wektorów na zdolności diagnostyczne modeli w wybranych zagadnieniach klasyfikacyjnych,

– zmodyfikowano architekturę sieci neuronowej wykorzystywanej do ekstrakcji x-wektorów oraz proces jej treningu w celu zwiększenia skuteczności detekcji wybranych schorzeń otolaryngologicznych i zaburzeń neurorozwojowych,

– zoptymalizowano hiperparametry modeli klasyfikacyjnych oraz architektury komitetów klasyfikatorów służących do detekcji rozważanych schorzeń i zaburzeń,

– zbadano wpływ wybranych metod przepróbkowania zbiorów danych na jakość modeli klasyfikacyjnych,

– przeanalizowano istotność statystyczną różnic pomiędzy wynikami uzyskiwanymi poprzez zastosowanie zaproponowanych wariantów x-wektorów,

– porównano uzyskane wyniki z rezultatami opisanymi w literaturze.

1.2. Teza pracy

Realizacja opisanego w rozdziale 1.1 celu badawczego miała na celu sprawdzenie prawdziwości założonej tezy pracy, którą sformułowano w sposób następujący:
Wektory własnościowe mówcy umożliwiają detekcję wybranych schorzeń otolaryngologicznych oraz zaburzeń neurorozwojowych na podstawie mowy ciągłej z wyższą skutecznością niż wcześniej stosowane metody.

1.3. Struktura pracy

W rozdziale 2 opisano zastosowane w pracy algorytmy i metody klasyfikacyjne, przedstawiono ich zasadę działania oraz występujące pomiędzy nimi różnice.

W rozdziale 3 streszczono dostępne w literaturze rozwiązania dotyczące diagnostyki schorzeń otolaryngologicznych i zaburzeń neurorozwojowych na podstawie sygnału mowy oraz dotychczas zaprezentowane zastosowania x-wektorów w diagnostyce medycznej.

Rozdział 4 przestawia wykorzystane podczas prac badawczych korpusy mowy: korpus mowy niemieckiej *Saarbrücken Voice Database* (SVD), zawierający mowę osób cierpiących na szerokie spektrum schorzeń otolaryngologicznych, korpus mowy czeskiej zawierający mowę dzieci ze zdiagnozowanymi specyficznymi zaburzeniami językowymi oraz korpus mowy niderlandzkiej zawierający mowę dzieci ze zdiagnozowanymi zaburzeniami ze spektrum autyzmu.

W rozdziale 5 zaprezentowano schemat prowadzonych prac badawczych, zaproponowano modyfikacje architektury sieci neuronowej wykorzystywanej do ekstrakcji x-wektorów oraz opisano wyniki przeprowadzonych klasyfikacji sygnałów. Następnie przeprowadzono analizę istotności statystycznej różnic pomiędzy skutecznością modeli klasyfikacyjnych uczonych na różnych wariantach x-wektorów. Wyłoniono również modele klasyfikacyjne służące do detekcji rozważanych w pracy jednostek chorobowych, których rezultaty uznano zgodnie z przyjętym kryterium za najbardziej obiecujące oraz przedstawiono sposób ich ewaluacji.

W rozdziale 6 przeprowadzono dyskusję dotyczącą otrzymanych wyników oraz ich znaczenia praktycznego.

Rozdział 7 zawiera podsumowanie przeprowadzonych prac badawczych.

2. Algorytmy i narzędzia wykorzystane w pracy

Dobór algorytmów wykorzystywanych w systemie automatycznej diagnostyki medycznej ma kluczowe znaczenie. W zależności od użytego algorytmu rezultaty mogą być diametralnie różnie – wybór modelu klasyfikacyjnego ma wpływ na skuteczność klasyfikacji, technika ekstrakcji cech i redukcji wymiarowości może wpływać na stabilność uzyskanego modelu, a wybór algorytmu optymalizacyjnego decyduje o złożoności czasowej i obliczeniowej prowadzonych analiz oraz decyduje o jakości otrzymanego modelu. Wybór algorytmów używanych podczas poszczególnych etapów przetwarzania i analizy danych powinien więc być przemyślany i uwzględniać rodzaj i ilość analizowanych danych. W przypadku prac badawczych, gdy nie ma dostępnych rekomendacji wskazujących właściwy wybór tych algorytmów, należy wypróbować możliwie szerokie spektrum rozwiązań w celu wyłonienia najbardziej adekwatnego do rozwiązania problemu będącego przedmiotem badań.

Podczas prowadzenia prac badawczych opisanych w niniejszej rozprawie wykorzystano liczne algorytmy, które można podzielić następująco:

- algorytmy ekstrakcji cech służące do parametryzacji sygnałów poddawanych analizie,
- modele klasyfikacyjne – grupa algorytmów służących do rozwiązania zagadnienia klasyfikacji, czyli przyporządkowania analizowanych obiektów do odpowiednich klas reprezentujących dystynktywne zgrupowania tych obiektów,
- algorytmy optymalizacyjne – metody obliczeniowe służące do wyznaczenia wartości hiperparametrów modeli klasyfikacyjnych, które umożliwią uzyskanie jak największej skuteczności klasyfikacji,
- metody redukcji wymiarowości – algorytmy służące do zredukowania liczby cech opisujących pojedynczy obiekt należący do zbioru danych,
- metody przepróbkowania zbioru danych – techniki służące do zmniejszenia problemu niezrównoważenia liczebności klas tworzących analizowany zbiór danych,
- testy statystyczne służące do porównania skuteczności zbudowanych modeli klasyfikacyjnych i zbadania istotności statystycznej różnic w uzyskiwanych przez nie wartościach metryk sukcesu.

W dalszej części rozdziału przybliżono podstawy teoretyczne technik należących do poszczególnych grup algorytmów wykorzystanych w niniejszej pracy.

2.1. Modele klasyfikacyjne

Algorytmy uczenia maszynowego są grupą algorytmów, którą potrafią uczyć się rozpoznawać wzorce na podstawie dostarczonego zbioru danych. Stosowane są z reguły w przypadku zadań, które są zbyt skomplikowane, by było możliwe znalezienie ich rozwiązania przez człowieka. Wśród tych zadań można wyróżnić wiele kategorii, np. [2]:

– klasyfikację – zadanie polegające na przyporządkowaniu obiektu do jednej z k kategorii zwanych klasami,

– regresję – zadanie polegające na przewidywaniu wartości liczbowej, którą przyjmie analizowany parametr na podstawie danych wejściowych,

– detekcję anomalii – zadanie polegające na wykryciu zdarzeń lub obiektów nietypowych, różniących się znacząco od pozostałych obiektów tworzących zbiór danych,

– syntezę – zadanie polegające na generowaniu nowych obiektów o określonych cechach, przypominających obiekty należące do zbioru uczącego.

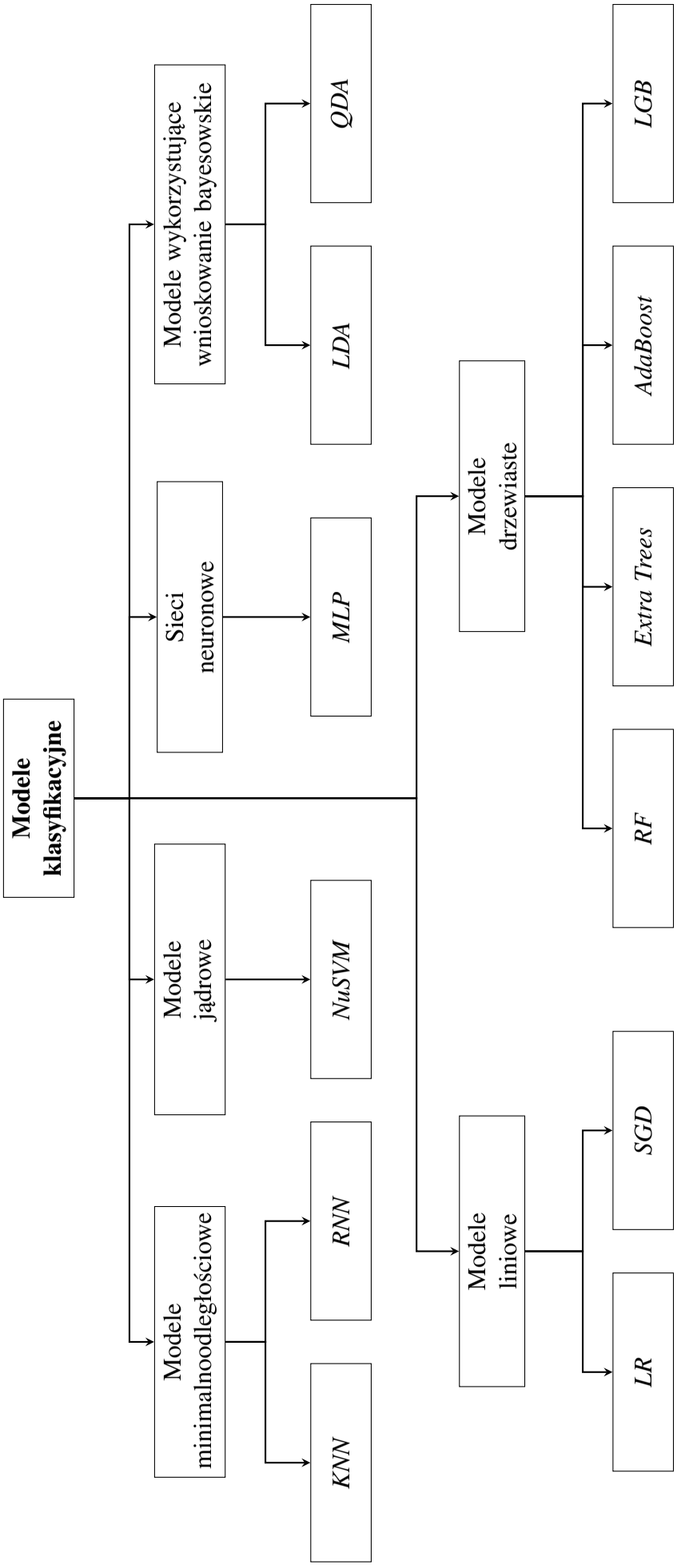
W niniejszej pracy skupiono się na zagadnieniu klasyfikacji – w tym celu zastosowano szereg modeli realizujących przyporządkowanie obiektów do odpowiednich grup. Modele te różniły się między sobą zasadą działania, stopniem skomplikowania struktury oraz złożonością obliczeniową. Na rys. 2.1 przedstawiono schemat podziału zastosowanych metod klasyfikacyjnych.

2.1.1. Modele minimalnoodległościowe

Modele minimalnoodległościowe to grupa modeli oceniających podobieństwo obiektów na podstawie ich odległości wyliczonej zgodnie z przyjętą metryką odległości – obiekty leżące blisko siebie, dla których wyznaczona odległość przyjmuje niewielkie wartości, uznawane są za podobne do siebie, zaś obiekty, dla którym obliczona odległość przyjmuje duże wartości uznawane są za różniące się od siebie. Tak określona miara podobieństwa służy do oceny podobieństwa nowych obiektów do obiektów należących do zbioru uczącego oraz przeprowadzenia jego klasyfikacji.

2.1.1.1. Klasyfikator *k*-najbliższych sąsiadów

Klasyfikator k-najbliższych sąsiadów (ang. *k nearest neighbours classifier*, KNN) jest jednym z najstarszych i najprostszych modeli klasyfikacyjnych. Jego działanie polega na przyporządkowaniu obiektu pochodzącego z nieznanej klasy do klasy wyznaczanej na podstawie jego najbliższych sąsiadów, czyli obserwacji ze zbioru uczącego znajdujących się w najmniejszej odległości od niego. Nowy obiekt przyporządkowany zostaje do tej samej klasy, do której należy większość z k jego najbliższych sąsiadów [3].



Rys. 2.1. Podział algorytmów klasyfikacyjnych wykorzystanych w pracy.

Rozważmy model klasyfikacyjny, w którym klasyfikacja nowego obiektu odbywa się w oparciu o przynależność klasową jednego najbliższego sąsiada (k=1). Sąsiad ten wyznaczany jest zgodnie z zależnością:

$$NN(q, P) = argmin_{x \in P} d(q, x) \tag{2.1}$$

gdzie $P = \{p_1, p_2, \ldots, p_n\}$ jest zbiorem obiektów uczących w przestrzeni metrycznej M, q oznacza nowy obiekt, którego najbliższy sąsiad jest poszukiwany, zaś d oznacza odległość nowego obiektu od rozważanego sąsiada wyznaczoną zgodnie z przyjętą metryką odległości [4].

Uogólniając powyższy zapis na przypadek, w którym rozważana jest liczba sąsiadów większa niż jeden ($k \geq 2$) możemy zapisać:

$$kNN(q, P, k) = A \tag{2.2}$$

$$|A| = k, A \subseteq P \tag{2.3}$$

$$\forall x \in A, y \in P - A, d(q, x) \leq d(q, y) \tag{2.4}$$

gdzie k oznacza liczbę sąsiadów, zaś x i y oznaczają obserwacje należące do zbioru uczącego [4].

Wyniki klasyfikacji prowadzonej przy użyciu modelu KNN zależne są od wybranej miary odległości oraz wartości parametru k oznaczającego liczbę rozważanych najbliższych sąsiadów – nieprawidłowy dobór wartości tych hiperparametrów może doprowadzić do uzyskania niesatysfakcjonujących rezultatów. Do wad metody można zaliczyć również fakt, że podczas klasyfikacji nowego obiektu konieczne jest wyznaczenie jego odległości od wszystkich obiektów należących do zbioru uczącego, co sprawia, że złożoność czasowa algorytmu jest liniowa i wynosi $O(n)$, gdzie n to liczba obiektów w zbiorze uczącym [3], [5].

2.1.1.2. Klasyfikator RNN

Klasyfikator RNN (ang. *radius nearest neighbours*), podobnie jak klasyfikator KNN, przyporządkowuje obiekty do klasy w oparciu o przynależność klasową obiektów ze zbioru uczącego będących jego najbliższymi sąsiadami. Jednakże w przypadku modelu RNN liczba rozważanych najbliższych sąsiadów nie jest stała i wyznaczana jest w oparciu o zdefiniowany obszar poszukiwań – promień sferycznego obszaru poszukiwań R jest hiperparametrem dobieranym empirycznie na podstawie analizowanych danych. Takie podejście prowadzi do uwzględniania większej liczby sąsiadów w przypadku obszarów przestrzeni cech, w których nagromadzenie obiektów uczących jest duże oraz mniejszej liczby sąsiadów w sytuacji, gdy liczba obiektów uczących zawartych w danym obszarze jest niewielka [4].

Analizowany zbiór najbliższych sąsiadów w metodzie RNN może zostać zapisany jako:

$$RNN(q, P, R) = \{p \in P, d(q, p) < R\} \tag{2.5}$$

gdzie $P = \{p_1, p_2, \ldots, p_n\}$ jest zbiorem obiektów uczących w przestrzeni metrycznej M, q oznacza nowy obiekt, którego najbliższy sąsiad jest poszukiwany, d oznacza odległość nowego obiektu od rozważanego sąsiada wyznaczoną zgodnie z przyjętą metryką odległości, zaś R oznacza promień obszaru poszukiwań [4].

2.1.2. Modele liniowe

Modele liniowe to grupa modeli, która dokonuje klasyfikacji w oparciu o wartość funkcji będącej liniową kombinacją cech opisujących obiekt. Ich główną zaletą jest szybkość i wydajność oraz mniejsza złożoność obliczeniowa niż w przypadku modeli nieliniowych, dokonujących transformacji przestrzeni cech do przestrzeni o wyższym wymiarze [6].

2.1.2.1. Regresja logistyczna

Regresja logistyczna (ang. *logistic regression*, LR) jest modelem zwracającym prawdopodobieństwo przynależności obiektu do jednej z dwóch klas, z reguły do klasy pozytywnej oznaczającej wystąpienie badanego zjawiska, np. wystąpienia choroby w przypadku analizy danych medycznych. Prawdopodobieństwo obliczane jest zgodnie z zależnością:

$$P(y_i = 1|x_i) = \frac{1}{1 + exp(-x_i\beta - \beta_0)} \tag{2.6}$$

gdzie y_i oznacza klasę, do której przynależy analizowany obiekt, x_i oznacza wektor cech opisujących obiekt, β oznacza wektor wag odpowiadających poszczególnym cechom, zaś β_0 oznacza wyraz wolny [7].

2.1.2.2. Klasyfikatory SGD

Metoda gradientu prostego służy do znalezienia wektora parametryzującego przestrzeń funkcji liniowych wykorzystywanych do opisu procesu decyzyjnego w modelach liniowych. Poszukiwania tego wektora odbywają się na podstawie gradientu funkcji kosztu, czyli wektora, którego kierunek pokrywa się z kierunkiem najszybszych zmian funkcji, zaś jego zwrot pozwala określić, w którą stronę następuje wzrost funkcji [8]. Algorytm zejścia gradientowego w sposób iteracyjny wyznacza kolejne punkty zlokalizowane na funkcji kosztu do momentu znalezienia minimum lokalnego funkcji. Proces ten można zapisać jako:

$$\theta = \theta - \eta\nabla J(\theta) \tag{2.7}$$

gdzie η oznacza parametr kontrolujący wielkość kroku, θ oznacza wektor parametrów modelu, zaś $J(\theta)$ oznacza funkcję kosztu [8].

Modele oznaczone w bibliotece *scikit-learn* jako SGD tworzą grupę modeli liniowych optymalizowanych z użyciem algorytmu stochastycznego zejścia gradientowego (ang. *stochastic gradient descent*). Jest to metoda zbliżona do metody gradientu prostego – obie metody różnią się sposobem wyliczania gradientu: w metodzie gradientu prostego gradient wyliczany jest

w oparciu o cały zbiór danych, natomiast w metodzie SGD gradient jest estymowany i aktualizowany na podstawie jednego, losowo wybranego obiektu ze zbioru danych, dzięki czemu koszty czasowe i obliczeniowe tej metody są mniejsze [9]. Aktualizację parametrów modelu w metodzie SGD można zapisać jako:

$$\theta = \theta - \eta \nabla J(\theta; x_i, y_i) \tag{2.8}$$

gdzie x_i oznacza i-ty obiekt ze zbioru uczącego, zaś y_i oznacza jego etykietę określającą przynależność do klasy [8].

2.1.3. Modele wykorzystujące wnioskowanie bayesowskie

Modele wykorzystujące wnioskowanie bayesowskie prezentują podejście alternatywne do modeli wykorzystujących algorytm estymacji największej wiarygodności – modele bazujące na estymacji największej wiarygodności wykorzystują estymatory punktowe poszukiwanego parametru θ, podczas gdy w podejściu bayesowskim model dokonuje predykcji z użyciem pełnego rozkładu względem parametru θ [2].

2.1.3.1. Liniowa analiza dyskryminacyjna

Liniowa analiza dyskryminacyjna (ang. *linear discriminant analysis*, LDA) jest metodą służącą głównie do rozwiązywania problemu klasyfikacji binarnej, jednakże może zostać uogólniona na przypadek wieloklasowy. Założeniem analizy LDA jest separowalność liniowa klas obiektów – celem analizy jest wyznaczenie hiperpłaszczyzn oddzielających obiekty należące do poszczególnych klas w przestrzeni cech [10].

W przypadku klasyfikacji binarnej wyznaczana jest jedna hiperpłaszczyzna, na którą rzutowane są klasyfikowane obiekty. Hiperpłaszczyzna ta jest wyznaczana w taki sposób, by zmaksymalizować odległość pomiędzy środkami ciężkości obu klas przy równoczesnej minimalizacji zmienności wewnątrzklasowej [10].

Predykcje zwracane przez model klasyfikacyjny wykorzystujący analizę LDA mogą zostać wyznaczone zgodnie z równaniem:

$$P(y = k|x) = \frac{P(x|y = k)P(y = k)}{P(x)} = \frac{P(x|y = k)P(y = k)}{\sum_l P(x|y = l) \cdot P(y = l)} \tag{2.9}$$

gdzie k oznacza numer klasy, x oznacza wektor cech opisujących obiekt, y oznacza klasę, do której przynależy obiekt, zaś $P(x|y)$ modelowane jest jako wielowymiarowy rozkład Gaussa, co można zapisać za pomocą zależności:

$$P(x|y = k) = \frac{1}{(2\pi)^{\frac{d}{2}}|\sum_k|^{\frac{1}{2}}} exp(-\frac{1}{2}(x - \mu_k)^t \sum_K^{-1}(x - \mu_k)) \tag{2.10}$$

gdzie d oznacza liczbę cech opisujących obiekt, zaś μ_k oznacza środek ciężkości k-tej klasy [11].

Prawdopodobieństwo opisane równaniem 2.9 poddawane jest operacji logarytmowania w celu uproszczenia obliczeń. Ponieważ w przypadku analizy LDA przyjmowane jest założenie, że macierze kowariancji każdej z klas są takie same, możemy zapisać:

$$logP(y=k|x) = -\frac{1}{2}(x-\mu_k)^t \sum^{-1}(x-\mu_k) + logP(y=k) + const \tag{2.11}$$

gdzie $(x-\mu_k)^t \sum^{-1}(x-\mu_k)$ oznacza odległość Mahalanobisa pomiędzy obiektem x a środkiem ciężkości k-tej klasy μ_k [11].

W wyniku przeprowadzonej analizy obiekt przyporządkowany zostaje do klasy, dla której uzyskany logarytm prawdopodobieństwa *a posteriori* jest największy.

2.1.3.2. Kwadratowa analiza dyskryminacyjna

Kwadratowa analiza dyskryminacyjne (ang. *quadratic distriminan analysis*, QDA) różni się od analizy LDA kształtem wyznaczanej płaszczyzny decyzyjnej – w przypadku QDA jest ona uogólnieniem funkcji kwadratowej. Proces wyznaczania predykcji przez model QDA jest analogiczny do opisanego powyżej procesu wyznaczania predykcji przez model LDA. Ponieważ jednak nie można przyjąć założenia o równości macierzy kowariancji poszczególnych klas, równanie 2.12 przyjmuje postać [11], [12]:

$$logP(y=k|x) = -\frac{1}{2}log|\sum_k| - \frac{1}{2}(x-\mu_k)^t \sum_k^{-1}(x-\mu_k) + logP(y=k) + const \tag{2.12}$$

2.1.4. Modele drzewiaste

Modele drzewiaste zaliczane są do grupy algorytmów nieparametrycznych. Ich działanie polega na poszukiwaniu kryteriów decyzyjnych, które umożliwią wydzielenie grup obiektów w oparciu o przyjmowane przez nie wartości cech. Kryteria decyzyjne wyznaczane są poprzez podział przestrzeni cech na obszary odpowiadające poszczególnym klasom. Dużą zaletą modeli drzewiastych jest łatwość interpretacji wynikająca z jawności przyjętych przez model kryteriów decyzyjnych [13].

2.1.4.1. Las losowy

Las losowy (ang. *random forest*, RF) zaliczany jest do komitetów klasyfikatorów i tworzony jest poprzez agregację wielu drzew klasyfikacyjnych. Ostateczna decyzja dotycząca przyporządkowania obiektu do klasy podejmowana jest na drodze głosowania większościowego: każde z drzew wchodzących w skład lasu losowego przewiduje przynależność klasową obiektu, następnie zaś wybierana jest ta klasa, która zwracana była najczęściej [14]. W celu zminimalizowania korelacji pomiędzy drzewami tworzącymi las losowy wprowadzany jest element losowości – każde z drzew uczone jest na losowo wybrany podzbiorze obiektów należących do zbioru uczącego. Obiekty losowane są ze zwracaniem, co oznacza, że dany obiekt może zostać wykorzystany do treningu wielu drzew klasyfikacyjnych, ale też może nie posłużyć do treningu

żadnego z nich. Ponadto każde z drzew uczone jest z użyciem losowo wybranego podzbioru cech opisujących obiekty. Takie postępowanie prowadzi do zmniejszenia obciążenia wynikowego modelu [15].

Podziały dokonywane w poszczególnych węzłach decyzyjnych drzewa klasyfikacyjnego wchodzącego w skład lasu losowego oparte są zazwyczaj o współczynnik Giniego. Współczynnik ten służy do znalezienia najlepszego kryterium podziału dla danego węzła: może on przyjmować wartości z przedziału [0, 1], gdzie wartość 0 oznacza, że wszystkie obiekty zostały przyporządkowane do tej samej klasy. Współczynnik Giniego dla zbioru S wyznaczany jest zgodnie z zależnością:

$$Gini(S) = 1 - \sum_{i=1}^{m} P_i^2 \tag{2.13}$$

gdzie P_i oznacza prawdopodobieństwo przynależności losowego obiektu ze zbioru S do podzbioru (klasy) C_i, zaś m oznacza liczbę podzbiorów (klas) wydzielonych w zbiorze S [16].

Alternatywnym podejściem do wyznaczenia optymalnego podziału w węźle jest zastosowanie jako kryteriów podziału entropii i zysku informacyjnego. W pierwszym kroku należy wyznaczyć wartość entropii H dla predykcji zwracanych przez model (równanie 2.14), następnie entropię podziału zbioru ze względu na rozważaną cechę opisującą obiekty (równanie 2.15), a w ostatnim kroku zysk informacyjny będący różnicą tych dwóch wartości (równanie 2.17). Maksymalizacja zysku informacyjnego prowadzi do uzyskania najlepszego podziału [17].

$$H = -\sum_{i=1}^{p} P(x_i) log_2(P(x_i)) \tag{2.14}$$

gdzie x_i oznacza rozważaną cechę, p oznacza liczbę cech opisujących obiekt.

$$E = \sum_{j=1}^{r} \frac{|S_j|}{\sum_{j=1}^{r} |S_j|} H_j \tag{2.15}$$

$$H_j = -\sum_{i=1}^{p} P(x_i|S_j) log_2(P(x_i|S_j)) \tag{2.16}$$

gdzie S_j oznacza podzbiór wydzielony ze zbioru S.

$$Gain = H - E \tag{2.17}$$

2.1.4.2. Klasyfikator Extra Trees

Model dostępny w bibliotece *scikit-learn* pod nazwą *Extra Trees* jest modelem zbliżonym do lasu losowego, jednak posiadającym dodatkowy element losowości. W przeciwieństwie do lasu losowego, w modelu *Extra Trees* podział danych w poszczególnych węzłach decyzyjnych nie jest dokonywany w oparciu o współczynnik Giniego lub zysk informacyjny, lecz prowadzony jest w sposób losowy. Takie postępowanie pozwala na zmniejszenie wariancji klasyfikatora przy niewielkim wzroście jego obciążenia [18], [19].

2.1.4.3. Klasyfikator AdaBoost

Metoda AdaBoost jest metodą wzmacniania (ang. *boosting*) modelu klasyfikacyjnego, zwiększającą jego skuteczność klasyfikacji. Polega na iteracyjnej zmianie wag przypisywanych poszczególnym obiektom ze zbioru uczącego podczas kolejnych etapów uczenia. Początkowo drzewo klasyfikacyjne uczone jest na zbiorze uczącym, w którym każdy obiekt ma przypisaną taką samą wagę wynoszącą $w = \frac{1}{N}$, gdzie N oznacza liczbę obiektów w zbiorze. W kolejnych krokach wagi te ulegają modyfikacji – wagi obiektów, które zostały nieprawidłowo zaklasyfikowane ulegają zwiększeniu, zaś wagi obiektów zaklasyfikowanych prawidłowo ulegają zmniejszeniu. W ten sposób w każdym kolejnym kroku rośnie znaczenie obiektów trudnych do zaklasyfikowania. Uzyskany w ten sposób zbiór drzew klasyfikacyjnych zmuszony jest więc do skupienia się na przypadkach trudnych, dzięki czemu jego zdolności generalizacyjne oraz dokładność klasyfikacji ulegają poprawie [11], [20].

2.1.4.4. Klasyfikator LGB

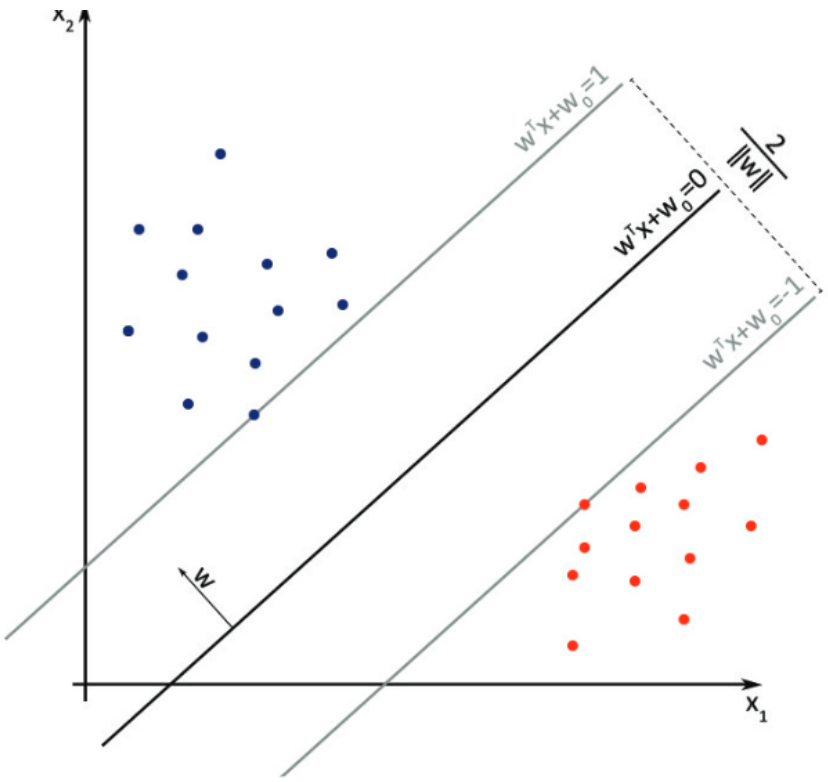
Model klasyfikacyjny znany jako LightGBM (w dalszej części pracy oznaczany jako LGB) uczony jest z użyciem technik mających na celu zredukowanie złożoności obliczeniowej algorytmów drzewiastych związanej z dużą liczebnością zbiorów uczących oraz dużą liczbą opisujących je cech [21].

Problemy związane z liczebnością zbiorów rozwiązywane są z użyciem techniki GOSS (ang. *Gradient-based One-Side Sampling*) opartej o analizę gradientu oraz jego związku z zyskiem informacyjnym. Autorzy tej metody wykazali, że obiekty, z którymi związane są większe wartości bezwzględne gradientu mają większy udział w zysku informacyjnym. Usunięcie ze zbioru danych losowych obiektów cechujących się niewielkimi wartościami gradientu spowoduje więc zmniejszenie ponoszonych kosztów obliczeniowych przy równoczesnym zachowaniu możliwie dużego zysku informacyjnego [21].

Problemy wynikające z dużej wymiarowości danych są natomiast rozwiązywane z użyciem techniki EFB (ang. *Exclusive Feature Bundling*). Technika ta wykorzystuje fakt, że w przypadku danych jakościowych o dużej wymiarowości przestrzeń cech jest z reguły przestrzenią rzadką, co oznacza, że stosunkowo rzadko występuje sytuacja, w której wiele cech przyjmuje równocześnie wartości niezerowe. Bazując na tym założeniu możliwe jest przeprowadzenie redukcji wymiarowości poprzez agregację cech wykluczających się w taki sposób, by ich wartości przedstawić za pomocą jednej nowej zmiennej [21].

2.1.5. Modele jądrowe

Modele jądrowe wykorzystują tzw. funkcję jądra dokonującą transformacji przestrzeni cech. Celem zastosowania tej funkcji jest uzyskanie nowej przestrzeni cech o wyższym wymiarze, w której możliwe jest takie zrzutowanie obiektów należących do różnych klas, by były one liniowo separowalne [22].



Rys. 2.2. Hiperpłaszczyzna i margines decyzyjny w modelu maszyny wektorów nośnych dla obiektów opisanych dwiema cechami i należących do dwóch klas. Źródło: [23].

Przykładem modelu jądrowego, który został użyty w niniejszej pracy, jest maszyna wektorów nośnych (ang. *support vector machine*, SVM). Model SVM poszukuje optymalnej hiperpłaszczyzny decyzyjnej oddzielającej od siebie obiekty należące do różnych klas z zachowaniem maksymalnego marginesu. W przypadku klasyfikacji binarnej rozwiązuje on następujący problem optymalizacyjny:

$$\min_{w,b,\zeta} \frac{1}{2} w^t w + C \sum_{i=1}^{n} \zeta_i \tag{2.18}$$

przy warunkach

$$y_i(w^T \phi(x_i) + w_0 \geq 1 - \zeta_i$$

$$\zeta_i \geq 0, i = 1, \ldots, n$$

gdzie w_0 jest przesunięciem (ang. *bias*) hiperpłaszczyzny rozdzielającej, w oznacza wektor wag modelu, C jest parametrem kontrolującym siłę regularyzacji i położenie hiperpłaszczyzn marginesowych, zaś ζ_i określa odległość rozważanego obiektu od hiperpłaszczyzny marginesowej wyznaczającej jego rzeczywistą klasę [22], [24], [25]. Maksymalizacja szerokości marginesu (rys 2.2) odbywa się poprzez minimalizację wyrażenia $||w||^2 = w^T w$.

Zadaniem dualnym (ang. *dual problem*) do przedstawionego powyżej jest minimalizacja wyrażenia:

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \tag{2.19}$$

przy warunkach

$$y^T\alpha = 0$$

$$0 \leq \alpha_i \leq C, i = 1, \ldots, n$$

gdzie α_i oznacza mnożniki Lagrange'a, Q jest macierzą nieujemnie określoną, której elementy dane są wzorem $Q_{ij} = y_i, y_j(K(x_i, x_j)$, gdzie $(K(x_i, x_j) = \phi(x_i)^T\phi(x_j)$ jest funkcją jądra [22], [24].

Rozwiązanie problemu optymalizacyjnego prowadzi do uzyskania funkcji decyzyjnej dla danego obiektu x wyrażonej w postaci:

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b. \tag{2.20}$$

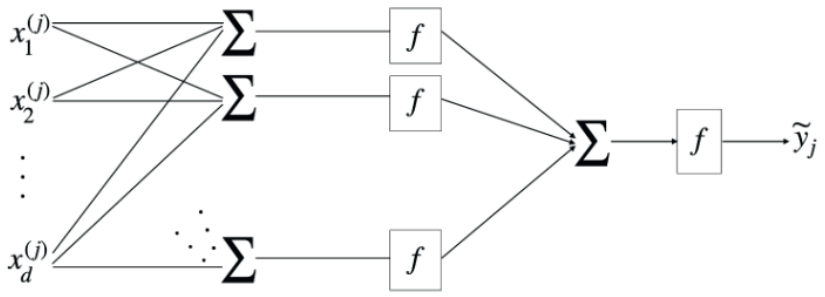
$$w_0 = 1 - w^T x_{SV} \tag{2.21}$$

gdzie SV oznacza zbiór wektorów nośnych X_{SV}, czyli obserwacji leżących najbliżej hiperpłaszczyzny rozdzielającej [22], [24].

2.1.6. Sieci neuronowe

Sieci neuronowe to rodzaj modeli, których budowa i działanie inspirowane są biologicznymi sieciami neuronowymi występującymi w układzie nerwowym. Najprostszym przykładem wielowarstwowej sieci neuronowej jest perceptron wielowarstwowy.

Perceptron wielowarstwowy (ang. *multilayer perceptron*, MLP) jest rodzajem jednokierunkowej sieci neuronowej złożonej z co najmniej trzech warstw: warstwy wejściowej, co najmniej jednej warstwy ukrytej oraz warstwy wyjściowej. Każda warstwa połączona jest z warstwą poprzednią, a kierunek przepływu danych jest zawsze taki sam: od warstwy wejściowej, poprzez kolejne warstwy ukryte, do warstwy wyjściowej. Ponadto sieć MLP jest siecią gęstą, co oznacza, że każdy neuron w danej warstwie połączony jest z każdym neuronem warstwy poprzedniej (rys. 2.3) [26].



Rys. 2.3. Schemat perceptronu wielowarstwowego z jedną warstwą ukrytą. Źródło: [23].

Proces uczenia sieci MLP odbywa się z użyciem algorytmu propagacji wstecznej minimalizującego popełniany błąd predykcji z użyciem metody największego spadku. Algorytm propagacji wstecznej można podzielić na dwa etapy:

1. propagację w przód (ang. *forward propagation*), podczas której generowana jest wartość na wyjściu sieci w oparciu o dane wejściowe i wagi poszczególnych neuronów:

$$y_j^l = \phi(v_j^l(n)) \tag{2.22}$$

gdzie y_j^l jest sygnałem na wyjściu j-tego neuronu w l-tej warstwie i w n-tej epoce, ϕ jest funkcją aktywacji, zaś v_j^l jest zdefiniowane jako:

$$v_j^l = \sum_{i=1}^{m} w_{ji}^l y_i^{l-1}(n) \tag{2.23}$$

gdzie $w_{ji}^l(n)$ oznacza jest wektorem wag pomiędzy j-tym neuronem l-tej warstwy a i-tym neuronem w warstwie $l-1$, zaś m oznacza liczbę neuronów w warstwie $l-1$,

2. propagację wsteczną (ang. *backward propagation*), podczas której sygnał zawierający informację o popełnianym błędzie jest przesyłany od warstwy wyjściowej w kierunku warstwy wyjściowej w celu aktualizacji wag neuronów zgodnie ze wzorem:

$$w_{ji}^l(n+1) = w_{ji}^l(n) + \eta \Delta w_{ji}^l(n) \tag{2.24}$$

$$\Delta w_{ji}^l(n) = \delta_j^l(n) \cdot y_i^{l-1}(n) \tag{2.25}$$

gdzie η jest współczynnikiem uczenia określającym wielkość kroku wykonywanego podczas zmian wartości wag, $\delta_j^l(n)$ jest gradientem zdefiniowanym jako:

$$\delta_j^l(n) = \begin{cases} (d_j(n) - o_j^l)\phi'(v_j^l(n)) & \text{dla warstwy wyjściowej} \\ \phi'(v_j^l(n)) \sum_{i=1}^{p} \delta_i^{l+1} w_{ij}^{l+1}(n) & \text{dla warstwy ukrytej} \end{cases} \tag{2.26}$$

gdzie p jest liczbą neuronów w warstwie $l+1$, zaś wyrażenie $d_j(n) - o_j^l$ określa wartość błędu popełnianego przez perceptron przy aktualnych wartościach wag.

Po przeprowadzeniu propagacji wstecznej błędu ponownie wykonywana jest propagacja w przód, w celu wyznaczenia nowego błędu predykcji popełnianego przez sieć. Algorytm wykonywany jest iteracyjnie przez określoną liczbę epok lub do momentu osiągnięcia dopuszczalnego błędu predykcji [26].

Poprzez wprowadzenie nieliniowych funkcji aktywacji w warstwach ukrytych oraz warstwie wyjściowej możliwe jest klasyfikowanie obiektów należących do klas, które nie są liniowo separowalne. [27] Do najczęściej stosowanych funkcji aktywacji zaliczają się [26], [28]:

– funkcja logistyczna:

– funkcja sigmoidalna:

$$\phi(v_j(n)) = \frac{1 - e^{-v_j(n)}}{1 + e^{-v_j(n)}} \tag{2.27}$$

– poprawiona funkcja liniowa (ang. *rectified linear unit*, ReLU):

$$\phi(v_j(n)) = min(1, max(0, v_j(n))) \tag{2.28}$$

– tangens hiperboliczny:

$$\phi(v_j(n)) = tanh(2v_j(n)) = \frac{e^{-v_j(n)} - e^{-v_j(n)}}{e^{-v_j(n)} + e^{-v_j(n)}} \tag{2.29}$$

2.2. Algorytmy optymalizacyjne

Na dobór odpowiedniego modelu umożliwiającego rozwiązanie rozważanego problemu klasyfikacyjnego składa się nie tylko wybór odpowiedniego algorytmu, ale również określenie jego architektury poprzez dobór najlepszego zestawu wartości hiperparametrów. Dobór tych wartości, nazywany optymalizacją hiperparametrów, jest procesem złożonym i czasochłonnym - ręczne ustalenie wartości hiperparametrów jest nieefektywne i ograniczone poprzez liczne czynniki, takie jak liczba hiperparametrów, zachodzące pomiędzy nimi nieliniowe inteakcje oraz konieczność ewaluacji modelu po każdej zmianie wartości hiperparametrów. Problem ten rozwiązywany jest poprzez użycie odpowiednich algorytmów optymalizacyjnych, które w sposób zautomatyzowany przeszukują przestrzeń hiperparametrów w poszukiwaniu zestawu wartości prowadzącego do osiągnięcia możliwie małego błędu predykcji [29]. W niniejszej rozprawie zastosowane zostały dwa rodzaje optymalizacji hiperparametrów modelu: optymalizacja bayesowska oraz optymalizacja ewolucyjna.

2.2.1. Optymalizacja bayesowska

Optymalizacja bayesowska to rodzaj optymalizacji mający na celu znalezienie ekstremum funkcji celu (maksimum lub minimum, w zależności od zastosowanej funkcji) w jak najmniejszej liczbie wykonanych iteracji. Swoją nazwę zawdzięcza zastosowaniu wnioskowania bayesowskiego do wyznaczania kolejnych zestawów wartości hiperparametrów x_{opt}, dla których wyznaczana jest wartość funkcji celu. Stosując tę strategię optymalizacji zakłada się, że możliwe jest przyjęcie założeń dotyczących niektórych własności funkcji celu, np. jej gładkości czy też występowania punktów zmiany (ang. *changepoint*) i ich lokalizacji. Na podstawie tych założeń określane jest prawdopodobieństwo *a priori* dotyczące przestrzeni funkcji celu uzyskanych dla rozważanego x_{opt}. Następnie w kolejnych iteracjach algorytmu optymalizacyjnego prawdopodobieństwo to jest aktualizowane w oparciu o funkcję wiarygodności $P(D_{1:n}|f)$ wyznaczaną na podstawie dodawanych kolejnych obserwacji $D_{1:n} = \{x_i, f(x_i)\}|_{i=1}^{n}$ w celu uzyskania prawdopodobieństwa *a posteriori* (równanie 2.30) lepiej przybliżającego rzeczywistą funkcję celu [30].

$$P(f|(D_{1:n}) \propto P(D_{1:n}|f)P(f) \tag{2.30}$$

Optymalizacja bayesowska wykorzystuje dwa rodzaje przeszukiwania przestrzeni hiperparametrów:

– eksplorację (ang. *exploration*) - przeszukiwanie obszarów znacznie oddalonych od dotychczas przeanalizowanych,

– eksploatację (ang. *exploitation*) - przeszukiwanie obszarów znajdujących się w pobliżu tych, dla których uzyskano najbardziej satysfakcjonujące rezultaty.

Równowagę pomiędzy udziałem tych strategii w procesie poszukiwania optymalnego zestawu wartości hiperparametrów zapewnia funkcja akwizycji (ang. *aquisition function*), która może zostać zdefiniowana za pomocą oczekiwanego spadku popełnianego błędu (ang. *expected improvement*, równanie 2.31) lub prawdopodobieństwa spadku popełnianego błędu (ang. *probability of improvement*, równanie 2.32) [31].

$$EI_{y^*}[x|D] := \int_{-\infty}^{y^*} (y^* - y)p(y|x, D)\, dy \tag{2.31}$$

$$P(y \leq y^*|x, D) := \int_{-\infty}^{y^*} p(y|x, D)\, dy \tag{2.32}$$

gdzie D oznacza zbiór obserwacji o liczebności N, x jest zestawem wartości hiperparametrów modelu, zaś y jest funkcją celu [31].

W celu obliczenia wartości funkcji akwizycji konieczne jest zamodelowanie składnika $p(y|x, D)$ – zazwyczaj jest to czynione przy użyciu procesów gaussowskich, lasów losowych bądź jądrowych estymatorów gęstości, do których zaliczane są estymatory Parzena (ang. *tree-structured Parzen estimator*, TPE) wykorzystywane w niniejszej rozprawie [31].

Aby zamodelować składnik $p(y|x, D)$ przy użyciu TPE przyjmowane jest następujące założenie:

$$p(x|y, D) := \begin{cases} p(x|D^{(l)}) & gdy \quad (y \leq y^{\gamma}) \\ p(x|D^{(g)}) & gdy \quad (y > y^{\gamma}) \end{cases} \tag{2.33}$$

gdzie $D^{(l)} = \{(x_n, y_n)\}_{n=1}^{N^l}$ oznacza podzbiór obserwacji w zbiorze D prowadzących do uzyskania mniejszego błędu, $D^{(g)} = \{(x_n, y_n)\}_{n=N^l+1}^{N}$ oznacza podzbiór obserwacji w zbiorze D prowadzących do uzyskania większego błędu, $N^{(l)}$ oznacza wielkość zbioru $D^{(l)}$, $\gamma \in (0, 1]$ oznacza kwantyl wyznaczony dla zbioru $D^{(l)}$ podczas każdej iteracji, zaś y^{γ} oznacza wartość funkcji celu wyznaczoną dla kwantyla γ zbioru D [31], [32].

Funkcje gęstości prawdopodobieństwa dla zbiorów $D^{(l)}$ i $D^{(g)}$ występujące w równaniu 2.33 są wyznaczane zgodnie z zależnościami:

$$p(x|D^{(l)}) = w_0^l p_0(x) + \sum_{n=1}^{N^{(l)}} w_n k(x, x_n|b^{(l)}) \tag{2.34}$$

$$p(x|D^{(g)}) = w_0^g p_0(x) + \sum_{n=N^{(l)}+1}^{N} w_n k(x, x_n|b^{(g)}) \tag{2.35}$$

gdzie $w_n \in [0,1], n = 1, 2, ..., N$ oznaczają wagę każdej bazy w estymatorze KDE, $b^{(l)} \in \mathbb{R}_+$ oznacza szerokość funkcji jądra wyznaczoną na podstawie zbioru $D^(l)$, $b^{(g)} \in \mathbb{R}_+$ oznacza szerokość funkcji jądra wyznaczoną na podstawie zbioru $D^(g)$, k oznacza funkcję jądra, zaś p_0 oznacza prawdopodobieństwo *a priori* odzwierciedlające niewiedzę badacza (*non-informative prior*) [31].

Zgodnie z założeniem przedstawionym w równaniu 2.33, funkcja akwizycji przyjmuje następującą postać:

$$P(y \leq y^{\gamma}|x, D) \overset{\text{rand}}{\simeq} r(x|D) := \frac{p(x|D^{(l)})}{p(x|D^{(g)})} \tag{2.36}$$

gdzie $\overset{\text{rand}}{\simeq}$ oznacza izomorfizm porządkowy [31].

Przebieg wyznaczania wartości estymatora TPE zaimplementowany w pakiecie Optuna [33] można opisać za pomocą następującego algorytmu [31]:

1. utworzenie zbioru pustego obserwacji D
2. wybranie losowego zestawu wartości hiperparamtrów x,
3. ewaluacja funkcji celu $y_n := f(x_n) + \epsilon_n$,
4. dodanie pary $\{(x_n, y_n)\}$ do zbioru D,
5. N_{init}-krotne powtórzenie kroków 2-4, gdzie N_{init} jest parametrem definiowanym przez użytkownika, oznaczającym początkową liczbę generowanych zestawów wartości hiperparametrów,
6. wyznaczenie kwantyla γ na podstawie zbioru D,
7. podział zbioru D na podzbiory $D^{(l)}$ i $D^{(l)}$ zawierające odpowiednio obserwacje prowadzące do uzyskania mniejszego i większego błędu,
8. wyznaczenie wag bazy estymatora KDE,
9. wyznaczenie szerokości funkcji jądra $b^{(l)}$ i $b^{(g)}$ na podstawie zbiorów $D^{(l)}$ i $D^{(g)}$,
10. wyznaczenie $p(x|D^{(l)})$ i $p(x|D^{(g)})$ zgodnie z równaniami 2.34-2.35,
11. utworzenie zbioru S poprzez wybór N_s obserwacji na podstawie funkcji gęstości prawdopodobieństwa $p(x|D^{(l)})$, które posłużą do optymalizacji funkcji akwizycji,

12. wybór obserwacji x_{N+1} pozwalającej uzyskać największy wartość $\frac{p(x|D^{(l)})}{p(x|D^{(g)})}$ spośród obserwacji zawartych w zbiorze S,

13. ewaluacja funkcji celu $y_{N+1} := f(x_{N+1}) + \epsilon_{N+1}$,

14. dodatnie pary $\{(x_{N+1}, y_{N+1})\}$ do zbioru D,

15. K-krotne powtórzenie kroków 6-14, gdzie K oznacza zdefiniowaną przez użytkownika liczbę prób.

2.2.2. Optymalizacja ewolucyjna

Optymalizacja ewolucyjna, należąca do grupy heurystycznych metod stochastycznej optymalizacji modeli, bazuje na mechanizmach obserwowanych w systemach biologicznych – ewolucji gatunków oraz ich adaptacji do zmiennych środowiskowych. Inspiracja mechanizmami biologicznymi znajduje odzwierciedlenie w stosowanej nomenklaturze dotyczącej algorytmów ewolucyjnych, takiej jak populacja, mutacja czy krzyżowanie. Przebieg poszukiwania optymalnego zestawu hiperparametrów modelu można opisać za pomocą następującego algorytmu [34]:

1. Wygenerowanie populacji początkowej (zbioru modeli wykorzystywanych do rozwiązania rozważanego problemu klasyfikacyjnego) o losowych wartościach hiperparametrów.

2. Wyznaczenie wartości funkcji przystosowania (funkcji celu) dla każdego z osobników (modeli).

3. Wybór osobników, które posłużą do reprodukcji (utworzenia nowych modeli) na drodze krzyżowania lub mutacji.

4. Wygenerowanie nowych osobników tworzących populację potomną o takiej samej liczebności, jak populacja początkowa.

5. Powtórzenie kroków 2-4 do momentu znalezienia ekstremum funkcji przystosowania lub warunku przerwania określanego zazwyczaj poprzez dopuszczalny czas trwania procedu optymalizacji lub maksymalną liczbę iteracji.

Liczebność populacji, prawdopodobieństwo zajścia mutacji i krzyżowania oraz możliwość pozostawienia w populacji osobników, dla których w dnej popualacji osiągnięto najlepsze wartości funkcji przystosowania są parametrami określanymi podczas implementacji algorytmu [35].

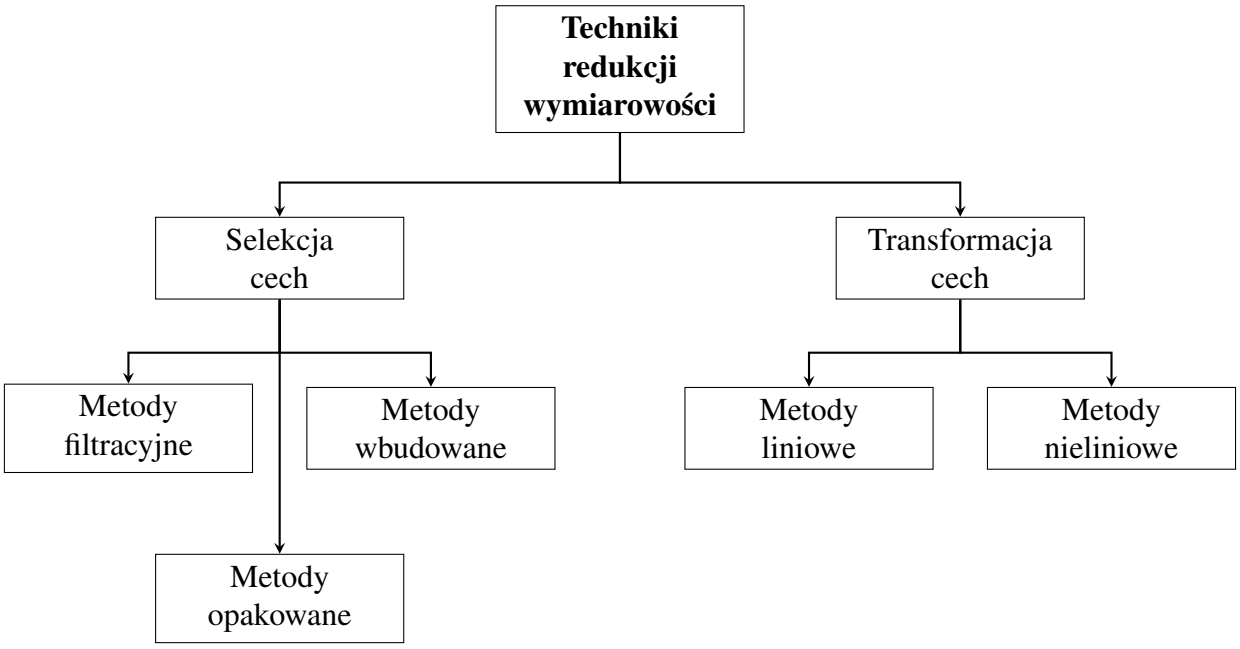
Ewolucyjne algorytmy optymalizacyjne znajdują zastosowanie przede wszystkim w sytuacji, gdy rozważany jest problem opisany za pomocą więcej niż jednej funkcji celu. Otrzymywane jest wówczas tzw. rozwiązanie Pareto-optymalne, czyli takie, w którym dalsza poprawa wartości jednej z funkcji celu powoduje pogorszenie wartości co najmniej jednej z pozostałych funkcji celu [36].

Chociaż zastosowanie optymalizacji ewolucyjnej umożliwia bardziej wyczerpujące przeszukanie przestrzeni cech niż użycie algorytmów deterministycznych, należy wziąć pod uwagę,

że często wiąże się to z nadmiernymi kosztami czasowymi i obliczeniowymi. W przypadku problemów związanych z danymi o dużej wymiarowości problematyczne stają się również kwestie związane z niestabilnością algorytmów ewolucyjnych, możliwością przypadkowego wzmocnienia szumu informacyjnego oraz zaobserwowaniem fałszywych korelacji pomiędzy analizowanymi zmiennymi [37].

2.3. Metody redukcji wymiarowości

Analiza sygnału mowy wiąże się z problemem znanym jako *przekleństwo wymiarowości* – sygnał rejestrowany jest z częstotliwością próbkowania rzędu kilku lub kilkudziesięciu kiloherców, co oznacza, że każde nagranie składa się z dziesiątek tysięcy próbek. Równocześnie medyczne korpusy mowy składają się często z bardzo niewielkiej liczby sygnałów, zawierając jedynie po kilkadziesiąt lub kilkaset nagrań odpowiadających poszczególnym schorzeniom. Tak duża dysproporcja pomiędzy liczebnością zbioru danych oraz liczbą parametrów opisujących pojedynczy sygnał stanowi duże utrudnienie w budowie wiarygodnego modelu klasyfikacyjnego. Proces ekstrakcji cech, polegający zwykle na wyznaczeniu z sygnału mowy parametrów czasowych, częstotliwościowych lub czasowo-częstotliwościowych zmniejsza skalę tego problemu, jednak nie redukuje go całkowicie – pojedyncze nagrania nadal opisywane jest przez bardzo dużą liczbę cech [38].



Rys. 2.4. Podział algorytmów redukcji wymiarowości.

Metody służące do zmniejszenia liczby cech opisujących pojedynczy obiekt nazywane są technikami redukcji wymiarowości. Ich zastosowanie ma na celu doprowadzenie do sytuacji,

w której obiekt opisywany jest przez stosunkowo niewielką liczbę cech przy równoczesnym zachowaniu jak największej części informacji zawartej w oryginalnym wektorze cech. Skutkiem uczenia modeli klasyfikacyjnych na zredukowanej macierzy cech jest usunięcie z modelu cech redundantnych oraz szumu, zmniejszenie złożoności obliczeniowej algorytmów, wyznaczenie przez nie granic decyzyjnych lepiej oddających rzeczywiste różnice pomiędzy analizowanymi klasami obiektów, a tym samym wzrost dokładności klasyfikacji [39].

Techniki redukcji wymiarowości można podzielić na dwie grupy (rys. 2.4):

– algorytmy selekcji cech,

– algorytmy transformacji cech.

Pierwsza grupa algorytmów ma za zadanie wybrać najbardziej istotne cechy opisujące obiekt spośród wszystkich dostępnych cech. Wybór tych cech może przebiegać w sposób rekurencyjny lub być dokonywany na drodze filtracji w oparciu o wartości wyznaczonych statystyk. Algorytmy należące do drugiej grupy mają na celu transformację cech opisujących obiekt w taki sposób, by przy użyciu możliwie małej liczby cech transformowanych opisać jak największą część informacji zawartej w cechach oryginalnych [40].

Liczba opracowanych algorytmów redukcji wymiarowości jest duża i nadal rośnie – opracowywane są zarówno nowe techniki, jak i wprowadza się modyfikacje technik już istniejących, prowadzące do zwiększenia ich skuteczności. Poniżej przedstawiono techniki, które wykorzystano w niniejszej pracy.

2.3.1. Metoda *SelectKBest*

Algorytm *SelectKBest* zaimplementowany w pakiecie *scikit-learn* zaliczany jest do algorytmów filtracyjnej selekcji cech. Jego działanie polega na obliczeniu wartości pewnej metryki dla każdej z cech opisujących obiekty zawarte w zbiorze danych, a następnie wybraniu określonej liczby najlepszych cech na podstawie wartości tej metryki. Wybór metryki uzależniony jest od zastosowanej funkcji oceny. Najczęściej stosowanymi funkcjami oceny są [11]:

– ANOVA F-test – funkcja ta jest stosowana w przypadku rozwiązywania problemów regresji i klasyfikacji binarnej,

– chi-kwadrat - funkcja ta jest stosowana w przypadku rozwiązywania problemów klasyfikacji,

– informacja wzajemna (ang. *mutual information*) - funkcja ta jest stosowana w przypadku rozwiązywania problemów klasyfikacji i regresji.

W niniejszej pracy zastosowano funkcję bazującą na statystyce chi-kwadrat. W metodzie tej obliczana jest metryka chi-kwadrat między zmienną zależną a ilościową zmienną niezależną, a następnie wybierane jest n zmiennych niezależnych o maksymalnych wartościach statystyki chi-kwadrat. Metoda ta bazuje na założeniu, że brak zależności pomiędzy zmienną zależną i niezależną (występujący dla małych wartości statystyki chi-kwadrat) oznacza, że dana zmienna

niezależna nie jest przydatna dla budowanego modelu, może więc zostać usunięta bez znaczącego pogorszenia wyników klasyfikacji [11].

2.3.2. Analiza składowych niezależnych

Analiza składowych niezależnych (ang. *independent component analysis*, ICA) jest algorytmem stosowanym przede wszystkim do ślepej separacji źródeł sygnału, może jednak posłużyć do przeprowadzenia redukcji wymiarowości. Jej zastosowanie umożliwia przekształcenie oryginalnej przestrzeni cech w przestrzeń o niższym wymiarze poprzez transformację zmiennych niezależnych (cech) opisujących obiekt w nowe, wzajemne niezależne zmienne [41], [42].

Załóżmy, że obserwowany jest wektor $x = [x_1, x_2, \ldots, x_N]^T$. Wektor ten może zostać przedstawiony w postaci iloczynu macierzy mieszającej A oraz wektora niezależnych źródeł sygnału $s = [s_1, s_2, s_L]^T$ opisanych rozkładem niegaussowskim [42].:

$$x = As \tag{2.37}$$

Głównym celem analizy ICA jest znalezienie macierzy separującej W, która umożliwia uzyskanie przybliżonych przebiegów sygnałów źródłowych tworzących wektor u:

$$u = Wx = WAs \tag{2.38}$$

Macierz W powinna gwarantować jak największą niezależność składowych wektora u [42].

Z punktu widzenia teorii estymacji maksymalizacja niezależności składowych wektora u jest równoważna z minimalizacją informacji wzajemnej pomiędzy tymi składowymi. Jak zostało wykazane w [43], minimalizację informacji wzajemnej pomiędzy składowymi obserwowanymi na wyjściu układu $y = [y_1, y_2, \ldots, y_N]^T$ można zastąpić maksymalizacją entropii wspólnej $H(y)$ tych składowych:

$$H(y) = -\int p(y) log p(y), dy = -\int p(x) \frac{p(x)}{log|detJ(x)|}, dx \tag{2.39}$$

gdzie $J(x)$ oznacza jakobian, którego element (i, j) jest wyznaczany jako pochodna cząstkowa $\frac{\partial y_j}{\partial x_i}$, $p(y)$ oznacza funkcję gęstości prawdopodobieństwa wektora y [42].

Na podstawie równania 2.39 można wyznaczyć zależność opisującą proces uczenia algorytmu ICA:

$$\Delta W \propto [I - \phi(u)u^T]W \tag{2.40}$$

gdzie

$$\phi(u) = \left[-\frac{\frac{\partial p_1(u_1)}{\partial u_1}}{p_1(u_1)}, \ldots, -\frac{\frac{\partial p_1(u_N)}{\partial u_N}}{p_1(u_N)} \right]^T \tag{2.41}$$

$$I(y) = \int log \frac{p(y)}{\prod_{i=1}^{N} p_i(y_i)}, dy \tag{2.42}$$

gdzie $p_i(y_i)$ oznacza funkcję gęstości rozkładu prawdopodobieństwa brzegowego zmiennej y_i [42].

2.3.3. Metoda SVD

Metoda SVD oparta jest o algorytm rozkładu macierzy według wartości osobliwych (ang. *singular value decomposition*, SVD). Polega ona na dekompozycji macierzy A i przedstawieniu jej w postaci trzech macierzy zgodnie z zależnością

$$A_{mxn} = U_{mxr}\Sigma_{rxr}V^T_{nxr} \tag{2.43}$$

gdzie U_{mxr} i V_{nxr} są macierzami ortonormalnymi, Σ_{rxr} jest macierzą diagonalną, której elementy stanowią wartości osobliwe, m i n oznaczają liczbę wierszy i kolumn macierzy A, zaś r oznacza rząd macierzy A [44].

Istotną cechą macierzy Σ_{rxr} jest fakt, że elementy umieszczone na jej przekątnej są uporządkowane w sposób malejący. Ponieważ elementy te odpowiadają kolejnym wymiarom nowej przestrzeni można stwierdzić, że każdy kolejny wymiar niesie mniejszą zawartość informacji niż wymiary go poprzedzające. Dzięki temu możliwe jest uzyskanie przybliżonej postaci macierzy A poprzez pozostawienie jedynie k najistotniejszych wartości osobliwych tworzących macierz Σ (pozostałym elementom macierzy przypisywana jest wartość 0) oraz odpowiadających im wierszy i kolumn macierzy U i V [44].

Algorytm SVD wykorzystywany jest do kompresji danych oraz do redukcji wymiarowości poprzez ekstrakcję nowych cech (wartości osobliwych) z macierzy cech opisujących obiekty [45], [46].

2.3.4. Analiza składowych głównych

Analiza składowych głównych (ang. *principal component analysis*, PCA) jest metodą liniowej transformacji zmiennych polegającą na utworzeniu nowych zmiennych, będących kombinacją liniową zmiennych oryginalnych, w taki sposób, by były one wzajemnie ortogonalne. Nowe zmienne, zwane składowymi głównymi, uporządkowane są w sposób malejący ze względu na opisywaną wariancję danych – pierwsza składowa główna opisuje największą część zmienności, a każda kolejna składowa wyjaśnia coraz mniejszą część wariancji danych. Suma wariancji składowych głównych jest równa sumie wariancji zmiennych oryginalnych, transformacja ta nie prowadzi więc do utraty informacji [47].

Niech x będzie wektorem N cech opisujących obiekty, dla których znana jest wariancja oraz macierz kowariancji dana wzorem [48]:

$$\Sigma = \sum_{i=1}^{N}(x_i - \mu)(x_i - \mu)^T \tag{2.44}$$

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{2.45}$$

Pierwsza składowa główna wyznaczana jest zgodnie z zależnością:

$$z_1 = \alpha_1^T x = \alpha_{11}x_1 + \alpha_{12}x_2 + ... + \alpha_{1N}x_N = \sum_{i=1}^{N} \alpha_{1i}x_i \tag{2.46}$$

gdzie α_{1i} stanowią współczynniki dobrane w taki sposób, by zmaksymalizować wariancję pierwszej składowej głównej [47], [49].

Kolejne składowe główne wyznaczane są iteracyjnie w taki sposób, by uzyskać maksymalną wariancję przy równowczesnym zachowaniu awrunku ortogonalności tworzonych składowych. Formalnie składowe główne można zapisać za pomocą wzoru:

$$z_n = \alpha_n^T x \tag{2.47}$$

gdzie α_n jest wektorem własnym macierzy kowariancji zmiennych tworzących wektor cech x, odpowiadającej n-tej największej wartości własnej λ_n [47].

Redukcja wymiarowości przy użyciu algorytmu PCA polega na odrzuceniu części składowych głównych cechujących się najmniejszą wariancją. Prowadzi to do zmniejszenia wymiarowości analizowanych danych przy jednoczesnej stosunkowo niewielkiej utracie informacji [47].

2.3.4.1. Jądrowa analiza składowych głównych

Klasyczna analiza składowych głównych ze względu na fakt, że jest przekształceniem liniowym, nie daje dobrych rezultatów w przypadku, gdy dane nie są liniowe – uzyskiwana wówczas przestrzeń nie jest optymalna. Rozwiązaniem tego problemu jest zastosowanie jądrowej analizy składowych głównych (ang. *kernel principal component analysis*, KernelPCA), która rzutuje dane na przestrzeń o wyższym wymiarze, w której dane są liniowo separowalne korzystając z funkcji jądra danej poniższym wzorem:

$$K(x_a, x_b) = \phi(x_a)^T \phi(x_b) \tag{2.48}$$

gdzie x_a i x_b są punktami danych, ϕ oznacza nieliniowe przekształcenie $\phi : \mathbb{R}^p -> H$, p jest liczbą cech opisujących pojedynczy obiekt, zaś H jest przestrzenią Hilberta [49].

Do najczęściej wykorzystywanych funkcji jądra zaliczają się [46]:

– funkcja liniowa:

$$K(x_a, x_b) = x_a^T x_b + const \tag{2.49}$$

– funkcja wielomianowa:

$$K(x_a, x_b) = (\alpha x_a^T x_b + const)^d \tag{2.50}$$

gdzie d oznacza rząd wielomianu

– radialna funkcja bazowa (ang. *radial basis function*, RBF):

$$K(x_a, x_b) = exp(-\frac{|x_a - x_b|^2}{2\sigma^2}) \tag{2.51}$$

– funkcja sigmoidalna:

$$K(x_a, x_b) = tanh(x_a^T x_b + const) \tag{2.52}$$

2.3.4.2. Przyrostowa analiza składowych głównych

Przyrostowa analiza składowych głównych (ang. *incremental principal component analysis*, IncrementalPCA) umożliwia aktualizację wyznaczanych głównych składowych wraz z dodawaniem do zbioru danych nowych obiektów. Jest to szczególnie przydatne podczas pracy z dużymi zbiorami danych, w przypadku których niemożliwe jest równoczesne wczytanie całego zbioru ze względu na ograniczenia sprzętowe dotyczące pamięci. Kolejnym istotnym zastosowaniem przyrostowej metody PCA jest praca ze strumieniowymi bazami danych, w których napływ informacji jest ciągły i gwałtowny [50] – jej użycie jest więc zasadne w przypadku, gdy prowadzona redukcja wymiarowości ma posłużyć do budowy modelu klasyfikującego sygnał mowy będącego elementem systemu dokonującego analizy w czasie rzeczywistym.

2.3.5. Analiza czynnikowa

U podstaw analizy czynnikowej (ang. *factor analysis*, FA) leży założenie, że związki obserwowane pomiędzy powiązanymi wewnętrznie cechami mogą wynikać z istnienia czynników powiązanych z poszczególnymi cechami, będących źródłem wzajemnej informacji. Eksploracyjna analiza czynnikowa ma na celu wykrycie tych czynników i pogrupowanie cech w skupienia cechujące się wysoką korelacją obiektów należących do tego samego skupienia oraz niewielką korelacją obiektów należących do różnych skupień [51]. W przypadku zastosowania FA do redukcji wymiarowości wielowymiarowa przestrzeń cech może zostać zredukowana do przestrzeni o mniejszej liczbie wymiarów, tworzonej przez wyznaczone czynniki.

Załóżmy, że każdy obiekt w zbiorze danych opisany jest za pomocą n cech $x_i, i = 1, 2, \ldots, n$, które mogą zostać przedstawione jako kombinacja liniowa m czynników $Z_i, j = 1, 2, \ldots, m$:

$$X_i = \sum_{j=1}^{m} a_{ij} Z_j + b_i U_i \tag{2.53}$$

$$X_{nx1} = A_{nxm} Z_{mx1} + B_{nxn} U_{nx1} \tag{2.54}$$

$$B = diag(b_1, b_2, \ldots, b_n) \tag{2.55}$$

$$b_i^2 = 1 - h_i^2 \tag{2.56}$$

$$h_i^2 = \sum_{j=1}^{m} a_{ij}^2 \tag{2.57}$$

gdzie a_{ij} oznacza ładunek czynnikowy czynnika Z_j na cechę x_i, czyli nasilenie występowania j-tego czynnika w i-tej zmiennej, U_i jest czynnikiem specyficznym w zmiennej x_i, h_i^2 oznacza zasób wspólnej zmienności cechy x_i, zaś b_i^2 oznacza wariancję specyficzną odpowiadającą czynnikowi specyficznemu U_i [52].

Przyjmując założenie, że czynniki U_i i Z_j nie są skorelowane wewnętrznie oraz między sobą możemy zapisać macierz korelacji pomiędzy cechami opisującymi obiekt jako

$$R = A_{nxm}A_{mxn}^T + B_{nxn}^2 \tag{2.58}$$

Powyższe równanie stanowi równanie podstawowe analizy czynnikowej [52].

Istotną cechą analizy FA jest możliwość dokonania rotacji czynników wynikająca z faktu, że pomnożenie macierzy ładunków czynnikowych Z przez dowolną macierz ortogonalną T nie zmienia struktury macierzy korelacji R. Ortogonalna rotacja czynników prowadzi do uzyskania maksymalnego zróżnicowania ładunków w ramach danego czynnika, co powoduje prostszą interpretację wyników analizy [53]. Zastosowanie rotacji ukośnej, dokonywanej poprzez pomnożenie macierzy Z przez macierz nieortogonalną prowadzi natomiast do uzyskania czynników, które mogą być ze sobą skorelowane, dzięki czemu lepiej reprezentują złożoność analizowanych zmiennych [54].

2.3.6. Metoda *dictionary learning*

Metoda *dictionary learning* ma na celu znalezienie rzadkiej reprezentacji (ang. textitsparse representation) analizowanych danych. Wykorzystuje w tym celu macierz zwaną słownikiem (ang. textitdictionary) – zgodnie z zależnością przedstawioną w równaniu 2.59) wielowymiarowy wektor cech x może zostać aproksymowany jako iloczyn słownika D oraz wektora γ, stanowiącego reprezentację rzadką [55].

$$x \approx D\gamma \tag{2.59}$$

Wyróżniane są dwa rodzaje słowników:

- wyznaczane na drodze analitycznej – zaliczają się do nich słowniki wyznaczane zgodnie z formułą matematyczną, np. poprzez przeprowadzenie transformacji Fouriera lub transformacji falkowej,

- wyznaczane poprzez proces uczenia z użyciem dostępnych danych.

Słowniki analityczne cechują się mniejszą kosztownością obliczeniową ze względu istnienie efektywnych algorytmów ich wyznaczania, natomiast ich stała postać nie umożliwia adaptacji do analizowanych danych. Słowniki wyznaczane w oparciu o dostępne dane z użyciem technik uczenia maszynowego umożliwiają lepsze dopasowanie do danych, jednak operacje prowadzone z ich użyciem wiążą się z większymi nakładami obliczeniowymi [55].

Celem algorytmu dokonującego redukcji wymiarowości z użyciem metody *dictionary learning* jest znalezienie macierzy D oraz wektora γ gwarantujących jak najmniejszy błąd $||x - D\gamma||_F^2$, czyli zapewniających jak najlepszą aproksymację wektora x [56].

Rozwiązywany problem optymalizacyjny można zatem zapisać jako

$$\min_{D,\Gamma} ||X - D\Gamma||_F^2 + \lambda||\Gamma||_F^2 \tag{2.60}$$

gdzie λ jest parametrem regularyzacji określających równowagę pomiędzy dokładnością rekonstrukcji macierzy X oraz rzadkością elementów macierzy Γ, zaś $||.||_F$ oznacza normę macierzową Frobeniousa [57].

W przypadku, gdy metoda *dictionary learning* ma posłużyć do redukcji wymiarowości, prowadzone jest równoczesne rzutowanie danych na przestrzeń o mniejszej liczbie wymiarów oraz poszukiwanie pożądanego słownika D poprzez minimalizację poniższej funkcji

$$\min_{P,D,\Gamma} ||PX - D\Gamma||_F^2 + ||X - P^T PX||_F^2 + \lambda||\Gamma||_F^2 \tag{2.61}$$

gdzie P oznacza macierz transformacji użytą do zrzutowania danych na przestrzeń o mniejszej liczbie wymiarów [57].

2.4. Metody postępowania z danymi niezrównoważonymi

Częstym problemem występującym podczas uczenia modeli klasyfikacyjnych jest niezrównoważenie zbiorów danych, czyli duża dysproporcja w liczebności poszczególnych klas. Uczony na takich danych klasyfikator może mieć problemy z wyznaczeniem prawidłowych reguł klasyfikacyjnych, co skutkuje jego niewielką przydatnością w zastosowaniach praktycznych. Jedną z metod radzenia sobie z tym problemem jest przepróbkowanie zbioru danych, które ma na celu wyrównanie liczebności poszczególnych klas lub zmniejszenie jej dysproporcji. Algorytmy przepróbkowania zbioru danych można podzielić na dwie grupy: algorytmy nadpróbkowania oraz algorytmy podpróbkowania.

2.4.1. Algorytmy nadpróbkowania zbioru danych

Algorytmy nadpróbkowania zbioru danych prowadzą do uzyskania równolicznych klas poprzez generację syntetycznych obserwacji z klasy mniej licznej. Sposób generowania tych obserwacji zależny jest od konkretnego algorytmu. W dalszej części rozdziału opisano najpopularniejsze metody nadpróbkowania danych, zaimplementowane w bibliotece *imblearn* [58].

2.4.1.1. Random Minority Oversampling

Random Minority Oversampling (RMO) jest podstawową, najprostszą w implementacji metodą nadpróbkowania danych. Polega na powielaniu obserwacji należących do klasy mniej licznej – spośród obserwacji należących do klasy mniej licznej losowany jest obiekt i tworzony jest

nowy, syntetyczny obiekt będący jego kopią, który dodawany jest do zbioru danych. Operacja ta powtarzana jest do momentu, w którym liczebność klas zrówna się lub zostanie osiągnięty inny sprecyzowany stosunek liczebności klas.

Zastosowanie metody RMO prowadzi do zwiększenia dokładności klasyfikacji modelu, jednakże należy mieć na uwadze fakt, że może również spowodować przeuczenie modelu. Losowy wybór powielanych obiektów może skutkować uzyskaniem wielu obiektów opisanych identycznym zestawem cech. Klasyfikator uczony na takim zbiorze danych może nadmiernie dopasować się do rozkładu wartości cech obiektów przykładając nadmierną wagę do tych obiektów, które zostały powielone największą liczbę razy. To może z kolei doprowadzić do uzyskania modelu uzyskującego niesatysfakcjonujące wyniki na zbiorze testowym i, w konsekwencji, nieprzydatnego w rzeczywistych zastosowaniach [59].

2.4.1.2. SMOTE

Zastosowanie metody *SMOTE* (ang. *Synthetic Minority Over-sampling Technique*) skutkuje uzyskaniem nowych, syntetycznych obserwacji należących do klasy mniej licznej, różniących się od obserwacji rzeczywistych. Obserwacje syntetyczne generowane są zgodnie z następującym algorytmem [60]:

1. Wyznaczenie sumarycznej liczby obserwacji syntetycznych G, które należy wygenerować.

2. Określenie liczby najbliższych sąsiadów k, którzy posłużą do wyznaczenia obserwacji syntetycznej na podstawie wybranego obiektu oraz liczby syntetycznych obiektów N, które mają zostać wyznaczone na podstawie każdego obiektu należącego do zbioru A obiektów z klasy mniej licznej.

3. Dla każdego obiektu $\overrightarrow{a_i}$ należącego do zbioru A:

   (a) Wyznaczenie k najbliższych sąsiadów obiektu $\overrightarrow{a_i}$ poprzez obliczenie odległości pomiędzy nim i każdym innym obiektem ze zbioru A i wybór k obiektów o najmniejszej odległości.

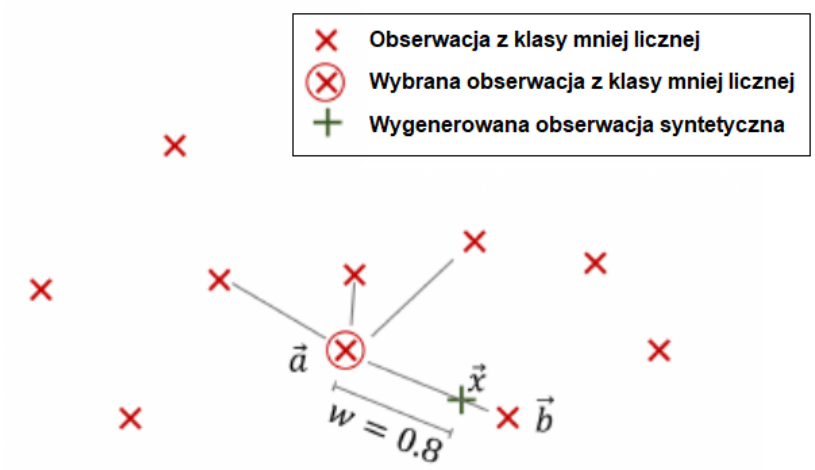
   (b) Wybór losowego obiektu $\overrightarrow{b_i}$ należącego do najbliższych sąsiadów rozważanego obiektu $\overrightarrow{a_i}$.

   (c) Utworzenie nowego obiektu zgodnie z równaniem:

   $$\overrightarrow{x_i} = \overrightarrow{a_i} + w \cdot (\overrightarrow{b_i} - \overrightarrow{a_i}) \tag{2.62}$$

   gdzie gdzie $\overrightarrow{b_i} - \overrightarrow{a_i}$ jest różnicą wektorów cech opisujących obiekty $\overrightarrow{b_i}$ i $\overrightarrow{a_i}$, zaś w oznacza wagę o losowej wartości należącej do przedziału [0, 1] (Rys. 2.5).

   (d) N-krotne powtórzenie kroków 3b-3c.



Rys. 2.5. Schemat tworzenia obserwacji syntetycznych z użyciem metody SMOTE. Źródło: [61].

Główną zaletą metody *SMOTE* w stosunku do metody RMO jest fakt, że uczenie modelu klasyfikacyjnego na przepróbkowanym w ten sposób zbiorze danych skutkuje uzyskaniem klasyfikatora o większych obszarach decyzyjnych związanych z klasą mniej liczną [60].

Do wad metody *SMOTE* można natomiast zaliczyć możliwość zwiększenia szumu obecnego w danych. Jest to szczególnie prawdopodobne, gdy pojedyncze obiekty z klasy mniej licznej znajdują się w obrębie skupienia dużej liczby obiektów z klasy bardziej licznej. Wygenerowane na ich podstawie obserwacje syntetyczne przyczyniają się wówczas do zwiększenia zawartości szumu w zbiorze danych [61].

Kolejną wadą metody *SMOTE* jest fakt, że jest to metoda nieuwzględniająca niezrównoważenia wewnątrzklasowego występującego w sytuacji, gdy zbiór danych jest niereprezentatywny w niektórych obszarach przestrzeni cech. W przypadku, gdy obiekty nie są równomiernie rozłożone w przestrzeni cech i można zaobserwować obszary o dużym i małym zagęszczeniu obiektów, po przeprowadzeniu nadpróbkowania w obszarach o dużym zagęszczeniu znajdzie się duża liczba obiektów syntetycznych, a w obszarach o małym zagęszczeniu niewielka liczba obiektów syntetycznych. To doprowadzi z kolei do jeszcze większej nierównomierności rozmieszczenia obiektów w przestrzeni cech, co może doprowadzić do uzyskania klasyfikatora o nieoptymalnych granicach decyzyjnych [61].

Należy również zwrócić uwagę na fakt, że dla każdego obiektu należącego do klasy mniej licznej generowana jest taka sama liczba obserwacji syntetycznych, bez względu na ich odległość od granicy decyzyjnej klasyfikatora, co nie jest zjawiskiem korzystnym [61].

2.4.1.3. Borderline-SMOTE

Algorytm *Borderline-SMOTE* jest modyfikacją algorytmu *SMOTE*, która uwzględnia odległość obiektów od granic decyzyjnych klasyfikatora. Generowanie obiektów syntetycznych przebiega zgodnie z następującym algorytmem [62]:

1. Wyznaczenie sumarycznej liczby obserwacji syntetycznych G, które należy wygenerować.

2. Wyznaczenie m najbliższych sąsiadów dla każdego obiektu obiektu $\overrightarrow{a_i}$ należącego do zbioru A obiektów z klasy mniej licznej i określenie liczby sąsiadów należących do klasy bardziej licznej m'.

3. Jeżeli $m' = m$, $\overrightarrow{a_i}$ uznawany jest za szum i dalsze kroki nie są dla niego wykonywane. Jeżeli $0 \leq m' \leq m/2$, $\overrightarrow{a_i}$ uznawany jest za obiekt o małym ryzyku nieprawidłowego zaklasyfikowania i dalsze kroki nie są dla niego wykonywane. Jeżeli $m/2 \leq m' \leq m$, $\overrightarrow{a_i}$ uznawany jest za obiekt o dużym ryzyku nieprawidłowego zaklasyfikowania i zostaje on dodany do zbioru obiektów, które zostaną poddane nadpróbkowaniu A'.

4. Określenie liczby obiektów N, które mają zostać wygenerowane dla każdego obiektu $\overrightarrow{a_i}$ należącego do zbioru A'.

5. Wykonanie kroków 3b-3d algorytmu SMOTE (podrozdział 2.4.1.2) dla każdego obiektu należącego do zbioru A'.

Uzyskane w ten sposób obserwacje syntetyczne zlokalizowane są w pobliżu granicy decyzyjnej klasyfikatora, co prowadzi do zwiększenia prawdopodobieństwa poprawnego zaklasyfikowania obiektów granicznych [62].

2.4.1.4. KMeans-SMOTE

Algorytm *KMeans-SMOTE* wykorzystuje algorytm k-średnich do wyznaczenia skupień obiektów należących do klasy mniej licznej w przestrzeni cech oraz określenia liczby próbek syntetycznych, które powinny zostać wygenerowane dla poszczególnych skupień. Zasadę działania metody *KMeans-SMOTE* można opisać za pomocą poniższego algorytmu [61]:

1. Wyznaczenie sumarycznej liczby obserwacji syntetycznych G, które należy wygenerować.

2. Wyznaczenie skupień obiektów za pomocą algorytmu k-średnich:

   (a) Określenie liczby skupień k_m.

   (b) Losowy wybór k_m obiektów, które uznawane są za środki ciężkości (centroidy) skupień.

   (c) Obliczenie odległości pomiędzy środkami ciężkości a pozostałymi obiektami.

   (d) Przypisanie obiektów do skupień, których centroidy znajdują się najbliżej.

   (e) Ustalenie nowych środków ciężkości na podstawie utworzonych skupień.

   (f) Ponowne obliczenie odległości pomiędzy obiektami a środkami ciężkości i w razie konieczności zmiana przypisania obiektów do skupień.

   (g) Powtórzenie kroków 1d-1f do momentu uzyskania stabilnych rezultatów lub wykonania ustalonej liczby iteracji.

3. Określenie gęstości poszczególnych skupień:

(a) Obliczenie proporcji obiektów należących do klasy mniej licznej do wszystkich obiektów należących do danego skupienia oraz porównanie jej z wartością progową c (domyślnie wynoszącą 0,5).

(b) W przypadku przekroczenia wartości progowej (to znaczy, gdy obiekty z klasy mniej licznej stanowią nie mniejszą niż c część wszystkich obiektów należących do skupienia), należy przejść do kroku 2c. W przeciwnym przypadku skupienie nie zostanie poddane nadpróbkowaniu i nie należy wykonywać kroku 2c.

(c) Obliczenie liczby obiektów syntetycznych, które zostaną wygenerowane dla danego skupienia:

i. Wyznaczenie macierzy odległości pomiędzy wszystkimi obiektami z klasy mniej licznej należącymi do skupienia.

ii. Obliczenie średniej odległości d_n poprzez zsumowanie odległości pomiędzy wszystkimi obiektami i podzielenie jej przez liczbę obiektów.

iii. Wyznaczenie miary gęstości skupienia D_{k_m} poprzez podzielenie liczby należących do niej obiektów z klasy mniej licznej $n_{minority}$ przez średnią odległość d podniesioną do potęgi m, gdzie m oznacza liczbę cech opisujących obiekt:

$$D_{k_m} = \frac{n_{minority}}{d_n^m} \tag{2.63}$$

(d) Powtórzenie kroków 3a-3c dla wszystkich skupień.

4. Wyznaczenie liczby obiektów syntetycznych, które zostaną wygenerowane dla k_m-tego skupienia zgodnie z zależnością $N = w * G$, gdzie $W = \frac{1}{D_{k_m}}$. 5. Wykonanie kroków 3-7 algorytmu SMOTE (podrozdział 2.4.1.2).

Algorytm *KMeans-SMOTE* prowadzi do zwiększenia zrównoważenia wewnątrzklasowego wynikowego zbioru danych w stosunku do zbioru uzyskanego za pomocą klasycznego algorytmu SMOTE. Dzięki wyznaczaniu liczby obiektów syntetycznych, które zostaną wygenerowane dla danego skupienia w oparciu o gęstość skupienia uzyskiwana jest różna liczba obserwacji syntetycznych w każdym ze skupień: dla skupień cechujących się dużą gęstością liczba syntetycznych obserwacji będzie mniejsza, zaś dla skupień rzadkich większa. Niepoddawanie przepróbkowaniu skupień o niewielkim udziale klasy mniej licznej pozwala natomiast zminimalizować ryzyko zwiększenia szumu obecnego w danych [61].

2.4.1.5. SVM-SMOTE

Algorytm *SVM-SMOTE* wykorzystuje maszynę wektorów nośnych (SVM) do wyznaczenia obserwacji, na podstawie których należy dokonać przepróbkowania – klasyfikator SVM trenowany jest na danych nieprzepróbkowanych, a obszary poddawane przepróbkowaniu wyznaczane są wzdłuż granic decyzyjnych uzyskanego modelu. Obserwacje syntetyczne generowane są zgodnie z następującym algorytmem [63]:

1. Wyznaczenie liczby syntetycznych obserwacji G, które należy wygenerować.

2. Trening modelu SVM na nieprzepróbkowanym zbiorze danych.

3. Wyznaczenie zbioru wektorów nośnych dla każdego obiektu z klasy mniej licznej.

4. Wyznaczenie liczby obserwacji syntetycznych N, które mają zostać wyznaczone dla pojedynczego wektora nośnego poprzez podzielenie liczby G przez liczę wektorów nośnych SV.

5. Wyznaczenie k najbliższych sąsiadów należących do klasy mniej licznej dla wybranego wektora nośnego $\overrightarrow{SV_i}$.

6. Wybór najbliższego obiektu $\overrightarrow{b_i}$ należącego do najbliższych sąsiadów rozważanego wektora nośnego $\overrightarrow{SV_i}$.

7. Wyznaczenie m najbliższych sąsiadów dla wybranego wektora nośnego $\overrightarrow{SV_i}$ należących do dowolnej klasy. Jeżeli liczba najbliższych sąsiadów należących do klasy bardziej licznej $< m/2$, obserwacja syntetyczna wyznaczana jest poprzez ekstrapolację zgodnie z zależnością (rys. 2.6c):
$$\overrightarrow{x_i} = \overrightarrow{SV_i} + w \cdot (\overrightarrow{SV_i} - \overrightarrow{b_i}) \tag{2.64}$$
W przeciwnym przypadku obserwacja syntetyczna wyznaczana jest poprzez interpolację zgodnie z zależnością (rys. 2.6b):
$$\overrightarrow{x_i} = \overrightarrow{SV_i} + w \cdot (\overrightarrow{b_i} - \overrightarrow{SV_i}) \tag{2.65}$$
gdzie w jest losową wartością z przedziału [0,1].

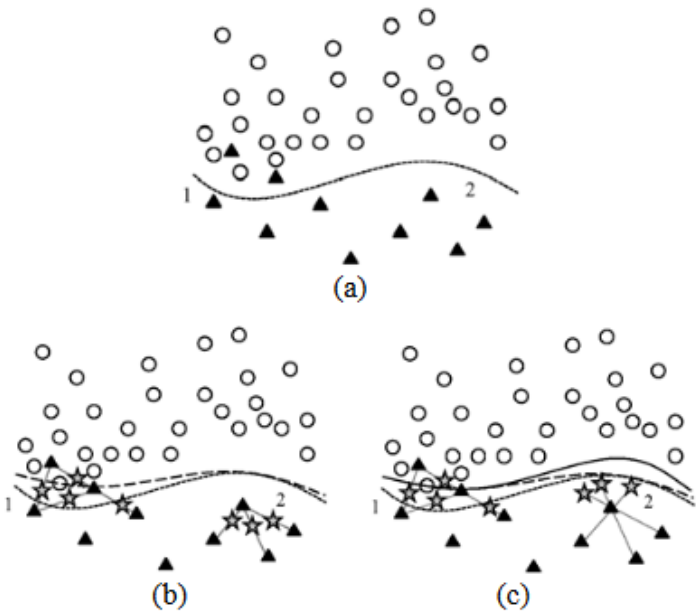
8. N-krotne wybranie kolejnego najbliższego sąsiada wektora nośnego $\overrightarrow{SV_i}$ i wykonanie kroku 7.

Zastosowanie ekstrapolacji w przypadku, gdy większość najbliższych sąsiadów danego wektora nośnego należy do klasy mniej licznej umożliwia zwiększenie obszaru decyzyjnego klasyfikatora odpowiadającego klasie mniej licznej w stosunku do obszaru uzyskanego z użyciem interpolacji, jak ma to miejsce w klasycznej metodzie *SMOTE* [63].

2.4.1.6. ADASYN

ADASYN jest adaptacyjnym algorytmem nadpróbkowania danych, w którym liczba obserwacji syntetycznych generowanych w danym obszarze przestrzeni cech uzależniona jest od rozkładu obiektów w tej przestrzeni. Obserwacje syntetyczne tworzone są na podstawie następującego algorytmu [64]:

1. Wyznaczenie sumarycznej liczby obiektów, które należy wygenerować zgodnie z zależnością:
$$G = (n_{majority} - n_{minority})\beta \tag{2.66}$$



Rys. 2.6. Schemat tworzenia obserwacji syntetycznych z użyciem metody SVM-SMOTE: a. dane nieprzepróbkowane, b. utworzenie danych syntetycznych z użyciem interpolacji, c. utworzenie danych syntetycznych z użyciem ekstrapolacji. Źródło: [63].

gdzie $\beta \in [0, 1]$ jest parametrem określającym pożądany stosunek liczebności klas, $n_{minority}$ oznacza liczbę obiektów należących do klasy mniej licznej, zaś $n_{majority}$ oznacza liczbę obiektów należących do klasy bardziej licznej.

2. Wyznaczenie k najbliższych sąsiadów dla każdej obserwacji z klasy mniej licznej i obliczenie stosunku r_i jej liczby sąsiadów z klasy bardziej licznej do całkowitej liczby sąsiadów k.

3. Normalizacja wartości r_i zgodnie z zależnością:

$$\hat{r_i} = \frac{r_i}{\sum_{i=1}^{n_{minority}} r_i} \tag{2.67}$$

4. Obliczenie liczby syntetycznych obserwacji, które należy wygenerować dla każdego obiektu z klasy mniej licznej:

$$N = \hat{r_i} \cdot G \tag{2.68}$$

5. Wygenerowanie odpowiedniej liczby N syntetycznych obserwacji dla każdego obiektu $\overrightarrow{a_i}$ z klasy mniej licznej:

   (a) Losowy wybór obiektu $\overrightarrow{b_i}$ spośród k najbliższych sąsiadów obiektu $\overrightarrow{a_i}$.

   (b) Utworzenie obserwacji syntetycznej zgodnie z zależnością:

$$\overrightarrow{x_i} = \overrightarrow{a_i} + w(\overrightarrow{b_i} - \overrightarrow{a_i}) \tag{2.69}$$

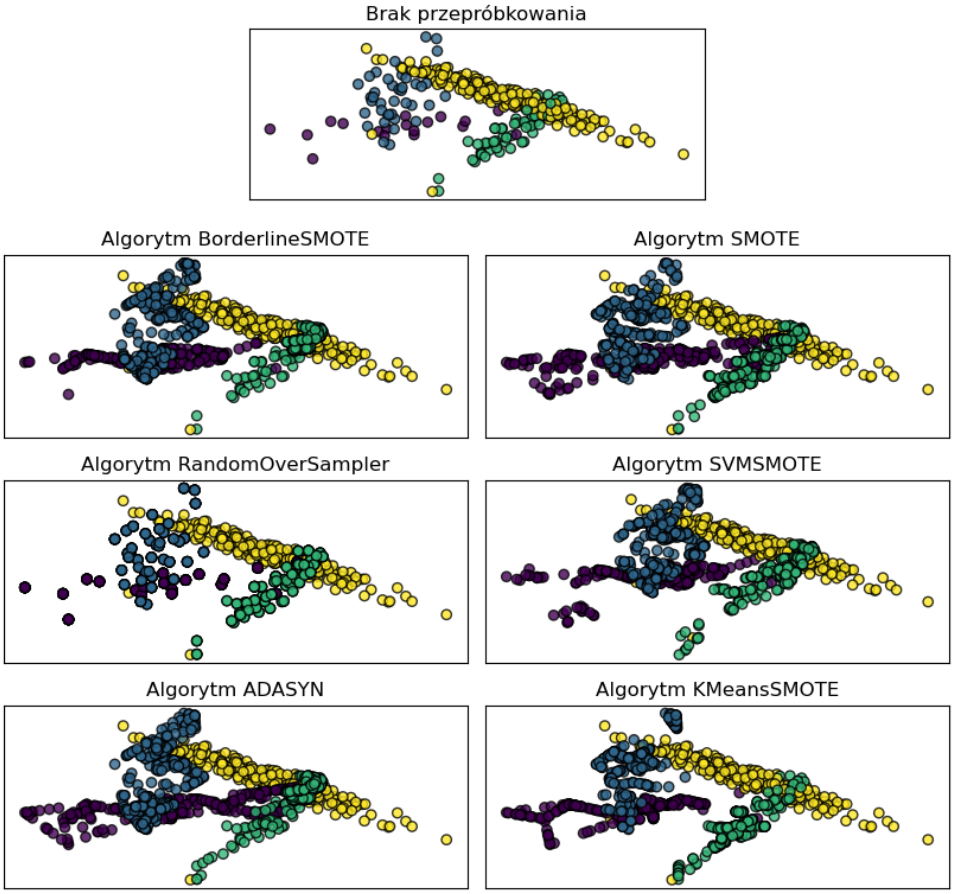
gdzie $\overrightarrow{b_i} - \overrightarrow{a_i}$ jest różnicą wektorów cech opisujących obiekty $\overrightarrow{b_i}$ i $\overrightarrow{a_i}$, zaś w jest losową wartością z przedziału [0, 1].

(c) N-krotne powtórzenie kroków 5a-5b.

Dzięki zastosowaniu adaptacyjnego wyznaczania liczby obserwacji syntetycznych generowanych na podstawie pojedynczej obserwacji rzeczywistej uzyskiwany jest zbiór danych, w którym większa liczba obiektów z klasy mniej licznej zostaje dodana w pobliżu obserwacji rzeczywistych trudniejszych do poprawnej klasyfikacji, tzn. takich, których duża liczba najbliższych sąsiadów należy do klasy bardziej licznej [64].

2.4.1.7. Porównanie metod nadpróbkowania danych

Różnice w zbiorach danych uzyskanych na drodze nadpróbkowania z użyciem opisanych powyżej algorytmów przedstawiono na rys. 2.7.



Rys. 2.7. Przykładowy nieprzepróbkowany zbiór danych oraz zbiory danych uzyskane na drodze nadpróbkowania z użyciem sześciu algorytmów: algorytmu ADASYN oraz pięciu wariantów algorytmu SMOTE.

Różnice w rozmieszczeniu obiektów syntetycznych w przestrzeni cech skutkuje zmianą położenia granic decyzyjnych modeli klasyfikacyjnych uczonych na zbiorach danych przepróbkowanych z użyciem różnych algorytmów. Na rys. 2.8-2.9 przedstawiono granice decyzyjne modelu regresji logistycznej oraz modelu SVM uzyskane dla zbiorów danych przedstawionych na rys. 2.7.

Jak można zauważyć na rys. 2.8-2.9 w przypadku modeli liniowych różnice w przebiegu granic decyzyjnych mogą być stosunkowo niewielkie, jednak w przypadku modelu SVM operującego na przestrzeni $N + 1$-wymiarowej, gdzie N to liczba cech opisujących każdy obiekt, granice zrzutowane na przestrzeń N-wymiarową różnią się w sposób znaczący. Oznacza to, że dobór algorytmu nadpróbkowania jest zagadnieniem istotnym i w przypadku każdego modelu klasyfikacyjnego konieczne jest osobne dobranie algorytmu nadpróbkowania pozwalającego uzyskać najbardziej satysfakcjonujące rezultaty.

2.4.2. Algorytmy podpróbkowania zbioru danych

Algorytmy podpróbkowania zbioru danych zazwyczaj nie prowadzą do uzyskania klas o takiej samej liczebności, pozwalają natomiast zredukować liczbę obiektów należących do klasy bardziej licznej. W zależności od stosowanego algorytmu wynikowy stosunek liczebności klas może być różny – wspólnym celem realizowanym przez wszystkie algorytmy z tej grupy jest natomiast taki wybór obiektów z klasy mniej licznej, których usunięcie nie pogorszy lub wręcz poprawi zdolności klasyfikacyjne uczonego na nich modelu. W dalszej części rozdziału przedstawiono algorytmy podpróbkowania zbioru danych dostępne w bibliotece *imblearn* [58].

2.4.2.1. Random Majority Undersampling

Random Majority Undersampling (RMU) jest metodą podpróbkowania danych polegającą na usuwaniu ze zbioru danych losowych obserwacji należących do klasy bardziej licznej. Wynikiem działania algorytmu RMU jest zrównoważony zbiór uczący, zawierający taką samą liczbę obserwacji należących do wszystkich występujących w zbiorze klas [59].

Zaletą metody RMU jest przede wszystkim jej prostota i łatwość implementacji. Ponadto zastosowanie tej metody powoduje zmniejszenie rozmiaru zbioru uczącego, co przyspiesza proces uczenia modelu. Należy jednak zwrócić uwagę, że usunięcie ze zbioru danych losowych obserwacji może spowodować utratę informacji i wzorców możliwych do zaobserwowania w pełnym zbiorze danych. To może z kolei doprowadzić do wyznaczenia nieprawidłowych reguł klasyfikacyjnych przez model, a w konsekwencji jego zmniejszoną przydatność [59].

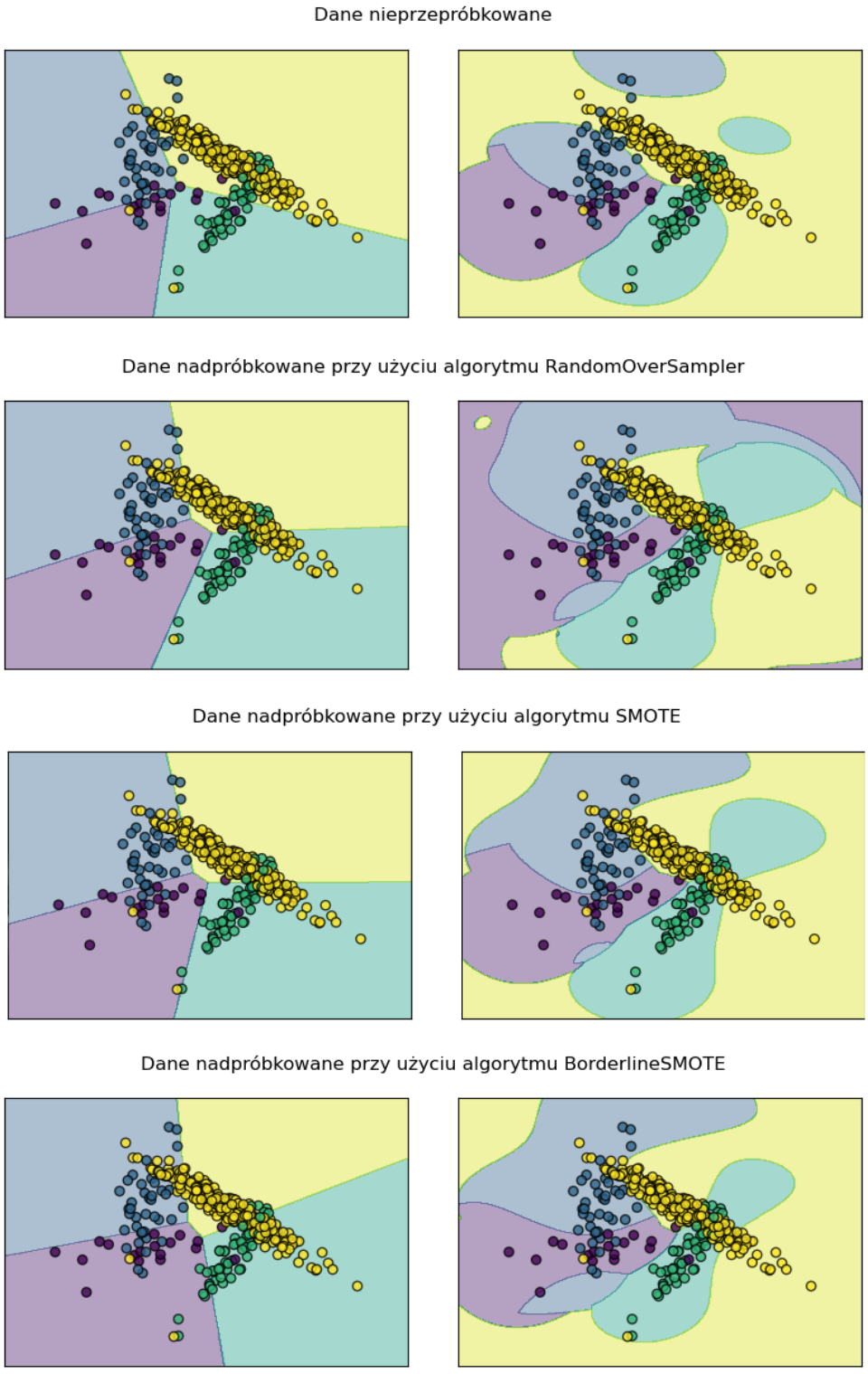
2.4.2.2. Cluster Centroids

Algorytm *Cluster Centroids* umożliwia uzyskanie idealnie zrównoważonego zbioru danych. Selekcja obiektów, które powinny zostać usunięte z oryginalnego zbioru danych dokonywana jest przy użyciu algorytmu k-średnich. Procedurę tworzenia podpróbkowanego zbioru danych można opisać w następujący sposób [65]:

1. Wyznaczenie liczby obserwacji należących do klasy mniej licznej $n_{minority}$ oraz liczby obiektów należących do klasy bardziej licznej $n_{majority}$.

2. Wyznaczenie skupień obiektów należących do klasy bardziej licznej za pomocą algorytmu k-średnich zgodnie z procedurą opisaną w sekcji 2.4.1.4.

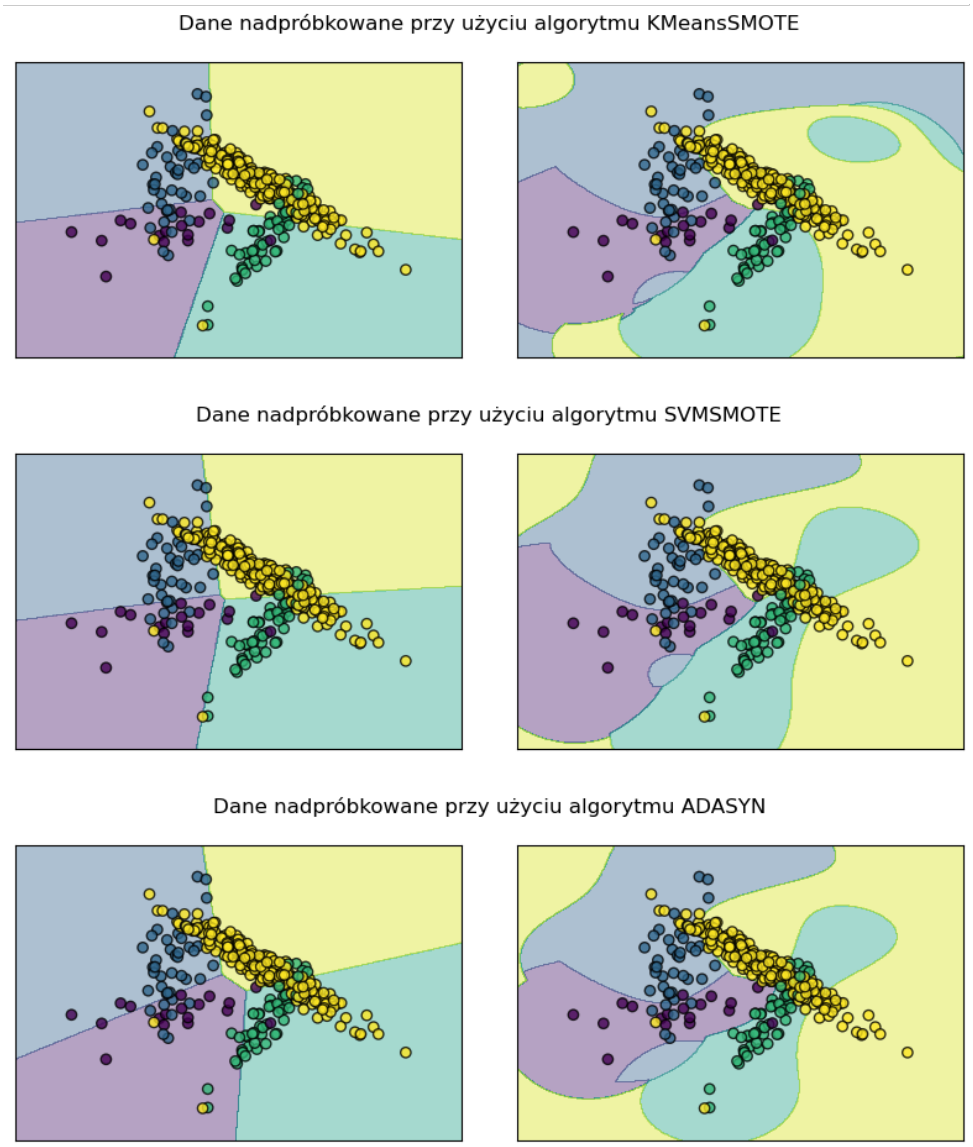
3. Dla każdego z k wyznaczonych skupień:



Rys. 2.8. Granice decyzyjne modelu regresji logistycznej (po lewej) oraz modelu SVM (po prawej) uczonych na zbiorze nieprzepróbkowanym oraz na zbiorach przepróbkowanych z użyciem różnych algorytmów nadpróbkowania danych.

(a) Wyznaczenie stosunku liczebności obiektów należących do skupienia n_k do całkowitej liczby obiektów z klasy bardziej licznej:

$$r_i = \frac{n_k}{n_{majority}} \tag{2.70}$$



Rys. 2.9. Granice decyzyjne modelu regresji logistycznej (po lewej) oraz modelu SVM (po prawej) uczonych na zbiorach przepróbkowanych z użyciem różnych algorytmów nadpróbkowania danych.

(b) Obliczenie odległości euklidesowej pomiędzy obiektami należącymi do danego skupienia a środkiem ciężkości skupienia.

(c) Zachowanie r_i obiektów o najmniejszej odległości od środka ciężkości oraz usunięcie pozostałych obiektów ze skupienia.

W przypadku wyboru liczby skupień k równej liczbie obiektów należących do klasy mniej licznej $n_{minority}$, wynikowy zbiór danych zawiera obiekty z klasy mniej licznej oraz środki ciężkości wyznaczonych skupień [58].

2.4.2.3. Near Miss

Algorytm *Near Miss* występuje w trzech wersjach różniących się między sobą sposobem wyboru obiektów, które należy pozostawić w zbiorze danych [66]:

– *Near Miss 1* - pozostawione zostają obiekty z klasy bardziej licznej, których średnia odległość od k najbliższych obserwacji z klasy mniej licznej jest najmniejsza (rys. 2.10a),

– *Near Miss 2* - pozostawione zostają obiekty z klasy bardziej licznej, których średnia odległość od k najdalej położonych obserwacji z klasy mniej licznej jest najmniejsza (rys. 2.10b),

– *Near Miss 3* - pozostawione zostają obiekty z klasy bardziej licznej wybierane na drodze selekcji dwuetapowej:

1. dla każdego obiektu z klasy mniej licznej pozostawiane jest m najbliższych sąsiadów,
2. pozostawione zostają obiekty z klasy bardziej licznej należące do zbioru utworzonego w punkcie 1, których średnia odległość od k najbliższych sąsiadów z klasy mniej licznej jest największa (rys. 2.10c).

2.4.2.4. Tomek Links

Algorytm *Tomek Links* pozwala na usunięcie ze zbioru danych obiektów znajdujących się w pobliżu granicy decyzyjnych klasyfikatora oraz obserwacji, które znajdują się w otoczeniu obserwacji należących do innej klasy. Algorytm ten wykorzystuje podział zbioru danych na dwa typy obserwacji [67]:

– zbiór obserwacji, których najbliższy sąsiad należy do tej samej klasy,

– zbiór obserwacji, których najbliższy sąsiad należy do innej klasy.

Obserwacje należące do drugiej grupy określane są mianem *Tomek link* i są usuwane ze zbioru danych podczas procesu podpróbkowania zgodnie z jedną z dwóch strategii:

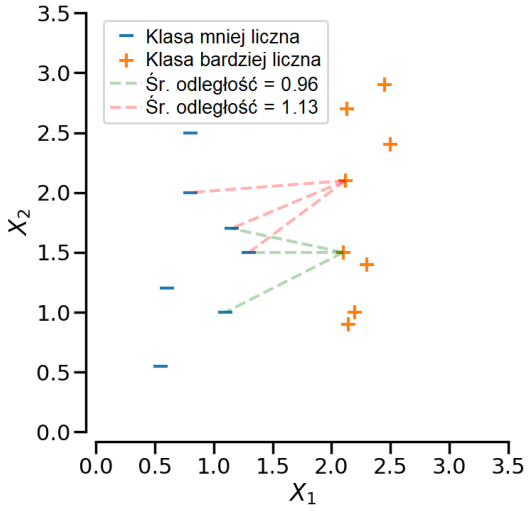
– usuwane są obie obserwacje wchodzące w skład pary,

– usuwana jest wyłącznie obserwacja należąca do klasy bardziej licznej.

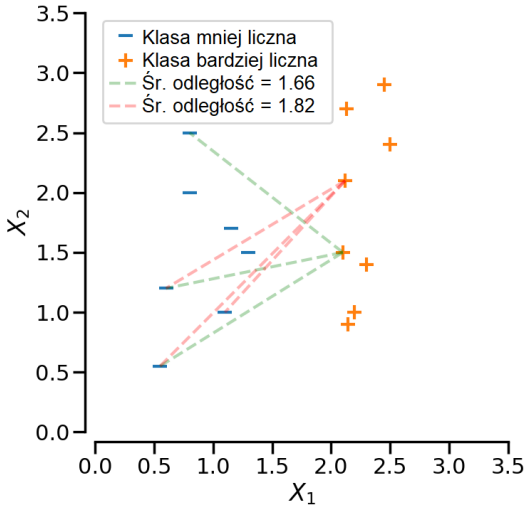
2.4.2.5. Edited Nearest Neighbours

Algorytm *Edited Nearest Neighbours* (ENN) pozwala na usunięcie ze zbioru danych obserwacji należących do klasy bardziej licznej, które mogą zostać uznane za obserwacje odstające. Detekcja obserwacji, które należy usunąć ze zbioru przeprowadzana jest z użyciem algorytmu *k* najbliższych sąsiadów. Działanie metody ENN można przedstawić za pomocą następującego algorytmu [68], [69]:

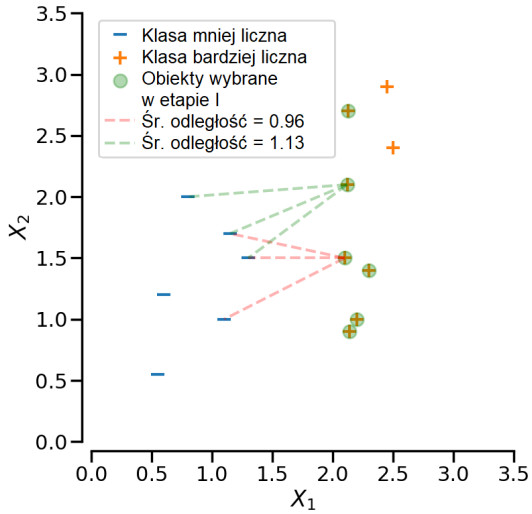
1. Utworzenie kopii zbioru danych.
2. Określenie liczby najbliższych sąsiadów k, którzy posłużą do oceny, czy obiekt ma zostać usunięty ze zbioru danych.



(a) Near Miss 1



(b) Near Miss 2



(c) Near Miss 3

Rys. 2.10. Schemat wyboru obiektów pozostawianych w zbiorze danych po przepróbkowaniu z użyciem trzech wariantach algorytmu Near Miss. Źródło: [58].

3. Dla każdego obiektu $\overrightarrow{b_i}$ należącego do klasy bardziej licznej B:
   (a) Wyznaczenie k najbliższych sąsiadów obiektu $\overrightarrow{b_i}$.
   (b) Jeżeli większość sąsiadów obiektu $\overrightarrow{b_i}$ należy do klasy bardziej licznej B, obiekt należy pozostawić w zbiorze danych. W przeciwnym przypadku obiekt $\overrightarrow{b_i}$ należy usunąć ze zbioru danych.

Implementacja algorytmu ENN w pakiecie *imblearn* umożliwia ponadto wybór klasy, która ma zostać poddana podpróbkowaniu - domyślnie jest to klasa bardziej liczna w przypadku klasyfikacji binarnej lub każda klasa z wyjątkiem klasy najmniej licznej w przypadku problemu klasyfikacji wieloklasowej. Inne dostępne opcje umożliwiają m.in. poddanie podpróbkowaniu wszystkich klas, w tym klasy najmniej licznej – takie podejście umożliwia usunięcie ze zbioru

danych obserwacji odstających, które mogą zostać uznane za szum informacyjny i znacząco pogarszać jakość zbioru danych [58], [68].

2.4.2.6. Repeated Edited Nearest Neighbours

Algorytm *Repeated Edited Nearest Neighbours* (RENN) jest rozszerzeniem algorytmu ENN (sekcja 2.4.2.5). Polega on na wielokrotnym, iteracyjnym zastosowaniu metody ENN do momentu, gdy nie jest już możliwe usunięcie kolejnych obiektów, tzn. dla każdego obiektu z klasy bardziej licznej spełnione jest założenie, że większość spośród jego k sąsiadów należy do klasy bardziej licznej [58], [70].

Zastosowanie metody RENN skutkuje usunięciem wszystkich obserwacji z klasy bardziej licznej, które mogą zostać uznane za obserwacje odstające, jednakże wiąże się z uzyskaniem mniej licznego zbioru danych oraz większą utratą informacji, niż ma to miejsce w przypadku algorytmu ENN [58].

2.4.2.7. AllKNN

Algorytm *AllKNN* jest kolejną modyfikacją algorytmu ENN (sekcja 2.4.2.5). Polega on na wielokrotnym wykonywaniu algorytmu ENN na zbiorze danych, jednak w każdej kolejnej iteracji liczba sąsiadów k wykorzystywana do detekcji obserwacji, które należy usunąć jest zwiększana [58].

2.4.2.8. Condensed Nearest Neighbours

Zastosowanie algorytmu *Condensed Nearest Neighbours* (CNN) ma na celu uzyskanie takiego zbioru danych, w którym każda z obserwacji należy do tej samej klasy, co jej najbliższy sąsiad [71]. Zasadę działania algorytmu CNN można przedstawić następująco [72]:

1. Wybór losowego obiektu ze zbioru danych U i określenie klasy, do której przynależy.
2. Przeszukanie zbioru utworzonego poprzez odjęcie od oryginalnego zbioru danych zbioru U, w celu znalezienia obiektu, którego najbliższy sąsiad należący do zbioru U należy do innej klasy.
3. Dodanie znalezionego obiektu do zbioru U.
4. Powtórzenie kroków 2-3 do momentu, w którym dodanie nowych obiektów do zbioru U nie jest możliwe.

Metoda CNN cechuje się nieodtwarzalnością uzyskanego przepróbkowanego zbioru danych ze względu na losowy charakter poszukiwań puktów w kroku 2 przedstawionej powyżej procedury. Ponadto ze względu na iteracyjny charakter oznaczający wielokrotne przeszukanie zbioru danych jest to algorytm stosunkowo czasochłonny [71].

Implementacja algorytmu CNN w pakiecie *imblearn* pozwala przeprowadzić podpróbkowanie w sposób zachowujący wszystkie obiekty należące do klasy mniej licznej. Co więcej, umożliwia również zachowanie odtwarzalności wyników ze względu na możliwość zdefiniowania ziarna losowości wewnętrznego generatora liczb pseudolosowych [58].

2.4.2.9. One Sided Selection

Algorytm *One Sided Selection* (OSS) umożliwia usunięcie z oryginalnego zbioru danych obserwacji redundantnych oraz takich, które mogą zostać uznane za szum. Dobór obserwacji, które powinny zostać usunięte przeprowadzany jest zgodnie z poniższym algorytmem [73]:

1. Utworzenie zbioru C zawierającego obserwacje z klasy mniej licznej oraz jedną, losowo wybraną obserwację z klasy bardziej licznej.
2. Trening modelu 1-NN (klasyfikatora k najbliższych sąsiadów, k=1) na zbiorze C.
3. Przeprowadzenie klasyfikacji wszystkich obiektów należących do zbioru oryginalnego.
4. Dodanie do zbioru C obserwacji z klasy bardziej licznej, które zostały zaklasyfikowane nieprawidłowo.
5. Usunięcie ze zbioru C obiektów granicznych wyznaczonych przy użyciu metody $TomekLinks$ opisanej w sekcji 2.4.2.4.

Zastosowanie algorytmu OSS nie prowadzi do uzyskania idealnie zrównoważonego zbioru danych, jednak dzięki ograniczeniu liczby obiektów w zbiorze pozwala zmniejszyć czas treningu modelu [74].

2.4.2.10. Neighbourhood Cleaning Rule

Algorytm *Neighbourhood Cleaning Rule* (NCR) może zostać uznany za kolejne rozszerzenie algorytmu ENN. Proces podpróbkowania zbioru danych z jego użyciem można opisać za pomocą następującego algorytmu [75]:

1. Podział zbioru danych na zbiór C zawierający obiekty należące do klasy, która nie ma być poddawana przepróbkowaniu oraz zbiór O zawierający obiekty należące do pozostałych klas.
2. Identyfikacja obserwacji odstających w zbiorze O za pomocą algorytmu ENN opisanego w sekcji 2.4.2.5 i utworzenie z nich zbioru D_1.
3. Dla każdej klasy O_i należącej do zbioru O:
   (a) Wyznaczenie liczebności klasy.
   (b) Sprawdzenie, czy liczebność tej klasy jest nie mniejsza niż niż połowa liczebności klasy C. Jeżeli warunek jest spełniony, należy wykonać krok 3.3. W przeciwnym przypadku należy przejść do kolejnej klasy O_i.
   (c) Wyznaczenie k najbliższych sąsiadów każdego obiektu należącego do zbioru C, który jest klasyfikowany nieprawidłowo i identyfikacja wśród nich obiektów należących do klasy O_i.
   (d) Umieszczenie wybranych w poprzednim kroku obiektów w zbiorze D_2.

4. Utworzenie przepróbkowanego zbioru danych poprzez usunięcie z oryginalnego zbioru danych obiektów należących do zbiorów D_1 oraz D_2.

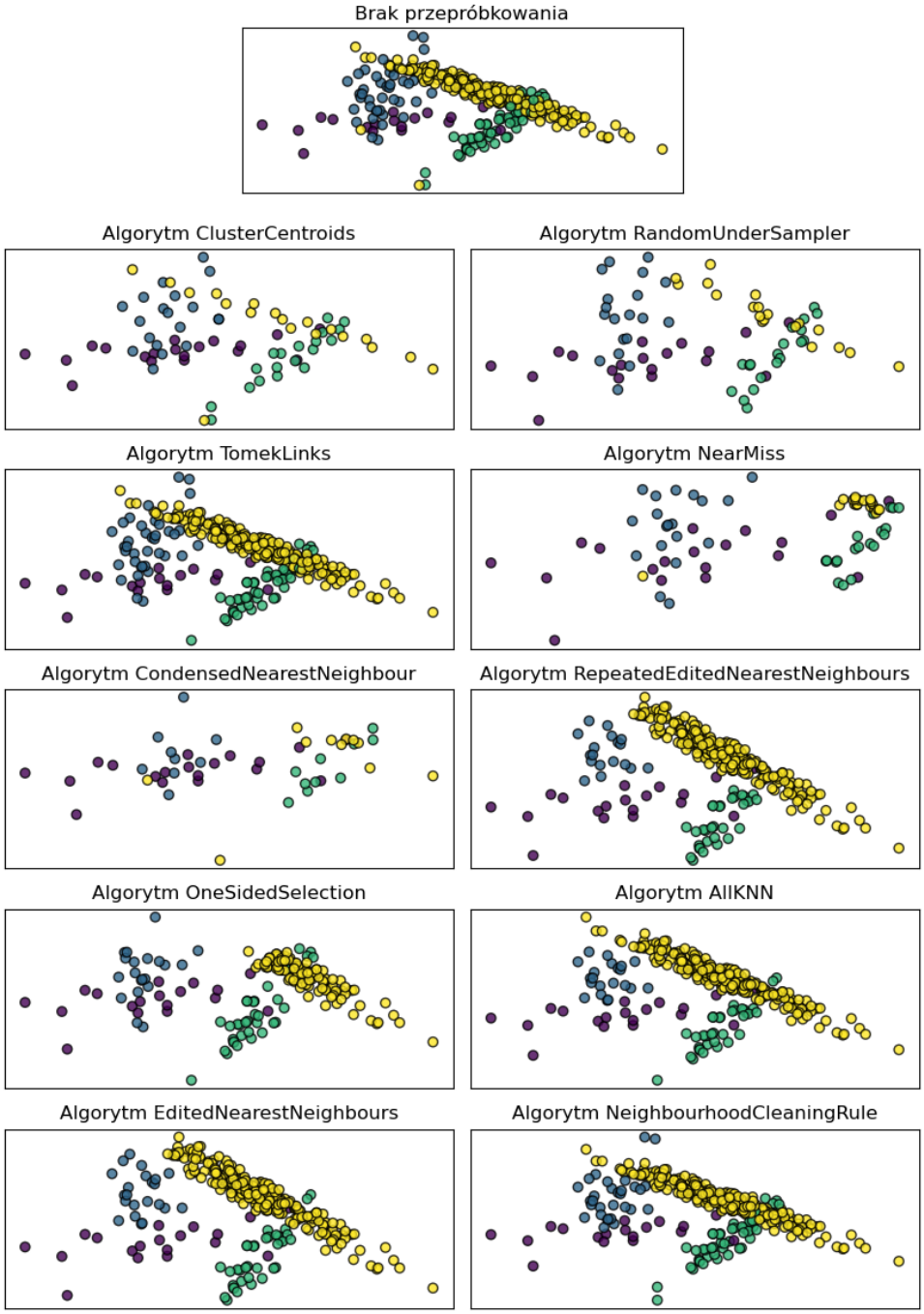
Główną zaletą algorytmu NCR jest skupienie na usuwaniu obserwacji odstających utrudniających wyznaczenie prawidłowych reguł decyzyjnych przy równoczesnym zachowaniu możliwie dużej liczby obiektów ze zbioru oryginalnego. Pozwala to zredukować ryzyko pogorszenia jakości klasyfikacji obiektów należących do klasy podpróbkowanej, co jest częstym problemem w przypadku nadmiernej redukcji liczebności zbioru [75].

2.4.2.11. Porównanie metod podpróbkowania danych

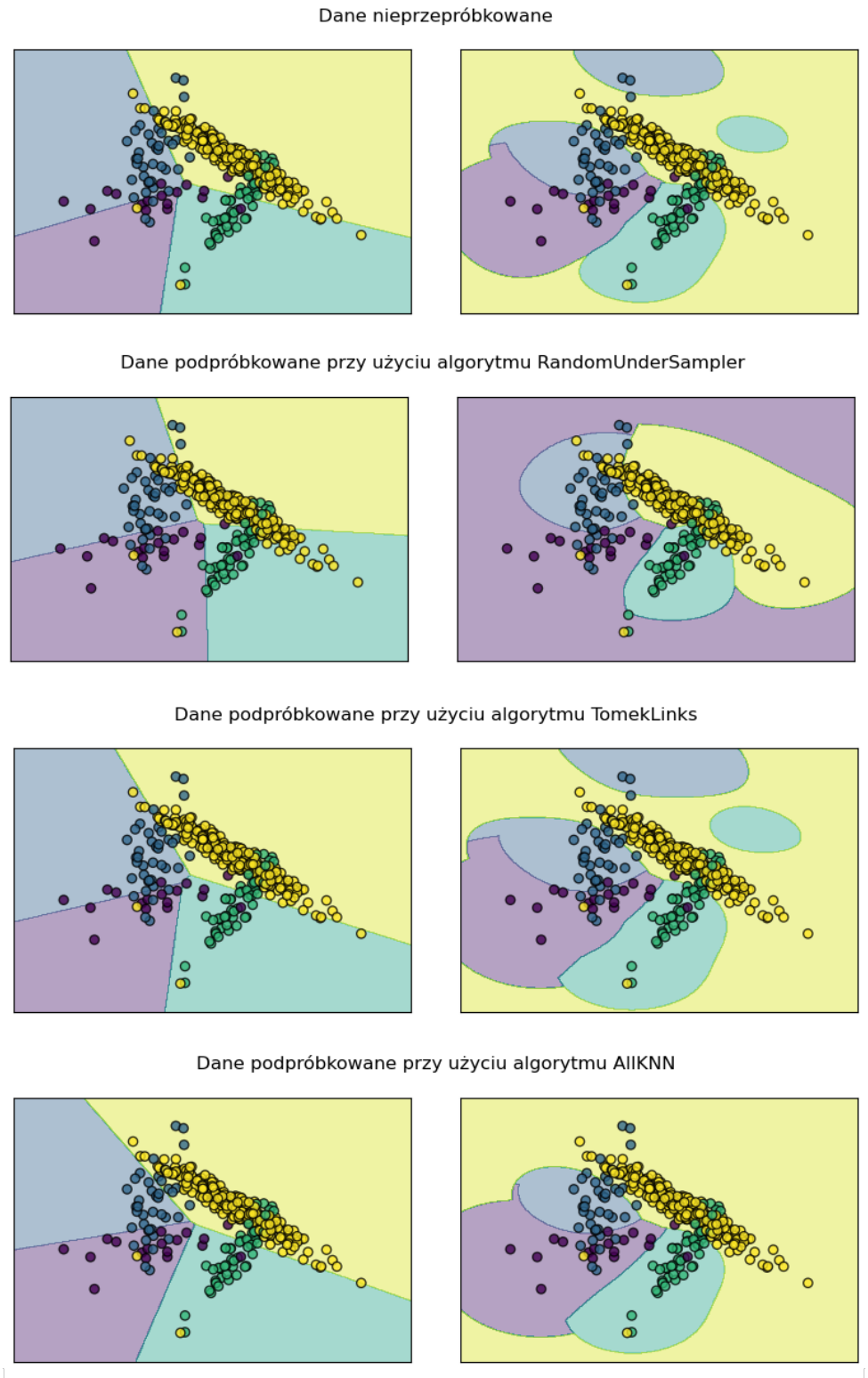
Różnice w zbiorach danych uzyskanych na drodze podpróbkowania z użyciem opisanych powyżej algorytmów przedstawiono na rys. 2.11.

Podobnie jak ma to miejsce w przypadku algorytmów nadpróbkowania, podpróbkowanie zbioru danych skutkuje uzyskaniem modeli klasyfikacyjnych o różnym przebiegu granic decyzyjnych, a tym samym dających różne rezultaty klasyfikacji obiektów. Na rys. 2.12-2.14 różnice pomiędzy granicami decyzyjnymi modelu regresji logistycznej oraz modelu SVM uzyskanymi dla zbiorów danych przedstawionych na rys.2.11.

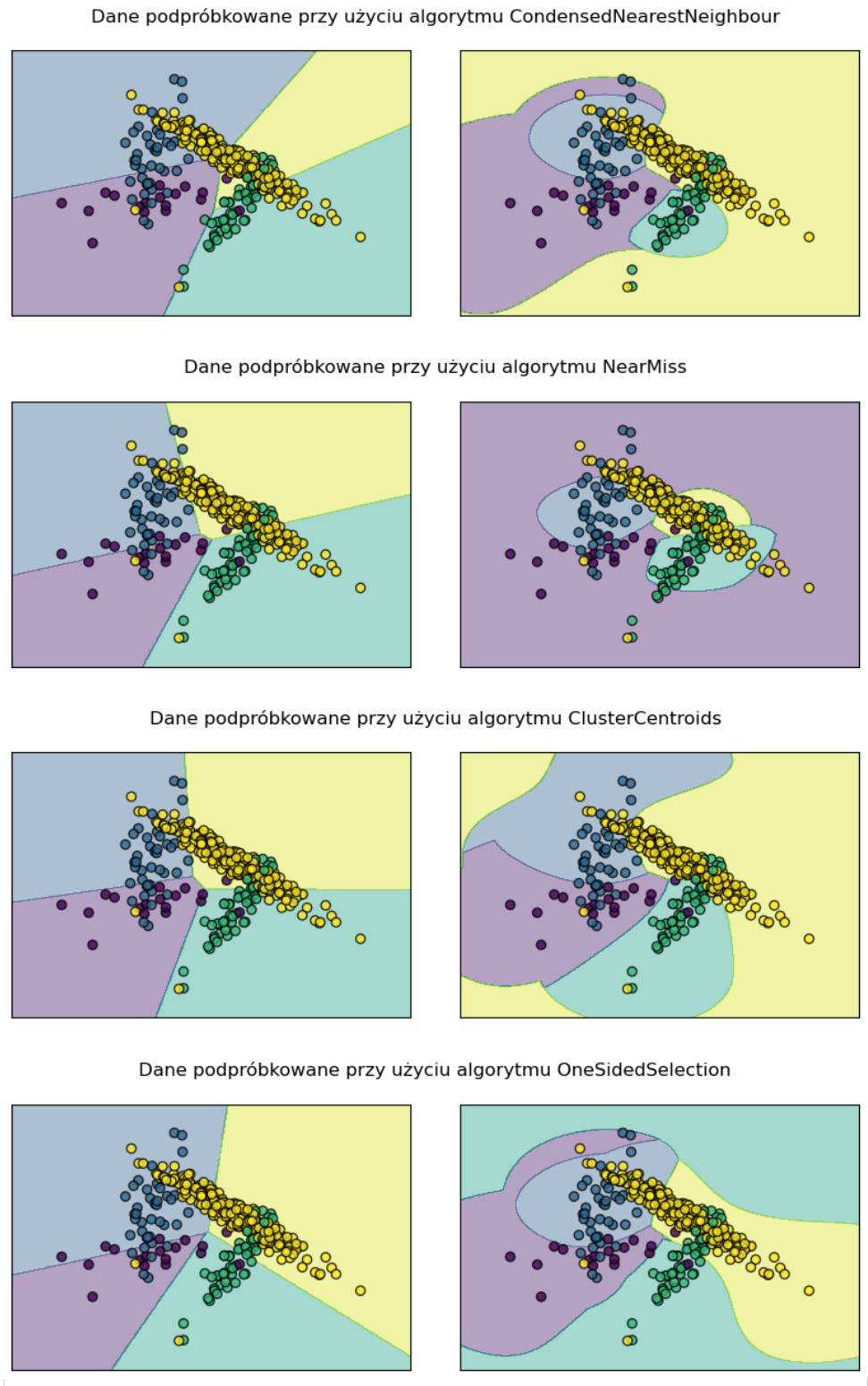
Również tym razem przebieg granic decyzyjnych modeli uczonych na zbiorach uzyskanych z użyciem poszczególnych metod przepróbkowania znacząco bardziej różni się w przypadku modelu SVM. Różnice obserwowane w przypadku modelu regresji logistycznej są mniejsze i widoczne przede wszystkim dla klasy najmniej licznej oznaczonej kolorem fioletowym.



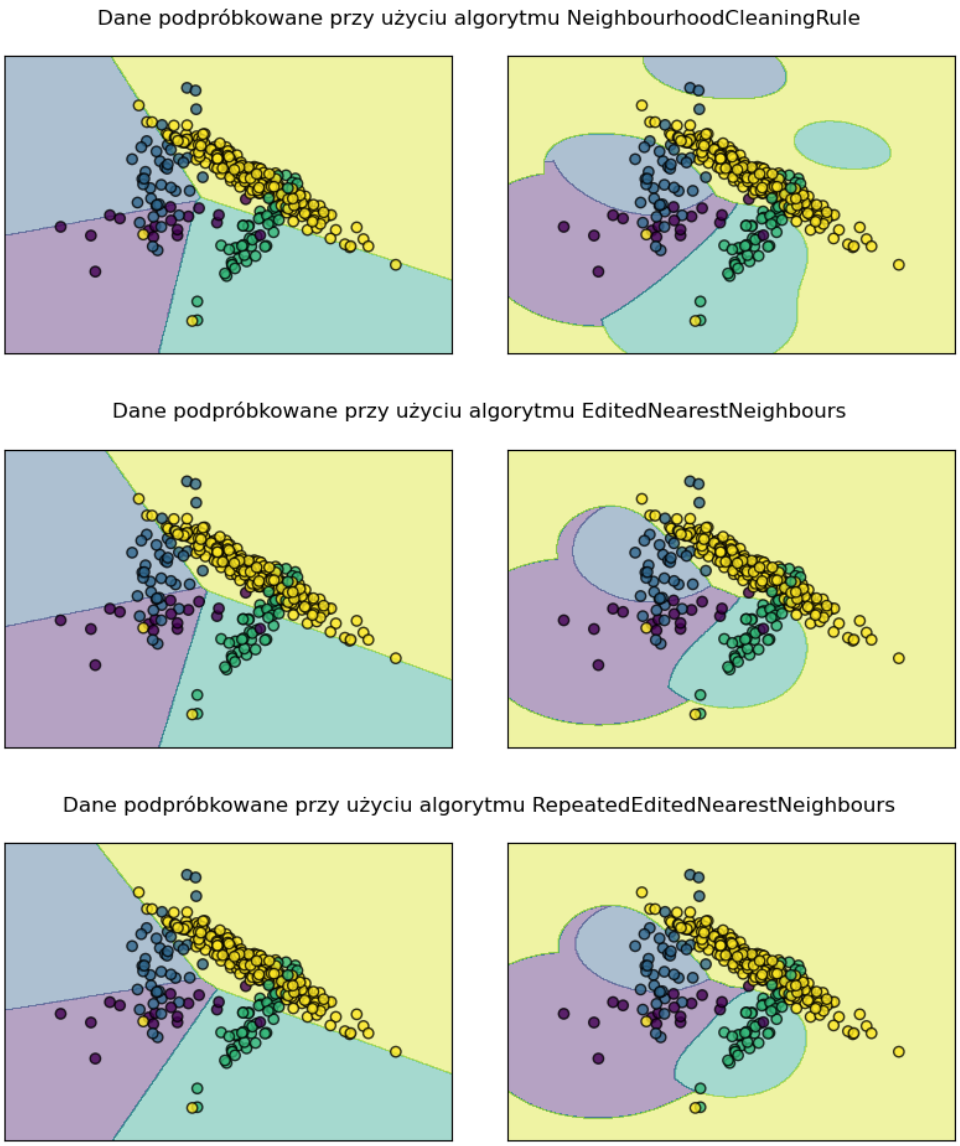
Rys. 2.11. Przykładowy nieprzepróbkowany zbiór danych oraz zbiory danych uzyskane na drodze podpróbkowania.



Rys. 2.12. Granice decyzyjne modelu regresji logistycznej (po lewej) oraz modelu SVM (po prawej) uczonych na zbiorach przepróbkowanych z użyciem różnych algorytmów podpróbkowania danych.



Rys. 2.13. Granice decyzyjne modelu regresji logistycznej (po lewej) oraz modelu SVM (po prawej) uczonych na zbiorach przepróbkowanych z użyciem różnych algorytmów podpróbkowania danych.



Rys. 2.14. Granice decyzyjne modelu regresji logistycznej (po lewej) oraz modelu SVM (po prawej) uczonych na zbiorach przepróbkowanych z użyciem różnych algorytmów podpróbkowania danych.

2.5. Porównanie modeli

Dobór strategii porównywania wyników uzyskiwanych przez różne modele predykcyjne ma kluczowe znaczenie dla wiarygodności tych porównań oraz ich interpretacji. Szczególnie istotne jest to w przypadku porównań wielokrotnych – niewłaściwy dobór metody może doprowadzić do znaczącego wzrostu ryzyka popełnienia błędu I rodzaju, czyli odrzucenia prawdziwej hipotezy zerowej [76]. Z tego względu metody wykorzystane do porównywania modeli i uzyskiwanych przez nie wyników powinny być wybrane w sposób starannie przemyślany i dostosowany do porównywanych metryk służących do ewaluacji modeli.

2.5.1. Bootstrapowe przedziały ufności

Większość testów statystycznych wykorzystywanych do porównywania wartości parametrów charakteryzujących dwie lub więcej grup bazuje na testowaniu hipotez statystycznych – w przypadku, gdy uzyskana z testu wartość p jest mniejsza niż przyjęty poziom istotności statystycznej odrzucana jest hipoteza zerowa i przyjmowana hipoteza alternatywna, w przeciwnym przypadku nie ma podstaw do odrzucenia hipotezy zerowej. Bazowanie jedynie na wartości p jest jednak złudne i niezalecane, szczególnie w przypadku dużych prób czy też porównywania pól pod krzywą ROC (AUC ROC) wyznaczanych dla poszczególnych modeli predykcyjnych. Jak wskazano w [76], zdażają się sytuacje, w których wartość p wskazuje na istotne statystycznie różnice pomiędzy wartościami AUC ROC wyznaczonymi dla dwóch modeli, podczas gdy rzeczywista wielkość efektu jest niewielka i wskazuje na bardzo zbliżone rezultaty uzyskiwane dla obu modeli. Z tego względu zalecane jest raportowanie nie tylko wartości p, ale również przedziałów ufności pozwalających ocenić wielkość efektu, a tym samym znaczenie praktyczne ewentualnych różnic pomiędzy modelami.

Wśród metod wyznaczania przedziałów ufności można wyznaczyć dwie główne grupy:

– metody bazujące na estymatorze największej wiarygodności (NW),

– metody bootstrapowe.

Metody bazujące na estymatorze NW pozwalają wyznaczyć przedział ufności parametru θ zgodnie z zależnością:

$$[\hat{\theta} + \hat{\sigma} z^{\alpha}, \hat{\theta} + \hat{\sigma} z^{1-\alpha}] \tag{2.71}$$

gdzie $\hat{\theta}$ oznacza estymator NW parametru θ, $\hat{\sigma}$ oznacza estymator odchylenia standardowego wartości tego parametru (często wyznaczaną z użyciem pochodnych funkcji log-wiarygodności), z^{α} jest kwantylem rozkładu normalnego rzędu α [77].

Przedziały ufności wyznaczane tą metodą znajdują zastosowanie w niemal każdej sytuacji związanej z estymacją parametryczną. Nie znajdują jednak zastosowania w sytuacji, gdy mamy do czynienia z testem nieparametrycznym, a założenie o normalności rozkładu danych nie jest spełnione [78].

W sytuacji, gdy założenie o normalności rozkładu nie jest spełnione, stosowane są metody bootstrapowe, które wykorzystują tzw. metodę bootstrap. Metoda ta polega na utworzeniu prób bootstrapowych o takiej samej liczebności jak analizowana próba poprzez wielokrotne losowanie ze zwracaniem obiektów z tej próby – oznacza to, że próby bootstrapowe zawierają duplikaty obiektów [79]. Wyznaczanie bootstrapowych przedziałów ufności może zostać opisane za pomocą następującego algorytmu [78]:

1. wybór liczby prób generowanych na podstawie danych *B*,
2. utworzenie B prób bootstrapowych poprzez losowanie obiektów ze zwracaniem,
3. wyznaczenie estymaty pożądanego parametru rozkładu opisującego próbę,
4. uporządkowanie uzyskanych estymat w sposób rosnący,
5. wyznaczenie przedziału ufności parametru zgodnie z zależnością

$$\left[\hat{\theta}_L, \hat{\theta}_U\right] = \left[\hat{\theta}^*_j, \hat{\theta}^*_k\right] \tag{2.72}$$

gdzie $\hat{\theta_L}$ oznacza dolną granicę przedziału ufności, $\hat{\theta_U}$ oznacza górną granicę przedziału ufności, $\hat{\theta}^*_j$ oznacza j-ty kwantyl, $j = \left[\frac{\alpha}{2} \cdot B\right]$, $\hat{\theta}^*_k$ oznacza k-ty kwantyl, $j = \left[(1 - \frac{\alpha}{2}) \cdot B\right]$.

Powyższa zależność wykorzystywana jest w procedurze wyznaczania percentylowych przedziałów ufności wykorzystanych w niniejszej pracy.

2.5.2. Metoda DeLonga

Przyjmijmy, że dla próby o liczebności N prowadzony jest test mający na celu określenie obecności lub braku schorzenia u osobnika należącego do tej próby. Test oparty jest o ciągłą zmienną ilościową, która określa prawdopodobieństwo występowania choroby, a większe wartości tej zmiennej związane są z występowaniem schorzenia. Przyjmijmy również, że w próbie występuje m osobników, u których wykrywana choroba występuje i oznaczmy tę grupę jako C_1. Analogicznie zdefiniujmy C_2 jako grupę o liczebności $n = N - m$ złożoną z osobników, u których choroba nie występuje [80].

Niech $X_i, i = 1, 2, \ldots, m$ oraz $Y_j, j = 1, 2, \ldots, n$ oznaczają zwracane przez model klasyfikacyjny wartości prawdopodobieństwa przynależności do klasy 1 (grupy osób chorych) odpowiednio osobników z grupy C_1 oraz osobników z grupy C_2. Wartości te mogą posłużyć do wyznaczenia empirycznej krzywej ROC, która posłuży do ewaluacji modelu [80]. Wartość pola pod krzywą ROC (AUC ROC) wyznaczana w sposób numeryczny obliczana jest często z użyciem metody trapezów. Jak zostało wykazane, wyznaczana w ten sposób wartość AUC ROC jest równa statystyce Manna-Whitneya wyznaczanej dla C_1 i C_2, liczonej zgodnie z zależnością

$$\hat{\theta} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \phi(X_i, Y_j) \tag{2.73}$$

gdzie

$$\phi(X,Y) = \begin{cases} 1 & \text{gdy } Y < X \\ \frac{1}{2} & \text{gdy } Y = X \\ 0 & \text{gdy } Y < X. \end{cases} \tag{2.74}$$

Uzyskiwany w ten sposób estymator $\hat{\theta}$ jest nieobciążonym estymatorem parametru θ, który określa prawdopodobieństwo tego, że dla losowo wybranej obserwacji z populacji C_2 uzyskiwana jest wartość mniejsza lub równa wartości dla losowo wybranej obserwacji z populacji C_1 [80].

Powyższe równanie (równanie 2.74) może zostać uogólnione na przypadek, w którym porównywane są co najmniej dwa modele klasyfikacyjne, dla których wyznaczono krzywe ROC. Empiryczna wartość metryki AUC ROC k-tego klasyfikatora dana jest wówczas wzorem

$$\hat{\theta^k} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \phi(X_i^k, Y_j^k) \tag{2.75}$$

gdzie X_i^k oznacza zwracane przez k-ty model prawdopodobieństwo przynależności do klasy 1 i-tego obiektu z populacji C_1, zaś Y_j^k oznacza zwracane przez k-ty model prawdopodobieństwo przynależności do klasy 1 j-tego obiektu z populacji C_2 [80].

Niech $\hat{\theta} = (\hat{\theta_1}, \ldots, \hat{\theta_K})^T \in \mathbb{R}^{\mathbb{K}}$ oznacza wektor K empirycznych wartości AUC ROC oraz $\theta = (\theta_1, \ldots, \theta_K)$ oznacza wektor rzeczywistych wartości AUC ROC, zaś $L \in \mathbb{R}^{\mathbb{K}}$ oznacza wektor współczynników modelu [80].

Ponieważ statystyka Manna-Whitneya jest uogólnioną statystyką U, zgodnie z teorią asymptotyczną możemy stwierdzić, że

$$\frac{L^T\hat{\theta} - L^T\theta}{\sqrt{L^T(\frac{1}{m}S_{10} + \frac{1}{n}S_{01})L}} \tag{2.76}$$

ma asymptotyczny rozkład normalny. S_{10} oraz S_{01} są macierzami o wymiarach KxK, w których elementy o współrzędnych (r, s) zdefiniowane są jako [80]

$$(S_{10})_{rs} = \frac{1}{m-1} \sum_{i=1}^{m} \left[V_{10}^r(X_i) - \hat{\theta}^r\right] \left[V_{10}^*(X_i) - \hat{\theta}^s\right] \tag{2.77}$$

$$(S_{01})_{rs} = \frac{1}{n-1} \sum_{i=1}^{m} \left[V_{01}^r(Y_j) - \hat{\theta}^r\right] \left[V_{01}^*(Y_j) - \hat{\theta}^s\right] \tag{2.78}$$

gdzie

$$V_{10}^r(X_i) = \frac{1}{n} \sum_{j=1}^{n} \phi(X_i^r, Y_j^r) \tag{2.79}$$

$$V_{01}^r(Y_j) = \frac{1}{m} \sum_{i=1}^{m} \phi(X_i^r, Y_j^r). \tag{2.80}$$

Dla powyższego asymptotycznego rozkładu normalnego można wyznaczyć przedziały ufności różnic pomiędzy metrykami AUC ROC uzyskiwanymi dla dwóch modeli klasyfikacyjnych oraz przeprowadzić test istotności tych różnic zgodnie z zależnością [80]

$$(\hat{\theta} - \theta)L^T \left[L(\frac{1}{m}S_{10} + \frac{1}{n}S_{01})L^T \right]^{-1} L(\hat{\theta} - \theta)^T. \tag{2.81}$$

Zawarte w niniejszej pracy wartości p uzyskane przy użyciu testu DeLonga oraz innych testów statystycznych przedstawiono zgodnie z rekomendacjami dotyczącymi raportowania wyników w badaniach medycznych zaprezentowanymi w [81]:

– wartości p mniejsze niż $0,001$ przedstawiono jako $p < 0,001$,

– wartości z przedziału $[0,001; 0,01]$ przedstawiono z dokładnością do części tysięcznych,

– wartości z przedziału $(0,01; 0,99]$ przedstawiono z dokładnością do części setnych,

– wartości p większe niż $0,99$ przedstawiono jako $p > 0,99$.

2.6. Wektorowe reprezentacje mówcy

2.6.1. i-wektory

i-wektory to rodzaj biometrycznej, wektorowej reprezentacji mówcy opracowany w celu wiarygodnej identyfikacji mówcy na podstawie sygnału mowy. System ekstrakcji i-wektorów z sygnału mowy wykorzystuje mieszane modele gaussowskie (GMM) uczone na na cechach uzyskanych na drodze parametryzacji sygnału (rys. 2.15), z reguły współczynnikach melcepstralnych (MFCC). Uczenie modelu GMM reprezentującego mowę danego mówcy jest prowadzone poprzez adaptację modelu uniwersalnego mówcy (ang. *universal backgroud model*, UBM) z użyciem algorytmu EM (ang. *expectation maximization*) [82].

Proces ekstrakcji i-wektorów przebiega w sposób nienadzorowany i bazuje na założeniu, że superwektor M reprezentujący analizowany sygnał mowy może zostać przedstawiony za pomocą wyrażenia

$$M = m + TW \tag{2.82}$$

gdzie m oznacza superwektor niezależny od mówcy i użytego rejestratora dźwięku wyznaczany za pomocą modelu UBM, T oznacza macierz całkowitej zmienności opisująca zmienność wynikającą z cech osobniczych mówcy oraz rejestratora dźwięku, zaś w jest losowym wektorem, którego wartości można opisać za pomocą standardowego rozkładu normalnego $N(0, I)$.

Wektory w wyznaczone dla analizowanych sygnałów nazywane są i-wektorami, a ich wartości są silnie zależne od mówcy i mogą zostać obliczone na podstawie wyrażenia

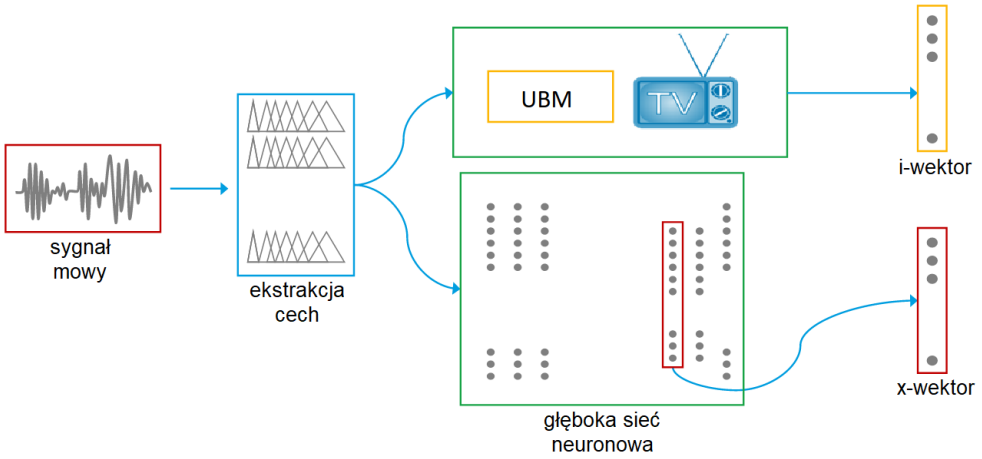
$$w = (I + T^t\Sigma^{-1}N(u)T)^{-1}.T^t\Sigma^{-1}\tilde{F}(u) \tag{2.83}$$

gdzie Σ jest diagonalną macierzą kowariancji o wymiarach $CFxCF$ wyestymowaną podczas treningu, C oznacza liczbę składowych modelu UBM, F oznacza liczbę cech opisujących sygnał, $\tilde{F}(u)$ oznacza superwektor o wymiarach $FCx1$ otrzymany poprzez scalenie wszystkich statystyk Bauma-Welcha pierwszego rzędu $\tilde{F}_c$ dla analizowanego sygnału (równanie 2.84), $N(u)$ jest macierzą diagonalną o wymiarach $CFxCF$, której przekątna może być wyznaczona jako $N_c I(c = 1, ..., C)$ (równanie 2.85).

$$\tilde{F}_c = \sum_{t=1}^{L} P(c|y_t, \Omega)(y_t - m_c) \tag{2.84}$$

$$N_c = \sum_{t=1}^{L} P(c|y_t, \Omega) \tag{2.85}$$

gdzie $c = 1, ..., C$ jest indeksem składowej w modelu GMM, $P(c|y_t, \Omega)$ oznacza prawdopodobieństwo *a posteriori* składowej c dla ramki y_t, m_c jest wartością średnią składowej c modelu UBM [82].



Rys. 2.15. Schemat ekstrakcji i-wektorów i x-wektorów z sygnału mowy. Źródło: [83].

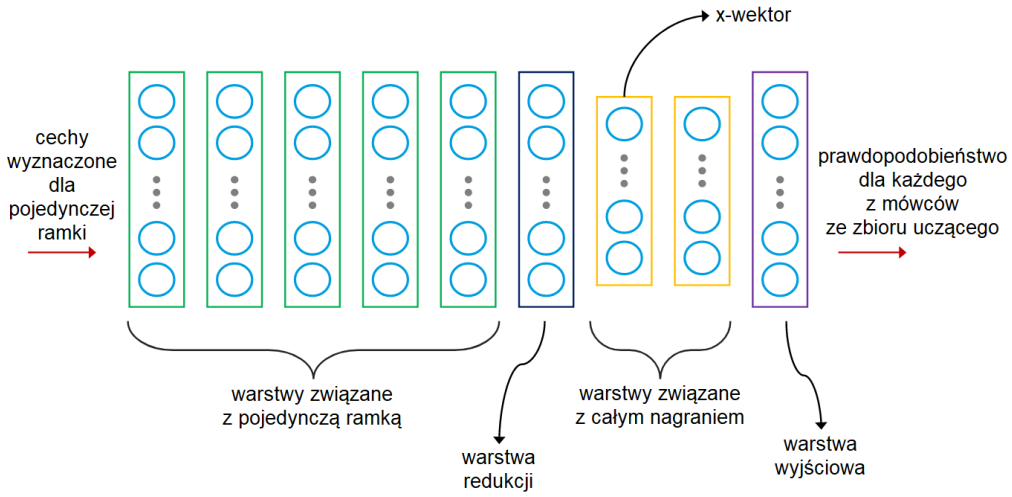
2.6.2. x-wektory

Do wektorów własnościowych mówcy zaliczają się również x-wektory, czyli wektorowe reprezentacje mówcy uzyskiwane na drodze uczenia nadzorowanego (ang. *supervised embeddings*), ekstrahowane przy użyciu głębokich sieci neuronowych (DNN). Standardowo, sieć DNN wykorzystywana do ekstrakcji x-wektorów składa się z trzech bloków różniących się pełnioną funkcją (rys. 2.16) [84]:

– enkodera złożonego z wielowarstwowej sieci z opóźnieniem czasowym (ang. *time-delay neural network*, TDNN) dokonującej operacji splotu na cechach wyznaczonych dla kolejnych ramek sygnału,

– warstwy redukcji (ang. *pooling layer*) obliczającej zagregowane miary cech wyznaczanych na poziomie ramek przez enkoder,

– dekodera dokonującego klasyfikacji sygnałów do klas odpowiadających poszczególnym mówcom. Dekoder operuje na zagregowanych cechach reprezentujących całość analizowanego sygnału i składa się z co najmniej dwóch warstw ukrytych oraz warstwy wyjściowej – jedna z warstw ukrytych, poprzedzająca warstwę o nieliniowej funkcji aktywacji, zwraca wektor zawierający cechy biometryczne mówcy nazywany x-wektorem [85].



Rys. 2.16. Schemat architektury sieci neuronowej służącej do ekstrakcji x-wektorów. Źródło: [83].

Pierwotnie sieć ekstrahująca x-wektory uczona była na współczynnikach MFCC ekstrahowanych z sygnału mowy [86]. Następnie zaproponowano rozwiązania, w których współczynniki MFCC zostały zastąpione przez współczynniki GFCC – cechy ekstrahowane z sygnału poprzez filtrację spektrogramu z użyciem banku filtrów gamma. Jedną z tych prac jest artykuł opublikowany przez autorkę niniejszej rozprawy [87] – prace nad tym rozwiązaniem były prowadzone równolegle do prac opisanych w [88] i nie były nimi inspirowane.

3. Tło literaturowe badań

Diagnostyka medyczna na podstawie sygnału mowy jest zagadnieniem budzącym rosnące zainteresowanie zarówno środowiska naukowego, jak i lekarskiego. Jej nieinwazyjny charakter pozwala na szybkie przeprowadzenie procesu diagnostycznego ze stosunkowo niskimi kosztami oraz brakiem niedogodności dla pacjenta. Nie bez znaczenia jest również fakt, że sygnał mowy może zostać wykorzystany w diagnostyce prowadzonej z użyciem telemedycyny, co jest szczególnie istotne w przypadku pacjentów zamieszkujących tereny znacznie oddalone od dużych ośrodków miejskich, mających ograniczony dostęp do specjalistycznej opieki medycznej [89]. Należy również podkreślić, że diagnostyka na podstawie sygnału mowy jest narzędziem umożliwiającym redukcję transmisji chorób zakaźnych, zminiejszając tym samym ryzyko infekcji zarówno wśród personelu medycznego, jak i pacjentów [90].

3.1. Diagnostyka schorzeń otolaryngologicznych na podstawie sygnału mowy

Oczywistym zastosowaniem diagnostyki medycznej na podstawie sygnału mowy jest diagnostyka chorób otolaryngologicznych, manifestujących się m.in. zmianami w emisji głosu. Duża część rozwiązań opisanych w literaturze bazuje na analizie cech sygnału ekstrahowanych z nagrań samogłosek o przedłużonej fonacji. Ma to związek z przyzwyczajeniami wyniesionymi z tradycyjnej diagnostyki schorzeń krtani, gdzie samogłoski o przedłużonej fonacji wykorzystywane są podczas badania lekarskiego – ich emisja jest możliwa podczas prowadzenia obserwacji wizualnej gardła, krtani oraz jamy ustnej. Kolejną zaletą samogłosek o przedłużonej fonacji jest fakt, że taki sygnał może zostać uznany za sygnał stacjonarny, co pozwala na zastosowanie wielu dobrze poznanych metod analizy sygnałów.

Jak wykazali Hemmerling i in. [91], zastosowanie samogłosek o przedłużonej fonacji pozwala na rozpoznanie schorzeń otolaryngologicznych ze 100% dokładnością. W swojej pracy autorzy wykorzystali 28 cech akustycznych ekstrahowanych z nagrań samogłoski $\backslash a \backslash$, m.in. 10 współczynników mel-cepstralnych (ang. *mel-frequency cepstral coefficients*, MFCC), jitter, shimmer i współczynniki formantowe, które posłużyły do treningu modelu lasu losowego. Fang i in. [92] zastosowali głęboką sieć neuronową (ang. *deep neural network*, DNN) uczoną na 13-wymiarowych współczynnikach MFCC ekstrahowanych z nagrań samogłoski $\backslash a \backslash$, uzyskując dokładność klasyfikacji wynoszącą 99,14%. Z kolei Al-Nasheri i in. [93] zbadali możliwość

diagnostyki na podstawie 22 parametrów akustycznych ekstrahowanych przy użyciu oprogramowania *Multidimensional Voice Program* [94] – wśród zastosowanych przez nich parametrów można wyróżnić parametry częstotliwościowe, związane z intensywnością sygnału mowy, czy też szumem. Parametry te posłużyły do przeprowadzenia statystycznego testu t-Studenta sprawdzającego równość wartości średnich w grupie osób chorych i zdrowych, co zaskutkowało dokładnością klasyfikacji równą 99,68%. Muhammad i in. [95] przedstawili natomiast rozwiązanie bazujące na polach powierzchni przekrojów modelu traktu głosowego i analizie nieregularności ich wartości. Wyekstrahowane cechy posłużyły do treningu maszyny wektorów nośnych (ang. *support vector machine*, SVM) i pozwoliły dokonać klasyfikacji sygnałów z dokładnością wynoszącą 94,7%. Wu i in. [96] zastosowali z kolei splotowe sieci neuronowe (ang. *convolutional neural network*, CNN) uczone na spektrogramach wyznaczonych na podstawie nagrań samogłoski $\backslash a \backslash$, uzyskując dokładność klasygikacji wynoszącą 77%. Podobne rozwiązanie zaproponowali Mohammed i in. [97], którzy wykorzystali DNN uczone na spektrogramach wyznaczonych z użyciem filtrów oktawowych. Proponowane przez nich rozwiązanie osiągnęło dokładność klasyfikacji równą 95,72%

Opisane powyżej prace dotyczą diagnostyki dowolnej choroby otolaryngologicznej – celem klasyfikacji jest wówczas określenie, czy dany sygnał mowy został zarejestrowany dla osoby zdrowej, czy nie. Istnieją jednak również publikacje, które opisują metody diagnostyki konkretnych schorzeń. Wśród takich prac można wymienić m.in. artykuł opublikowany przez Teixeirę i in. [98], w którym podjęto próbę detekcji dysfonii na podstawie nagrań trzech samogłosek o przedłużonej fonacji ($\backslash a \backslash$, $\backslash i \backslash$, $\backslash u \backslash$). Nagrania posłużyły do ekstrakcji łącznie 81 cech opisujących mowę każdego z pacjentów, które można podzielić na trzy grupy: dotyczące parametru jitter, parametru shimmer oraz stosunku harmonicznych do szumu (*harmonic to noise ratio*, HNR). Sieć neuronowa trenowana na tych parametrach uzyskała dokładność klasyfikacji wynoszącą 100% w przypadku kobiet oraz 90% w przypadku mężczyzn. Innym przykładem pracy dotyczącej diagnostyki konkretnego schorzenia jest metoda zaprezentowana przez Madrugę i in. [99], w której zastosowano maszynę wektorów nośnych. Model SVM został poddany treningowi przy użyciu wektora cech akustycznych, m.in. współczynników MFCC, częstotliwości przejść przez zero (ang. *zero-crossing rate*, ZCR), jitter oraz shimmer, ekstrahowanych z nagrań samogłoski $\backslash a \backslash$. Takie podejście zaskutkowało uzyskaniem dokładności klasyfikacji przekraczającej 90%.

Chociaż przy odpowiednim zastosowaniu samogłoski o przedłużonej fonacji umożliwiają uzyskanie bardzo wysokiej dokładności klasyfikacji, należy mieć na uwadze, że nie pozwalają one przeanalizować niektórych cech wykorzystywanych do oceny jakości emisji głosu, takich jak VOT (ang. *voice onset time*) określający czas pomiędzy rozpoczęciem emisji samogłoski, a początkiem emisji poprzedzającej ją spółgłoski w nagłosie, czy też cechy charakteryzujące zmiany wysokości głosu oraz ich fluktuacje. Informacje te zawarte są natomiast w sygnale mowy ciągłej [100].

W literaturze można znaleźć stosunkowo nieliczne prace opisujące metody detekcji schorzeń otolaryngologicznych bazujące na sygnale mowy ciągłej. Cordeiro i in. [101] w swoim

artykule zaproponowali algorytm detekcji porażenia nerwu krtaniowego oraz obrzęku fałdów głosowych oparty o 12-wymiarowe współczynniki MFCC oraz mieszany model gaussowski (ang. *gaussian mixture model*, GMM). Prezentowane przez nich podejście pozwoliło na uzyskanie dokładności klasyfikacji rzędu 74%. Ten sam zespół badaczy podjął również próbę diagnostyki porażenia nerwu krtaniowego wstecznego oraz guzków śpiewaczych na podstawie współczynników MFCC oraz współczynników LPC (ang. *linear predictive coding*) wykorzystywanych do treningu modelu SVM, modelu GMM oraz przeprowadzenia analizy dyskryminacyjnej Fishera [102]. Zastosowanie tej metody pozwoliło uzyskać model cechujący się czułością wynoszącą 76,3% i precyzją 82,4% w przypadku porażenia nerwu krtaniowego oraz czułością wynoszącą 83,1% i precyzją 80,3% w przypadku guzków śpiewaczych. Guedes i in. [103] zaprezentowali rozwiązanie pozwalające na detekcję dysfonii, przewlekłego zapalenia krtani oraz porażenia nerwu krtaniowego, uzyskujące wartość metryki F1 wynoszącą odpowiednio 66%, 67% i 80%. Do klasyfikacji sygnałów wykorzystane zostały pretrenowane sieci neuronowe. Z kolei Ali i in. [100] podjęli się klasyfikacji sygnałów mowy pochodzących od osób cierpiących na jedno z pięciu schorzeń: torbieli fałdu głosowego, guzków śpiewaczych, hiperkeratozy fałdów głosowych, porażenia nerwu krtaniowego oraz dysfonii spastycznej. W tym celu wykorzystali model GMM trenowany na spektrogramie przefiltrowanym przy użyciu banku filtrów słuchowych (ang. *auditory processed spectrum*). Takie podejście pozwoliło na uzyskanie dokładności klasyfikacji wynoszącej odpowiednio 89,33%, 85,53%, 85,53%, 78,42% i 93,33%.

Tabela 3.1. Dokładność klasyfikacji mowy osób chorych otolaryngologicznie na podstawie sygnału mowy ciągłej osiągnięta z użyciem metod opisanych w literaturze. GMM - mieszany model gaussowski, MFCC - współczynniki mel-cepstralne, APS - *auditory processed spectrum*, KNN - klasyfikator k-najbliższych sąsiadów, SVM - maszyna wektorów nośnych, ANN - sieć neuronowa.

Choroba	Cechy	Klasyfikator	Dokładność klasyfikacji
Dysfonia spastyczna [100]	APS	GMM	0,93
Guzki śpiewacze [102]	APS	GMM	0,89
Guzki śpiewacze [100]	APS	GMM	0,86
Hiperkeratoza fałdów głosowych [100]	APS	GMM	0,86
Nieokreślona [104]	miary złożoności trajektorii stanów	SVM	0,95
Nieokreślona [104]	miary złożoności trajektorii stanów	ANN	0,88
Nieokreślona [104]	miary złożoności trajektorii stanów	KNN	0,89
Porażenie nerwu krtaniowego [101]	MFCC	GMM	0,74
Porażenie nerwu krtaniowego [102]	MFCC	GMM	0,74
Porażenie nerwu krtaniowego [100]	APS	GMM	0,79
Torbiel fałdu głosowego [100]	APS	GMM	0,89

Podobnie jak w przypadku diagnostyki na podstawie samogłosek o przedłużonej fonacji, sygnał mowy ciągłej może również być wykorzystywany do odróżniania osób chorych od cierpiących na dowolną chorobę krtani. Do takich prac zalicza się artykuł opublikowany przez Terbeh i in. [105]. Autorzy przedstawili metodę opartą o modele fonetyczne tworzone dla języka arabskiego – klasyfikacja sygnałów dokonywana jest poprzez porównanie modelu wzorcowego utworzonego na podstawie korpusu mowy osób zdrowych z modelem utworzonym dla analizowanego mówcy. Opisane podejście pozwoliło na poprawną klasyfikację 100% sygnałów zarejestrowanych dla osób zdrowych oraz 95% sygnałów zarejestrowanych dla osób chorych. Orozco i in. [104] zaprezentowali natomiast podejście bazujące na modelowaniu układów dynamicznych: ekstrahowane przez nich cechy sygnału wyznaczane są na podstawie miar złożoności trajektorii stanów. Trzy modele – SVM, sieć neuronowa oraz klasyfikator k-najbliższych sąsiadów – zostały poddane treningowi w celu nauczenia ich detekcji osób chorych otolaryngologicznie i osiągnęły dokładność klasyfikacji wynoszącą odpowiednio 95%, 87,93% i 89,15%. Z kolei Vasilakis i Stylianou [106] opisali metodę bazującą na krótkoczasowych estymatorach parametru jitter, która pozwoliła na uzyskanie pola pod krzywą ROC wynoszącego 87,7%. Vicsi i in. [107] podjęli się próby detekcji schorzeń krtani na podstawie samogłosek *e* wyciętych z sygnału mowy ciągłej. Fragmenty sygnałów zawierające tę samogłoskę posłużyły do ekstrakcji dwóch parametrów: jitter ddp oraz shimmer dda. Uzyskane wektory wartości uśredniono i wykorzystano do treningu modelu SVM, który dokonał klasyfikacji sygnałów z dokładnością wynoszącą 84%.

3.2. Diagnostyka zaburzeń neurorozwojowych na podstawie sygnału mowy

Mowa osób ze zdiagnozowanymi zaburzeniami ze spektrum autyzmu (ang. textitautism spectrum disorder, ASD) cechuje się odmiennymi cechami prozodycznymi, niż mowa osób niezaburzonych – mowa osób z ASD często określana jest jako monotonna, robotyczna, a intonacja jest nieprawidłowa i nieadekwatna do sytuacji. Nieprawidłowa prozodia jest też jednym z kryteriów oceny wykorzystywanych podczas diagnostyki ASD dokonywanej przez lekarzy i terapeutów [108]. Chociaż liczba publikacji na ten temat jest niewielka, można zauważyć rosnące zainteresowanie tematyką wykrywania ASD w sposób nieinwazyjny i zapewniający osobom diagnozowanym – w szczególności dzieciom – naturalne, bezpieczne warunki. Do takich metod detekcji ASD zaliczają się metody bazujące na analizie sygnału mowy.

Wijesinghe i in. [109] zaproponowali rozwiązanie wykorzystujące splotowe sieci neuronowe (ang. *convolutional neural network*, CNN) trenowane na 20-wymiarowych współczynnikach MFCC ekstrahowanych z mowy ciągłej zarejestrowanej podczas rozmowy ze znaną dziecku osobą dorosłą. Zastosowane przez autorów podejście pozwoliło uzyskać dokładność klasyfikacji wynoszącą 78%. Należy jednak zwrócić uwagę na niewielkie rozmiary wykorzystanego zbioru danych, który zawierał mowę zaledwie 9 dzieci zdrowych i 8 dzieci ze zdiagnozowanym ASD. Marchi i in. [110] przedstawili wyniki uzyskane na podstawie trzech korpusów

mowy: hebrajskiego, szwedzkiego i angielskiego. Podobnie jak w przypadku wcześniej opisanego artykułu, również tym razem rozmiary użytych korpusów mowy są niewielkie – korpusy mowy zawierały odpowiednio mowę 17, 20 i 18 dzieci. Autorzy przeprowadzili klasyfikację sygnałów na podstawie cech akustycznych ekstrahowanych przy użyciu pakietu openSMILE [111], związanych m.in. z parametrami takimi jaki jitter, shimmer, częstotliwości formantowe, HNR, czy też współczynniki MFCC. Cechy te posłużyły do treningu modelu SVM, który następnie poddano ewaluacji za pomocą średniej wartości czułości (ang. *unweighted average recall*, UAR) wyznaczonej z użyciem sprawdzianu krzyżowego. Uzyskany w ten sposób model osiągnął wartości UAR wynoszące odpowiednio 82,7%, 86,4% oraz 78,3% UAR dla wymienionych powyżej korpusów mowy. Podobne rozwiązanie, również bazujące na modelu SVM i cechach ekstrahowanych za pomocą pakietu openSMILE zaprezentowali Schimmitt i in. [112], rozbudowując wspomniany korpus mowy o sygnały zarejestrowane dla 29 dzieci francuskojęzycznych. Przedstawiony przez nich model pozwolił osiągnąć wartości UAR wynoszące 89,6% dla języka angielskiego, 87,1% dla języka francuskiego, 71,6% dla języka hebrajskiego oraz 78,6% dla języka szwedzkiego. Ten sam zestaw cech został też użyty przez Deng i in. [113]. Autorzy tej pracy zdecydowali się jednak na użycie generatywnej sieci współzawodniczącej (ang. *generative adversarial network*, GAN) – aktywacje warstwy ukrytej sieci GAN posłużyły jako wektor cech wejściowych podawany do modelu SVM. Takie podejście pozwoliło im osiągnąć UAR wynoszące 46,93% dla zbioru danych liczącego 11 dzieci z ASD i 68 dzieci zdrowych. Rozwiązanie bazujące na połączeniu cech akustycznych i cech tekstowych zaprezentowali natomiast Cho i in. [114]. Proponowany przez autorów pracy model drzewiasty *Gradient Boosting* prawidłowo zaklasyfikował 66% sygnałów zarejestrowanych dla 35 dzieci ze zdiagnozowanym ASD i 86% sygnałów zarejestrowanych dla 35 dzieci zdrowych, osiągając tym samym dokładność klasyfikacji wynoszącą 76%. Z kolei Ringeval i in. [115] nie zdecydowali się na odróżnianie dzieci ze zdiagnozowanym ASD od dzieci zdrowych, a przeprowadzili klasyfikację z użyciem sygnałów zarejestrowanych dla dzieci zdrowych oraz dzieci ze zdiagnozowanym zaburzeniem neurorozwojowym – ASD lub specyficznymi zaburzeniami językowymi (ang. *specific language impairment*, SLI). Wykorzystując korpus mowy składający się z nagrań zarejestrowanych dla 35 dzieci z zaburzeniem oraz 68 dzieci zdrowych zdołali osiągnąć UAR wynoszącą 88,3% stosując model SVM uczony na cechach ekstrahowanych za pomocą pakietu openSMILE. Najlepsze rezultaty osiągnięte w wymienionych pracach zestawiono w tabeli 3.2.

Zagadnienie diagnostyki SLI na podstawie sygnału mowy zostało też poruszone w innych pracach. Grill i Tučková [116] zaprezentowali artykuł opisujący utworzony przez nich korpus mowy czeskiej zawierający mowę dzieci chorych i ze zdiagnozowanymi zaburzeniami SLI. Przedstawili również algorytm automatycznej diagnostyki tych zaburzeń w oparciu o cechy ekstrahowane przy użyciu pakietu openSMILE oraz samoorganizujące sieci neuronowe (ang. *self-organizing maps*, SOM). Takie podejście pozwoliło odróżniać sygnały zarejestrowane dla obu grup dzieci z dokładnością wynoszącą 96,94%. Jeden z autorów, Grill, opublikował również drugą pracę dotyczącą diagnostyki SLI – tym razem zdecydował się przeprowadzić klasyfikację w oparciu o liczbę błędów popełnianych podczas wymowy wybranego zestawu słów [117].

Tabela 3.2. Dokładność klasyfikacji mowy dzieci ze zdiagnozowanymi zaburzeniami ze spektrum autyzmu osiągnięta z użyciem metod opisanych w literaturze. GAN - generatywna sieć współzawodnicząca, MFCC - współczynniki mel-cepstralne, SVM - maszyna wektorów nośnych, CNN - splotowa sieć neuronowa.

Klasyfikator	Cechy	Dokładność klasyfikacji
Gradient Boosting [114]	openSMILE	0,76
SVM [112]	openSMILE	0,90
SVM [110]	openSMILE	0,86
SVM [113]	openSMILE + GAN	0,47
CNN [109]	MFCC	0,78

Tabela 3.3. Dokładność klasyfikacji mowy dzieci ze zdiagnozowanymi specyficznymi zaburzeniami językowymi osiągnięta z użyciem metod opisanych w literaturze. SOM - samoorganizujące sieci neuronowe, ELM - *extreme learning machine*, ULP - *Utterance Level Posteriograms*, FFNN - jednokierunkowa sieć neuronowa, AF - cechy akustyczne, EMS - *envelope modulation spectrum*, EBT - *ensemble boosted trees*, ResNet - rezydualna sieć neuronowa.

Klasyfikator	Cechy	Dokładność klasyfikacji
SOM [116]	openSMILE	0,97
ELM [118]	ULP	0,99
SVM [119]	openSMILE	0,97
FFNN [119]	openSMILE	0,98
EBT [120]	AF + EMS + LBP	0,97
ResNet [121]	spektrogram	1,00

Opisana metoda cechuje się nieco niższą dokładnością klasyfikacji, wynoszącą 93%, umożliwia jednak uzyskanie wstępnej diagnozy z wykorzystaniem opracowanego w tym celu oprogramowania. Z kolei Reddy i in. [119] zaproponowali zagadnienie bazujące na estymacie sygnału glotograficznego ekstrahowanej z sygnału mowy, z których następnie wyekstrahowano czasowe i częstotliwościowe parametry glotograficzne. Parametry te w połączeniu z 13-wymiarowymi współczynnikami MFCC posłużyły do treningu jednokierunkowej sieci neuronowej (ang. *feed-forward neural network*, FFNN) oraz maszyny wektorów nośnych (SVM) i pozwoliły uzyskać dokładność klasyfikacji wynoszącą odpowiednio 98,82% i 97,08%. Ramarao i in. [118] zdecydowali się natomiast na użycie sieci ELM (ang. *extreme learning machine*) uczonych na posteriogramach ekstrahowanych z 13-wymiarowych współczynników MFCC za pomocą mieszanych modeli gaussowskich (GMM), uzyskując dokładność klasyfikacji wynoszącą 99,41%. Z kolei Sharma i in. [120] opracowali metodę diagnostyki SLI bazującą na grupie drzew klasyfikacyjnych EBT (ang. *ensemble boosted trees*, EBT). Klasyfikator EBT został poddany procesowi uczenia na trzech rodzajach cech esktrahowanych z sygnałów akustycznych: cech akustycznych (ang. *acoustical features*, AF), cech obrazujących obwiednię sygnału i występującą

w nim modulację amplitudową (ang. *envelope modulation spectrum*, EMS) oraz cechy opisujące teksturę obrazu (ang. *local binary pattern*, LBP) – w tym przypadku spektrogramu – stosowane w technikach wizyjnych. Zaprezentowane podejście umożliwiło przeprowadzenie klasyfikacji z dokładnością wynoszącą 97,36%. Autorka niniejszej rozprawy jest natomiast współautorką pracy prezentującej sieć rezydualną (ang. *residual neural network*, ResNet) uczoną na spektrogramach ekstrahowanych z sygnałów mowy, która osiągnęła 100% dokładność klasyfikacji [121]. Co ważne, wszystkie wymienione powyżej prace wykorzystywały ten sam korpus mowy, co umożliwia porównanie zaprezentowanych w nich wyników – opisane rozwiązania podsumowano w tabeli 3.3.

3.3. Zastosowanie wektorów własnościowych w diagnostyce medycznej

x-wektory (rozdział 2, sekcja 2.6.2), jak stosunkowo nowa metoda wektorowego opisu mówcy, zostały dotychczas zastosowane w niewielu badaniach związanych z automatyczną diagnostyką metyczną na podstawie sygnału mowy. Większość przypadków ich użycia dotyczy wykrywania chorób neurodegeneracyjnych, mających wpływ na produkcję mowy: choroby Alzheimera oraz choroby Parkinsona. Problemem diagnostyki pierwszej z tych chorób zajęli się Pappagari i in. [122]. X-wektory posłużyły im do treningu modelu drzewiastego opartego o algorytm *XGBoost*, dzięki czemu osiągnięto dokładność klasyfikacji wynoszącą 71,1%. Połączenie modeli drzewiastych uczonych na x-wektorach z modelem językowym BERT (ang. *bidirectional encoder representations from transformers*) [123] pozwoliło natomiast zwiększyć dokładność klasyfikacji do 81,3%. Zamiana klasyfikatora drzewiastego na klasyfikator oparty o probabilistyczną liniową analizę dyskryminacyjną (ang. *probabilistic linear discriminant analysis*, PLDA) spowodowało wzrost dokładności klasyfikacji do 75,93%, zaś jego połączenie z modelem BERT wzrost do 81,48% [124]. Tematu diagnostyki choroby Parkinsona podjęli się natomiast Jeancolas i in. [125] – stosując liniową analizę dyskryminacyjną (ang. *linear discriminant analysis*) zdołali osiągnąć błąd EER (ang. *equal error rate*) wynoszący 32%. Autorka niniejszej rozprawy jest zaś współautorką artykułu dotyczącego możliwości zastosowania x-wektorów do diagnostyki chorób otolaryngologicznych. Zaprezentowana metoda wykorzystująca perceptron wielowarstwowy (ang. *multi-layer perceptron*, MLP) pozwoliła na uzyskanie dokładności klasyfikacji wynoszącej 81,84% [87]. W kolejnej pracy autorka zaprezentowała natomiast klasyfikator SGD (ang. *stochastic gradient descent*) uczony na x-wektorach, który umożliwił odróżnienie osób chorych na obrzęk Reinkego od osób zdrowych z dokładnością wynoszącą 89% [126].

W niektórych pracach zastosowano nie tylko x-wektory, ale również i-wektory opisane w rozdziale 2, sekcji 2.6.1. Przykładem takiej pracy jest artykuł opublikowany poruszający tematykę automatycznej diagnostyki choroby Alzheimera. Haj Zargarbashi i Babaali [127] przedstawili metodę wykorzystującą model SVM trenowany na x-wektorach, i-wektorach oraz estymatach zwracanych przez językowy model n-gramowy [128], osiągając dokładność klasyfikacji wynoszącą 83,6%. Pompili i in. [129] uzyskali natomiast dokładność klasyfikacji 81,25%, również stosując model SVM trenowany na wektorach własnościowych oraz model językowy oparty o sieć rekurencyjną LSTM (ang. *long-short term memory*). Z kolei Pérez-Toro i in. [130] zastosowali model SVM z radialną funkcją bazową (ang. *radial basis function-support vector machine*, RBF-SVM), który pozwolił odróżnić osoby chore od osób chorych na chorobę Alzheimera w 59% przypadków. Metodę wykrywania choroby Parkinsona przedstawili Moro-Valazquez i in. [131], którzy zastosowali modele PLDA uczone na i-wektorach oraz x-wektorach , osiągające odpowiednio dokładność klasyfikacji 87% oraz 90%. Botelho i in. [132] użyli wektorów własnościowych w systemach detekcji dwóch schorzeń: choroby Parkinsona oraz bezdechu sennego. W przypadku pierwszej choroby i-wektory umożliwiły osiągnięcie lepszych rezultatów niż x-wektory – precyzja modelu wyniosła odpowiednio 75,6% oraz 74,4%, zaś czułość odpowiednio 75,7% oraz 74,5%. Metodę diagnostyki bezdechu sennego przedstawili również Perero-Codosero i in. [133]. W swojej pracy przedstawili badania dotyczące oceny stopnia zaawansowania choroby na podstawie i-wektorów, x-wektorów oraz połączenia obu wektorów własnościowych. Sieć neuronowa złożona z dwóch warstw gęstych posłużyła do oceny, czy dany pacjent cierpi na zaawansowany bezdech senny, czy lżejszą postać tej choroby. Przedstawione przez autorów rezultaty wskazują na niewielki związek cech sygnału mowy zakodowanych za pomocą wektorów własnościowych a stopniem zaawansowania choroby - osiągnięty współczynnik korelacji wartości zwracanych przez sieć oraz wartości rzeczywistych wynosi nie więcej niż 0,32. Z kolei Pan i in. [134] zaprezentowali metodę oceny stopnia zaawansowania zaburzeń poznawczych w przebiegu demencji. Użyty przez nich model SVM trenowany na i-wektorach i x-wektorach osiągnął pierwiastkowy błąd średniokwadratowy (ang. *root mean square error*, RMSE) wynoszący odpowiednio 7,25 otaz 4,35. Moro-Valazquez i in. Moro-Valazquez i in. [135] przedstawili natomiast badania dotyczące wpływu trzech laryngologicznych procedur chirurgicznych – tonsillektomii (usunięcia migdałków podniebiennych), operacji zatok przynosowych metodą FESS (ang. *functional endoscopic sinus surgery*, FESS) oraz septoplastyki (korekcji skrzywienia przegrody nosowej) – na sygnał mowy. Uzyskane rezultaty wskazują na istotny wpływ pierwszych dwóch procedur na sygnał mowy, co zostało wykazane za pomocą modelu PLDA uczonego na i-wektorach oraz x-wektorach oraz testu Kruskala-Wallisa, który zbadał różnice w przeciętnym poziomie predykcji zwracanych przez model dla osób cierpiących na daną chorobę oraz osób zdrowych tworzących grupę kontrolną. W przypadku septoplastyki nie wykazano istotnego wpływu przeprowadzonego zabiegu na produkcję mowy.

4. Materiał badawczy

Korpusy mowy użyte w niniejszej pracy zostały dobrane w taki sposób, aby w szerokim zakresie przebadać i przeanalizować przydatność wektorów własnościowych mówcy w systemach wspomagania diagnostyki medycznej. W badaniach wykorzystano ogólnodostępne korpusy mowy, których liczba jest bardzo ograniczona, stąd dobór stosunkowo rzadko analizowanych języków takich jak język czeski i niderlandzki. Różnorodność języków tworzących użyte bazy można jednak uznać za zaletę, ponieważ umożliwiła zbadanie skuteczności proponowanych rozwiązań w przypadku użycia mowy w językach należących do różnych grup językowych: języków germańskich oraz języków słowiańskich.

Oczywistym zastosowaniem sygnału mowy w diagnostyce medycznej jest diagnostyka schorzeń aparatu mowy – wszelkie jego nieprawidłowości mają wpływ na emisję głosu, często zauważalny nie tylko przez lekarza, ale też niewykształconego w tym kierunku człowieka. Zbadanie możliwości użycia cech ekstrahowanych z sygnału mowy do detekcji schorzeń otolaryngologicznych jest więc uzasadnionym pierwszym etapem badań.

Kolejną grupą chorób mających wpływ na emisję sygnału mowy oraz jego jakość są zaburzenia neurorozwojowe. Zaburzenia te mają istotny wpływ na rozwój mowy u dzieci i często wiążą się z koniecznością podjęcia terapii logopedycznej. Wydaje się więc prawdopodobne, że wektory własnościowe mówcy zawierające w sobie informacje o cechach osobniczych mówcy mają potencjał zagwarantować dobre rezultaty w detekcji tych zaburzeń na podstawie sygnału mowy.

Warto zauważyć, że wykorzystanie sygnału mowy do diagnostyki zaburzeń neurorozwojowych jest zagadnieniem dużo mniej trywialnym, niż zastosowanie go do diagnostyki chorób otolaryngologicznych. Analiza porównawcza wyników uzyskanych dla szerokiego spektrum problemów medycznych – opisanych w niniejszej pracy oraz przedstawionych w rozdziale 3, sekcji 3.3 – może stanowić cenny materiał pozwalający ocenić przydatność x-wektorów do rozwiązywania zagadnień klasyfikacyjnych w medycynie o różnym stopniu trudności.

4.1. Korpus mowy osób ze schorzeniami krtani

Pierwszą bazą, na której przeprowadzono badania jest korpus mowy niemieckiej *Saarbrücken Voice Database* (SVD) udostępniony przez Instytut Fonetyki w Saarland w Niemczech [136]. W bazie SVD zawarte są nagrania pochodzące od ponad 2000 osób wypowiadających frazę *Guten Morgen, wie geht es Ihnen?* oraz trzy samogłoski o przedłużonej fonacji – \a\, \i\

oraz \u\ – w tonacji normalnej, wysokiej, niskiej oraz modulowanej. Wszystkie nagrania zostały zarejestrowane z częstotliwością próbkowania 50 kHz oraz rozdzielczością 16 bitów.

W badaniach wykorzystano nagrania zawierające frazę – są one dostępne dla 1897 osób, w tym 634 osób zdrowych oraz 1263 chorych cierpiących na łącznie 71 schorzeń narządu mowy. W celu detekcji osób chorych, bez odróżniania chorób, wykorzystano wszystkie sygnały. W celu detekcji pojedynczego schorzenia wykorzystano jedynie sygnały pochodzące od osób zdrowych oraz osób, u których zdiagnozowano jedną z pięciu najbardziej licznych chorób obecnych w bazie: dysfonię, ostre zapalenie krtani, porażenie nerwu krtaniowego wstecznego, obrzęk Reinkego oraz pachydermię kontaktową. Ponadto, ponieważ w bazie wyszczególniono pięć typów dysfonii – funkcjonalną, hiperfunkcjonalną, hipofunkcjonalną, spastyczną oraz psychogenną – analizy przeprowadzono zarówno z podziałem na typy, jak i bez takiego podziału. Liczbę pacjentów cierpiących na wymienione schorzenia zestawiono w tabeli 4.1.

Tabela 4.1. Liczność nagrań w bazie SVD pochodzących od osób zdrowych oraz cierpiących na pięć wybranych schorzeń krtani.

Nazwa choroby		Liczba nagrań
Dysfonia	bez określonego typu	100
	funkcjonalna	111
	hiperfunkcjonalna	211
	hipofunkcjonalna	16
	spastyczna	65
	psychogenna	76
Obrzęk Reinkego		39
Ostre zapalenie krtani		123
Pachydermia kontaktowa		49
Porażenie nerwu krtaniowego wstecznego		200
Zdrowi		634

4.2. Korpus mowy dzieci ze specyficznymi zaburzeniami językowymi

Detekcja zaburzeń neurorozwojowych przeprowadzona została przy użyciu dwóch korpusów mowy. Jednym z nich jest korpus mowy czeskiej zawierający mowę dzieci zdrowych oraz cierpiących na specyficzne zaburzenia językowe (ang. *specific language impairment*, SLI) opublikowany przez zespół badawczy z laboratorium LANNA z Pragi w Republice Czeskiej [137]. W skład korpusu wchodzą nagrania 44 zdrowych dzieci w wieku od 4 do 10 lat, zarejestrowane w sali lekcyjnej lub na terenie Szpitala Klinicznego Uniwersytetu Karola Motol w Pradze, oraz 54 chorych dzieci w wieku od 4 do 13 lat, zarejestrowane na terenie Szpitala Klinicznego lub w gabinecie prywatnego terapeuty. Dla każdego uczestnika badania zarejestrowano następujące wypowiedzi:

– samogłoski o przedłużonej fonacji: \a\, \e\, \i\, \o\, \u\,

– spółgłoski: m, b, t, d, r, l, k, g, h, ch,
– sylaby: pe, la, vla, pro, bě, nos, ber, krk, prst,
– słowa dwusylabowe: kolo, pivo, sokol, papír, trdlo,
– słowa trzysylabowe: dědeček, pohádka, pokémon, květina,
– słowa czterowylabowe: motovidlo, televize, popelnice,
– słowa pięciosylabowe: různobarevný, mateřídouška.

W grupie dzieci zdrowych zarejestrowano łącznie 1658 wypowiedzi, natomiast w grupie dzieci chorych 2103 wypowiedzi. Podczas badań wykorzystano jedynie wypowiedzi zawierające sylaby oraz słowa: 998 wypowiedzi dzieci zdrowych oraz 1293 wypowiedzi dzieci chorych.

Nagrania dzieci cierpiących na SLI zarejestrowane zostały z częstotliwością próbkowania 44,1 kHz i rozdzielczością 16 bitów, natomiast nagrania dzieci zdrowych z częstotliwością 16 kHz i rozdzielczością 16 bitów. W celu ujednolicenia parametrów dokonano zmiany częstotliwości próbkowania nagrań dzieci chorych na 16 kHz.

4.3. Korpus mowy dzieci z zaburzeniami ze spektrum autyzmu

Ostatnią bazą wykorzystaną podczas badań jest holenderski korpus mowy utworzony w ramach projektu Assymetries realizowanego w Uniwersytecie w Groningen w Holandii [138]. W skład bazy wchodzą nagrania mowy swobodnej 38 dzieci zdrowych w wieku od 6 do 12 lat oraz 46 dzieci ze zdiagnozowanymi zaburzeniami ze spektrum autyzmu (ang. *autism spectrum disorder*, ASD) w wieku od 6 do 12 lat. Nagrania zostały zarejestrowane w trakcie sesji terapeutycznej z wykwalifikowanym terapeutą, podczas której badane dzieci miały opowiedzieć cztery historie przedstawione w formie graficznej, załączone w dodatku A (Rys. A.1).

Ponieważ nagrania zawierają nie tylko wypowiedzi dzieci, ale też wypowiedzi terapeuty oraz zakłócenia, zostały poddane ręcznej segmentacji. Segmentację przeprowadzono w taki sposób, by uzyskane fragmenty nagrania były możliwie długie – koniec danego segmentu wyznaczany był poprzez wystąpienie w nagraniu znaczącego zakłócenia lub zmianę mówcy. Dalszym analizom poddano wyłącznie segmenty zawierające mowę dziecka.

Nagrania dzieci zdrowych zarejestrowane zostały z częstotliwością próbkowania 24 kHz, natomiast dzieci ze zdiagnozowanym ASD z częstotliwością próbkowania 44,1 kHz. Podobnie jak w przypadku korpusu mowy czeskiej, dokonano zmiany częstotliwości próbkowania wszystkich nagrań na 16 kHz.

5. Schemat prowadzenia badań

Skuteczna klasyfikacja danych medycznych uznawana jest za skomplikowane, problematyczne zagadnienie. Wyniki przedstawiane w publikacjach naukowych są często bardzo obiecujące, jednak po wdrożeniu proponowanych rozwiązań do praktyki klinicznej okazuje się, że uzyskiwane rezultaty są dalekie od zadowalających. Ma to związek z trudnym dostępem do dużych zbiorów danych, które zawierałyby reprezentatywną próbę z całej populacji osób cierpiących na dane schorzenie – niewielka liczba danych wykorzystywanych podczas prac nad algorytmami sztucznej inteligencji i ich zastosowaniami w medycynie skutkuje uzyskaniem modeli o słabych zdolnościach generalizacyjnych. Szczególnie duże różnice pomiędzy wynikami uzyskiwanymi w czasie treningu modeli klasyfikacyjnych i wynikami uzyskiwanymi w praktycznym ich zastosowaniu obserwuje się w przypadku analizowania danych o wysokiej wymiarowości. Do takich danych zaliczają się dane akustyczne, w szczególności sygnał mowy [38].

5.1. Uzasadnienie wyboru użytych narzędzi

W przypadku niewielkich rozmiarów analizowanych baz danych często przyjmowaną strategią jest k-krotny sprawdzian krzyżowy, w którym dane dzielone są na k podzbiorów. Proces uczenia klasyfikatora prowadzony jest k-krotnie w taki sposób, by każdy z k podzbiorów został jednokrotnie wykorzystany jako zbiór testowy, podczas gdy pozostałe k-1 podzbiorów tworzą zbiór uczący. Selekcja cech, redukcja wymiarowości i optymalizacja hiperparametrów modelu są wówczas prowadzone na całym zbiorze danych, co może prowadzić do uzyskania niemiarodajnych wyników, w szczególności zaniżenia rzeczywistego błędu predykcji [139], [140]. Rozwiązaniem tego problemu jest podział danych na zbiór uczący i testowy przed przystąpieniem do sprawdzianu krzyżowego oraz przeprowadzenie go jedynie z wykorzystaniem zbioru uczącego. Co ważne, niedopuszczalne jest używanie błędu popełnianego na zbiorze testowym jako funkcji celu minimalizowanej w procesie optymalizacji modelu – aby zachować wiarygodność prowadzonych analiz, zbiór testowy powinien służyć jedynie do oceny zdolności predykcyjnych modelu uzyskanego w wyniku treningu i optymalizacji na zbiorze uczącym [38], [139], [141].

Istotnym zagadnieniem związanym z optymalizacją modeli klasyfikacyjnych jest dobór odpowiedniej funkcji celu. W wielu implementacjach algorytmów uczenia maszynowego domyślną stosowaną metryką służącą do oceny modelu jest dokładność klasyfikacji (równanie 5.1). Dokładność jest jednak uznawana za metrykę mogącą prowadzić do nieprawidłowego wyboru modelu – jak wskazano w [142], dokładność klasyfikacji jest zależna od arbitralnie przyjętej wartości punktu odcięcia wykorzystywanego do dyskretyzacji ciągłych predykcji zwracanych przez model. Może to spowodować utratę informacji potencjalnie istotnych z punktu widzenia diagnostyki medycznej, a w konsekwencji prowadzić do wyciągnięcia nieprawidłowych wniosków z prowadzonych analiz i podjęcia decyzji zagrażającej zdrowiu lub życiu pacjenta. Modele używane w diagnostyce medycznej powinny więc zwracać ciągłe prawdopodobieństwo przynależności do klasy, a dyskretyzacja powinna być przeprowadzona tylko jednokrotnie, na najpóźniejszym możliwym etapie analizy [142], [143].

W przypadku modeli używanych w zagadnieniach binarnych, zwracających prawdopodobieństwo przynależności obiektu do jednej z dwóch klas, rekomendowaną i często stosowaną funkcją celu jest pole pod krzywą ROC (ang. *area under receiver-operator characteristic curve*, AUC ROC, równanie 5.5). Jednakże w literaturze dostępne są opracowania wskazujące, że w przypadku danych o niezrównoważonej liczebności klas wysoka wartość AUC ROC jest możliwa do uzyskania również wtedy, gdy poprawność klasyfikacji klasy mniej licznej będzie znikoma. Jest to szczególnie problematyczne w przypadku danych medycznych, w których liczba osób zdrowych często znacznie przewyższa liczbę osób chorych [144], [145]. Zgodnie z przedstawionymi w cytowanych pracach analizami, najbardziej wiarygodne modele diagnostyczne można uzyskać maksymalizując pole pod krzywą precyzja-czułość (ang. *area under curve precision-recall*, AUC PR, równanie 5.6) podczas treningu oraz optymalizacji. Udowodniono również, że optymalizacja AUC ROC nie gwarantuje równoczesnej optymalizacji AUC PR, co potwierdza istotność odpowiedniego wyboru funkcji celu w przypadku niezrównoważenia liczebności klas [146].

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.1}$$

$$TPR = \frac{TP}{TP + FN} \tag{5.2}$$

$$TNR = \frac{TN}{FP + TN} \tag{5.3}$$

$$PPV = \frac{TP}{TP + FP} \tag{5.4}$$

gdzie TP - predykcje prawdziwie pozytywne, FP - fałszywie pozytywne, TN - prawdziwie negatywne, FN - fałszywie negatywne, TPR - czułość klasyfikatora, PPV - precyzja klasyfikatora, TNR - swoistość klasyfikatora.

$$AUCROC = \int_0^1 \mathrm{ROC}(t)\,\mathrm{d}t = \int_0^1 \mathrm{TPR}((1-\mathrm{TNR})^{-1}(\mathrm{t}))\,\mathrm{d}t \tag{5.5}$$

$$AUCPR = \int_0^1 \mathrm{PR}(t)\,\mathrm{d}t = \int_0^1 \mathrm{PPV}(\mathrm{TPR}^{-1}(\mathrm{t}))\,\mathrm{d}t \tag{5.6}$$

W wykorzystywanym podczas pierwszego etapu badań korpusie *Saarbrücken Voice Database* (SVD) można zaobserwować wyraźne różnice w liczebności poszczególnych grup pacjentów – duża różnica jest widoczna szczególnie pomiędzy liczbą osób zdrowych (634), a liczbą osób ze zdiagnozowaną dysfonią hipofunkcjonalną (16), spastyczną (65) oraz psychogenną (76) (tabela 4.1). Z tego względu zdecydowano o zastosowaniu AUC PR jako funkcji celu, która ma być maksymalizowana w procesie optymalizacji.

Odpowiedni dobór strategii uczenia, optymalizacji i walidacji modeli nie jest jednak wystarczający w sytuacji, gdy dane cechują się wysoką wymiarowością przy równoczesnej małej liczebności. Równie istotny jest właściwy sposób pozyskania oraz wstępnego przetwarzania danych [38]. Ponieważ w niniejszej pracy wykorzystano jedynie ogólnodostępne korpusy mowy, nie było możliwe zadbanie o właściwe warunki rejestracji sygnałów oraz zapewnienie odpowiedniej liczby nagrań, gwarantującej reprezentatywność danych. Skupiono się więc na doborze cech ekstrahowanych z sygnałów, które zapewnią możliwie duże uogólnienie modeli klasyfikacyjnych i pozwolą zminimalizować negatywne skutki małych rozmiarów analizowanych baz danych.

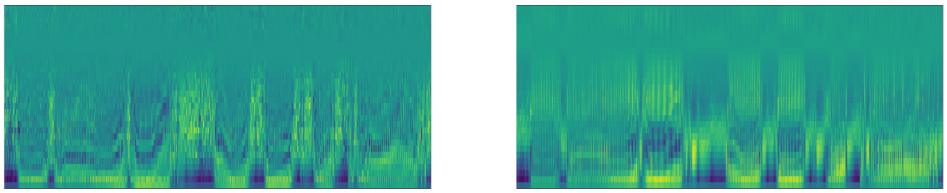
Diagnostyka medyczna na podstawie mowy najczęściej przeprowadzana jest w oparciu o analizę akustyczną samogłosek o przedłużonej fonacji. Ma to na celu zapewnienie stabilności wyznaczanych parametrów akustycznych związanej z występowaniem stanu ustalonego podczas fonacji samogłosek [147]. Jednakże opublikowane w 2020 roku analizy powtarzalności uzyskiwanych wartości cech akustycznych [148] wskazują na bardzo dużą zmienność popularnych cech takich jak częstotliwości formantowe, jitter czy shimmer. Co więcej, mniejszą zmiennością w czasie charakteryzują się cechy wyznaczane nie na podstawie samogłosek o przedłużonej fonacji, lecz dłuższych wypowiedzi. Autorzy wskazują ponadto na konieczność wykorzystywania w diagnostyce medycznej cech akustycznych charakteryzujących się dużą powtarzalnością w obrębie wypowiedzi danego mówcy.

5.2. Opis wykonanych prac oraz analiza wyników

5.2.1. Detekcja schorzeń otolaryngologicznych

Pierwszym etapem przeprowadzonych badań była analiza przydatności x-wektorów ekstrahowanych z wykorzystaniem współczynników MFCC i GFCC wyznaczonych na podstawie

mowy ciągłej (rys. 5.1) w detekcji schorzeń otolaryngologicznych. Prace rozpoczęto od przeprowadzenia klasyfikacji binarnej mającej na celu odróżnienie osób zdrowych od osób ze zdiagnozowaną chorobą krtani. Klasyfikację przeprowadzono wykorzystując korpus SVD opisany w rozdziale 4, sekcji 4.1.



Rys. 5.1. Przykładowe współczynniki MFCC (po lewej) i GFCC (po prawej) ekstrahowane z sygnału mowy osoby ze zdiagnozowanym schorzeniem otolaryngologicznym.

5.2.1.1. Pretrenowana sieć neuronowa

X-wektory ekstrahowane z MFCC zostały wyznaczone przy użyciu pretrenowanej głębokiej sieci neuronowej (DNN) opisanej w [149] (rys. 5.4a) służącej do rozpoznawania mówcy, charakteryzującej się niższą wartością EER (ang. *equal error rate*) niż konkurencyjne DNN opisywane w literaturze w momencie publikacji artykułu. W tym celu postępowano zgodnie z algorytmem zaprezentowanym przez autorów: sygnały akustyczne podzielono na ramki o długości 25 ms z nakładkowaniem 15 ms, a następnie z każdej ramki wyekstrahowano 30 współczynników mel-cepstralnych. Na uzyskanych w ten sposób współczynnikach MFCC przeprowadzono detekcję aktywności mówcy, używając algorytmu opartego na analizie energii sygnału zaimplementowanego w pakiecie Kaldi [150] oraz usunięto ramki niezawierające mowy. Tak przygotowane dane zostały podane do sieci neuronowej, w wyniku czego uzyskano 512-wymiarowe wektory własnościowe mówcy.

Przed przystąpieniem do wyznaczenia x-wektorów ekstrahowanych z GFCC sygnałów zawartych w bazie SVD, konieczne było przeprowadzenie procesu uczenia sieci neuronowej wykorzystywanej do rozpoznawania mówcy. Skorzystano w tym celu z bazy danych Voxceleb2 [151], zawierającej ponad milion wypowiedzi około 7300 osób. Sygnały podzielono na ramki o długości 25 ms z nakładkowaniem 15 ms i dla każdej ramki wyznaczono 30 współczynników GFCC. Następnie przeprowadzono filtrację filtrem ARMA (ang. *autoregressive moving average*), wyliczono długoczasową zmienność sygnału (ang. *long-term signal variability*, LSTV) i na jej podstawie przeprowadzono detekcję aktywności mówcy. Wszystkie operacje przeprowadzono przy użyciu pakietu Featxtra [152].

Przygotowane w ten sposób dane wykorzystano do treningu głębokiej sieci neuronowej zaimplementowanej z użyciem biblioteki PyTorch [153]. Wykorzystano sieć o architekturze takiej samej, jaką ma pretrenowana sieć użyta do ekstrakcji x-wektorów z MFCC - dzięki temu różnice w skuteczności detekcji schorzeń krtani w oparciu o x-wektory ekstrahowane z MFCC i GFCC nie powinny wynikać z innych czynników niż różnice w parametryzacji sygnałów akustycznych przeprowadzonej przed ekstrakcją wektorów własnościowych. Po zakończeniu

treningu sieci neuronowej wykorzystano ją do ekstrakcji x-wektorów z sygnałów zawartych w bazie SVD zgodnie z procedurą opisaną powyżej.

Uzyskane dwa warianty x-wektorów podzielono na zbiory uczące i testowe, stanowiące odpowiednio 80% i 20% danych (tabela 5.1). Zbiory uczące wykorzystano do przeprowadzenia treningu modeli, które posłużyły do klasyfikacji osób zdrowych i cierpiących na schorzenia krtani. Wybrano 12 modeli należących do pięciu rodzajów algorytmów klasyfikacyjnych:

- modele liniowe:
  - regresję logistyczną (LR),
  - klasyfikator oparty o stochastyczne zejście gradientowe (SGD),
- modele wykorzystujące wnioskowanie bayesowskie:
  - liniową analizę dyskryminacyjną (LDA),
  - kwadratową analizę dyskryminacyjną (QDA),
- modele drzewiaste:
  - AdaBoost,
  - ExtraTrees,
  - klasyfikator drzewiasty zaimplementowany przy użyciu pakietu LightGBM (LGB) [154],
  - las losowy (RF),
- modele jądrowe:
  - modyfikację maszyny wektorów nośnych (NuSVM),
- modele minimalnoodległościowe:
  - klasyfikator k najbliższych sąsiadów (KNN),
  - klasyfikator najbliższych sąsiadów analizujący otoczenie o zadanym promieniu (RNN),
- sieci neuronowe:
  - perceptron wielowarstwowy (MLP).

Nie zdecydowano się na użycie sieci neuronowych o bardziej skomplikowanej strukturze ze względu na zbyt wysokie ryzyko przeuczenia związane z niewielkim rozmiarem analizowanego korpusu mowy. Wszystkie wymienione powyżej modele, z wyjątkiem LGB, zaimplementowano w bibliotece scikit-learn [11] w języku Python.

Tabela 5.1. Liczebność zbiorów uczącego i testowego użytych podczas treningu i ewaluacji modeli klasyfikacyjnych wykorzystywanych do detekcji schorzeń krtani.

	Liczba osób	
Zbiór	Chorzy	Zdrowi
Uczący	1010	507
Testowy	259	127

Każdy z wymienionych modeli poddano optymalizacji mającej na celu dobór hiperparametrów, które zagwarantują uzyskanie możliwie dużej wartości AUC PR – użyto w tym celu algorytmu bazującego na estymatorze Parzena o strukturze drzewiastej [32] zaimplementowanego w pakiecie Optuna [33]. Równocześnie optymalizowano metodę redukcji wymiarowości, dobierając metodę gwarantującą najlepsze wyniki spośród metod opisanych w rozdziale 2, sekcji 2.3, oraz liczbę cech, które otrzymywano w jej wyniku i które posłużyły do treningu optymalizowanego modelu – użycie pełnych 512-wymiarowych wektorów cech mogłoby skutkować znacznym przeuczeniem modeli i pogorszeniem ich zdolności do generalizacji, szczególnie w przypadku stosunkowo niewielkiego zbioru danych uczących [38], [155]. Optymalizację przeprowadzono stosując 5-krotny stratyfikowany sprawdzian krzyżowy.

Najlepszy model spośród modeli trenowanych na tych samych cechach wybrany został na podstawie uzyskanych wartości AUC ROC. Metryka ta została wybrana ze względu na fakt, że rozwiązania opisane w literaturze ewaluowane były zazwyczaj właśnie przy użyciu AUC ROC lub dokładności - jednak, jak opisano wcześniej, dokładność klasyfikacji nie jest uznawana za metrykę dobrze charakteryzującą jakość klasyfikatora, więc zdecydowano się na zastosowanie AUC ROC. Dzięki temu możliwe było porównanie uzyskiwanych wyników z tymi dostępnymi w literaturze oraz, w dalszym etapie badań, wyliczenie na jej podstawie innych metryk takich jak dokładność klasyfikacji (równanie 5.1), czułość (równanie 5.2) czy swoistość (równanie 5.3).

Aby określić, który wariant x-wektorów pozwala uzyskać większą skuteczność detekcji schorzeń krtani, porównano wartości AUC ROC uzyskane przez najlepsze modele – w obu przypadkach były to modele MLP (tabela 5.2). Proste porównanie wartości nie gwarantuje jednak tego, że model o wyższym AUC ROC faktycznie jest istotnie lepszy, przeprowadzono więc analizę istotności statystycznej uzyskanych różnic, stosując w tym celu bootstrapowy przedział ufności różnic w metryce AUC ROC. Zdecydowano się na użycie percentylowych bootstrapowych przedziałów ufności – cechują się one dużą efektywnością w porównaniu do innych metod wyznaczania bootstrapowych przedziałów ufności oraz małą wrażliwością na rozmiar próby, co jest szczególnie istotne w przypadku analizowania niewielkich zbiorów danych [78]. Ponadto zdecydowano, że w przypadku braku istotnych statystycznie różnic w wartościach AUC ROC, na przewagę modelu wskazywać będzie istotna statystycznie różnica w uzyskiwanych wartościach straty logarytmicznej (równanie 5.7) – model o mniejszej wartości straty

logarytmicznej cechuje się lepszymi zdolnościami generalizacyjnymi. Schemat postępowania prowadzącego do wyboru najlepszego modelu przedstawiono na rys. 5.2.

W analizowanym przypadku bootstrapowy przedział ufności różnic pola pod krzywą ROC wynosi [-0.0364; -0.0131]. Oznacza to, że różnice pomiędzy AUC ROC uzyskiwanymi przez oba modele są istotne statystycznie. Fakt, że w przedziale znajdują się wyłącznie wartości ujemne świadczy ponadto o tym, że AUC ROC uzyskane przez MLP uczony na x-wektorach ekstrahowanych z GFCC jest istotnie większe. Zgodnie z przyjętą metodą porównywania modeli uznano więc, że model ten jest lepszy pod względem zdolności do odróżniania osób zdrowych od osób cierpiących na schorzenia krtani.

Tabela 5.2. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji osób chorych otolaryngologicznie. Najwyższe wartości metryki dla każdego rodzaju cech użytych do ekstrakcji x-wektorów oznaczono czcionką pogrubioną.

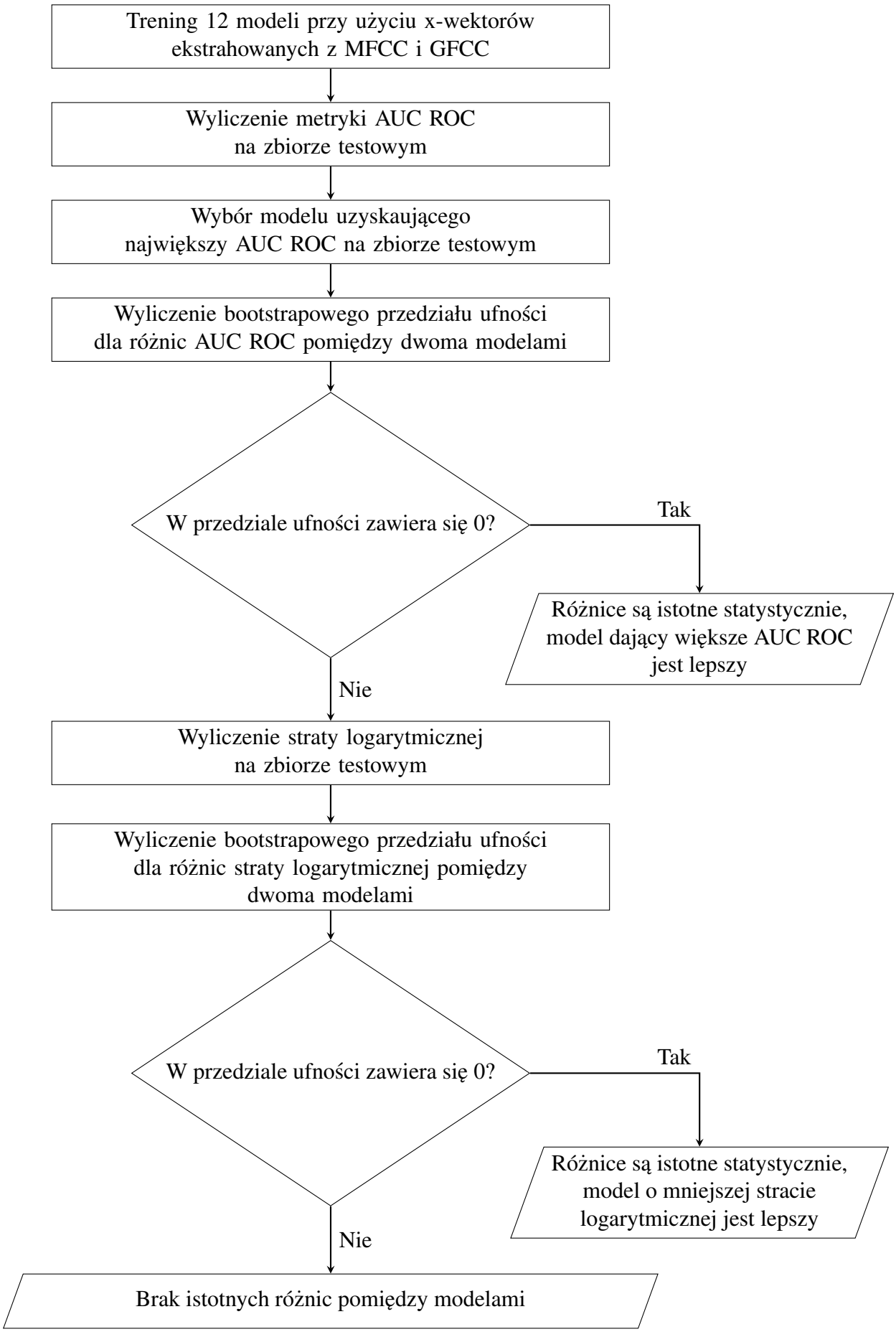
	AUC ROC	
Model	MFCC	GFCC
AdaBoost	0.8145	0.8250
Extra Trees	0.7475	0.8043
KNN	0.8162	0.8588
LDA	0.7494	0.7832
LGB	0.7671	0.8397
LR	0.7497	0.7950
MLP	**0.8494**	**0.8748**
NuSVM	0.8406	0.8619
QDA	0.7026	0.7890
RF	0.6825	0.7892
RNN	0.7260	0.8284
SGD	0.8389	0.8717

$$L_{log}(y, p) = -(y \cdot log(p) + (1 - y) \cdot log(1 - p)) \tag{5.7}$$

gdzie L_{log} – strata logarytmiczna, y – klasa, do które przynależy obiekt, p – prawdopodobieństwo przynależności do klasy pozytywnej [11].

5.2.1.2. Modyfikacje sieci neuronowej

Kolejnym etapem prac była modyfikacja sieci neuronowej służącej do ekstrakcji x-wektorów. Ponieważ GFCC okazały się być cechami pozwalającymi na uzyskanie x-wektorów dających istotnie lepsze rezultaty w omawianym zagadnieniu klasyfikacyjnym,



Rys. 5.2. Schemat procedury wyboru najlepszego modelu – postępowanie w przypadku dwóch wariantów x-wektorów.

wszelkie opisywane dalej modyfikacje dotyczą jedynie sieci przyjmującej na wejściu współczynniki GFCC. Analogiczne eksperymenty nie zostały przeprowadzone z użyciem MFCC ze względu na duże koszty obliczeniowe oraz ograniczone zasoby sprzętowe i czasowe.

Początkowo funkcja kosztu stosowana podczas treningu i optymalizacji wag DNN zdefiniowana była jako entropia krzyżowa ze znormalizowaną funkcją wykładniczą (ang. *softmax*) (równanie 5.8), będącą uogólnieniem funkcji logistycznej. Funkcja *softmax* zwraca wartości z przedziału [0; 1], które sumują się do 1, a więc mogą być interpretowane jako prawdopodobieństwo przynależności klasyfikowanego obiektu do poszczególnych klas w sytuacji, gdy *softmax* jest funkcją aktywacji wartwy wyjściowej sieci. Tak określona funkcja kosztu jest powszechnie stosowana w DNN wykorzystywanych w wielu zagadnieniach klasyfikacyjnych, takich jak rozpoznawanie mówcy, rozpoznawanie emocji czy rozpoznawanie obrazów. Należy jednak zauważyć, że w niektórych przypadkach funkcja kosztu oparta o funkcję *softmax* nie jest wystarczająca do uzyskania wystarczającej separacji klas [156], [157]. Z tego względu podjęte zostały prace nad modyfikacją funkcji kosztu w sposób zwiększający separowalność klas [156]–[159]. Pierwszą wprowadzoną w niniejszej pracy modyfikacją względem pierwotnej architektury sieci była więc zmiana funkcji kosztu. Zdecydowano o użyciu funkcji ArcFace (ang. *additive angular margin loss*, równanie 5.9) opisanej w [157], ze względu na fakt, że maksymalizuje ona odległość geodezyjną wektorów własnościowych obiektów należących do różnych klas, rozmieszczonych na znormalizowanej hipersferze. Przewagą funkcji ArcFace nad innymi funkcjami zwiększającymi separowalność klas, takich jak SphereFace [156] czy CosFace [158], [160] jest fakt, że uzyskiwane w ten sposób hiperpłaszczyzny decyzyjne oraz tworzony przez nie margines klasyfikatora w przestrzeni kątowej są liniowe (rys. 5.3).

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N} log\frac{exp(W_{y_i}^T x_i + b_{y_i})}{\sum_{j=1}^{n} exp(W_j^T x_i + b_j)} \tag{5.8}$$

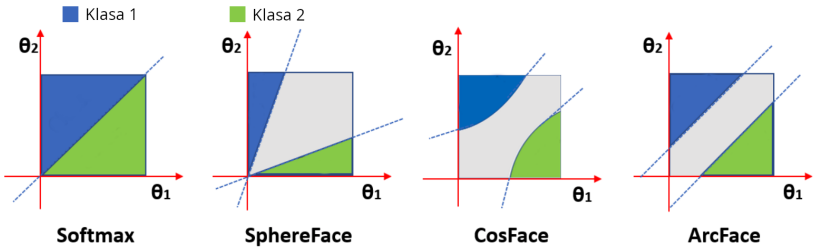
gdzie $x_i \in \Re^d$ – wektor własnościowy i-tej próbki należącej do y_i klasy, d - rozmiar wektora własnościowego x_i, $W_j \in \Re^d$ – j-ta kolumna macierzy wag $W \in \Re^{dxn}$, $B_j \in \Re^n$ – bias, N – rozmiar wsadu, n – liczba klas [157].

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N} log\frac{exp(s(cos(\theta_{y_i} + m)))}{exp(s(cos(\theta_{y_i} + m))) + \sum_{j=1, j\neq y_i}^{n} exp(s(cos\theta_j))} \tag{5.9}$$

gdzie θ_j – kąt pomiędzy wektorem wag W_j i wektorem własnościowym x_i, s – promień znormalizowanej hipersfery, na której rozmieszczone są wektory własnościowe, m – parametr marginesu pomiędzy x_i i W_j [157].

W dalszej części pracy w odniesieniu do sieci z modyfikacją funkcji kosztu uczonej na GFCC oraz do wyekstrahowanego w ten sposób wariantu x-wektorów stosowane będzie określenie GFCC-Arc.

Kolejną wprowadzoną modyfikacją była zmiana rodzaju użytej sieci. W opisywanych dotychczas wariantach występowała sieć neuronowa z opóźnieniem czasowym (ang. *time-delay*



Rys. 5.3. Porównanie marginesów klasyfikatorów binarnych z funkcjami kosztu bazującymi na czterech funkcjach: *softmax*, SphereFace, CosFace, ArcFace. Źródło: [157].

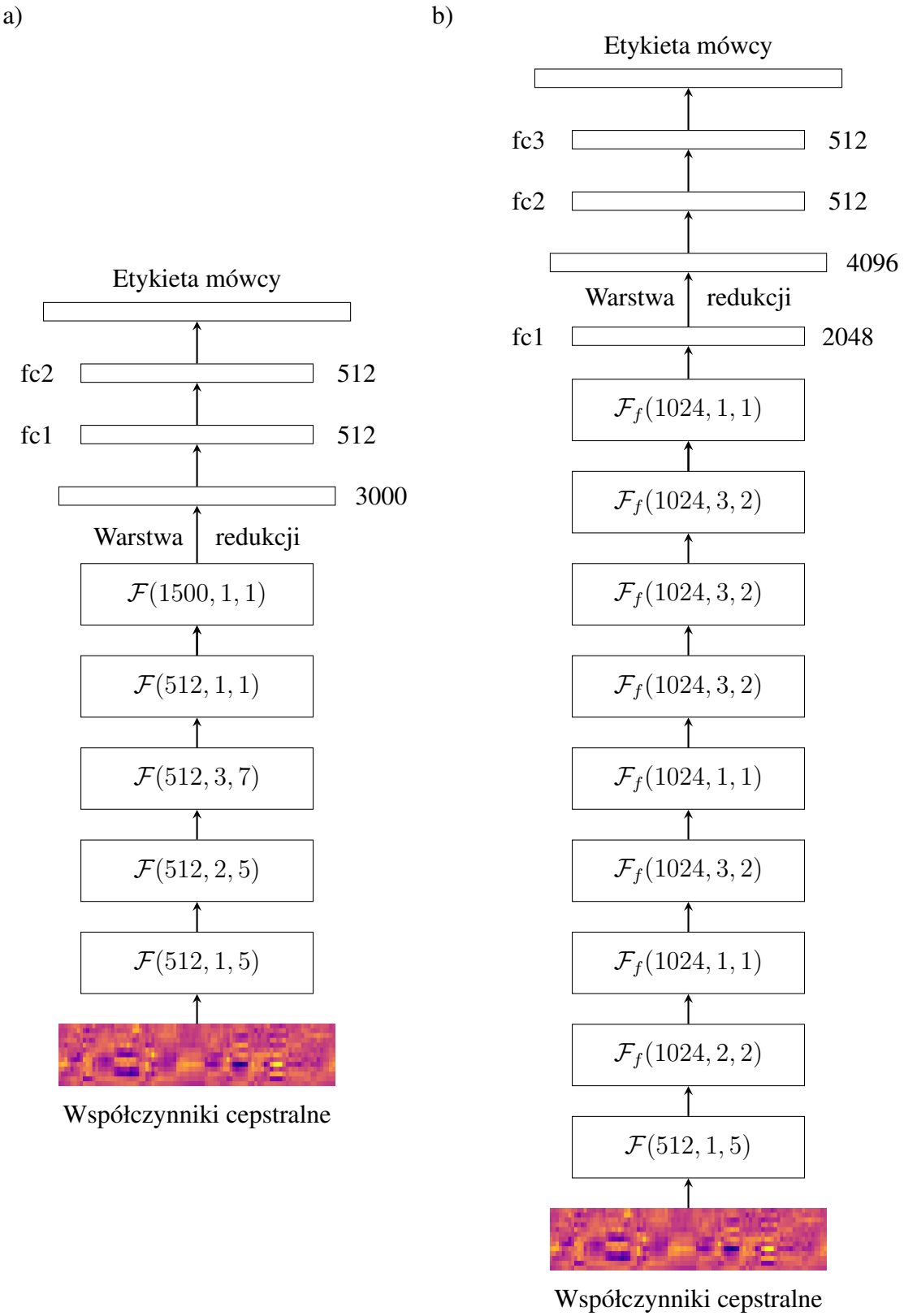
neural network, TDNN). Sieć typu TDNN umożliwia analizę danych sekwencyjnych z zachowaniem jednokierunkowego przepływu danych poprzez zastosowanie splotu na sekwencji czasowej. Dzięki zastosowaniu splotu ze stałym jądrem występuje możliwość współdzielenia wag w czasie – sekwencja wyjściowa warstwy splotowej zależna jest od określonej liczby sąsiadujących ze sobą składowych wejściowych [161].

W zmodyfikowanej wersji DNN (rys. 5.4b) użyto faktoryzowanych warstw TDNN (ang. *factorized time-delay neural network*, FTDNN). Faktoryzacja polega na przedstawieniu macierzy wag warstwy TDNN za pomocą iloczynu dwóch macierzy niższego rzędu, co skutkuje zmniejszeniem liczby parametrów sieci w stosunku do sieci, w której występuje niefaktoryzowana warstwa TDNN (rys. 5.5) [162], [163]. Zmniejszenie liczby parametrów skutkuje zaś skuteczniejszym i bardziej wydajnym uczeniem się sieci.

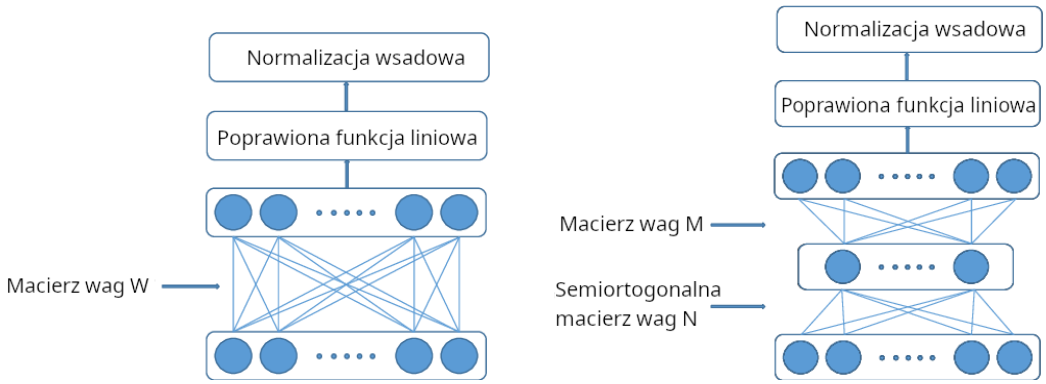
W dalszej części pracy w odniesieniu do powyższej modyfikacji DNN wykorzystanej do ekstrakcji x-wektorów oraz do wyekstrahowanego w ten sposób wariantu x-wektorów stosowane będzie określenie GFCC-FTDNN.

Ostatnią wprowadzoną zmianą była modyfikacja danych uczących, polegająca na wprowadzeniu zerowania losowego współczynnika spośród 20 współczynników GFCC opisujących każdą ramkę sygnału podawanych na wejście sieci, zaispirowana stosowaną w technikach wizyjnych metodą augmentacji danych *random erasing* [164]. Miało to na celu spowodowanie większego uogólnienia modelu – sieć ucząca się na niepełnych danych zmuszona jest uczyć się bardziej ogólnych wzorców występujących w tych danych. Ponieważ podczas każdej epoki i w każdej macierzy cech wyzerowane było inne losowe pasmo, uniknięto sytuacji, w której pojedynczy współczynnik GFCC miałby decydujący wpływ na decyzję klasyfikacyjną sieci. Podejście to jest podobne do powszechnie stosowanej techniki *dropout*, która polega na wyzerowaniu wyjść losowo wybranych neuronów w warstwie, dzięki czemu osiągana jest większa regularyzacja modelu, a pojedynczy neuron nie może mieć decydującego wpływu na wynik klasyfikacji [165]. Główna różnica pomiędzy techniką zastosowaną w niniejszej pracy a techniką *dropout* jest brak skalowania wag, które występuje w technice *dropout*.

W dalszej części pracy w odniesieniu do powyższej modyfikacji DNN wykorzystanej do ekstrakcji x-wektorów oraz do wyekstrahowanego w ten sposób wariantu x-wektorów stosowane będzie określenie GFCC-FTDNN-RC0.



Rys. 5.4. Schematy sieci neuronowych użytych do ekstrakcji x-wektorów: a) sieć pretrenowana oparta o warstwy TDNN, zaproponowana w [149], b) sieć oparta o warstwy FTDNN [162]. fc - warstwa gęsta (ang. fully connected), $\mathcal{F}(N, D, K)$ - warstwa TDNN, $\mathcal{F}_f(N, D, K)$ - warstwa FTDNN, gdzie N - długość wyjściowego tensora danych, D - dylatacja, K - szerokość okna kontekstowego.



Rys. 5.5. Porównanie warstwy TDNN (po lewej) i warstwy FTDNN (po prawej). Źródło: [162].

Opisane powyżej modyfikacje doprowadziły do otrzymania pięciu wariantów x-wektorów, które następnie wykorzystano do treningu modeli klasyfikacyjnych. W dalszej części pracy w odniesieniu do tych wariantów stosowane będą określenia MFCC, GFCC, GFCC-Arc, GFCC-FTDNN, GFCC-FTDNN-RC0.

5.2.1.3. Analiza porównawcza wyników uzyskiwanych przy użyciu zaproponowanych pięciu wariantów x-wektorów

Wartości AUC ROC uzyskane przez poszczególne modele na x-wektorach wyekstrahowanych przy użyciu wszystkich wariantów sieci zestawiono w tabeli 5.3. Jak można zauważyć, w przypadku sieci trenowanej na cechach GFCC-Arc największą wartość AUC ROC uzyskano przy użyciu modelu SGD, natomiast w przypadku pozostałych wariantów przy użyciu MLP. Wszystkie cztery warianty x-wektorów ekstrahowanych z GFCC pozwoliły na uzyskanie większych wartości, niż w przypadku użycia MFCC, jednak różnice pomiędzy nimi nie są zbyt wielkie – największą różnicę można zaobserwować pomiędzy wariantami opartymi o MFCC oraz GFCC-FTDNN-RC0 i wynosi ona 0,0283. Istotność statystyczną tych różnic zbadano za pomocą testu DeLonga opisanego w rozdziale 2, sekcji 2.5.2, wykorzystując funkcję zaimplementowaną w pakiecie Daim w języku R [166]. W tym celu sformułowano następujące hipotezy:

H_0: wartości AUC ROC uzyskane przez wszystkie porównywane modele są takie same

H_1: wartość AUC ROC uzyskana przez co najmniej jeden model jest istotnie różna od pozostałych

Test przeprowadzono przyjmując poziom istotności α = 0,05. Uzyskana wartość p wyniosła mniej niż $0,001$, co oznacza, że istnieją podstawy do odrzucenia hipotezy zerowej i przyjęcia hipotezy alternatywnej – przynajmniej jeden z modeli uzyskał AUC ROC istotnie różne od pozostałych.

W celu sprawdzenia, pomiędzy którymi modelami występują istotne różnice przeprowadzono test istotności różnic dla każdej pary modeli – w tym celu wyznaczono przedziały ufności

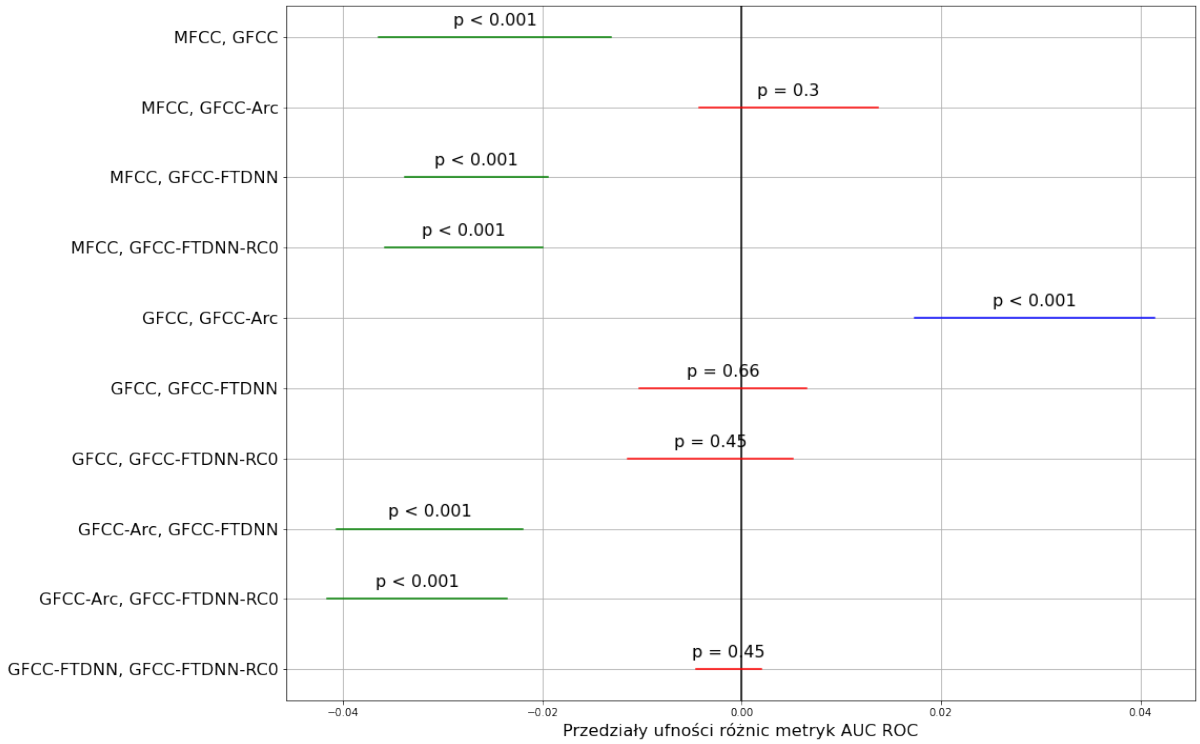
Tabela 5.3. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji osób chorych otolaryngologicznie.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.8145	0.8250	0.8395	0.8614	0.8747
Extra Trees	0.7475	0.8043	0.7989	0.8226	0.8668
KNN	0.8162	0.8588	0.8447	0.8522	0.8724
LDA	0.7494	0.7832	0.7162	0.7950	0.7910
LGB	0.7671	0.8397	0.8063	0.8149	0.8472
LR	0.7497	0.7950	0.7497	0.7950	0.8049
MLP	**0.8494**	**0.8748**	0.8421	**0.8759**	**0.8777**
NuSVM	0.8406	0.8619	0.8409	0.8572	0.8634
QDA	0.7026	0.7890	0.7575	0.7713	0.8029
RF	0.6825	0.7892	0.8031	0.8306	0.8521
RNN	0.7260	0.8284	0.8450	0.8331	0.8447
SGD	0.8389	0.8717	**0.8451**	0.8705	0.8676

różnic oraz wartości p dla testu DeLonga. Należy podkreślić, że test DeLonga zwraca skorygowane wartości p, uwzględniając fakt, że dokonywane są porównania wielokrotne. Uzyskane wyniki przedstawiono na rys. 5.6.

Istotne statystycznie różnice wykazano dla sześciu par. Trzy warianty x-wektorów – ekstrahowane z GFCC, GFCC-FTDNN oraz GFCC-FTDNN-RC0 – pozwalają uzyskać istotnie większą wartość AUC ROC niż x-wektory ekstrahowane z MFCC oraz GFCC-GFCC-Arc, zaś pomiędzy wariantami MFCC i GFCC-Arc nie wykazano istotnych różnic. Istotne różnice nie występują również pomiędzy GFCC, GFCC-FTDNN oraz GFCC-FTDNN-RC0, co potwierdzono wykonując kolejny test DeLonga, tym razem jedynie dla tych trzech modeli – uzyskano p = 0,6108, co świadczy o tym, że nie ma podstaw do odrzucenia hipotezy zerowej o równości wszystkich porównywanych AUC ROC. W celu wybrania modelu uzyskującego najlepsze rezultaty było więc konieczne przeprowadzenie dalszych analiz porównawczych.

Kolejnym kryterium pozwalającym na wybór najlepszego modelu było porównanie uzyskiwanych przez badane modele strat logarytmicznych, podobnie jak miało to miejsce w przypadku przedstawionego algorytmu porównywania dwóch modeli (rys. 5.2). W tym celu wygenerowano 500 prób bootstrapowych oraz wyliczono dla nich wartości strat logarytmicznych, zgodnie z metodą przedstawioną w [167]. Posłużyły one do porównania rozkładów strat logarytmicznych uzyskiwanych przez poszczególne modele – równość uzyskiwanych wyników należało sprawdzić za pomocą testu nieparametrycznego, ponieważ nie wszystkie próby miały rozkład normalny (dodatek B). Należy jednak zauważyć, że w cytowanej publikacji porównywano tylko dwa modele i skorzystano w tym celu z testu par Wilcoxona, który nie może być użyty do sprawdzenia równości strat logarytmicznych większej liczby modeli. W podobnych



Rys. 5.6. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji osób chorych otolaryngologicznie. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedział wskazujący na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

przypadkach opisywanych w literaturze [168]–[171] do porównania wielu prób wykorzystywany jest test Friedmana [172], [173], zdecydowano się więc na jego użycie, formułując następujące hipotezy:

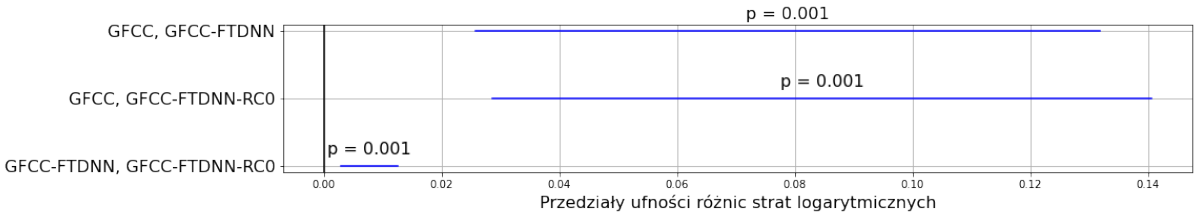
H_0: straty logarytmiczne uzyskane przez wszystkie porównywane modele pochodzą z tego samego rozkładu

H_1: straty logarytmiczne uzyskane przez wszystkie porównywane modele nie pochodzą z tego samego rozkładu

Test przeprowadzono przy użyciu funkcji zaimplementowanej w bibliotece scipy [174], przyjmując poziom istotności α = 0,05. Uzyskana wartość $p < 0,001$ jest mniejsza od przyjętego poziomu istotności, a więc są podstawy do odrzucenia hipotezy zerowej i przyjęcia hipotezy alternatywnej mówiącej o tym, że występują istotne różnice pomiędzy stratami logarytmicznymi uzyskanymi przez porównywane modele.

W celu identyfikacji różniących się modeli przeprowadzono test *post hoc* Nemenyiego [175], [176], wykorzystując funkcję zaimplementowaną w bibliotece *scipy-posthocs* [177]. Uzyskane wartości p oraz przedziały ufności różnic strat logarytmicznych przedstawiono na rys. 5.7. Analiza tych wyników pozwala stwierdzić, że strata logarytmiczna modelu trenowanego na wariancie GFCC jest istotnie większa od pozostałych dwóch modeli, co świadczy o jego

gorszych zdolnościach generalizacyjnych. Pomiędzy modelami trenowanymi na wariantach GFCC-FTDNN oraz GFCC-FTDNN-RC0 również stwierdzono istotne statystycznie różnice. Dodatnie wartości w przedziale ufności różnic wskazują na mniejsze straty logarytmiczne uzyskiwane przez model trenowany na wariancie GFCC-FTDNN-RC0. Oznacza to, że usunięcie losowego współczynnika GFCC podczas treningu DNN ekstrahującej x-wektory pozytywnie wpłynęło na zdolności generalizacyjne modelu MLP wykorzystywanego do detekcji schorzeń krtani.



Rys. 5.7. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji osób chorych otolaryngologicznie, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC.

Dotychczas przeprowadzane analizy dotyczyły diagnostyki dowolnej choroby krtani. W bazie SVD można jednak wyróżnić grupę pięciu chorób o stosunkowo dużej liczebności: dysfonię, ostre zapalenie krtani, porażenie nerwu krtaniowego wstecznego, obrzęk Reinkego oraz pachydermię kontaktową (tabela 4.1). Choroby te cechują się różną patogenezą oraz symptomami, co może wpływać na dobór cech i modelu zapewniających maksymalizację skuteczności ich detekcji. Postanowiono więc przeprowadzić analizy analogiczne do opisanych powyżej w celu zwiększenia poprawności klasyfikacji oraz sprawdzenia, czy rodzaj użytych cech zapewniających najlepsze rezultaty będzie zależny od choroby. Liczebność zbiorów uczących i testowych zestawiono w tabeli 5.4.

5.2.1.3.1 Detekcja dysfonii

U części pacjentów ze zdiagnozowaną dysfonią nie został określony jej typ. U pozostałych pacjentów diagnoza była bardziej szczegółowa – lekarze sprecyzowali, czy pacjent cierpi na dysfonię czynnościową, psychogenną czy spastyczną. Co więcej, w przypadku części pacjentów ze zdiagnozowaną dysfonią czynnościową doprecyzowano jej rodzaj zależny od tego, czy dysfonia związana jest z nadmiernym napięciem mięśni zewnętrznych i wewnętrznych krtani (dysfonia hiperfunkcjonalna), czy z ich osłabieniem i – w konsekwencji – niedomykalnością fałdów głosowych (dysfonia hipofunkcjonalna). W związku z tym uznano, że najlepszym wyborem będzie trening modeli na wszystkich dostępnych sygnałach zawierających dowolny typ dysfonii, a następnie trening osobnych modeli dla dysfonii podzielonych ze względu na charakter zaburzenia, czyli dysfonii psychogennej oraz dysfonii czynnościowej. W celu zwiększenia liczby analizowanych sygnałów, do zbioru zawierającego pacjentów z dysfonią czynnościową zaliczono trzy grupy pacjentów: z dysfonią czynnościową bez określonego rodzaju oraz z dysfonią hiper- i hipofunkcjonalną (tabela 5.5). Postanowiono również przeprowadzić osobne analizy

Tabela 5.4. Liczbeność zbiorów uczącego i testowego użytych podczas treningu i ewaluacji modeli klasyfikacyjnych wykorzystywanych do detekcji poszczególnych schorzeń otolaryngologicznych.

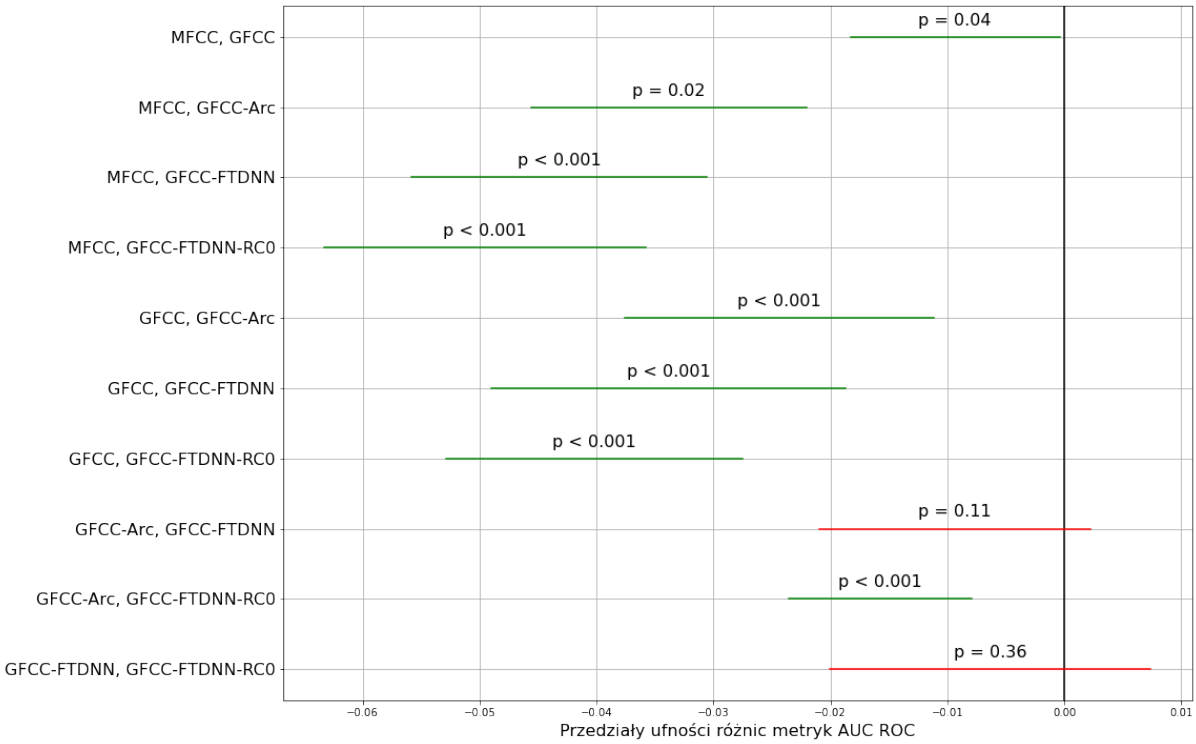
Nazwa choroby	Zbiór	Liczba nagrań
Dysfonia	uczący	430
	testowy	116
Obrzęk Reinkego	uczący	31
	testowy	8
Ostre zapalenie krtani	uczący	98
	testowy	25
Pachydermia kontaktowa	uczący	39
	testowy	10
Porażenie nerwu krtaniowego wstecznego	uczący	160
	testowy	40
Zdrowi	uczący	507
	testowy	127

dla dysfonii spastycznej – określenie to jest stosowane w stosunku do dysfonii hiperfunkcjonalnej [178] lub do ciężkiej postaci przewlekłych zaburzeń psychogennych [179]. Nie wiadomo, które ze znaczeń zostało przyjęte przez twórców bazy SVD, nie można więc było przypisać pacjentów do żadnej z grup. Nie podjęto się natomiast przeprowadzenia osobnych analiz dla dwóch rodzajów dysfonii czynnościowej, ponieważ liczba zdiagnozowanych przypadków dysfonii hipofunkcjonalnej zawartych w bazie SVD jest jednak zbyt mała – podział 16 sygnałów należących do tej grupy na zbiór uczący i testowy skutkowałby uzyskaniem zbioru testowego zawierającego zaledwie 3 sygnały osób chorych i jakiekolwiek próby uogólnienia uzyskanych rezultatów na całą populację byłyby pozbawione podstaw.

Analizy danych dotyczących pacjentów z dysfonią rozpoczęto od grupy bez podziału na typy schorzenia. Wartości metryki AUC ROC uzyskane na zbiorze testowym przez poszczególne modele uczone na pięciu wariantach x-wektorów zestawiono w tabeli 5.6. Jak można zauważyć, w przypadku wariantów opartych o MFCC i GFCC-Arc najwyższe wartości uzyskano przy użyciu modelu SGD, natomiast w przypadku wariantów opartych o GFCC, GFCC-FTDNN oraz GFCC-FTDNN-RC0 przy użyciu odpowiednio modeli MLP, AdaBoost oraz NuSVM. Widoczna jest dużo większa różnorodność modeli wykazujących najlepsze działanie – we wcześniej analizowanym przypadku dla aż czterech z pięciu wariantów x-wektorów najlepszym modelem okazał się być MLP. Również tym razem okazało się, że użycie MFCC do ekstrakcji x-wektorów prowadzi do uzyskania najgorszych rezultatów, zaś największą wartość AUC ROC uzyskano dla cech GFCC-FTDNN-RC0. Różnica pomiędzy wartościami uzyskanymi dla MFCC i GFCC-FTDNN-RC0 jest nieco większa w przypadku dysfonii, niż dowolnego schorzenia krtani i wynosi 0,0496.

Tabela 5.5. Liczbeność zbiorów uczącego i testowego użytych podczas treningu i ewaluacji modeli klasyfikacyjnych wykorzystywanych do detekcji poszczególnych typów dysfonii.

Typ dysfonii	Zbiór	Liczba nagrań
Czynnościowa	uczący	267
	testowy	71
Psychogenna	uczący	58
	testowy	18
Spastyczna	uczący	52
	testowy	13
Zdrowi	uczący	507
	testowy	127



Rys. 5.8. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji dysfonii. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.

Wartość p uzyskana w teście równości wszystkich pięciu uzyskanych przez najlepsze modele wartości AUC ROC wyniosła mniej niż $0,001$, co oznacza, że występują pomiędzy nimi istotne statystycznie różnice. Wartości p uzyskane w teście DeLonga dla poszczególnych par cech przedstawiono na rys. 5.8. Tylko dla dwóch par – GFCC-Arc i GFCC-FTDNN oraz GFCC-FTDNN i GFCC-FTDNN-RC0 uzyskano p większe od przyjętego poziomu istotności i nie stwierdzono istotnych statystycznie różnic. Równocześnie można stwierdzić, że cechy te

Tabela 5.6. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji dysfonii.

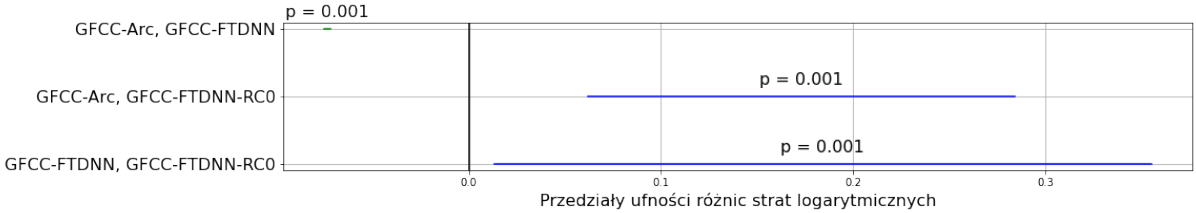
	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.7432	0.7897	0.8468	**0.8563**	0.8140
Extra Trees	0.7012	0.7212	0.7583	0.7260	0.8125
KNN	0.7838	0.8002	0.8184	0.7986	0.8442
LDA	0.6873	0.6776	0.7683	0.7392	0.7515
LGB	0.6876	0.7615	0.8292	0.7799	0.8445
LR	0.7024	0.7472	0.7569	0.7507	0.7565
MLP	0.7794	**0.8225**	0.8434	0.8370	0.8598
NuSVM	0.7564	0.8135	0.8333	0.8315	**0.8627**
QDA	0.6969	0.7659	0.7593	0.7389	0.7655
RF	0.6420	0.6641	0.7174	0.7055	0.7643
RNN	0.6175	0.5000	0.7961	0.6719	0.8071
SGD	**0.8131**	0.8074	**0.8469**	0.8385	0.8545

dają istotnie większe wartości AUC ROC niż MFCC i GFCC. Postanowiono więc przeprowadzić test równości strat logarytmicznych dla tych trzech modeli i na tej podstawie wybrać najlepszy z nich.

Tylko dwie z trzech porównywanych prób bootstrapowych mają rozkład normalny (Dodatek C), więc również tym razem do ich porównania wykorzystano test Friedmana. W jego wyniku uzyskano wartość $p < 0,001$, co świadczy o podstawach do odrzucenia hipotezy zerowej mówiącej o równości rozkładów opisujących badane próby. Na podstawie wyników testu *post hoc* Nemenyiego przedstawionych na rys. 5.9 można stwierdzić, że pomiędzy wszystkimi trzema próbami występują istotne statystycznie różnice. Najmniejszą stratę logarytmiczną uzyskano dla modelu trenowanego na wariancie GFCC-FTDNN-RC0, o czym świadczą dodatnie wartości w przedziałach ufności różnic pomiędzy GFCC-FTDNN i GFCC-FTDNN-RC0 oraz GFCC-Arc i GFCC-FTDNN-RC0. Zgodnie z przyjętymi kryteriami, model NuSVM trenowany na GFCC-FTDNN-RC0 może zostać uznany za najlepszy w rozwiązywaniu problemu klasyfikacji osób cierpiących na dysfonię bez określonego typu.

W kolejnym kroku analogiczne analizy przeprowadzono dla trzech typów dysfonii: czynnościowej, psychogennej i spastycznej. Wartości metryki AUC ROC uzyskane dla modeli trenowanych na rozważanych pięciu wariantach x-wektorów zestawiono odpowiednio w tabelach 5.7-5.9. W przypadku wszystkich trzech typów choroby można zauważyć wyraźną przewagę modeli trenowanych na czterech wariantach x-wektorów wykorzystujących GFCC nad modelami trenowanymi na wariancie opartym o MFCC. Szczególnie duże różnice można zauważyć

w przypadku dysfonii spastycznej – MFCC pozwoliło na uzyskanie AUC ROC = 0,6172, podczas gdy pozostałe cztery warianty na wartości o ponad 0,35 większe. Różnice pomiędzy pozostałymi czterema wariantami są niewielkie i wynoszą maksymalnie 0,0055. Dla wariantów GFCC-FTDNN i GFCC-FTDNN-RC0 uzyskano takie same wartości AUC ROC wynoszące 0.9988.



Rys. 5.9. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji dysfonii, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.

Tabela 5.7. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji dysfonii funkcjonalnej.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.5255	0.8703	0.8542	0.8331	**0.8742**
Extra Trees	0.4980	0.7881	0.7703	0.8350	0.7679
KNN	0.5034	0.7693	0.8795	0.8194	0.8441
LDA	0.5000	0.7590	0.6542	0.7313	0.7412
LGB	0.5122	0.8490	0.7467	0.6874	0.7862
LR	0.4921	0.5000	0.6464	0.6522	0.6799
MLP	**0.5786**	**0.8819**	**0.8685**	0.8552	0.8477
NuSVM	0.5211	0.8661	0.8442	**0.8744**	0.8497
QDA	0.5000	0.5527	0.5000	0.5000	0.5000
RF	0.4497	0.8146	0.7299	0.8525	0.7713
RNN	0.5000	0.5000	0.6782	0.7624	0.5000
SGD	0.5132	0.8340	0.8528	0.8442	0.8511

Tabela 5.8. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji dysfonii psychogennej.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	**0.6083**	**0.8832**	0.8780	0.8712	0.8843
Extra Trees	0.5527	0.7314	**0.8998**	0.8495	0.8644
KNN	0.3878	0.7375	0.8723	0.8169	0.8207
LDA	0.5000	0.5637	0.6509	0.5516	0.7384
LGB	0.5534	0.6214	0.8075	0.7721	0.8528
LR	0.5000	0.5914	0.5120	0.6111	0.7025
MLP	0.5420	0.8828	0.8456	0.8745	0.9204
NuSVM	0.5088	0.8132	0.8871	0.8574	0.8513
QDA	0.5000	0.5000	0.5000	0.7544	0.7741
RF	0.4646	0.7458	0.8294	0.7826	0.8476
RNN	0.5000	0.8047	0.5000	0.5000	0.5000
SGD	0.5000	0.8644	0.8758	**0.8837**	**0.9261**

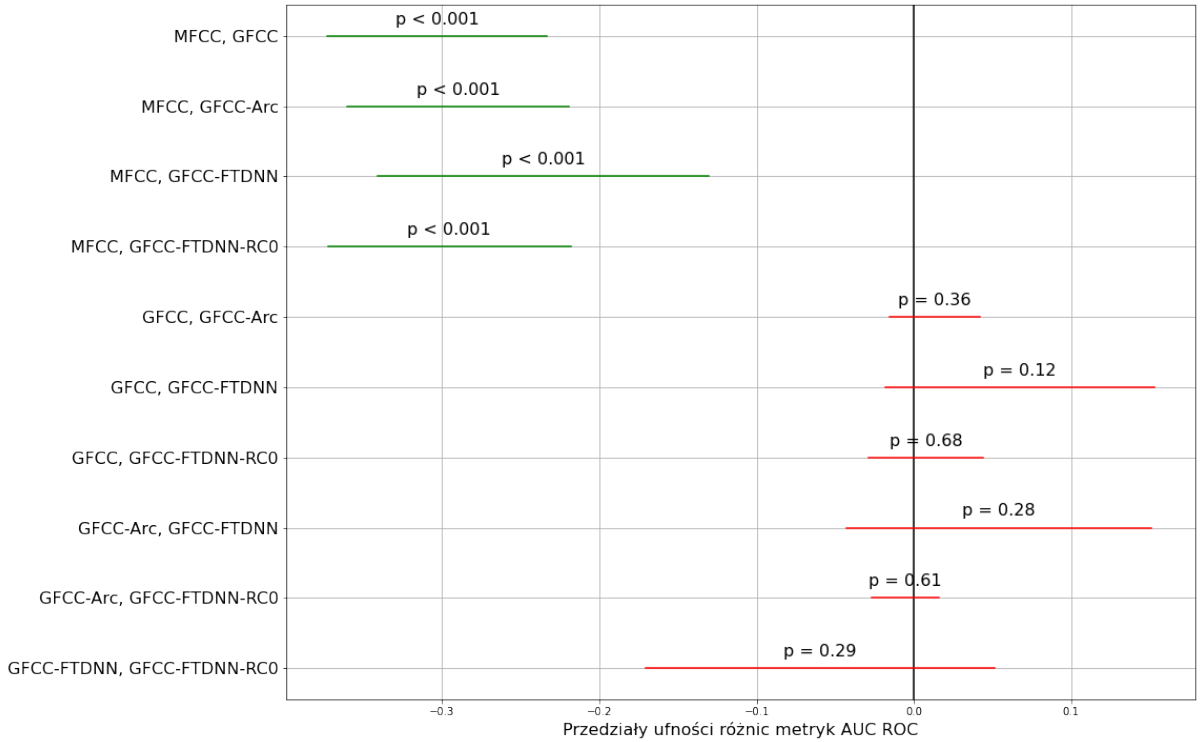
Tabela 5.9. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji dysfonii spastycznej.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.5145	**0.9981**	0.9685	**0.9988**	0.9933
Extra Trees	0.6045	0.9706	0.9924	0.9873	0.9964
KNN	0.4782	0.9746	0.9712	0.9652	0.9588
LDA	0.5000	0.9191	0.9113	0.9152	0.9231
LGB	0.4073	0.9946	0.9673	0.9921	0.9691
LR	0.5000	0.9615	0.9113	0.9152	0.9152
MLP	**0.6172**	0.9946	**0.9933**	0.9970	0.9976
NuSVM	0.5906	0.9879	0.9897	0.9952	**0.9988**
QDA	0.4764	0.8422	0.8422	0.9152	0.9576
RF	0.5415	0.9426	0.9779	0.9909	0.9873
RNN	0.5000	0.7117	0.7798	0.5000	0.5000
SGD	0.5000	0.9945	0.9764	0.9915	0.9891

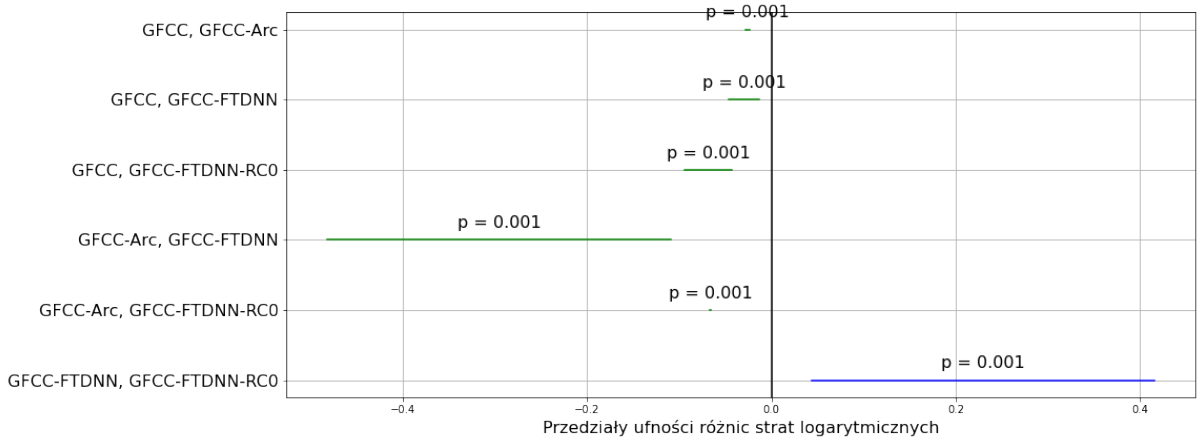
W przypadku dysfonii psychogennej tylko dla jednego wariantu x-wektorów – wariantu opartego o cechy GFCC-FTDNN-RC0 – uzyskano AUC ROC przekraczające 0,9. Pozostałe trzy warianty wykorzystujące GFCC dały nieco gorsze wyniki, jednak również w ich przypadku wartości AUC ROC można uznać za dobre – wszystkie przekraczają 0,88 i są lepsze od AUC ROC uzyskanego na wariancie wykorzystującym MFCC o co najmniej 0,2749.

W przypadku dysfonii czynnościowej żaden z wariantów nie pozwolił na uzyskanie AUC ROC przekraczającego 0,9 – największą wartość uzyskano dla wariantu GFCC i wynosi ona 0,8819. Warto zauważyć, że jest to pierwszy z analizowanych w pracy przypadków, w którym dla GFCC-FTDNN-RC0 nie uzyskano największej wartości tej metryki. Różnica pomiędzy wartością uzyskaną dla wariantu MFCC i wariantu GFCC wynosi 0,3033, zaś pomiędzy wartością uzyskaną dla wariantu MFCC i wariantu GFCC-GFCC-Arc, dającego najgorsze rezultaty spośród czterech wariantów bazujących na GFCC, różnica wynosi 0,2899. Można więc uznać, że użycie wariantu opartego o MFCC daje rezultaty wyraźnie gorsze. Jednak by jednoznacznie stwierdzić, że różnice są istotne statystycznie, konieczne jest jednak przeprowadzenie odpowiedniego testu statystycznego.

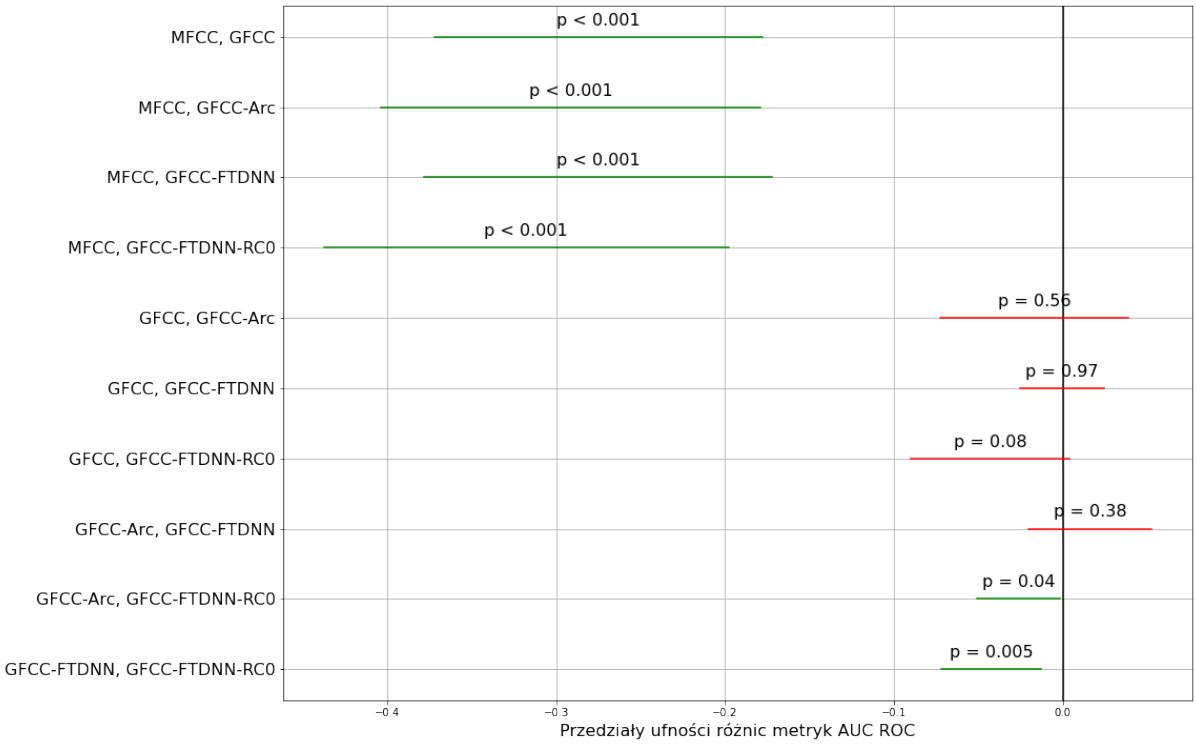
Wartości p uzyskane w testach DeLonga przeprowadzonych dla analizowanych trzech typów dysfonii wskazują na konieczność odrzucenia hipotezy o równości analizowanych wartości metryki AUC ROC - wartości p wyniosły mniej niż $0,001$ w przypadku wszystkich trzech typów schorzenia, czyli dysfonii czynnościowej, psychogennej i spastycznej. Wyniki testu dla poszczególnych par wariantów x-wektorów przedstawiono na rysunkach 5.10, 5.12, 5.14. W przypadku dysfonii czynnościowej i spastycznej potwierdziły się przypuszczenia dotyczące istotności różnic pomiędzy wartościami AUC ROC uzyskanymi dla różnych wariantów x-wektorów – wariant oparty o MFCC uzyskał rezultaty istotnie gorsze. W przypadku pozostałych czterech wariantów różnice są na tyle niewielkie, że nie wykazano ich istotności statystycznej. W przypadku dysfonii psychogennej również uzyskano wyniki wskazujące na istotnie mniejsze wartości AUC ROC uzyskane dla wariantu opartego o MFCC. Ponadto istotne różnice zaobserwowano dla dwóch innych par: wariantów GFCC-Arc i GFCC-FTDNN-RC0 oraz wariantów GFCC-FTDNN i GFCC-FTDNN-RC0. W obu przypadkach uzyskano przedziały ufności różnic zawierające jedynie wartości ujemne, co wskazuje na to, że wariant GFCC-FTDNN-RC0 daje lepsze rezultaty niż pozostałe dwa warianty – podobne wnioski można wyciągnąć na podstawie analizy wartości AUC ROC przedstawionych w tabeli 5.8. Warto również zauważyć, że dla wszystkich trzech typów dysfonii wartości AUC ROC uzyskane dla wariantu MFCC nie różnią się od wartości uzyskiwanych przez model zwracający losowe predykcje, co nie miało miejsca w żadnym innym przypadku rozważanym w niniejszej pracy – świadczące o tym wartości p uzyskane przy użyciu testu DeLonga przedstawiono w dodatku B.



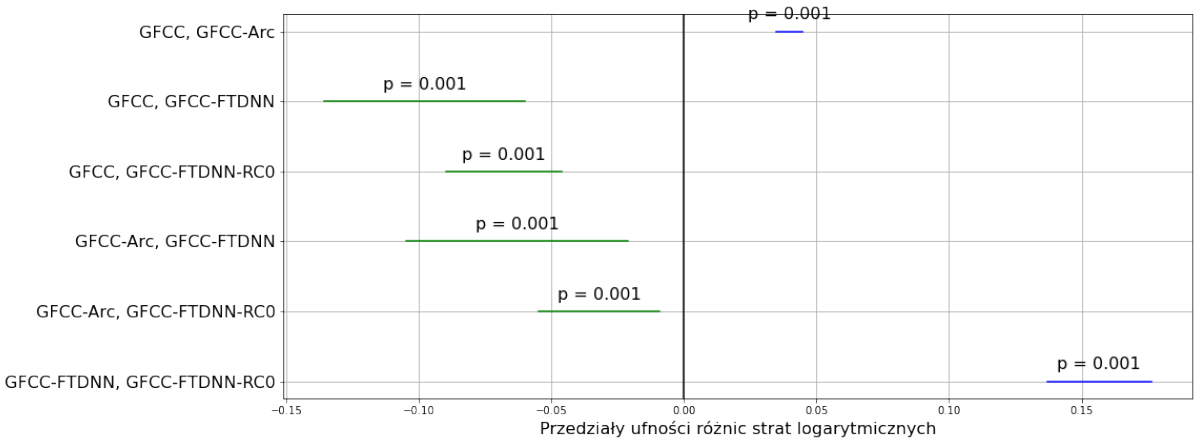
Rys. 5.10. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji dysfonii funkcjonalnej. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.



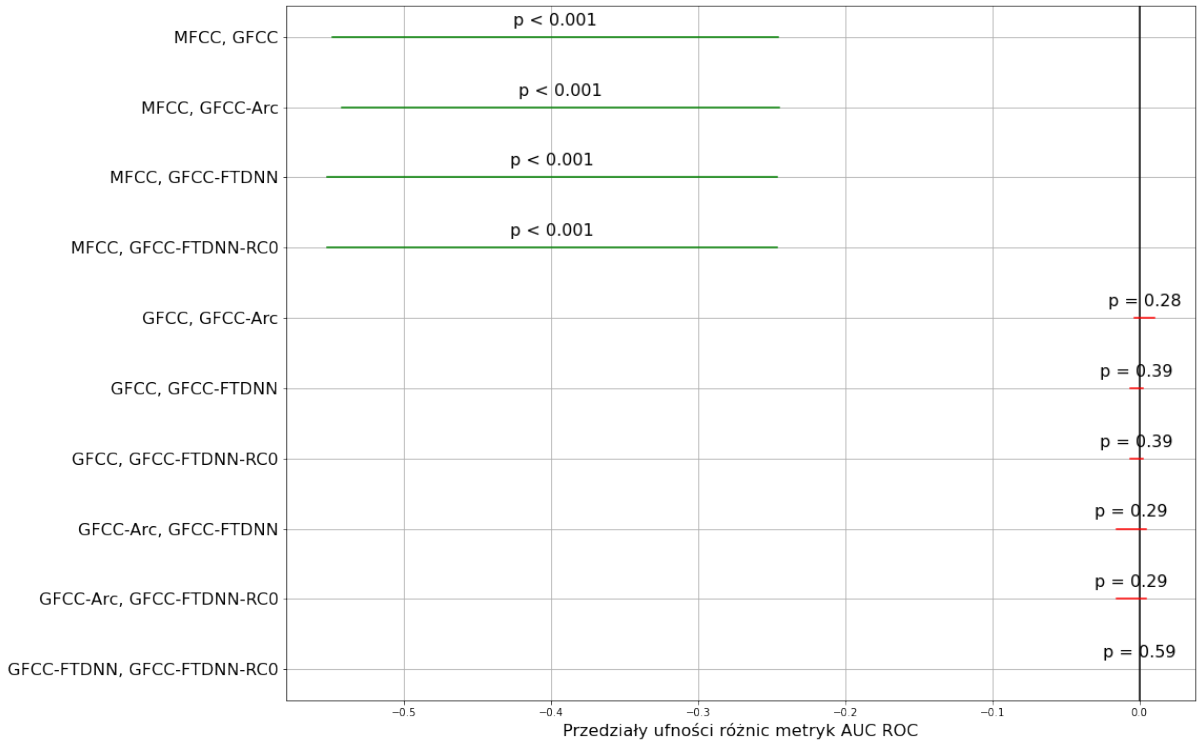
Rys. 5.11. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji dysfonii czynnościowej, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedział wskazujący na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.



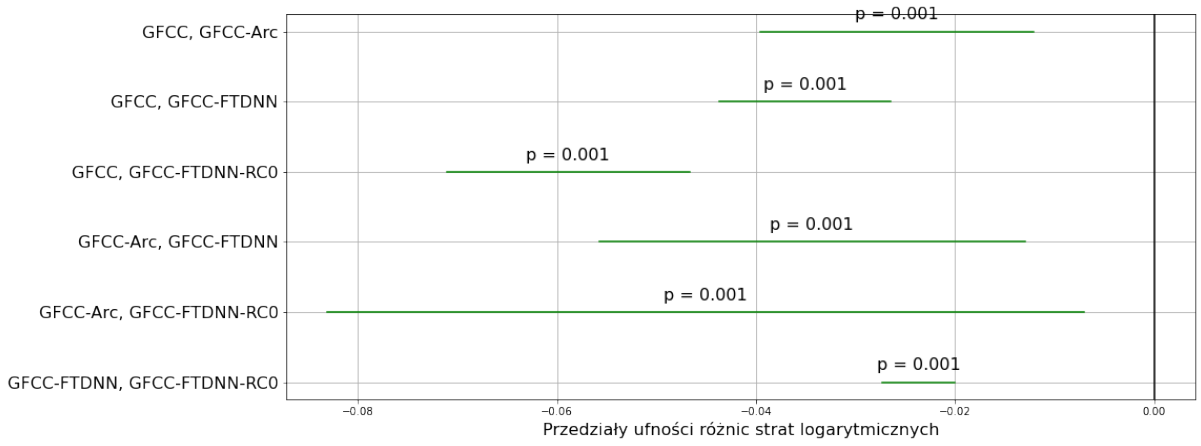
Rys. 5.12. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji dysfonii psychogennej. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.



Rys. 5.13. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji dysfonii psychogennej, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.



Rys. 5.14. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji dysfonii spastycznej. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.



Rys. 5.15. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji dysfonii spastycznej, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC.

Ponieważ analiza AUC ROC nie pozwoliła na wyłonienie wariantu x-wektorów dające najlepsze rezultaty, wykonano dalsze testy. W przypadku dysfonii czynnościowej i spastycznej konieczne było przeprowadzenie testów dla czterech wariantów bazujących na GFCC, natomiast dla dysfonii psychogennej dla trzech wariantów. Podobnie jak w poprzednich przypadkach, przeprowadzono test Friedmana, ponieważ nie wszystkie porównywane próby bootstrapowe miały rozkład normalnych. Uzyskane wartości p wskazują na to, że istnieją podstawy do stwierdzenia, że nie wszystkie porównywane warianty x-wektorów pozwalają na uzyskanie takich samych strat logarytmicznych – we wszystkich trzech przypadkach wartość p wyniosła mniej niż $0,001$.

Wyniki testu *post hoc* Nemenyiego dla trzech typów dysfonii przeprowadzonych w celu identyfikacji różniących się prób bootstrapowych przedstawiono na rysunkach 5.11, 5.13, 5.15. Jak można zauważyć, istotne statystycznie różnice występują pomiędzy wszystkimi porównywanymi próbami. Analiza uzyskanych przedziałów ufności różnic pozwala stwierdzić, że w przypadku dysfonii czynnościowej najmniejszą stratę logarytmiczną uzyskano dla wariantu opartego o GFCC – wszystkie cztery przedziały ufności uzyskane dla par, w których występuje wariant GFCC zawierają wyłącznie wartości ujemne. Ponieważ różnice wyznaczono w taki sposób, że od wyników uzyskanych dla wariantu GFCC odejmowano wyniki uzyskane dla drugiego wariantu w parze, ujemne wartości oznaczają, że wariant GFCC zapewniał mniejsze wartości strat logarytmicznych. Podobnie sytuacja wygląda w przypadku dysfonii spastycznej – również w tym przypadku najmniejszą stratę logarytmiczną uzyskano dla wariantu opartego o GFCC. W przypadku dysfonii psychogennej najmniejsze wartości straty logarytmicznej uzyskano natomiast dla wariantu GFCC-Arc.

Zgodnie z przyjętymi kryteriami wyboru najlepszych modeli i wariantów, za klasyfikatory dające najlepsze rezultaty w klasyfikacji trzech typów dysfonii należy więc uznać MLP trenowany na wariancie GFCC w przypadku dysfonii czynnościowej, Extra Trees trenowany na wariancie GFCC-Arc w przypadku dysfonii psychogennej oraz AdaBoost trenowany wariancie GFCC w przypadku dysfonii spastycznej.

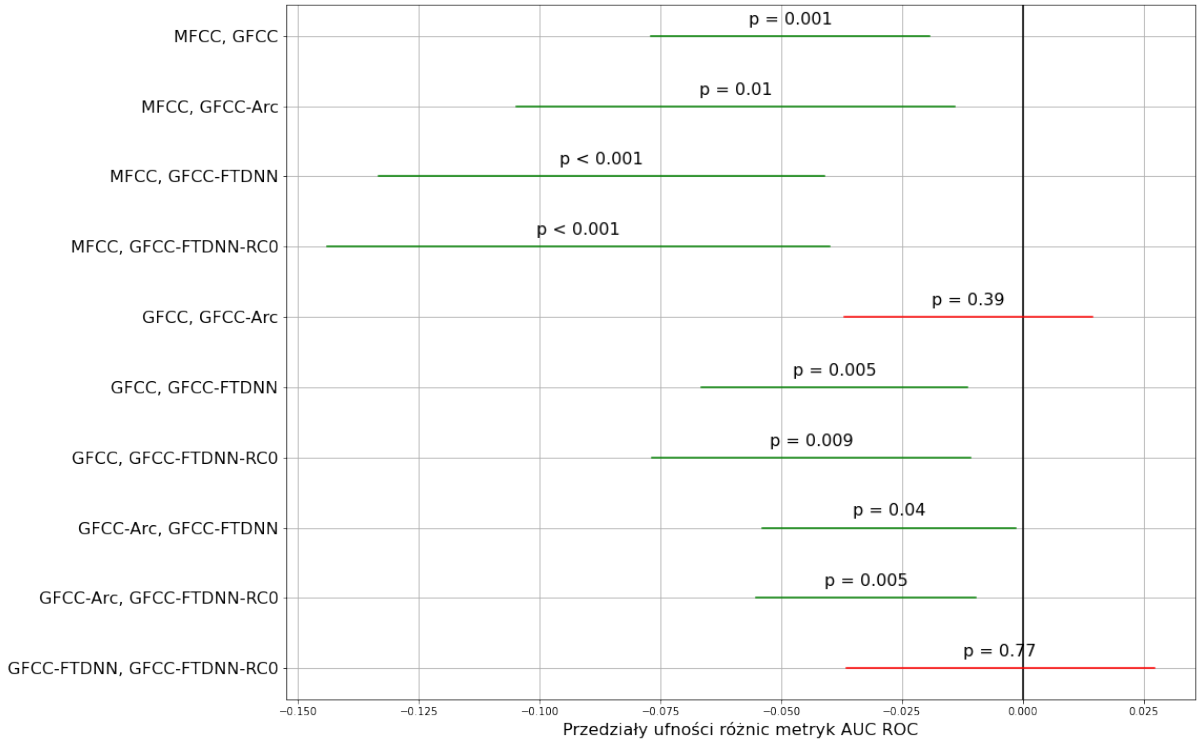
5.2.1.3.2 Detekcja ostrego zapalenia krtani

Kolejną chorobą, dla której przeprowadzono analizy jest ostre zapalenie krtani. Analizując uzyskane przez poszczególne modele wartości AUC ROC (tabela 5.10) można zauważyć, że również tym razem wariant MFCC dał najgorsze rezultaty. Są one jednak dużo lepsze niż w przypadku klasyfikacji typów dysfonii oraz zbliżone do wartości uzyskiwanych w klasyfikacji niesprecyzowanych schorzeń krtani oraz dysfonii bez określonego typu – AUC ROC wynosi 0,8173. Pozostałe cztery warianty pozwoliły na uzyskanie AUC ROC przekraczającego 0,86: najmniejsza z tych wartości uzyskana została dla wariantu GFCC i wynosi 0,8655, zaś największa dla wariantu GFCC-FTDNN i wynosi 0,9046. Również użycie wariantu GFCC-FTDNN-RC0 skutkowało osiągnięciem AUC ROC większego niż 0,9.

Przeprowadzony test DeLonga wskazuje na występowanie istotnych statystycznie różnic pomiędzy porównywanymi wartościami - uzyskana wartość p = $0,004$. Analizując wartości p

Tabela 5.10. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji ostrego zapalenia krtani.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.7767	0.8288	0.8299	**0.9046**	0.8649
Extra Trees	0.7490	0.7512	0.7658	0.8139	0.8608
KNN	0.7466	0.7954	0.8737	0.8427	0.8913
LDA	0.6961	0.7443	0.7003	0.7282	0.7882
LGB	0.7761	0.8142	0.8624	0.7365	0.8433
LR	0.7321	0.7031	0.6164	0.6561	0.7561
MLP	0.8035	0.8189	0.8551	0.8904	**0.9093**
NuSVM	0.7735	0.8139	0.7776	0.8806	0.8772
QDA	0.5600	0.5000	0.6521	0.7495	0.8052
RF	0.7334	0.8280	0.7090	0.8602	0.8194
RNN	0.7381	0.7934	**0.8769**	0.8309	0.8789
SGD	**0.8173**	**0.8655**	0.8614	0.8787	0.8989



Rys. 5.16. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji ostrego zapalenia krtani. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.

uzyskane dla poszczególnych par wariantów (rys. 5.16) można zauważyć, że brak istotnych różnic występuje tylko dla dwóch par: wariantów GFCC i GFCC-Arc oraz GFCC-FTDNN i GFCC-FTDNN-RC0. Dalsze analizy porównujące straty logarytmiczne postanowiono jednak przeprowadzić tylko dla drugiej z par. Zarówno wariant GFCC, jak i wariant GFCC-Arc prowadzą do uzyskania istotnie mniejszych wartości AUC ROC niż warianty GFCC-FTDNN i GFCC-FTDNN-RC0. W związku z tym nie tylko nie ma konieczności uwzględniania ich w kolejnym teście, ale jest to wręcz niewskazane – wraz ze wzrostem liczby prób, które analizowane są podczas porównań wielokrotnych maleje moc testu, co jest zjawiskiem niekorzystnym. Należy więc w miarę możliwości porównywać jak najmniejszą liczbę prób [180].

Do zbadania równości strat logarytmicznych uzyskiwanych przy użyciu obu wariantów x-wektorów wykorzystano tym razem test par Wilcoxona, który dostosowany jest do porównywania dwóch prób. Zdecydowano się na test Wilcoxona, a nie test t-Studenta dla danych zależnych ze względu na fakt, że obie próby bootstrapowe nie mają rozkładu normalnego (dodatek C). Hipotezy zostały sformułowane w sposób analogiczny do hipotez w teście Friedmana:

H_0: straty logarytmiczne uzyskane przez oba porównywane modele pochodzą z tego samego rozkładu
H_1: straty logarytmiczne uzyskane przez oba porównywane modele nie pochodzą z tego samego rozkładu

Uzyskano wartość $p < 0,001$, a więc na przyjętym poziomie istotności $\alpha = 0,05$ istnieją podstawy do odrzucenia hipotezy zerowej i przyjęcia hipotezy alternatywnej mówiącej o braku równości rozkładów strat logarytmicznych wyznaczonych dla obu wariantów. Do określenia, który wariant daje lepsze rezultaty posłużył wyznaczony przedział ufności różnic: [0,0664; 0,0781]. Wartości dodatnie wskazują na większe wartości strat logarytmicznych uzyskanych dla wariantu GFCC-FTDNN. Na tej podstawie można stwierdzić, że najlepszym z porównywanych modeli okazał się być MLP trenowany na wariancie GFCC-FTDNN-RC0.

5.2.1.3.3 Detekcja porażenia nerwu krtaniowego wstecznego

Kolejną chorobą, dla której przeprowadzono analizy mające na celu znalezienie modelu i wariantu x-wektorów zapewniających najbardziej skuteczną klasyfikację było porażenie nerwu krtaniowego wstecznego. Wartości pola pod krzywą ROC uzyskane przez poszczególne modele zestawiono w tabeli 5.11. Tym razem dla każdego wariantu x-wektorów najlepsze rezulaty uzyskano przy użyciu innego modelu. Wszystkie warianty z wyjątkiem wariantu bazującego na MFCC pozwoliły na uzyskanie AUC ROC przekraczającego 0,9. Największą wartość uzyskano przy użyciu wariantu GFCC-FTDNN - w tym przypadku AUC ROC wynosi 0,9323. Należy jednak zauważyć, że różnice pomiędzy poszczególnymi wariantami bazującymi na GFCC nie są duże i wynoszą od zaledwie 0,0005 do 0,0074. Można się więc spodziewać, że różnice te nie będą istotne statystycznie, natomiast różnice pomiędzy wariantem opartym o MFCC i pozostałymi wariantami będą istotne statystycznie.

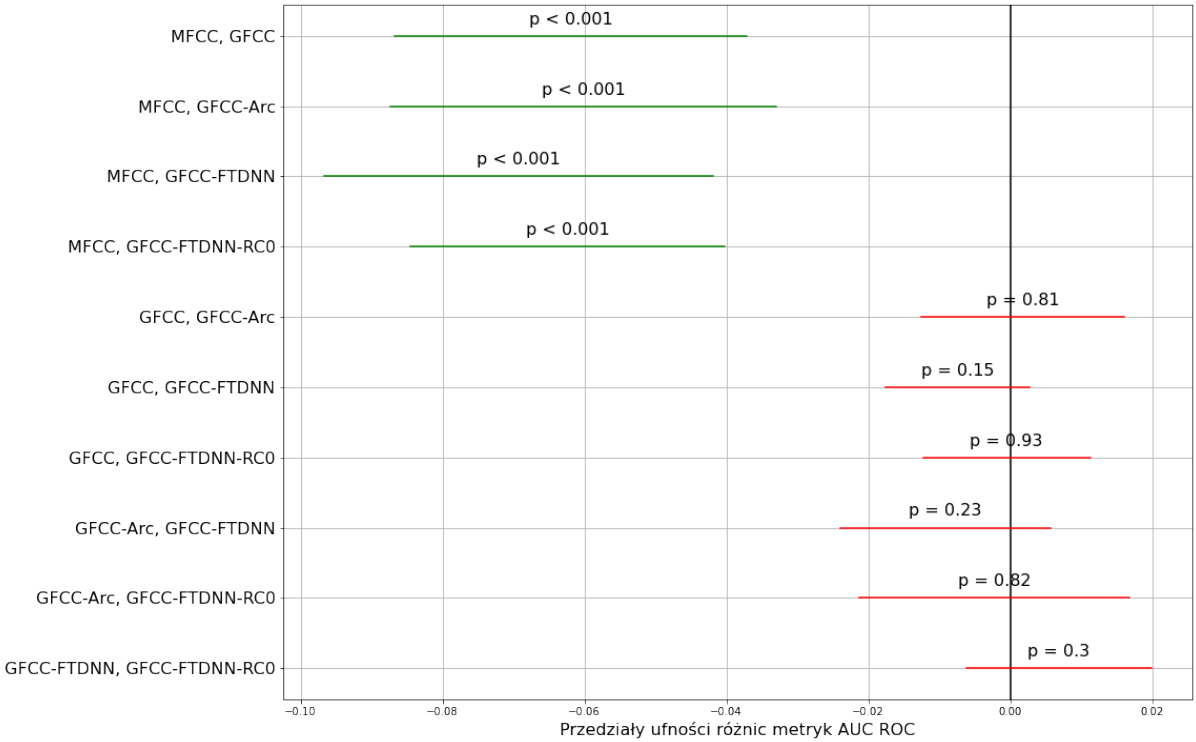
Tabela 5.11. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.8298	0.9051	**0.9231**	0.8918	0.9102
Extra Trees	0.7791	**0.9249**	0.8998	0.8612	0.9134
KNN	0.8366	0.8774	0.8830	0.8803	0.9089
LDA	0.7660	0.8449	0.7856	0.7856	0.8278
LGB	0.8398	0.9037	0.8870	0.8074	0.8715
LR	0.6981	0.7567	0.7021	0.7849	0.7974
MLP	0.8626	0.9076	0.8919	0.9142	**0.9254**
NuSVM	0.8502	0.9032	0.9114	**0.9323**	0.9106
QDA	0.6968	0.7535	0.7845	0.7514	0.8139
RF	0.7503	0.8890	0.7918	0.8417	0.8998
RNN	0.7525	0.8807	0.8594	0.8315	0.9278
SGD	**0.8630**	0.9185	0.9091	0.9152	0.9236

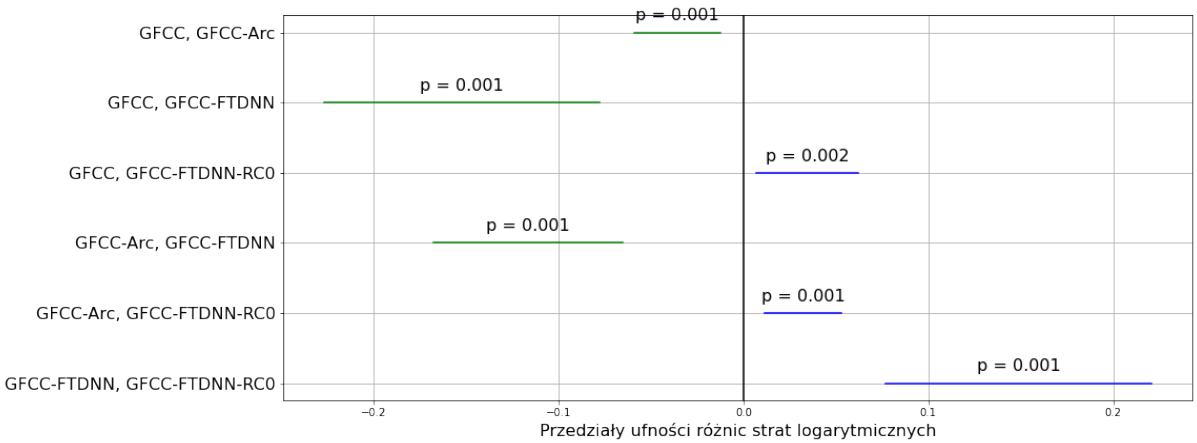
Przeprowadzony test DeLonga pozwolił stwierdzić, że nie wszystkie warianty x-wektorów pozwoliły na uzyskanie takich samych wartości AUC ROC – wartość p wyniosła mniej niż $0,001$. Analiza uzyskanych przedziałów ufności różnic wartości metryki AUC ROC pomiędzy poszczególnymi parami wariantów potwierdziły wyrażone wcześniej przypuszczenia – istotność statystyczną różnic pomiędzy uzyskanymi przez poszczególne modele wartościami pola pod krzywą ROC wykazano jedynie pomiędzy wariantem opartym o MFCC i pozostałymi czterema wariantami. W przypadku czterech wariantów bazujących na GFCC różnice są nieistotne statystycznie (rys. 5.17).

Również tym razem równość strat logarytmicznych poszczególnych modeli zbadano testem Friedmana, ponieważ wygenerowane próby bootstrapowe nie mają rozkładu normalnego (Dodatek C). Uzyskano wartość $p < 0,001$, więc odrzucono hipotezę zerową o równości rozkładów strat logarytmicznych i przyjęto hipotezę alternatywną mówiącą, że rozkłady prób nie są takie same. Przeprowadzony test *post hoc* Nemenyiego pozwolił stwierdzić, że istotne statystycznie różnice występują pomiędzy wszystkimi porównywanymi wariantami (rys. 5.18). Można też zauważyć, że wariant GFCC-FTDNN-RC0 pozwolił na uzyskanie mniejszych wartości strat logarytmicznych niż warianty GFCC, GFCC-Arc i GFCC-FTDNN, o czym świadczą dodatnie wartości w wyznaczonych przedziałach ufności różnic. Wariant GFCC-FTDNN spowodował natomiast uzyskanie większych strat logarytmicznych niż pozostałe trzy oparte o GFCC warianty – w przedziałach ufności znajdują się wyłącznie wartości ujemne. Na podstawie tych obserwacji można stwierdzić, że zgodnie z przyjętymi kryteriami wyboru model MLP uczony na wariancie GFCC-FTDNN-RC0 uzyskuje istotnie lepsze rezultaty od pozostałych

analizowanych modeli wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego i cechuje się lepszymi zdolnościami generalizacyjnymi.



Rys. 5.17. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.



Rys. 5.18. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.

5.2.1.3.4 Detekcja obrzęku Reinkego

Kolejną rozważaną chorobą jest obrzęk Reinkego. Również w tym przypadku wariant oparty o MFCC pozwolił na uzyskanie wyników gorszych od pozostałych czterech wariantów x-wektorów. Uzyskane wartości są jednak zauważalnie wyższe niż w przypadku wcześniej analizowanych chorób – używając wariantu opartego o MFCC uzyskano AUC ROC wynoszące 0,8858, zaś używając czterech wariantów opartych o GFCC uzyskano wartości przekraczające 0,97. Największą wartość uzyskano dla wariantu GFCC-FTDNN-RC0 – wynosi ona 0,9961 i została uzyskana dla modelu MLP.

Przeprowadzając test DeLonga uzyskano wartość p = $0,1502$, co świadczy o tym, że nie ma podstaw do odrzucenia hipotezy zerowej mówiącej o równości wszystkich porównywanych wartości AUC ROC. Oznacza to, że w przypadku obrzęku Reinkego porównywane pięć wariantów x-wektorów nie ma istotnego wpływu na uzyskiwane wartości pola pod krzywą ROC. Jest to pierwszy taki przypadek w niniejszej pracy – dla analizowanych dotychczas chorób istotne różnice można było zauważyć zawsze przynajmniej pomiędzy wariantem opartym o MFCC i pozostałymi wariantami.

Analiza przedziałów ufności różnic pomiędzy wartościami AUC ROC uzyskanymi przez poszczególne porównywane modele pozwala stwierdzić, że istotne różnice można zauważyć jedynie pomiędzy wariantami GFCC-Arc i GFCC-FTDNN oraz GFCC-Arc i GFCC-FTDNN-RC0 (rys. 5.19). Różnice pomiędzy pozostałymi parami rozważanych wariantów nie są już jednak istotne statystycznie, konieczne więc było przeprowadzenie testu równości strat logarytmicznych uwzględniającego wszystkie pięć zaproponowanych wariantów x-wektorów.

Wartość p uzyskana w teście Friedmana wyniosła mniej niż $0,001$, co oznacza, że pomiędzy co najmniej dwoma analizowanymi wariantami występują istotne statystycznie różnice rozkładów strat logarytmicznych dla wygenerowanych prób bootstrapowych.

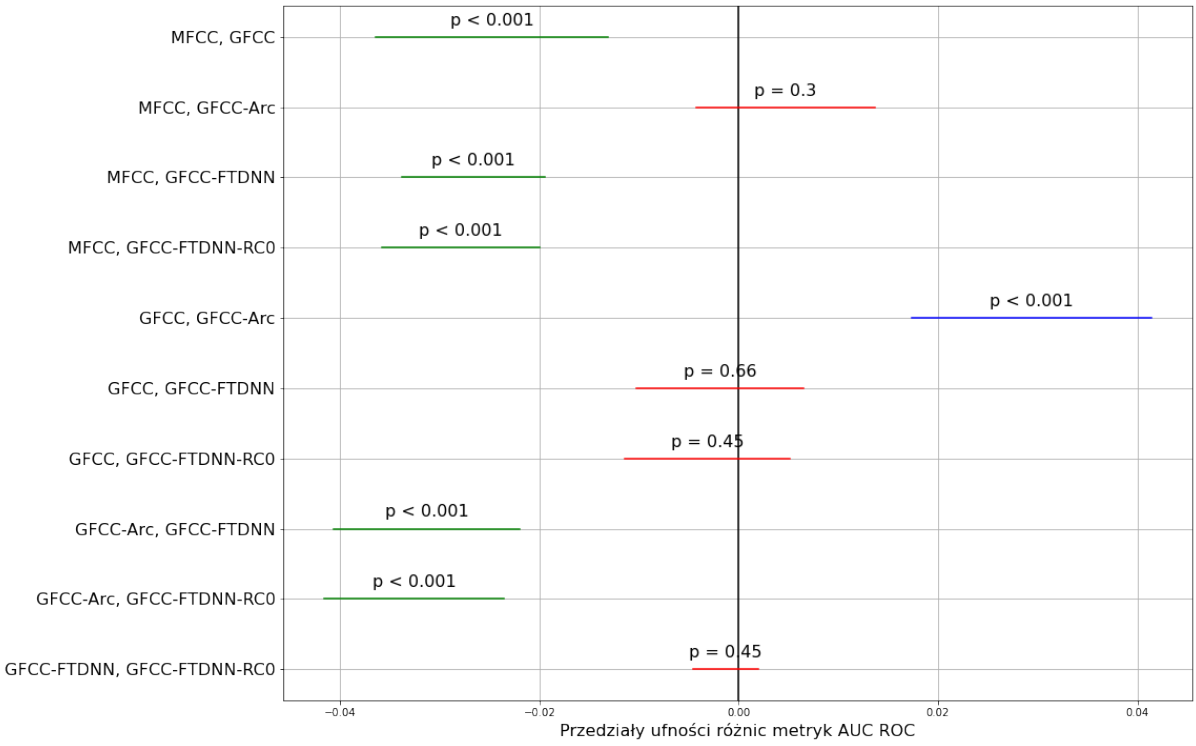
Test Nemenyiego wykazał, że brak istotnych różnic występuje dla par wariantów MFCC i GFCC oraz GFCC i GFCC-FTDNN-RC0 (rys. 5.20). Analizując uzyskane przedziały ufności można stwierdzić, że najlepszym z porównywanych wariantów jest wariant GFCC-Arc – wartości w przedziałach ufności wskazują na to, że straty logarytmiczne uzyskane przy użyciu tego wariantu są mniejsze od strat uzyskiwanych przy użyciu pozostałych wariantów.

5.2.1.3.5 Detekcja pachydermii kontaktowej

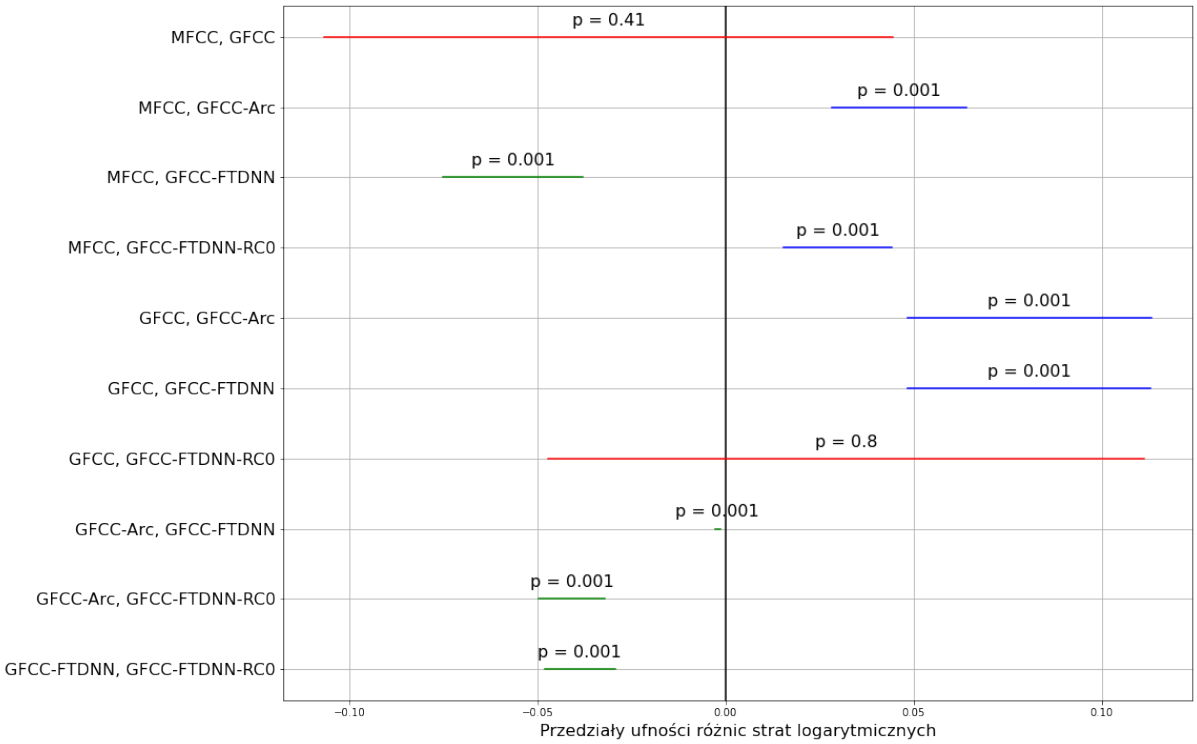
Ostatnią chorobą otolaryngologiczną rozważaną w niniejszej pracy jest pachydermia kontaktowa. Wartości metryki AUC ROC uzyskane przez poszczególne modele zestawiono w tabeli 5.13. Również w tym przypadku wartości AUC ROC przekraczające 0,9 uzyskano dla czterech wariantów opartych o GFCC. Najmniejszą wartość wynoszącą 0,8866 uzyskano dla modelu MLP trenowanego przy użyciu wariantu opartego o MFCC. Największe pole pod krzywą ROC uzyskano dla modelu NuSVM trenowanego na wariancie GFCC-Arc.

Tabela 5.12. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji obrzęku Reinkego.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.8730	0.9724	0.8612	0.9823	0.9916
Extra Trees	0.8189	0.9144	0.9286	0.9710	0.9907
KNN	0.8209	0.9749	0.9050	0.9892	0.9926
LDA	0.7303	0.7421	0.8593	0.9336	0.8711
LGB	**0.8858**	0.9587	0.9518	0.9282	0.9626
LR	0.6132	0.7343	0.6836	0.7382	0.9296
MLP	0.9232	0.9705	0.9675	0.9764	**0.9961**
NuSVM	0.8839	**0.9803**	0.9705	0.9734	0.9921
QDA	0.5000	0.8583	0.7687	0.7968	0.8671
RF	0.7977	0.9454	0.8091	0.8927	0.8858
RNN	0.8415	0.9685	0.7677	0.9784	0.9852
SGD	0.8740	0.9754	**0.9774**	**0.9931**	0.9951



Rys. 5.19. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji obrzęku Reinkego. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze.



Rys. 5.20. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji obrzęku Reinkego, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.

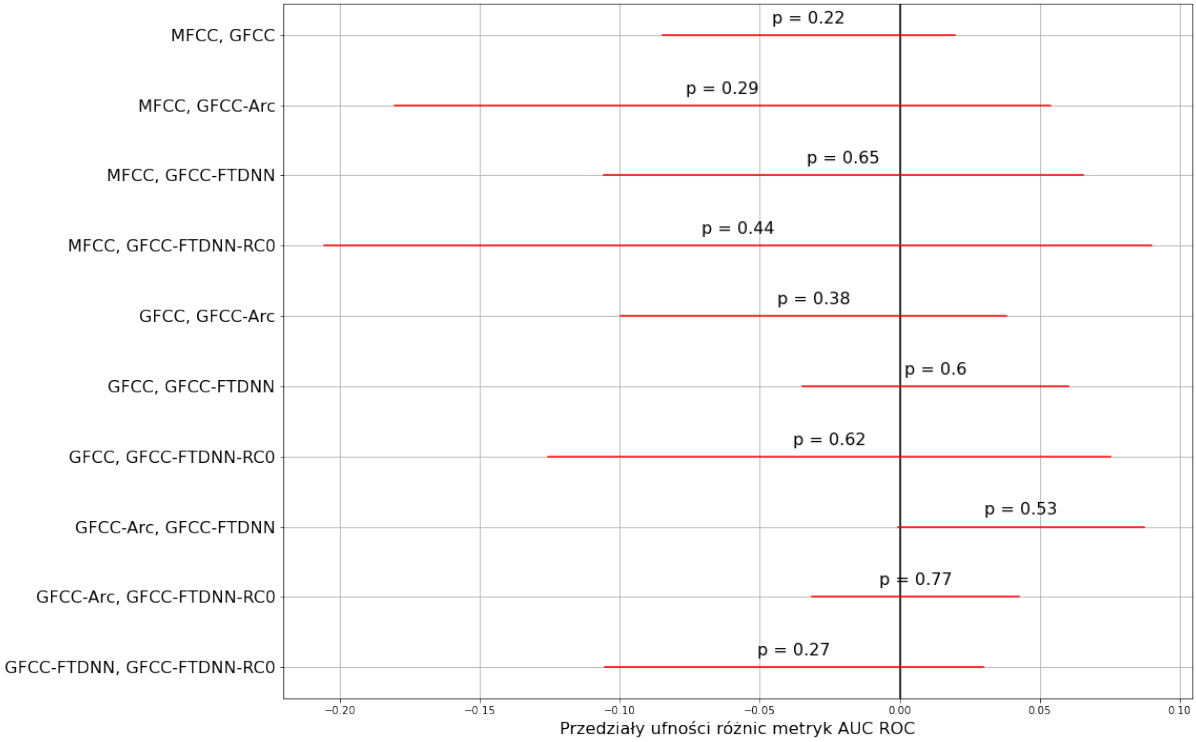
Test DeLonga wykazał brak istotnych różnic pomiędzy analizowanymi wartościami AUC ROC – uzyskana wartość p wynosi 0.7980. Analiza przedziałów ufności oraz wartości p uzyskanych dla poszczególnych par wskazuje na to, że pomiędzy żadną parą wariantów istotne statystycznie różnice nie występują (rys. 5.21). Konieczne jest więc przeprowadzenie testów równości strat logarytmicznych z użyciem wszystkich pięciu wariantów x-wektorów, podobnie jak miało to miejsce w przypadku obrzęku Reinkego. Przed ich przeprowadzeniem należało jednak wybrać tylko jeden model uczony na wariancie GFCC-FTDNN. Zdecydowano, że w testach uwzględniony zostanie model SGD, ponieważ jego strata logarytmiczna na zbiorze testowym wyniosła 0,0525, zaś modelu AdaBoost – 0,2040.

Przeprowadzając test Friedmana uzyskano wartość $p < 0,001$. Podjęto więc decyzję o odrzuceniu hipotezy zerowej mówiącej o równości rozkładów strat logarytmicznych prób bootstrapowych wygenerowanych dla poszczególnych wariantów x-wektorów. Na podstawie wyników testu *post hoc* Nemenyiego (rys. 5.22) można stwierdzić, że istotne statystycznie różnice występują pomiędzy wszystkimi analizowanymi wariantami. Wariantem zapewniającym najmniejsze straty logarytmiczne okazał się być wariant GFCC-FTDNN. Świadczy o tym fakt, że przedział ufności różnic pomiędzy wariantami GFCC-FTDNN i GFCC-FTDNN-RC0 zawiera

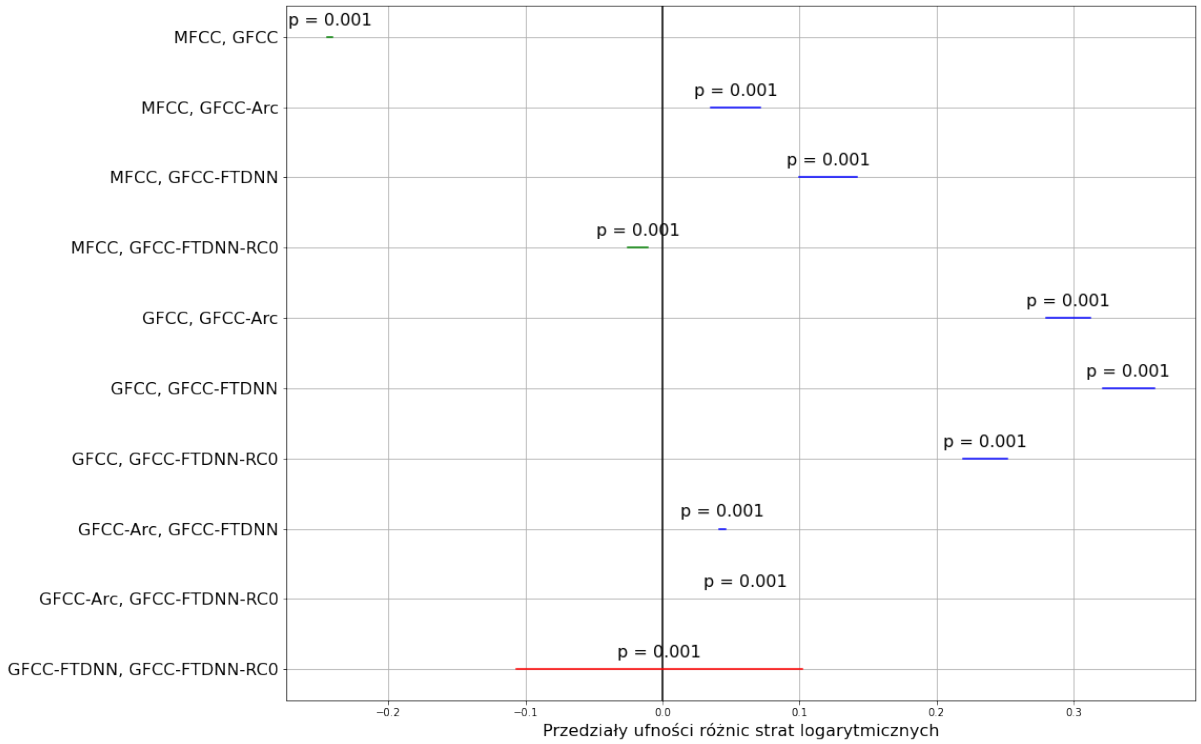
wyłącznie wartości ujemne, zaś przedziały ufności pomiędzy pozostałymi trzema wariantami a wariantem GFCC-FTDNN zawierają wyłącznie wartości dodatnie.

Tabela 5.13. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji pachydermii kontaktowej.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.8752	**0.9213**	0.9410	**0.9087**	**0.9465**
Extra Trees	0.8433	0.8780	0.9035	0.8713	0.9224
KNN	0.7417	0.8988	0.9331	0.8898	0.9319
LDA	0.6461	0.6500	0.7764	0.6461	0.7303
LGB	0.7748	0.8295	0.7669	0.8047	0.9307
LR	0.5000	0.6500	0.6421	0.5421	0.6000
MLP	**0.8866**	0.8866	0.9213	0.9008	0.9339
NuSVM	0.8835	0.8772	**0.9520**	0.8354	0.9252
QDA	0.4528	0.8055	0.7043	0.5461	0.5000
RF	0.8091	0.8189	0.8969	0.8539	0.8815
RNN	0.7910	0.8669	0.5000	0.8378	0.7189
SGD	0.8787	0.8591	0.9512	**0.9087**	0.9433



Rys. 5.21. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji pachydermii kontaktowej.



Rys. 5.22. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji pachydermii kontaktowej, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.

5.2.1.4. Analiza wyników uzyskanych przy użyciu modeli uczonych na zbiorach przeróbkowanych

Rozważane w niniejszej pracy problemy klasyfikacyjne należą do grupy problemów z dużym niezrównoważeniem zbiorów danych, co jest częste w przypadku danych medycznych. Duża dysproporcja w liczebności klas znacząco utrudnia proces uczenia klasyfikatorów i może doprowadzić do sytuacji, w której klasa mniej liczna będzie praktycznie pomijana w procesie uczenia ze względu na fakt, że nawet w sytuacji, gdy wszystkie obiekty z klasy mniej licznej będą zaklasyfikowane nieprawidłowo będzie możliwe uzyskanie zadowalającej wartości funkcji celu. Jest to problem tym bardziej istotny, że często to ta mniej liczna klasa jest głównym obiektem zainteresowania [181], [182] – jest tak m. in. w przypadku baz danych medycznych, w których liczba osób zdrowych znacząco przekracza liczbę osób chorych.

Metody mające na celu poprawę skuteczności klasyfikacji danych niezrównoważonych można podzielić na trzy grupy [183]:

– metody algorytmiczne polegające na modyfikacji istniejących algorytmów uczenia maszynowe w taki sposób, by przystosować je do pracy z danymi opisanymi rozkładem o dużej skośności. Do tej grupy zaliczają się metody polegające na modyfikacji funkcji

kosztu, w których błędne predykcje klasy mniej licznej wiążą się z większym kosztem oraz metody komitetowe polegające na tworzeniu komitetu wielu klasyfikatorów, które łącznie uzyskują bardziej trafne predykcje,

– metody modyfikacji danych wykorzystujące algorytmy nad- i podpróbkowania zbiorów danych,

– metody hybrydowe.

Do grupy metod algorytmicznych zaliczają się m.in. wykorzystywane w niniejszej pracy lasy losowe, algorytmy wzmocnienia adaptacyjnego (AdaBoost) oraz algorytmy wzmocnienia gradientowego (LGB) [184]–[186].

Metody modyfikacji danych były wykorzystywane w większości prac opisanych w rozdziale 3 – problem niezrównoważenia zbiorów danych został rozwiązany poprzez wykorzystanie jedynie części dostępnych danych tak, by uzyskać klasy równoliczne. Nie zostały jednak przybliżone algorytmy, które posłużyły do wyboru obiektów uwzględnianych w analizach, a ich nieprawidłowy dobór może mieć istotny wpływ na uzyskiwane wyniki i spowodować nadmierne dopasowanie modelu do analizowanego zbioru danych [181].

W niniejszej pracy postanowiono wypróbować zarówno algorytmy nadpróbkowania zbioru danych, polegające na replikacji obiektów z klasy mniej licznej lub wygenerowaniu syntetycznych obiektów w oparciu o obiekty rzeczywiste, jak i algorytmy podpróbkowania zbioru danych, polegające na eliminacji niektórych obiektów z klasy bardziej licznej. W tym celu wykorzystano implementacje sześciu metod nadpróbkowania oraz dziesięciu metod podpróbkowania dostępne w bibliotece imblearn [58]. Opis użytych algorytmów zamieszczono w rozdziale 2, sekcji 2.4.

Uczeniu na zbiorach przepróbkowanych poddano jedynie najlepsze modele służące do klasyfikacji poszczególnych schorzeń wybrane w sekcji 5.2.1. Algorytmy próbkujące zastosowano wyłącznie na zbiorach uczących. Zadbano również, by podczas optymalizacji przy użyciu sprawdzianu krzyżowego w zbiorach walidacyjnych znajdowały się jedynie obiekty oryginalne – umieszczenie w nich zreplikowanych bądź syntetycznych obiektów mogłoby spowodować uzyskanie niemiarodajnych rezultatów i doprowadzić do wyboru nieodpowiednich hiperparametrów. Uzyskane wartości AUC ROC zestawiono w tabelach 5.14-5.17.

Analizując wyniki przedstawione w tabeli 5.14 można zauważyć, że w czterech z sześciu przypadków najlepsze wyniki spośród zastosowanych algorytmów zagwarantował algorytm SVM-SMOTE. W przypadku ostrego zapalenia krtani najlepsze rezulaty zagwarantował algorytm Borderline SMOTE, zaś w przypadku porażenia nerwu krtaniowego wstecznego KMeans-SMOTE. Wszystkie trzy algorytmy niwelują dysproporcje w liczebności klas poprzez wygenerowanie syntetycznych nowych obiektów. Na uwagę zasługuje jednak fakt, że uzyskane na zbiorze testowym wartości AUC ROC są zbliżone do siebie bez względu na użyty algorytm nadpróbkowania – jedynym wyjątkiem jest wartość uzyskana dla obrzęku Reinkego przy użyciu algorytmu ADASYN, która wynosi zaledwie 0,5. Co więcej, tylko w przypadku dysfonii bez

sprecyzowanego typu oraz ostrego zapalenia krtani wartości AUC ROC są większe niż uzyskane poprzez trening modelu na oryginalnym, nieprzepróbkowanym zbiorze danych uczących.

Sytuacja wygląda inaczej w przypadku klasyfikacji poszczególnych typów dysfonii – dla wszystkich trzech typów największą wartość AUC ROC uzyskano stosując algorytm ADASYN (tabela 5.16). Największe różnice pomiędzy wartościami uzyskanymi przy zastosowaniu różnych algorytmów nadpróbkowania można zauważyć w przypadku dysfonii czynnościowej, zaś najmniejsze w przypadku dysfonii spastycznej. Co ważne, w przypadku dysfonii czynnościowej uzyskana przy użyciu algorytmu ADASYN wartość AUC ROC jest mniejsza niż ta uzyskana bez przepróbkowania danych. Warto również zauważyć, że w przypadku dysfonii spastycznej wszystkie algorytmy z wyjątkiem KMeans-SMOTE spowodowały osiągnięcie pola pod krzywą ROC równego 1, co wskazuje na uzyskanie klasyfikatora idealnego.

W przypadku dysfonii czynnościowej jeden z algorytmów podpróbkowania – Neighbourhood Cleaning Rule – spowodował uzyskanie nieco większej wartości AUC ROC niż uzyskana w wyniku treningu na danych nieprzepróbkowanych 5.17. W przypadku dysfonii psychogennej poprawę można zaobserwować stosując pięć z dziesięciu metod podpróbkowania: AllKNN, Cluster Centroids, NearMiss, RENN oraz Tomek Links. W przypadku dysfonii spastycznej ponownie udało się uzyskać AUC ROC wynoszące 1.

Podobnie jak miało to miejsce podczas analizy wyników uzyskanych z wykorzystaniem oryginalnych, nieprzepróbkowanych zbiorów danych, przeprowadzono analizę istotności statystycznej różnic w metrykach AUC ROC. Dla każdej rozważanej w niniejszej pracy choroby przeprowadzono test DeLonga sprawdzający równość wartości pola pod krzywą ROC uzyskanych przez model trenowany na danych nieprzepróbkowanych oraz na danych przepróbkowanych z użyciem algorytmu, dla którego uzyskano największą wartość pola pod krzywą ROC.

Tabela 5.14. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli uczonych na danych nadpróbkowanych przy użyciu poszczególnych technik nadpróbkowania, wykorzystywanych do detekcji poszczególnych schorzeń otolaryngologicznych. RMO - Random Minority Oversampling.

	AUC ROC					
Technika nadpróbkowania	Nieokreślone schorzenie krtani	Dysfonia	Ostre zapalenie krtani	Porażenie nerwu krtaniowego wstecznego	Obrzęk Reinkego	Pachydermia kontaktowa
ADASYN	0.8633	–	0.8828	0.9238	0.5000	0.8992
Borderline SMOTE	0.8722	0.8395	**0.9175**	0.9270	0.9685	0.8961
KMeans-SMOTE	0.8620	0.8654	0.8923	**0.9325**	0.97936	—
RMO	0.8650	0.8429	0.8948	0.9295	0.9833	0.8835
SMOTE	0.8741	0.8588	0.9055	0.9234	0.9843	0.8953
SVM-SMOTE	**0.8753**	**0.8655**	0.9002	0.9266	**0.9882**	**0.9213**
Model	MLP	NuSVM	MLP	NuSVM	SGD	NuSVM
Wariant	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-Arc	GFCC-FTDNN

Tabela 5.15. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli uczonych na danych podpróbkowanych przy użyciu poszczególnych technik nadpróbkowania, wykorzystywanych do detekcji poszczególnych schorzeń otolaryngologicznych. RENN - Repeated Edited Nearest Neighbours, RMU - Random Majority Undersampling, NCR - Neighbourhood Cleaning Rule, CNN - Condensed Nearest Neighbour, ENN - Edited Nearest Neighbours.

	AUC ROC					
Technika podpróbkowania	Nieokreślone schorzenie krtani	Dysfonia	Ostre zapalenie krtani	Porażenie nerwu krtaniowego wstecznego	Obrzęk Reinkego	Pachydermia kontaktowa
AllKNN	0.8713	0.8520	**0.8998**	0.9057	0.9902	0.9534
Cluster Centroids	0.8572	0.8638	0.8973	0.9264	0.9843	0.9252
CNN	0.8658	0.8437	0.8958	0.9256	0.9705	0.9260
ENN	0.8758	0.8399	0.8895	0.9065	0.9882	0.9417
NearMiss	0.8660	0.8611	0.8803	0.9244	0.9803	**0.9780**
NCR	0.8732	0.8516	0.8976	0.9059	0.9882	0.9347
One-Sided Selection	0.8678	0.8603	0.8838	0.9329	0.9902	0.8701
RMU	0.8698	0.8576	0.8756	0.9203	0.9852	0.8984
RENN	**0.8808**	0.8501	0.8945	0.9087	**0.9921**	0.9118
Tomek links	0.8670	**0.8649**	0.8935	**0.9333**	**0.9921**	0.8780
Model	MLP	NuSVM	MLP	NuSVM	SGD	NuSVM
Wariant	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-Arc	GFCC-FTDNN

Tabela 5.16. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli uczonych na danych nadpróbkowanych przy użyciu poszczególnych technik nadpróbkowania, wykorzystywanych do detekcji poszczególnych typów dysfonii.

	AUC ROC		
Technika nadpróbkowania	Dysfonia czynnościowa	Dysfonia psychogenna	Dysfonia spastyczna
ADASYN	**0.8234**	**0.9169**	**1.0000**
Borderline SMOTE	0.7491	0.9138	**1.0000**
KMeans-SMOTE	0.8038	–	0.9982
Random minority oversampling	0.8124	0.9103	**1.0000**
SMOTE	0.8223	0.9016	**1.0000**
SVM-SMOTE	0.7932	0.9003	**1.0000**
Model	MLP	SGD	NuSVM
Wariant	GFCC	GFCC-Arc	GFCC

Tabela 5.17. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli uczonych na danych podpróbkowanych przy użyciu poszczególnych technik nadpróbkowania, wykorzystywanych do detekcji poszczególnych typów dysfonii.

	AUC ROC		
Technika podpróbkowania	Dysfonia czynnościowa	Dysfonia psychogenna	Dysfonia spastyczna
AllKNN	0.8216	**0.9204**	0.9994
Cluster Centroids	0.7843	0.9103	0.9958
Condensed Nearest Neighbour	0.8110	0.8417	0.9885
Edited Nearest Neighbours	0.8227	0.8880	**1.0000**
NearMiss	0.8114	0.9116	**1.0000**
Neighbourhood Cleaning Rule	**0.8829**	0.8976	0.9988
One-Sided Selection	0.8268	0.8745	0.9988
Random majority undersampling	0.8148	0.8924	0.9988
Repeated Edited Nearest Neighbours	0.8251	0.9169	0.9994
Tomek links	0.8199	0.9007	0.9982
Model	MLP	SGD	NuSVM
Wariant	GFCC	GFCC-Arc	GFCC

W przypadku obrzęku Reinkego i dysfonii spastycznej więcej niż jeden algorytm przepróbkowania doprowadził do uzyskania takich samych wartości AUC ROC na zbiorze testowym, konieczne więc było zastosowanie dodatkowego kryterium wyboru najlepszego algorytmu. Postanowiono, że kryterium tym będzie mniejsza strata logarytmiczna uzyskiwana na zbiorze testowym, podobnie jak w przyjętej metodzie wyboru najlepszych modeli uczonych na danych nieprzepróbkowanych. Zgodne z tym kryterium najlepszymi algorytmami są:

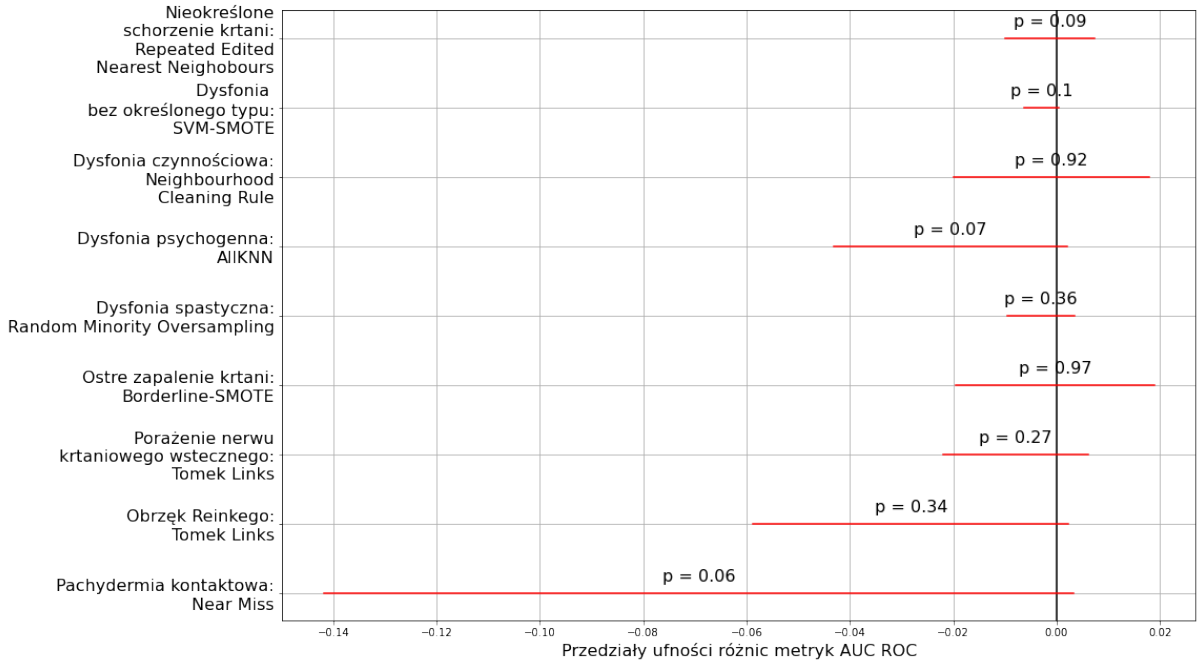
– w przypadku obrzęku Reinkego: algorytm Tomek Links, dla którego uzyskano stratę logarytmiczną równą 0,2321 – dla algorytmu RENN wyniosła ona 0,3987,

– w przypadku dysfonii spastycznej: algorytm Random Minority Oversampling, dla którego uzyskano stratę logarytmiczną równą 0,0691 – dla algorytmu ADASYN wyniosła ona 0,0983, dla Borderline SMOTE 0,0703, dla SMOTE 0,1105, dla SVM-SMOTE 0,1579, dla Edited Nearest Neighobours 0,4913, zaś dla NearMiss 0,3703.

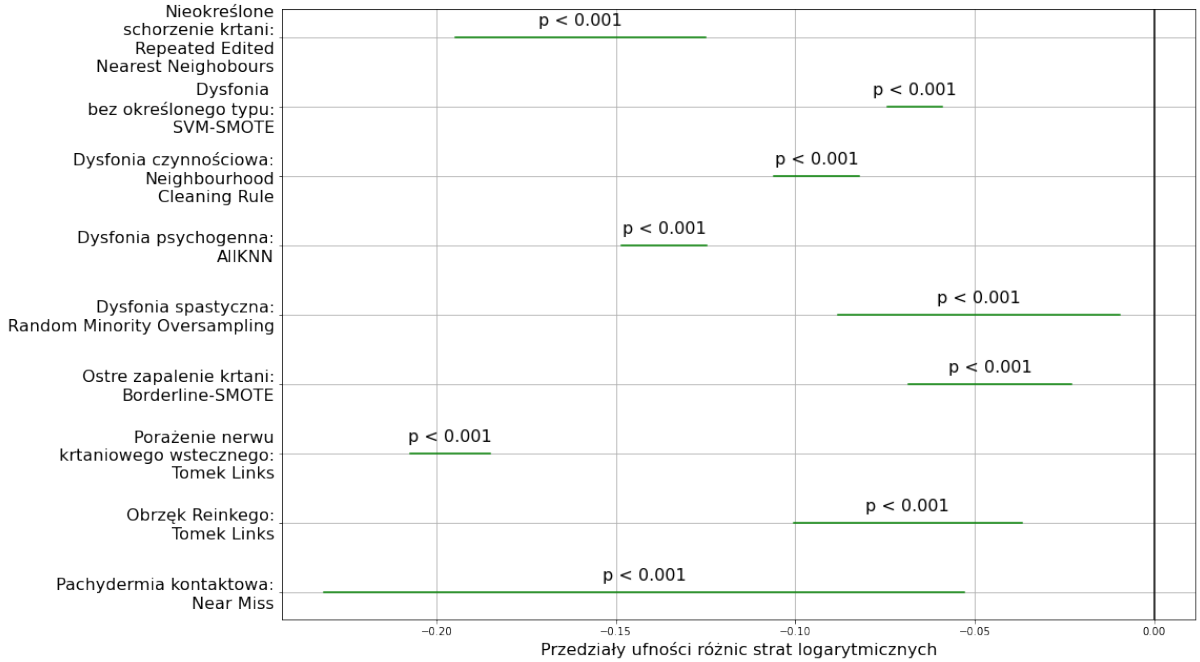
Wyniki testów DeLonga przedstawioo na rys. 5.23. Analizując uzyskane wartości p można zauważyć, że w żadnym przypadku nie wystąpiły istotne statystycznie różnice w wartościach AUC ROC, więc nie jest możliwy wybór lepszego modelu na podstawie tej metryki.

Kolejnym krokiem było wygenerowanie prób bootstrapowych i sprawdzenie równości uzyskiwanych dla nich strat logarytmicznych. W przypadku każdej choroby porównywano tym razem tylko dwa modele, więc do sprawdzenia równości strat logarytmicznych wykorzystano test par Wilcoxona. Wybór testu podyktowany był faktem, że żadna z wygenerowanych prób bootstrapowych nie miała rozkładu normalnego (Dodatek C), konieczne więc było wykorzystanie testu nieparametrycznego.

Wyniki testów Wilcoxona przedstawiono na rys. 5.24 – istotne statystycznie różnice strat logarytmicznych wykazano we wszystkich rozważanych przypadkach. Jednakże uzyskane wyniki świadczą na niekorzyść modeli uczonych na zbiorach przepróbkowanych, co można wywnioskować na podstawie faktu, że wszystkie wyznaczone przedziały ufności różnic zawierają wyłącznie wartości ujemne. Ponieważ wyznaczając przedziały ufności różnic zawsze odejmowano wartość straty logarytmicznej uzyskanej dla modelu uczonego na danych przepróbkowanych od wartości uzyskanej dla modelu uczonego na danych nieprzepróbkowanych oznacza to, że przepróbkowanie zbioru uczącego spowodowało uzyskanie modeli o gorszych właściwościach generalizacyjnych.



Rys. 5.23. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli uczonych na danych nieprzepróbkowanych oraz przepróbkowanych.



Rys. 5.24. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli uczonych na danych nieprzepróbkowanych oraz przepróbkowanych, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC.

5.2.1.5. Analiza wyników uzykanych przy użyciu komitetów klasyfikatorów optymalizowanych genetycznie

Stosowane do tej pory modele klasyfikacyjne w większości przypadków można zaliczyć do klasyfikatorów o nieskomplikowanej strukturze. Jedynie klasyfikatory drzewiaste – lasy losowe, AdaBoost, Extra Trees oraz LGB – mają strukturę złożoną i składają się z wielu drzew decyzyjnych tworzących komitet. Jednakże możliwe jest również tworzenie komitetów złożonych z różnych typów klasyfikatorów, co w niektórych zastosowaniach pozwoliło znacząco poprawić skuteczność klasyfikacji w stosunku do rezultatów uzyskiwanych z użyciem prostych modeli klasyfikacyjnych.

Istotnym zagadnieniem dotyczącym komitetów klasyfikatorów jest odpowiedni dobór modeli składowych, ich hiperparametrów oraz metod przetwarzania podawanych do nich danych. W niniejszej pracy postanowiono przeprowadzić optymalizację architektury komitetu klasyfikatorów używając w tym celu biblioteki TPOT [187], [188] w języku Python, wykorzystującej algorytmy genetyczne do znalezienia architektury maksymalizującej dokładność klasyfikacji na drodze optymalizacji wielokryterialnej. Podczas kolejnych generacji wybierane są rozwiązania optymalne w sensie Pareto, które cechują się wyższością na innymi potencjalnymi architekturami, gwarantując lepsze rezultaty, jednak nie jest możliwe wyłonienie najlepszej spośród nich. Dalsze modyfikacje tych architektur spowodowałyby poprawę jakości komitetu klasyfikatorów według niektórych kryteriów oceny oraz równoczesne pogorszenie według innych kryteriów. W bibliotece TPOT stosowanymi kryteriami oceny są złożoność komitetu oraz jego zdolności predykcyjne oceniane poprzez zdefiniowaną metrykę sukcesu – podobnie jak miało to miejsce podczas optymalizacji modeli opisanych w poprzednich sekcjach, ze względu na niezrównoważenie zbioru danych zdecydowano się maksymalizować metrykę AUC PR. Proces optymalizacji kończy się wyborem komitetu gwarantującego największą wartość wybranej metryki sukcesu spośród wszystkich wyłonionych rozwiązań Pareto-optymalnych [35].

Trening oraz optymalizację komitetów klasyfikatorów przeprowadzono dla każdej z rozważanych chorób otolaryngologicznych, wykorzystując w tym celu zaproponowane pięć wariantów x-wektorów – schematy otrzymanych architektur komitetów przedstawiono w załączniku E. W niektórych przypadkach uzyskane architektury były bardzo proste, przykładowo komitet wykorzystywany do detekcji pachydermii kontaktowej uczony na wariancie MFCC składa się z dwóch modeli połączonych szeregowo: modelu k najbliższych sąsiadów oraz modelu XGBoost wykorzystującego metodę wzmocnienia gradientowego. W niektórych przypadkach uzyskane architektury są jednak dużo bardziej złożone, przykładowo komitet wykorzystywany do detekcji porażenia nerwu krtaniowego wstecznego uczony na wariancie MFCC składa się z ośmiu elementów połączonych szeregowo (rys. E.1), zaś komitet wykorzystywany do detekcji ostrego zapalenia krtani uczony na wariancie GFCC składa się z trzech elementów połączonych równolegle, które następnie łączą się szeregowo z dwoma kolejnymi klasyfikatorami (rys. E.2).

Wartości metryki AUC ROC uzyskane przez komitety wykorzystywane do detekcji poszczególnych schorzeń zestawiono w tabelach 5.18 i 5.19. Jak można zauważyć, w przypadku dysfonii, ostrego zapalenia krtani, porażenia nerwu krtaniowego wstecznego oraz obrzęku Reinkego

Tabela 5.18. Wartości metryki AUC ROC uzyskane na zbiorach testowych dla poszczególnych typów dysfonii przy użyciu komitetów modeli o architekturze optymalizowanej genetycznie.

	AUC ROC		
Wariant x-wektorów	Dysfonia funkcjonalna	Dysfonia psychogenna	Dysfonia spastyczna
MFCC	0.5098	0.5000	0.5000
GFCC	0.5288	0.5000	**0.9615**
GFCC-Arc	0.5475	0.5000	0.5000
GFCC-FTDNN	0.5855	0.5000	0.5000
GFCC-FTDNN-RC0	**0.5991**	**0.8045**	0.5000

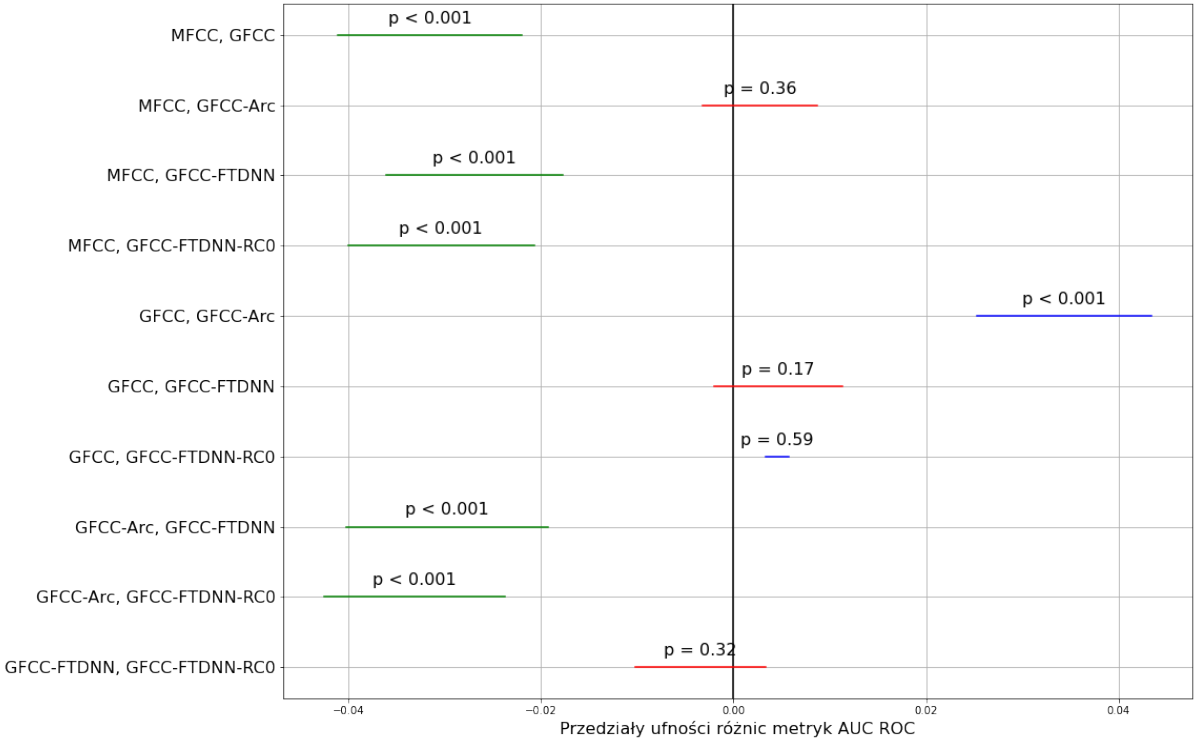
wartości pola pod krzywą ROC uzyskane z użyciem wariantu x-wektorów oparych o MFCC są mniejsze od uzyskanych z użyciem czterech wariantów opartych o GFCC. W przypadku pozostałych dwóch zagadnień przedstawionych w tabeli 5.19 – detekcji osób chorych otolaryngologicznie oraz detekcji pachydermii kontaktowej – wariant oparty o MFCC dał nieznacznie lepsze rezultaty niż wariant GFCC-Arc.

Rezultaty uzyskane dla najmniej licznych chorób – obrzęku Reinkego, pachydermii kontaktowej oraz trzech typów dysfonii – są wyraźnie gorsze niż te uzyskane dla pozostałych schorzeń, w których różnice w liczebności grup osób zdrowych i chorych są mniejsze. W przypadku obrzęku Reinkego wszystkie warianty x-wektorów pozwalają na uzyskanie zbliżonych rezultatów mieszczących się w przedziale [0,6722; 0,6949], zaś w przypadku dysfonii czynnościowej wszystkie wartości są jeszcze mniejsze i mieszczą się w przedziale [0,5098; 0,5991]. W przypadku pachydermii kontaktowej wariant GFCC-FTDNN-RC0 pozwolił na uzyskanie AUC ROC wynoszącego 0,7, jednakże pozostałe cztery warianty pozwoliły jedynie na uzyskanie AUC ROC nie przekraczającego 0,5. Podobnie wygląda sytuacja w przypadku dysfonii psychogennej i funkcjonalnej – tylko jeden z wariantów pozwolił na uzyskanie AUC ROC większego niż 0,5. Są to odpowiednio wariant GCC-FTDNN-RC0 oraz GFCC.

Tabela 5.19. Wartości metryki AUC ROC uzyskane na zbiorach testowych dla poszczególnych schorzeń otolaryngologicznych przy użyciu komitetów modeli o architekturze optymalizowanej genetycznie.

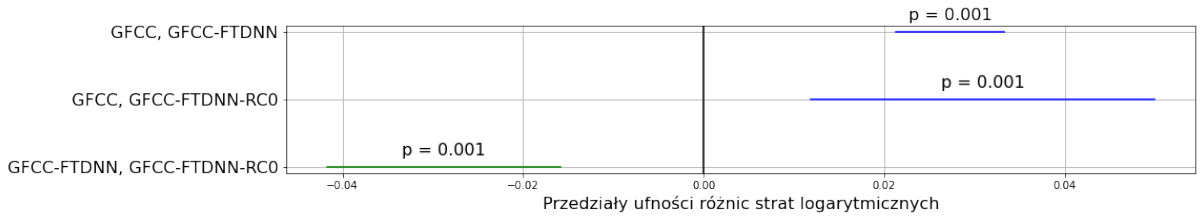
	AUC ROC					
Wariant x-wektorów	Nieokreślone schorzenie krtani	dysfonia	zapalenie krtani	porażenie nerwów krtaniowych	obrzęk Reinkego	pachydermia kontaktowa
MFCC	0.8569	0.7975	0.8233	0.8786	0.6722	0.5000
GFCC	**0.8875**	0.8484	0.8624	0.9091	0.6875	0.5000
GFCC-Arc	0.8531	0.8391	0.8285	**0.9219**	0.6820	0.4658
GFCC-FTDN	0.8828	0.8504	0.8333	0.8814	**0.6949**	0.5000
GFCC-FTDNN-RC0	0.8862	**0.8651**	**0.8754**	0.8624	0.6806	**0.7000**

Podobnie jak uczyniono podczas poprzednich przeprowadzonych analiz, najlepszy komitet służący do detekcji danego schorzenia wyłoniono na drodze analizy istotności statystycznej różnic wartości metryki AUC ROC lub, w przypadku braku istotnych różnic, poprzez analizę istotności różnic strat logarytmicznych. Analizy rozpoczęto od klasyfikacji osób ze zdiagnozowanym dowolnym schorzeniem otolaryngologicznym. Uzyskana w teście DeLonga wartość p mniejsza niż $0,001$ wskazuje na to, że pomiędzy co najmniej dwoma komitetami występują istotne statystycznie różnice w metrykach AUC ROC. Wyniki dla poszczególnych par wariantów przedstawiono na rys. 5.25. Istotnych statystycznie różnic nie wykazano dla pary wariantów MFCC i GFCC-Arc, jednakże oba warianty są istotnie mniejsze od pozostałych trzech wariantów, co można stwierdzić na podstawie analizy przedziałów ufności różnic. Różnice nieistotne statystycznie występują też pomiędzy wariantami GFCC, GFCC-FTDNN oraz GFCC-TDNN-RC0. Aby wybrać wariant, na którym nauczony został najlepszy komitet konieczne było przeprowadzenie testu równości strat logarytmicznych.



Rys. 5.25. Przadziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji osób chorych otolaryngologicznie. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

W teście Friedmana uzyskano wartość $p < 0,001$. Test Nemenyiego (rys. 5.26) pozwolił na stwierdzenie, że istotne statystycznie różnice strat logarytmicznych występują pomiędzy wszystkimi trzema modelami. Analizując przedziały ufności różnic można stwierdzić, że najmniejsze wartości strat logarytmicznych uzyskano na wariancie GFCC-FTDNN-RC0 – wartości w przedziałach wyznaczonych dla par GFCC i GFCC-FTDNN-RC0 oraz GFCC-FTDNN i GFCC-FTDNN-RC0 są dodatnie, co świadczy o tym, że straty logarytmiczne uzyskane dla wariantu GFCC-FTDNN-RC0 są mniejsze od tych uzyskanych dla pozostałych dwóch wariantów. Zgodnie z przyjętymi kryteriami, komitet wykorzystywany do detekcji schorzeń otolaryngologicznych uczony na wariancie GFCC-FTDNN-RC0 (rys. E.5) może zostać uznany za najlepszy.



Rys. 5.26. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji osób chorych otolaryngologicznie, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem niebieskim oznaczono przedziały wskazujące na istotnie większą wartość strat logarytmicznych uzyskaną dla modelu prostego, zaś kolorem zielonym przedział wskazujący na istotnie większą wartość strat logarytmicznych uzyskaną dla komitetu klasyfikatorów.

Test DeLonga przeprowadzony dla wartości AUC ROC uzyskanych dla komitetów służących do wykrywania dysfonii zwrócił wartość p mniejszą niż $0,001$, co oznacza, że również tym razem występują istotne statystycznie różnice pomiędzy co najmniej dwiema porównywanymi wartościami. Wartości p oraz przedziały ufności różnic uzyskane dla komitetów uczonych na poszczególnych parach wariantów x-wektorów przedstawiono na rys. 5.27. Jak można zauważyć, istotne statystycznie różnice występują pomiędzy wszystkimi wariantami z wyjątkiem par GFCC i GFCC-Arc oraz GFCC i GFCC-FTDNN. Wariant oparty o MFCC okazał się istotnie gorszy od czterech wariantów opartych o GFCC. Pomimo tego, że nie wykazano istotnych różnic pomiędzy wszystkimi rozważanymi wariantami, test DeLonga pozwoli wyłonić wariant, który zagwarantował uzyskanie największej wartości AUC ROC, a tym samym najlepszy komitet według przyjętych kryteriów i nie było konieczności analizy uzyskiwanych strat logarytmicznych. Analiza przedziałów ufności pozwoliła podjąć decyzję, że jest to komitet uczony na wariancie GFCC-FTDNN-RC0 (rys. E.5) – w przedziałach wyznaczonych dla par wariantów, w których się on znajduje wartości są wyłącznie ujemne, co oznacza, że AUC ROC uzyskane na tym wariancie jest istotnie większe od wszystkich pozostałych.

Wszystkie komitety trenowane w celu detekcji dysfonii czynnościowej uzyskały wartości AUC ROC niewiele większe od 0,5, czego nie można uznać za dobre wyniki. Co więcej, przeprowadzony test DeLonga zwrócił wartość p równą 0,8658, co oznacza, że żadna z wartości przedstawionych w tabeli 5.18 nie różni się istotnie od 0,5. Oznacza to, że wszystkie uzyskane

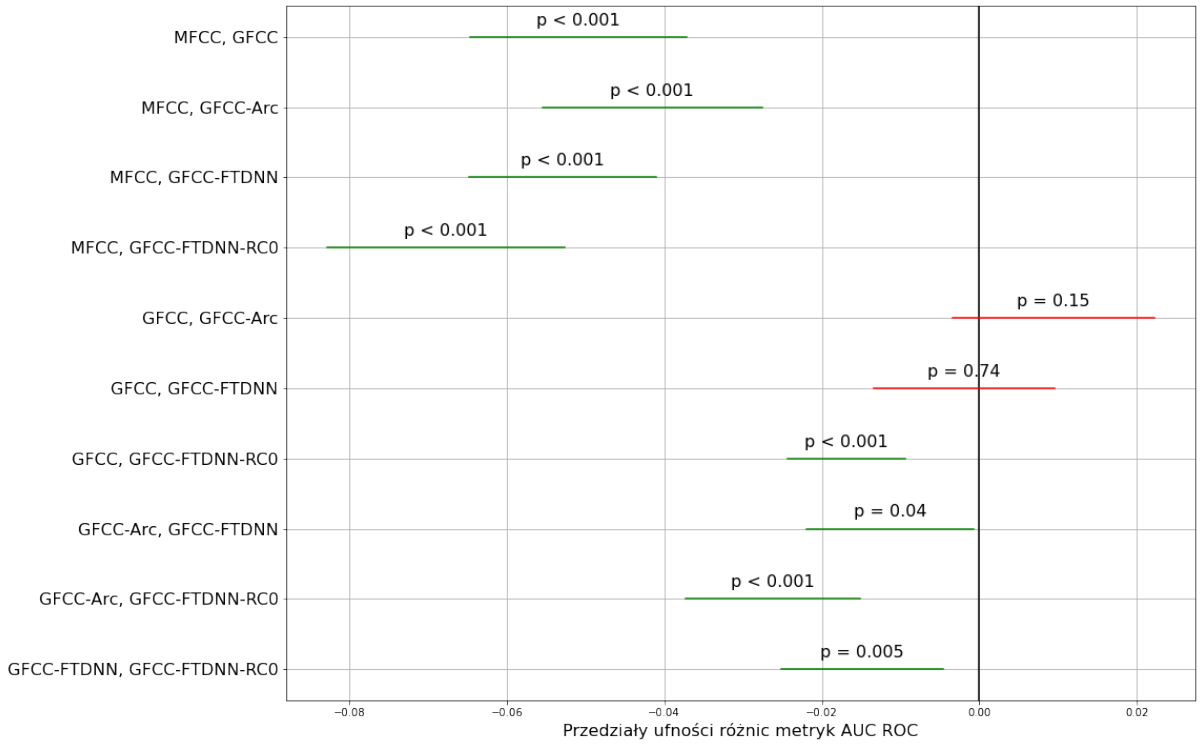
komitety nie różnią się istotnie od modelu zwracającego predykcje losowe. Z tego względu zrezygnowano z przeprowadzenia dalszych testów mających na celu wyłonienie najlepszego komitetu – nawet, gdyby wykazano istotne statystycznie różnice w uzyskiwanych przez poszczególne komitety stratach logarytmicznych, nie zmieniłoby to faktu, że nie pozwalają one na uzyskanie wiarygodnych wyników. Bez względu na to, który wariant x-wektorów cechuje się lepszymi zdolnościami generalizacyjnymi, wszystkie pięć komitetów można uznać za nieprzydatne w kontekście detekcji dysfonii czynnościowej.

W przypadku dysfonii psychogennej tylko jeden z porównywanych komitetów uzyskał wartości AUC ROC większe od 0,5. Z tego powodu zdecydowano, że zamiast testu DeLonga porównującego wartości uzyskane przez wszystkie komitety wykonany zostanie test DeLonga sprawdzający równość komitetu o największej wartości AUC ROC z modelem zwracającym predykcje losowe. Analogiczne testy przeprowadzone dla pozostałych komitetów byłyby pozbawione sensu – komitety uzyskujące AUC ROC równe 0,5 mają dokładnie taką samą wartość pola pod krzywą ROC jak model losowy, więc bez przeprowadzenia testu można stwierdzić, że nie ma pomiędzy nimi istotnych statystycznie różnic i predykcje zwracane przez taki komitet nie są wiarygodne. Analiza zwracanych przez komitety predykcji pozwoliła ponadto stwierdzić, że przyczyną tak złych wyników jest przyporządkowanie wszystkich obiektów do klasy bardziej licznej, czyli uznanie osób chorych za zdrowe. W przypadku dysfonii spastycznej sytuacja jest podobna – tylko jeden komitet uzyskał AUC ROC większe od 0,5, więc tylko on został poddany testowi DeLonga.

Dla dysfonii psychogennej oraz spastycznej uzyskano wartość p mniejsza niż $0,001$. Oznacza to, że oba komitety uzyskały wartości AUC ROC istotnie różne od AUC ROC modeli zwracających predykcje losowe. Można więc uznać, że komitet uczony na wariancie GFCC-FTDNN-RC0 jest najlepszym spośród porównywanych komitetów wykorzystywanych do detekcji dysfonii psychogennej, zaś komitet uczony na wariancie GFCC jest istotnie lepszy od pozostałych w rozwiązywaniu problemu detekcji dysfonii spastycznej.

Analiza schematów uzyskanych komitetów przedstawionych na rys. E.6-E.8 pozwala zauważyć interesującą zależność – wszystkie komitety służące do wykrywania poszczególnych typów dysfonii, dla których uzyskane pole pod krzywą ROC nie różni się istotnie od wartości modeli zwracających predykcje losowe są w rzeczywistości pojedynczymi modelami klasyfikacyjnymi, a nie komitetami klasyfikatorów. W wyniku optymalizacji genetycznej uzyskano tylko dwa faktyczne komitety – w przypadku dysfonii psychogennej jest to komitet złożony z klasyfikatora *k* najbliższych sąsiadów oraz klasyfikatora XGBoost, zaś w przypadku dysfonii spastycznej komitet złożony z klasyfikatorów AdaBoost oraz XGBoost. Są to równocześnie jedyne modele, dla których uzyskano AUC ROC istotnie większe od 0,5.

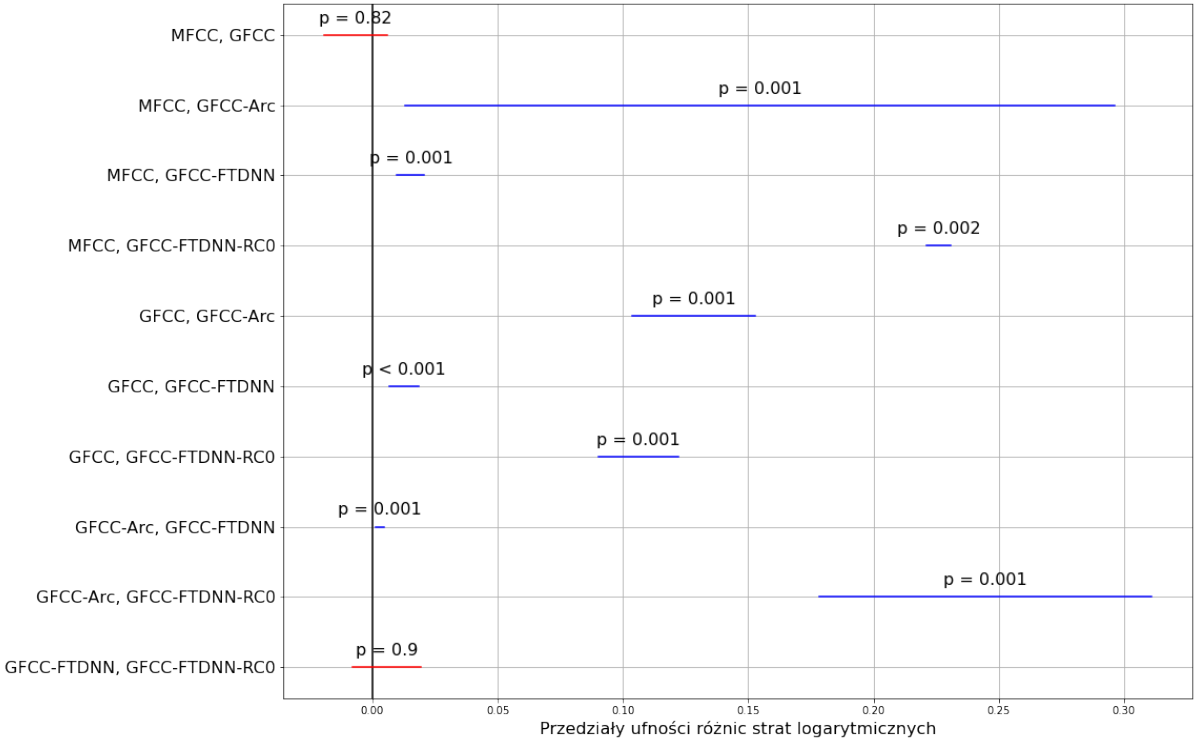
W przypadku testu DeLonga przeprowadzonego dla ostrego zapalenia krtani nie było podstaw do odrzucenia hipotezy zerowej mowiącej o równości wszystkich porównywanych wartości AUC ROC – wartość p wynosi $0,1736$. Wykonano więc test Friedmana, by zbadać różnice w uzyskiwanych przez poszczególne komitety stratach logarytmicznych, uzyskując wartość p



Rys. 5.27. Przadziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji dysfonii. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

mniejszą niż $0,001$ oznaczającą, że istnieją podstawy do odrzucenia hipotezy zerowej i pomiędzy co najmniej dwoma komitetami występują istotne różnice w rozkładach wartości strat logarytmicznych prób bootstrapowych. Wyniki testu *post hoc* Nemenyiego przedstawiono na rys. 5.28. Brak istotnych statystycznie różnic wykazano dla dwóch par wariantów x-wektorów: opartego na MFCC i opartego na GFCC oraz GFCC-FTDNN i GFCC-FTDNN-RC0. Analiza przedziałów ufności różnic pozwoliła określić, że najmniejsze straty logarytmiczne uzyskano dla wariantów GFCC-FTDNN i GFCC-FTDNN-RC0. Oznacza to, że komitety klasyfikatorów uczone na tych dwóch wariantach są zgodnie z przyjętymi kryteriami tak samo dobre w rozwiązywaniu problemu detekcji ostrego zapalenia krtani.

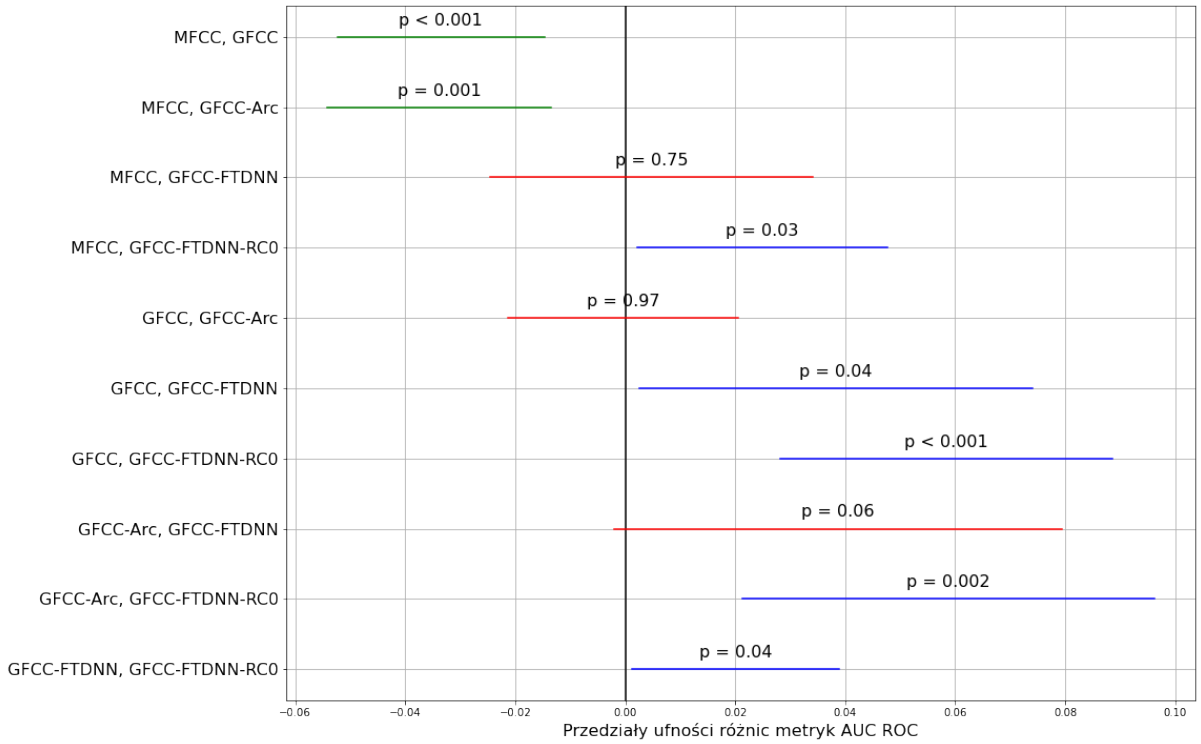
Ponieważ stosowana dotychczas metoda wyboru jednego modelu lub komitetu modeli przewyższającego pozostałe porównywane z nim modele nie sprawdziła się w tym przypadku, postanowiono wprowadzić dodatkowe kryterium, które pozwoli wyłonić lepszy komitet klasyfikatorów – ustalono, że w takiej sytuacji lepszym wyborem będzie komitet o mniej złożonej architekturze. Kryterium złożoności jest jednym z kryteriów oceny stosowanych podczas optymalizacji genetycznej w bibliotece TPOT, więc taki wybór pozwala zachować ciągłość zasad oceny komitetów. Ponadto należy mieć na uwadze fakt, że mniej złożona architekturze oznacza



Rys. 5.28. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji ostrego zapalenia krtani, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy modelami, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość strat logarytmicznych uzyskaną dla modelu prostego.

mniejszą złożoność obliczeniową, a więc jest korzystniejsza z punktu widzenia wykorzystania zasobów obliczeniowych. Zgodnie z tym kryterium lepszym komitetem jest ten uczony na wariancie GFCC-FTDNN (Dodatek E, rys. E.4).

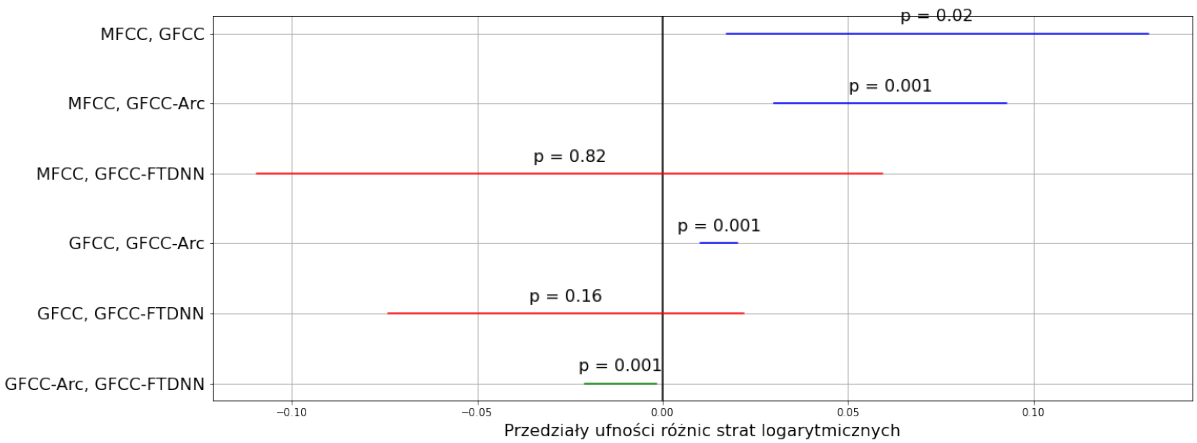
Test DeLonga przeprowadzony dla komitetów mających za zadanie wykryć porażenie nerwu krtaniowego wstecznego pozwolił stwierdzić, że pomiędzy co najmniej dwoma komitetami występują istotne statystycznie różnice w uzyskiwanych wartościach pola pod krzywą ROC – wartość p wyniosła $0,002$ i jest mniejsza od przyjętego poziomu istotności. Analiza przedziałów ufności różnic pomiędzy metrykami uzyskanymi przez poszczególne komitety (rys. 5.29) pozwoliła stwierdzić, że jedynym wariantem x-wektorów pozwalającym na uzyskanie AUC ROC istotnie różnego od pozostałych czterech wariantów jest wariant GFCC-FTDNN-RC0. Jednakże na podstawie wartości w tych przedziałach można stwierdzić, że są to wyniki istotnie gorsze, więc najlepszym komitetem jest któryś spośród komitetów trenowanych na pozostałych czterech wariantach. Żeby określić, który to komitet, konieczne było przeprowadzenie testu Friedmana i zweryfikowanie hipotezy o równości uzyskiwanych strat logarytmicznych.



Rys. 5.29. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

Test Friedmana zwrócił wartość p równą 0.0113, co oznacza, że istnieją podstawy do odrzucenia hipotezy zerowej mówiącej o równości rozkładów opisujących wartości strat logarytmicznych uzyskiwanych przez poszczególne komitety klasyfikatorów. Na podstawie testu Nemenyiego, którego wyniki przedstawiono na rys. 5.30 można stwierdzić, że istotne statystycznie różnice nie występują pomiędzy wariantami MFCC i GFCC-FTDNN oraz GFCC i GFCC-FTDNN. Wykazano natomiast różnice pomiędzy wariantem opartym o GFCC-Arc oraz pozostałymi trzema analizowanymi wariantami x-wektorów. Analiza przedziałów ufności różnic wskazuje na to, że komitet trenowany na wariancie opartym o GFCC-Arc (rys. E.3) uzyskał najmniejsze wartości strat logarytmicznych, więc może zostać uznany za najlepszy spośród porównywanych komitetów wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego.

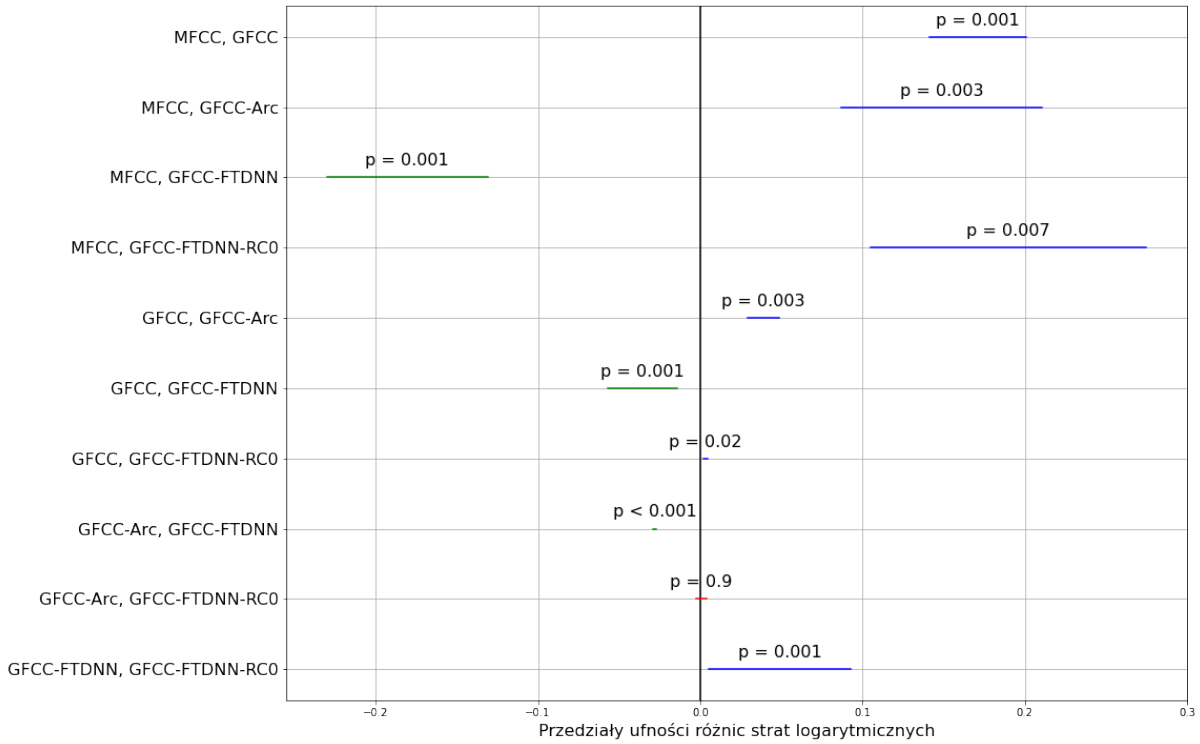
W przypadku obrzęku Reinkego test DeLonga nie wykazał istotnych różnic pomiędzy wartościami AUC ROC uzyskanymi przez pięć porównywanych komitetów, o czym świadczy wartość p równa $0,7701$. Wybór najlepszego komitetu został więc dokonany w oparciu o wyniki testu Friedmana i testu *post hoc* Nemenyiego. W teście Friedmana uzyskano wartość p mniejszą niż $0,001$ wskazującą na występowanie istotnych statystycznie różnic pomiędzy stratami logarytmicznymi porównywanych komitetów klasyfikatorów.



Rys. 5.30. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji porażenia nerwu krtaniowego wstecznego, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy modelami, kolorem niebieskim przedziały wskazujące na istotnie większą wartość strat logarytmicznych uzyskaną dla modelu prostego, zaś kolorem zielonym przedziaływskazujący na istotnie większą wartość strat logarytmicznych uzyskaną dla komitetu klasyfikatorów.

Wyniki testu Nemenyiego przedstawiono na rys. 5.31 – zgodnie z przedstawionymi wartościami p można stwierdzić, że różnice nie występują jedynie pomiędzy parą wariantów GFCC-Arc i GFCC-FTDNN-RC0. Analiza przedziałów ufności różnic pozwala ponadto stwierdzić, że wartości strat logarytmicznych uzyskanych przez komitety klasyfikatorów uczonych na tych dwóch wariantach są mniejsze niż wartości uzyskane przez komitety trenowane na pozostałych trzech wariantach. Postępując zgodnie z opisanym wcześniej schematem postępowania zdecydowano, że w przypadku braku różnic w uzyskiwanych wartościach metryk AUC ROC oraz strat logarytmicznych dwóch komitetów decydujące znaczenie będzie mieć złożoność architektury komitetu i za najlepszy komitet uznano ten trenowany na wariancie GFCC-Arc. Należy jednak zwrócić uwagę, że tak naprawdę nie można w tej sytuacji mówić o komitecie klasyfikatorów, ponieważ w jego skład wchodzi tylko jeden model oparty o algorytm SGD (rys. E.3).

W przypadku pachydermii kontaktowej jedynie dwa komitety modeli – uczone na wariantach GFCC-Arc i GFCC-FTDNN-RC0 – uzyskały wartość pola pod krzywą ROC różną od 0,5 (tabela 5.19). Należy jednak zauważyć, że w przypadku wariantu GFCC-Arc wartość ta jest mniejsza niż 0,5, więc jego predykcje nie mogą zostać uznane za wiarygodne. Stwierdzono więc, że najlepszym komitetem jest ten uczony na wariancie GFCC-FTDNN-RC0 (rys. E.5). Przy użyciu testu DeLonga sprawdzono, czy uzyskana przez niego wartość AUC ROC wynosząca 0,7 jest istotnie różna od wartości uzyskiwanej przez model zwracający predykcje losowe, czyli od wartości 0,5. Wartość p wyniosła 0.0072 co oznacza, że istnieją podstawy do odrzucenia hipotezy zerowej mówiącej o braku takich różnic. Z tego wniosek, że można przyjąć hipotezę alternatywną i stwierdzić, że AUC ROC komitetu istotnie różni się od 0,5, a więc jego predykcje nie są losowe.



Rys. 5.31. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji obrzęku Reinkego, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy modelami, kolorem niebieskim przedziały wskazujące na istotnie większą wartość strat logarytmicznych uzyskaną dla modelu prostego, zaś kolorem zielonym przedziały wskazujące na istotnie większą wartość strat logarytmicznych uzyskaną dla komitetu klasyfikatorów.

5.2.1.6. Zestawienie najlepszych wyników uzyskanych podczas klasyfikacji poszczególnych schorzeń krtani

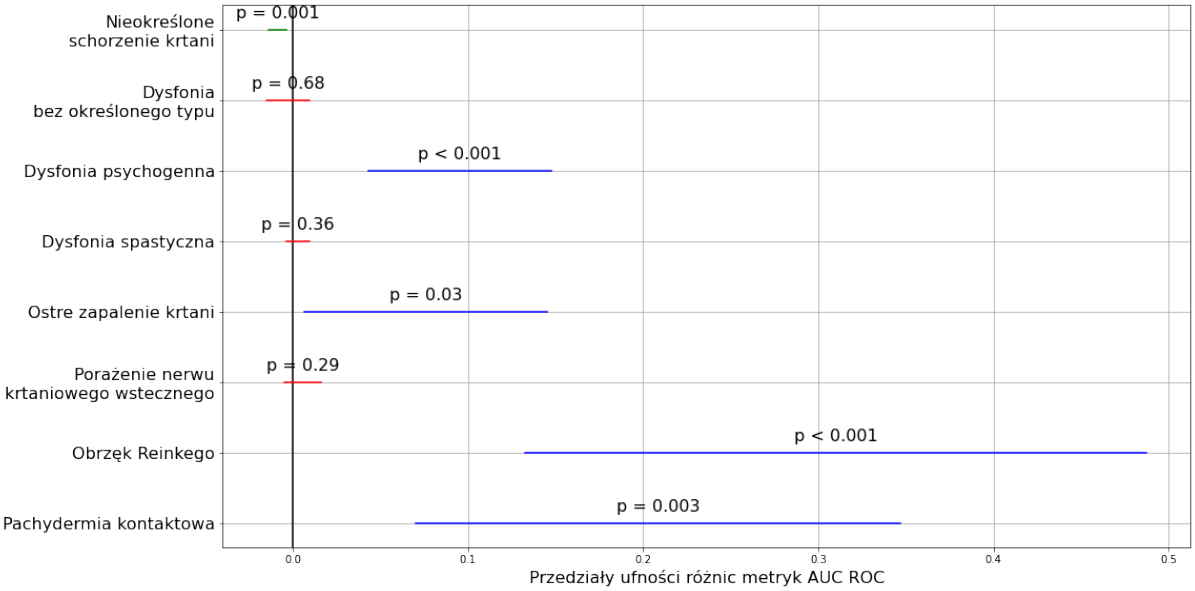
Ostatnim etapem prac mającym na celu wyłonienie najlepszego modelu dla każdej z analizowanych chorób otolaryngologicznych było porównanie wyników uzyskanych przez proste modele, uczone na danych nieprzepróbkowanych, z wynikami uzyskanymi przez komitety modeli o architekturze optymalizowanej genetycznie – modele uczone na danych przepróbkowanych zostały wykluczone z tego etapu prac ze względu na istotnie gorsze rezultaty niż modele proste, co wykazano w rozdziale 5, sekcji 5.2.1.4. W tym celu przeprowadzono testy DeLonga dla modeli uznanych za najlepsze w poprzednich sekcjach. Wyniki tych testów przedstawiono na rys. 5.32. Na podstawie uzyskanych wartości p można wyłonić modele dające istotnie większe wartości AUC ROC służące do klasyfikacji pięciu grup pacjentów: cierpiących na nieokreślone schorzenie otolaryngologiczne, dysfonię psychogenną, ostre zapalenie krtani, obrzęk Reinkego oraz pachydermię kontaktową – w tych przypadkach wartości p są mniejsze od przyjętego poziomu istotności wynoszącego 0,05. Analiza przedziałów ufności różnic pozwala stwierdzić, że we wszystkich przypadkach z wyjątkiem detekcji osób cierpiących na nieokreślone schorzenie krtani, istotnie większe wartości pola pod krzywą ROC uzyskano używając prostych modeli.

Ponieważ przedziały wyznaczano odejmując wartości metryki uzyskane przez komitet klasyfikatorów służący do detekcji danego schorzenia od wartości uzyskanej przez prosty, pojedynczy model, dodatnie wartości w przedziałach wskazują na przewagę klasyfikatorów prostych. Ujemne wartości występują jedynie w przedziale ufności różnic modeli służących do odróżnienia osób cierpiących na dowolne schorzenie od osób zdrowych – w tym przypadku lepszy okazał się komitet klasyfikatorów. Warto również zauważyć, że najlepsze rezultaty zostały uzyskane przez modele trenowane na danych nieprzepróbkowanych.

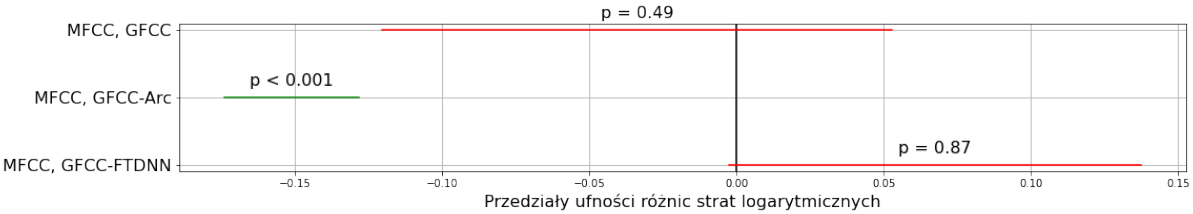
W przypadku dysfonii czynnościowej nie przeprowadzono testu DeLonga – wszystkie komitety modeli uzyskały wartości AUC ROC nie różniące się w sposób istotny od wartości 0,5, więc ich przydatność jest w praktyce znikoma. Model uczony z użyciem wariantu GFCC na danych nieprzepróbkowanych, uznany we wcześniejszych analizach za przewyższający model uczony na danych przeprókowanych mógł zostać uznany za najlepszy bez dalszych testów.

Test DeLonga nie wystarczył do określenia, który model daje najlepsze rezultaty w przypadku dysfonii bez określonego typu, dysfonii spastycznej i porażenia nerwu krtaniowego wstecznego. Dla tych trzech chorób przeprowadzono zatem testy par Wilcoxona, dzięki którym określono, który model uzyskuje istotnie mniejsze straty logarytmiczne. Wyniki tych testów przedstawiono na rys. 5.33. Jak można zauważyć, jedynie w przypadku dysfonii spastycznej udało się jednoznacznie rozstrzygnąć, który model daje lepsze rezultaty – uzyskana w teście Wilcoxona wartość p jest mniejsza od przyjętego poziomu istotności, co oznacza, że straty logarytmiczne modeli istotnie się różnią. Na podstawie przedziału ufności różnic można stwierdzić, że komitet klasyfikatorów uzyskał większe straty logarytmiczne. Za lepszy model można więc uznać AdaBoost trenowany na danych nieprzepróbkowanych.

W przypadku dysfonii bez określonego typu oraz porażenia nerwu krtaniowego wstecznego nie udało się określić, który model jest lepszy – wartości p są większe od przyjętego poziomu istotności, więc nie ma podstaw do odrzucenia hipotez zerowych mówiących o równości rozkładów opisujących straty logarytmiczne uzyskane przez oba porównywane modele. Postanowiono więc przyjąć trzecie kryterium oceny, wykorzystane już wcześniej do wyboru najlepszego komitetu klasyfikatorów. Ponieważ wartości metryk AUC ROC oraz straty logarytmiczne obu modeli nie różnią się istotnie, lepszym wyborem będzie model o mniejszej złożoności architektury oraz złożoności obliczeniowej. Warto zauważyć, że decyzja o zastosowaniu takiego kryterium znajduje potwierdzenie w literaturze – w sytuacji, gdy nie powoduje to znacznego pogorszenia jakości klasyfikatora ocenianej poprzez wybrane metryki sukcesu, rekomendowane jest używanie klasyfikatorów o możliwie prostej architekturze [189]. Zgodnie z tym kryterium należy wybrać modele niebędące komitetami klasyfikatorów, a więc w przypadku dysfonii NuSVM trenowany na danych nieprzepróbkowanych, zaś w przypadku porażenia nerwu krtaniowego wstecznego MLP trenowany na danych nieprzepróbkowanych.



Rys. 5.32. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych prostych modeli uczonych na danych nieprzepróbkowanych oraz komitetów modeli klasyfikacyjnych. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy modelami, kolorem zielonym przedział wskazujący na istotnie większą wartość AUC ROC uzyskaną dla komitetu klasyfikatorów, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość AUC ROC uzyskaną dla modelu prostego.



Rys. 5.33. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych najlepszych prostych modeli uczonych na danych nieprzepróbkowanych oraz komitetów modeli klasyfikacyjnych, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy modelami, zaś kolorem zielonym przedział wskazujący na istotnie większą wartość strat logarytmicznych uzyskaną dla komitetu klasyfikatorów.

5.2.1.7. Wybór metryk wykorzystywanych do ewaluacji modeli

Przedstawiając rezultaty klasyfikacji danych medycznych należy mieć na uwadze zapewnienie zrozumiałości uzyskanych wyników przez osoby niezwiązane ze środowiskiem inżynierskim, które mogą nie być zaznajomione z powszechnie stosowanymi do ewaluacji algorytmów klasyfikacyjnych metrykami sukcesu, takimi jak precyzja, F1 czy AUC ROC wykorzystywane w niniejszej pracy. Do takich osób zaliczają się w szczególności lekarze, będący równocześnie grupą docelową, mającą w przyszłości korzystać z nowoczesnych rozwiązań wspomagających proces diagnostyczny [190]. W środowisku lekarskim znane i zrozumiałe są natomiast inne

często stosowane metryki, które wykorzystywane są również do opisu zdolności diagnostycznych testów laboratoryjnych – czułość, specyficzność i dokładność [191]. Istotną przewagą przedstawienia wyników w postaci czułości i specyficzności, nie zaś precyzji i czułości, jest fakt, że raportując czułość i specyficzność metody diagnostycznej uwzględnia się wszystkie cztery miary przedstawiane w macierzy pomyłek: predykcje prawdziwie pozytywne, prawdziwie negatywne, fałszywie pozytywne oraz fałszywie negatywne. Raportując czułość i precyzję nie uwzględnia się natomiast predykcji prawdziwie negatywnych, w związku z czym nie jest możliwe odtworzenie macierzy pomyłek i wyliczenie pozostałych metryk. Biorąc pod uwagę powyższe zdecydowano się przedstawić wartości właśnie tych trzech metryk – dokładności klasyfikacji, czułości oraz specyficzności – uzyskiwane przez wybrane modele klasyfikacyjne.

Przed wyliczeniem wartości metryk sukcesu konieczne było przeprowadzenie binaryzacji uzyskiwanych predykcji. Wszystkie dotychczasowe analizy prowadzone były z użyciem prawdopodobieństw przynależności obiektów do klas. Aby wyliczyć z nich predykcje dychotomiczne, konieczne jest przyjęcie wartości odcięcia, która pozwoli na wyznaczenie klasy, do której dany obiekt przynależy według klasyfikatora [192]. Punkty odcięcia wyznaczono zgodnie z metodą opisaną w [193] jako punkt na krzywej ROC maksymalizujący iloczyn skalarny czułości i specyficzności. Metoda ta pozwala na uzyskanie węższych przedziałów ufności oraz mniejszych wartości odchylenia standardowego dokładności klasyfikacji [194], niż inne powszechnie stosowane metody, takie jak bazująca na wskaźniku Youdena [195] czy maksymalizująca statystykę χ^2 zaproponowana w [196].

W tabelach 5.21 i 5.20 przedstawiono wartości metryk sukcesu uzyskane przez najlepsze modele klasyfikacyjne wyłonione w analizach opisanych w poprzednich sekcjach rozdziału. Jak można zauważyć, wariant GFCC-FTDNN-RC0 gwarantuje najlepsze rezultaty spośród rozważanych wariantów x-wektorów w czterech przypadkach: klasyfikacji mającej na celu identyfikację osób cierpiących na dowolne schorzenie otolaryngologiczne, klasyfikacji osób cierpiących na dysfonię bez określonego typu, ostre zapalenie krtani oraz porażenie nerwu krtaniowego wstecznego. Analizując liczebność pacjentów cierpiących na poszczególne schorzenia przedstawioną w tabelach 5.4 oraz 5.1 można zauważyć, że są to te choroby, w przypadku których stosunek osób chorych do zdrowych jest nie mniejszy niż 1:5. Pozostałe przeprowadzone analizy – dotyczące obrzęku Reinkego, pachydermii kontaktowej oraz trzech typów dysfonii – dotyczą zbiorów danych o większym niezrównoważeniu i w ich przypadku najlepsze rezultaty zagwarantowały inne warianty x-wektorów. Jednakże, co istotne, w żadnym z analizowanych przypadków zastosowanie wariantu opartego o MFCC nie okazało się prowadzić do uzyskania najbardziej satysfakcjonujących wyników klasyfikacji.

Uzyskane wartości metryki AUC ROC we wszystkich przypadkach można uznać za satysfakcjonujące – we wszystkich przypadkach przekraczają one 0,85. Również wartości pozostałych wyliczonych metryk – dokładności klasyfikacji, czułości oraz specyficzności – można uznać za obiecujące. Jedynie w przypadku dysfonii bez określonego typu dokładność klasyfikacji nie przekroczyła 0,80, uzyskana wartość jest jednak niewiele mniejsza i wynosi 0,7984.

Tabela 5.20. Wartości metryk sukcesu modeli o największej wartości pola pod krzywą ROC uzyskane na zbiorach testowych dla poszczególnych typów dysfonii.

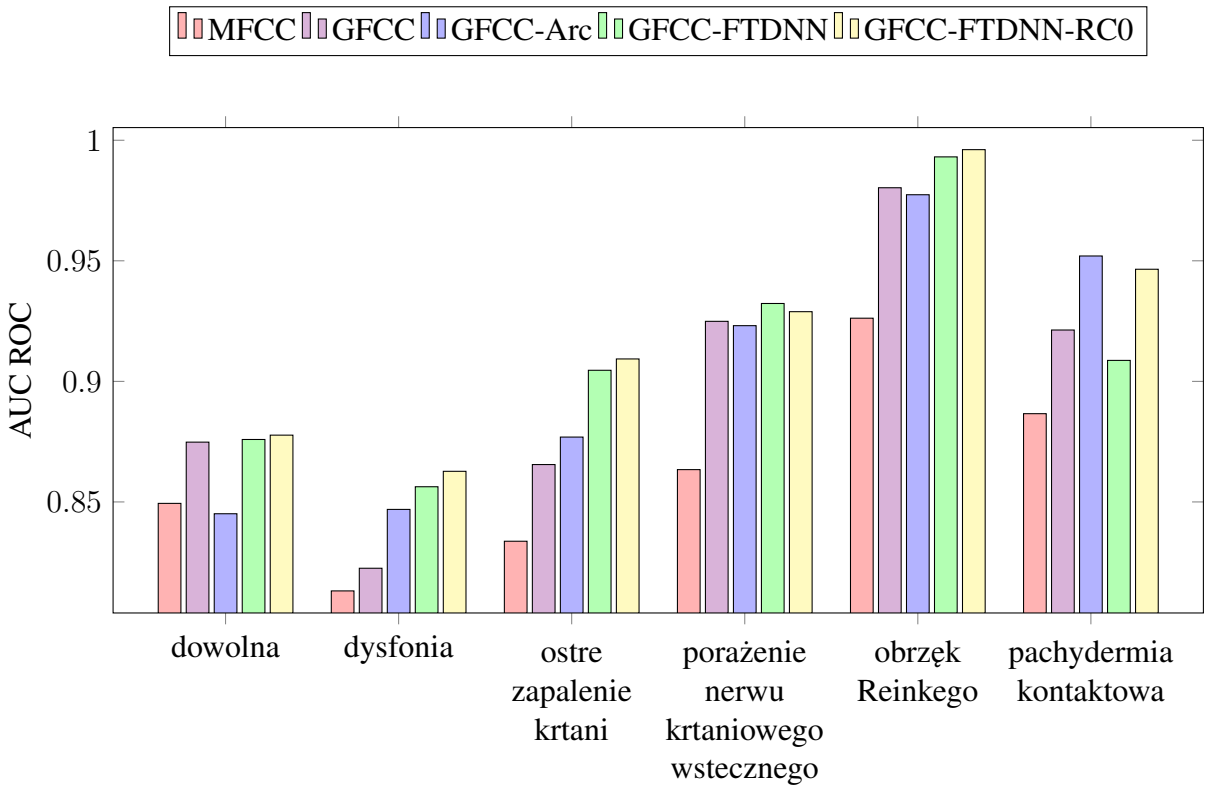
	Dysfonia		
Metryka	Funkcjonalna	Psychogenna	Spastyczna
Dokładność	0.8133	0.8069	0.9714
Czułość	0,7391	0.7778	0.9231
Specyficzność	0.8268	0.8110	0.9764
AUC ROC	0,8819	0,8998	0,9981
Model	MLP	Extra Trees	AdaBoost
Wariant	GFCC	GFCC-Arc	GFCC

Największą dokładność uzyskano dla dysfonii spastycznej – wartość 0,9714 wskazuje na niemal stuprocentową poprawność klasyfikacji obiektów znajdujących się w zbiorze testowym. Po przeanalizowaniu predykcji zwracanych przez model stwierdzono, że błędnie zaklasyfikowany został jeden sygnał odpowiadający osobie zdrowej oraz jeden sygnał odpowiadający osobie chorej. Z kolei najmniejszą specyficznością cechuje się komitet klasyfikatorów o architekturze optymalizowanej genetycznie wykorzystywany do detekcji dowolnego schorzenia krtani – wynosi ona 0,7952. We wszystkich pozostałych przypadkach specyficzność przekracza 0,80. Czułość uzyskana przez poszczególne modele klasyfikacyjne jest najmniej zadowalająca ze wszystkich analizowanych metryk sukcesu – w najgorszym przypadku, czyli detekcji dysfonii bez określonego typu, wynosi zaledwie 0,7069.

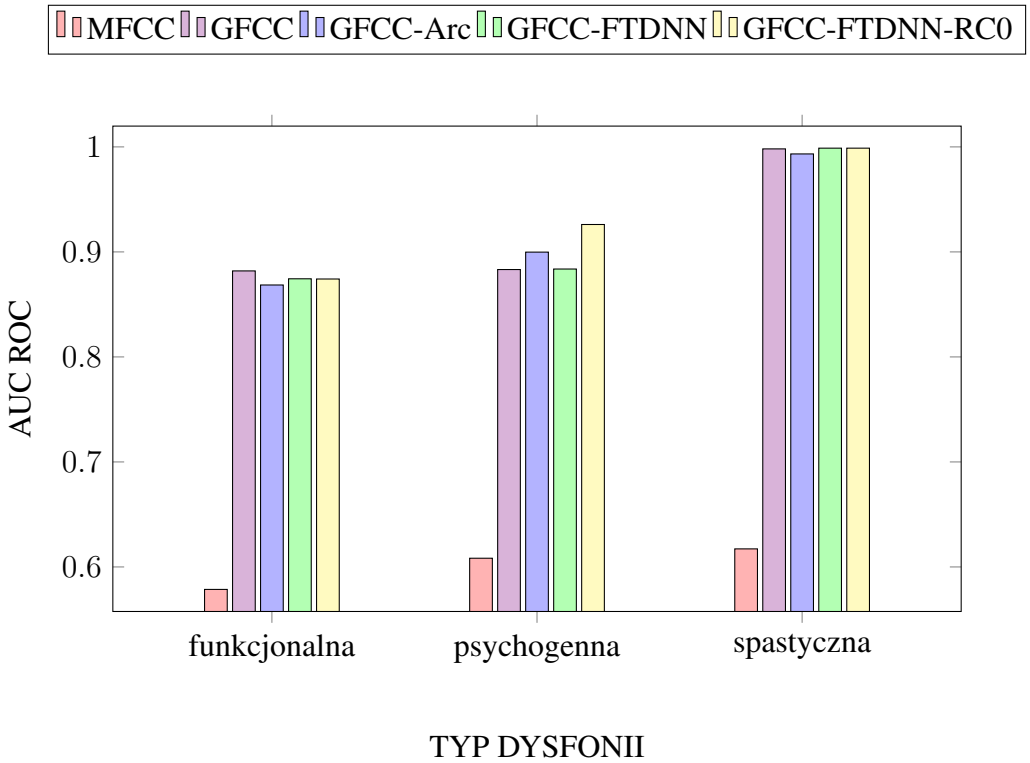
Na podstawie danych dotyczących metod redukcji wymiarowości użytych podczas treningu poszczególnych modeli, które zebrano w dodatku D (tabele D.2-D.9) można zauważyć, że większość modeli uznanych za najlepsze w rozwiązywaniu problemu detekcji danego schorzenia została nauczona na danych zredukowanych przy użyciu jednej z dwóch metod bazujących na dekompozycji macierzy. W przypadku dysfonii i obrzęku Reinkego był to algorytm tSVD (algorytm rozkładu według wartości osobliwych), zaś w przypadku pachydermii kontaktowej, dysfonii czynnościowej i dysfonii psychogennej algorytm KernelPCA (jądrowa analiza składowych głównych). W przypadku pozostałych trzech chorób – ostrego zapalenia krtani, porażenia nerwu krtaniowego wstecznego i dysfonii spastycznej – użytą metodą redukcji wymiarowości była bazująca na statystyce F uzyskanej w analizie wariancji metoda *SelectKBest* zaimplementowana w bibliotece *scikit-learn*.

Tabela 5.21. Wartości metryk sukcesu modeli o największej wartości pola pod krzywą ROC uzyskane na zbiorach testowych dla poszczególnych schorzeń otolaryngologicznych.

	Choroba					
Metryka	Dowolna	Dysfonia	Ostre zapalenie krtani	Porażenie nerwu krtaniowego wstecznego	Obrzęk Reinkego	Pachydermia kontaktowa
Dokładność	0.8237	0.7984	0.8750	0.8743	0.8889	0.8905
Czułość	0.8380	0.7069	0.7600	0.9000	0.8750	0.9000
Specyficzność	0.7952	0.8031	0.8897	0.8346	0.8898	0.8100
AUC ROC	0.8875	0.8627	0.9093	0.9254	0.9774	0.9087
Model	Komitet klasyfikatorów	NuSVM	MLP	MLP	SGD	SGD
Wariant	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-Arc	GFCC-FTDNN



Rys. 5.34. Największe wartości pola pod krzywą ROC uzyskane na zbiorach testowych dla poszczególnych schorzeń otolaryngologicznych.



Rys. 5.35. Największe wartości pola pod krzywą ROC uzyskane na zbiorach testowych dla poszczególnych typów dysfonii.

5.2.2. Detekcja zaburzeń neurorozwojowych

Detekcja zaburzeń neurorozwojowych została przeprowadzona przy użyciu dwóch korpusów mowy opisanych w rozdziale 4, sekcji 4.2 i sekcji 4.3. W przeciwieństwie do detekcji schorzeń otolaryngologicznych, detekcja zaburzeń neurorozwojowych dokonywana była na podstawie wielu nagrań zarejestrowanych dla tej samej osoby. Proces uczenia i optymalizacji modeli, a także ich wstępnej ewaluacji, prowadzone były w oparciu o pojedyncze nagrania lub segmenty. Podczas dzielenia danych na zbiory zagwarantowano przynależność wszystkich nagrań wykonanych dla danej osoby do tylko jednego zbioru: uczącego lub testowego (tabela 5.22).

Tabela 5.22. Liczebność zbiorów uczącego i testowego użytych podczas treningu i ewaluacji modeli klasyfikacyjnych wykorzystywanych do detekcji zaburzeń neurorozwojowych. SLI - specyficzne zaburzenia językowe, ASD - zaburzenia ze spoktrum autyzmu.

Zaburzenie		Zbiór	Liczba osób	Liczba segmentów
SLI	Chorzy	uczący	43	1034
		testowy	11	259
	Zdrowi	uczący	35	788
		testowy	9	200
ASD	Chorzy	uczący	37	500
		testowy	9	300
	Zdrowi	uczący	30	1285
		testowy	8	1005

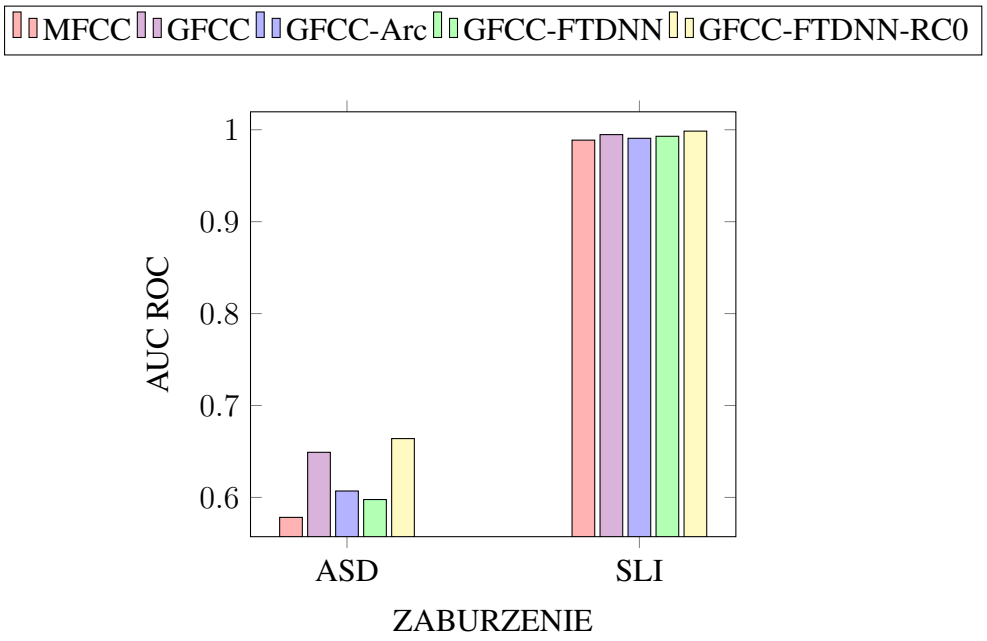
5.2.2.1. Wyniki uzyskane podczas klasyfikacji zaburzeń ze spektrum autyzmu

Podobnie jak w przypadku detekcji schorzeń krtani, trening modeli przeprowadzono używając pięciu rodzajów x-wektorów. Wartości metryki AUC ROC uzyskane przez poszczególne modele wykorzystywane do detekcji ASD zebrano w tabeli 5.23.Największe wartości pola pod krzywą ROC uzyskane na zbiorach testowych przez modele klasyfikujące pojedyncze nagrania przedstawiono na rys. 5.36.

Jak można zauważyć, uzyskane wartości pola pod krzywą ROC są stosunkowo niskie bez względu na cechy wykorzystane do uczenia sieci ekstrahującej x-wektory, jej architekturę, funkcję kosztu oraz sposób uczenia. Jednak również tym razem x-wektory ekstrahowane przy użyciu MFCC dały najgorsze rezultaty - w przypadku najlepszego modelu uczonego na tych cechach AUC ROC wynosi jedynie 0,5783.

Test DeLonga wskazuje na występowanie istotnych statystycznie różnic pomiędzy wartościami pola pod krzywą ROC uzyskanymi dla co najmniej dwóch modeli – wartość p wyniosła mniej niż $0,001$. Przedstawione na rys. 5.37 wartości pozwalają stwierdzić, że brak istotnych różnic pomiędzy wartościami metryki AUC ROC występuje tylko dla dwóch par cech: MFCC i GFCC-FTDNN oraz GFCC-Arc i GFCC-FTDNN. Wartości uzyskane dla wariantu GFCC-FTDNN-RC0 są istotnie różne od wartości uzyskanych dla wszystkich pozostałych wariantów,

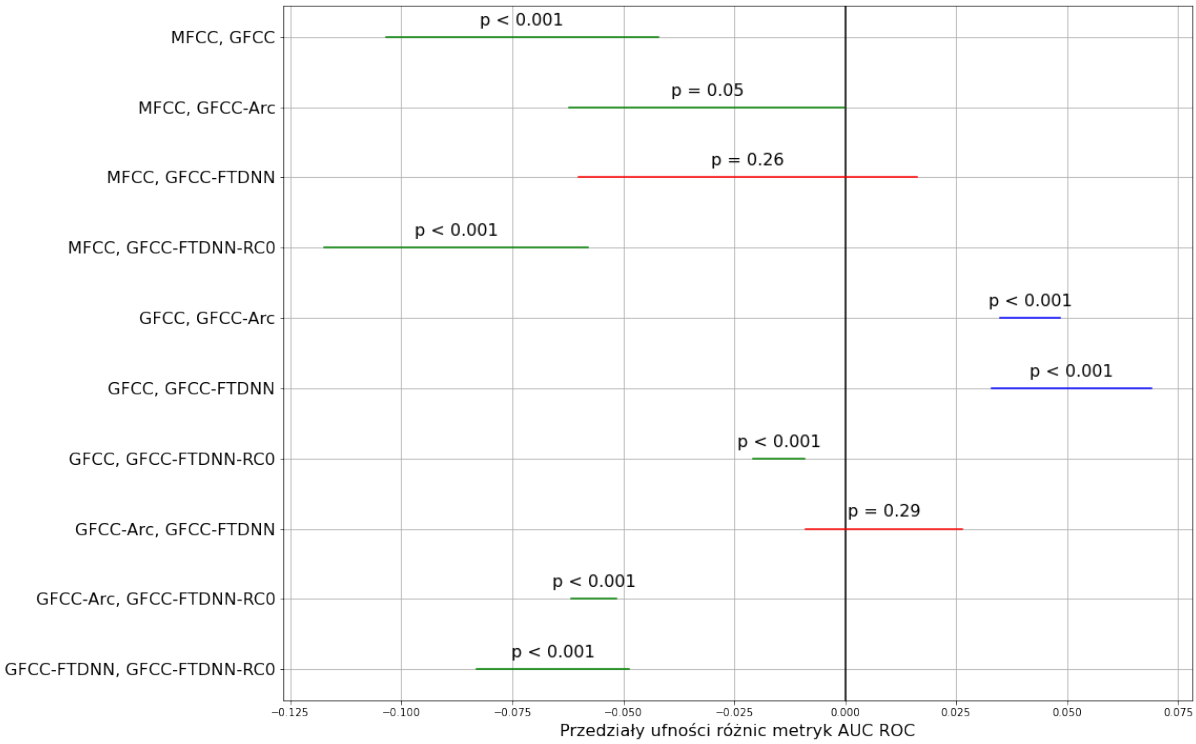
zaś ujemne wartości w przedziałach ufności różnic metryki AUC ROC pozwalają stwierdzić, że zgodnie z przyjętym kryterium model NuSVM uczony na tym wariancie może zostać uznany za najlepszy.



Rys. 5.36. Największe wartości pola pod krzywą ROC uzyskane na zbiorach testowych dla zaburzeń ze spektrum autyzmu (ASD) i specyficznych zaburzeń językowych (SLI).

Tabela 5.23. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji zaburzeń ze spektrum autyzmu.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.5010	0.5525	0.5394	0.5068	0.5425
Extra Trees	0.5179	0.5014	0.5183	0.4386	0.4590
KNN	0.5580	0.4997	0.5762	0.5655	0.5550
LDA	**0.5783**	0.5225	0.4917	0.4842	0.5542
LGB	0.5000	0.5330	0.5216	**0.5977**	0.4353
LR	0.5000	0.5000	0.5000	0.5000	0.5192
MLP	0.5360	0.5448	**0.6070**	0.5320	0.5402
NuSVM	0.5295	**0.6491**	0.5811	0.5298	**0.6640**
QDA	0.5000	0.5000	0.5208	0.5000	0.5000
RF	0.5706	0.5370	0.5283	0.4808	0.5111
RNN	0.5000	0.5000	0.5000	0.5000	0.5000
SGD	0.5561	0.5000	0.5000	0.5680	0.5413



Rys. 5.37. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji zaburzeń ze spektum autyzmu. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

Wybrany model został dodatkowo poddany treningowi na danych nad- i podpróbkowanych. W przypadku użycia danych nadpróbkowanych zaobserwowano jedynie niewielki wzrost wartości AUC ROC – wartość wzrosła z 0,6640 do 0,6672 (tabela 5.24). Ponadto różnica pomiędzy tymi wartościami nie jest istotna statystycznie, o czym świadczy wartość p uzyskana w teście DeLonga, wynosząca 0,2353. Istotne statystycznie okazały się natomiast różnice w stratach logarytmicznych uzyskiwanych przez porównywane modele – uzyskana w teście Wilcoxona wartość p wyniosła mniej niż $0,001$. Lepszymi zdolnościami generalizacyjnymi związanymi z mniejszymi wartościami strat logarytmicznych cechuje się model trenowany na danych nieprzepróbkowanych, co stwierdzono na podstawie analizy przedziału ufności różnic wynoszącego [-0,3001; -0,5825].

Zastosowanie algorytmów podpróbkowania spowodowało natomiast uzyskanie mniejszych wartości AUC ROC, niż w przypadku użycia do treningu modelu oryginalnych, nieprzepróbkowanych danych – największą wartość uzyskano przy użyciu algorytmu Tomek Links i wyniosła ona 0,6625 (tabela 5.25). Najgorsze rezultatu uzyskano stosując algorytm RENN – w tym przypadku wartość metryki AUC ROC wynosi zaledwie 0,4441.

Ponieważ test DeLonga wskazuje na brak podstaw do odrzucenia hipotezy mówiącej o równości wartości metryki AUC ROC uzyskanych przez model uczony na danych nieprzepróbkowanych oraz przepróbkowanych przy użyciu algorytmu Tomek Links (p = 0,2843), lepszy model wybrano na podstawie wartości strat logarytmicznych. Straty logarytmiczne uzyskane dla wygenerowanych prób bootstrapowych nie mają rozkładu normalnego, dlatego zdecydowano o przeprowadzeniu testu par Wilcoxona – otrzymana wartość p wynosząca 0,0014 pozwala stwierdzić, że różnica pomiędzy stratami logarytmicznymi uzyskanymi przez oba modele jest istotna statystycznie, zaś przedział ufności różnic wynoszący [-0,7570; -0,4699] wskazuje na przewagę modelu uczonego na danych nieprzepróbkowanych.

Tabela 5.24. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modelu NuSVM uczonego na danych nadpróbkowanych przy użyciu poszczególnych technik nadpróbkowania, wykorzystywanych do detekcji zaburzeń ze spektrum autyzmu.

Technika nadpróbkowania	AUC ROC
ADASYN	0.5857
Borderline SMOTE	0.5891
KMeans-SMOTE	0.6503
Random minority oversampling	**0.6672**
SMOTE	0.6108
SVM-SMOTE	0.6415
Model	NuSVM
Wariant	GFCC-FTDNN-RC0

Ostatnim etapem poszukiwań najlepszego modelu pozwalającego wykryć zaburzenia ze spektrum autyzmu był trening komitetów klasyfikatorów uczonych na poszczególnych wariantach x-wektorów. Schematy uzyskanych komitetów zamieszczono w załączniku E (rys. E.9), zaś uzyskane przez nie wyniki zestawiono w tabeli 5.26. Największe pole pod krzywą ROC uzyskano dla komitetu uczonego na wariancie GFCC – jego wartość wyniosła 0,5779. Również tym razem najgorszy rezultat osiągnął komitet uczony na wariancie MFCC – w jego przypadku wartość metryki AUC ROC jest mniejsza niż 0,5.

Aby potwierdzić, że model uczony na wariancie GFCC jest istotnie lepszy od pozostałych przeprowadzono test DeLonga. Uzyskana wartość p wynosząca mniej niż $0,001$ pozwala stwierdzić, że pomiędzy co najmniej dwoma komitetami występują istotne statystycznie różnice w wartościach metryki AUC ROC. Analizując przedziały ufności różnic w wartościach tej metryki oraz p wartości uzyskane dla poszczególnych par wariantów można zauważyć, że dla wszystkich par, w których występuje wariant GFCC uzyskane różnice są istotne statystycznie i wskazują na przewagę tego wariantu nad pozostałymi czterema wariantami (rys. 5.38).

W kolejnym kroku zbadano istotność różnic pomiędzy wartością metryki AUC ROC uzyskanej przez komitet klasyfikatorów uczony na wariancie GFCC oraz wartością uzyskaną przez

Tabela 5.25. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli NuSVM uczonych na danych podpróbkowanych przy użyciu poszczególnych technik podpróbkowania, wykorzystywanych do detekcji zaburzeń ze spektrum autyzmu.

Technika podpróbkowania	AUC ROC
AllKNN	0.6557
Cluster Centroids	0.6242
Condensed Nearest Neighbour	0.6535
Edited Nearest Neighbours	0.5501
NearMiss	0.5590
Neighbourhood Cleaning Rule	0.6359
One-Sided Selection	0.6526
Random majority undersampling	0.6248
Repeated Edited Nearest Neighbours	0.4441
Tomek links	**0.6625**
Model	NuSVM
Wariant	GFCC-FTDNN-RC0

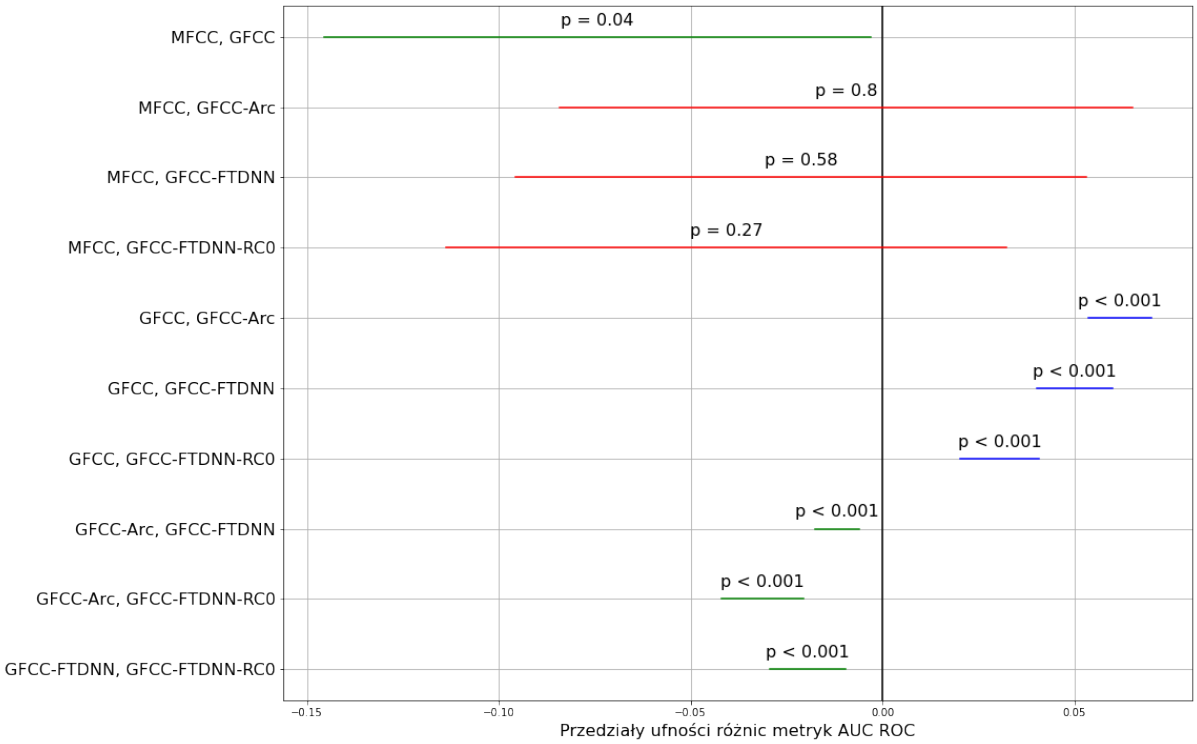
Tabela 5.26. Wartości metryki AUC ROC uzyskane na zbiorze testowym przy użyciu komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji zaburzeń ze spektrum autyzmu.

Wariant x-wektorów	AUC ROC
MFCC	0.4959
GFCC	**0.5779**
GFCC-Arc	0.5145
GFCC-FTDNN	0.5262
GFCC-FTDNN-RC0	0.5457

najlepszy model wyłoniony na wcześniejszym etapie analizy, tj. model NuSVM uczony na wariancie GFCC-FTDNN-RC0. Wartość p uzyskana w teście DeLonga wynosi mniej niż $0,001$ i wskazuje, że istnieją podstawy do odrzucenia hipotezy o równości pól pod krzywą ROC. Na podstawie wartości pól uzyskanych przez oba modele oraz przedziału ufności różnic wynoszącego [-0.1214; -0.0578] można stwierdzić, że model NuSVM zapewnia najlepsze rezultaty zgodnie z przyjętym kryterium opartym o metrykę AUC ROC.

5.2.2.2. Wyniki uzyskane podczas klasyfikacji specyficznych zaburzeń językowych

W przypadku SLI wszystkie warianty x-wektorów pozwoliły uzyskać bardzo dobre rezultaty – wartość pola pod krzywą ROC otrzymana dla modelu uczonego na MFCC wynosi 0,9887, zaś dla modeli uczonych na pozostałych cechach przekracza 0,99 (tabela 5.27). Również tym razem najmniejszą wartość uzyskano dla wariantu MFCC, jednak jest to różnica znikoma, nie



Rys. 5.38. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC komitetów modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji zaburzeń ze spektrum autyzmu. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

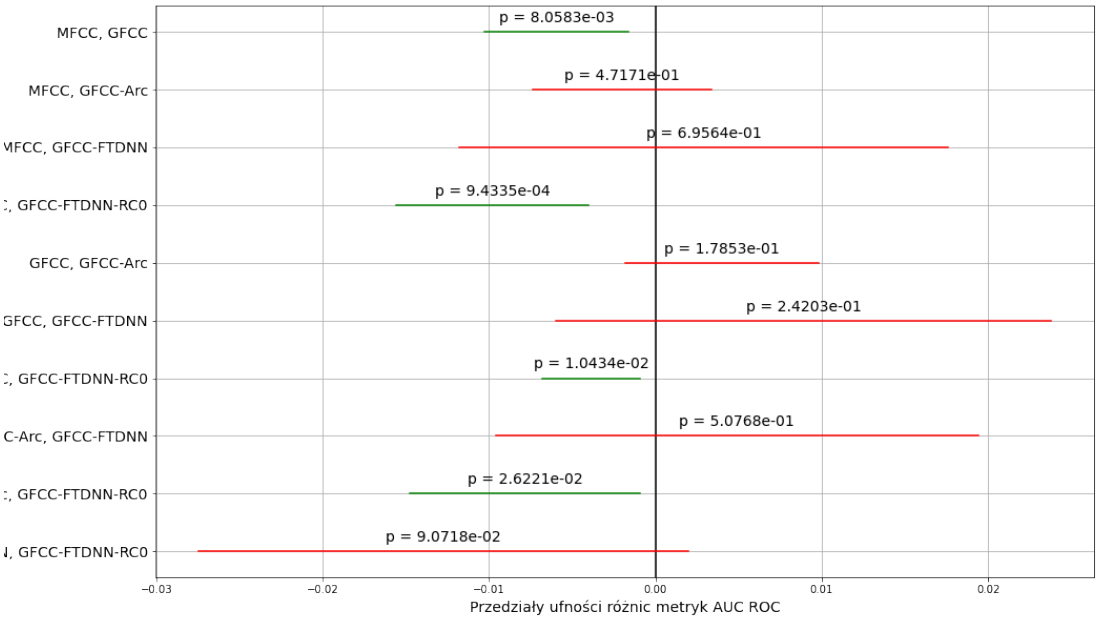
przekraczająca 0,001. Można się więc spodziewać, że różnice te nie są istotne statystycznie. Potwierdzono to przeprowadzając test DeLonga i otrzymując w jego wyniku wartość p wynoszącą 0,1341. Istotne statystycznie różnice wykazano dla czterech par wariantów x-wektorów: MFCC i GFCC, MFCC i GFCC-FTDNN-RC0, GFCC i GFCC-FTDNN-RC0 oraz GFCC-Arc i GFCC-FTDNN-RC0 (rys. 5.39). Wyniki te nie pozwalają jednak na wyłonienie najlepszego modelu, konieczne więc było przeprowadzenie analizy strat logarytmicznych modeli.

W wyniku testu Friedmana uzyskano wartość $p < 0,001$, co świadczy o występowaniu istotnych różnic pomiędzy stratami logarytmicznymi uzyskiwanymi przez co najmniej dwa modele. W celu identyfikacji tych modeli przeprowadzono test *post hoc* Nemenyiego – brak istotnych różnic w stratach logarytmicznych stwierdzono tylko dla jednej pary wariantów: MFCC i GFCC-Arc (rys. 5.40). Na podstawie analizy wartości przedziałów ufności różnic można stwierdzić, że najlepszym z porównywanych modeli jest model uczony na wariancie GFCC-FTDNN-RC0.

Ze względu na bardzo dobre rezultaty uzyskane przez modele uczone na danych nieprzepróbkowanych zrezygnowano z treningu modeli z użyciem danych przepróbkowanych oraz treningu komitetów klasyfikatorów, które miałyby służyć do detekcji specyficznych zaburzeń językowych. Biorąc pod uwagę, że różnice w wartościach metryki AUC ROC uzyskane przez

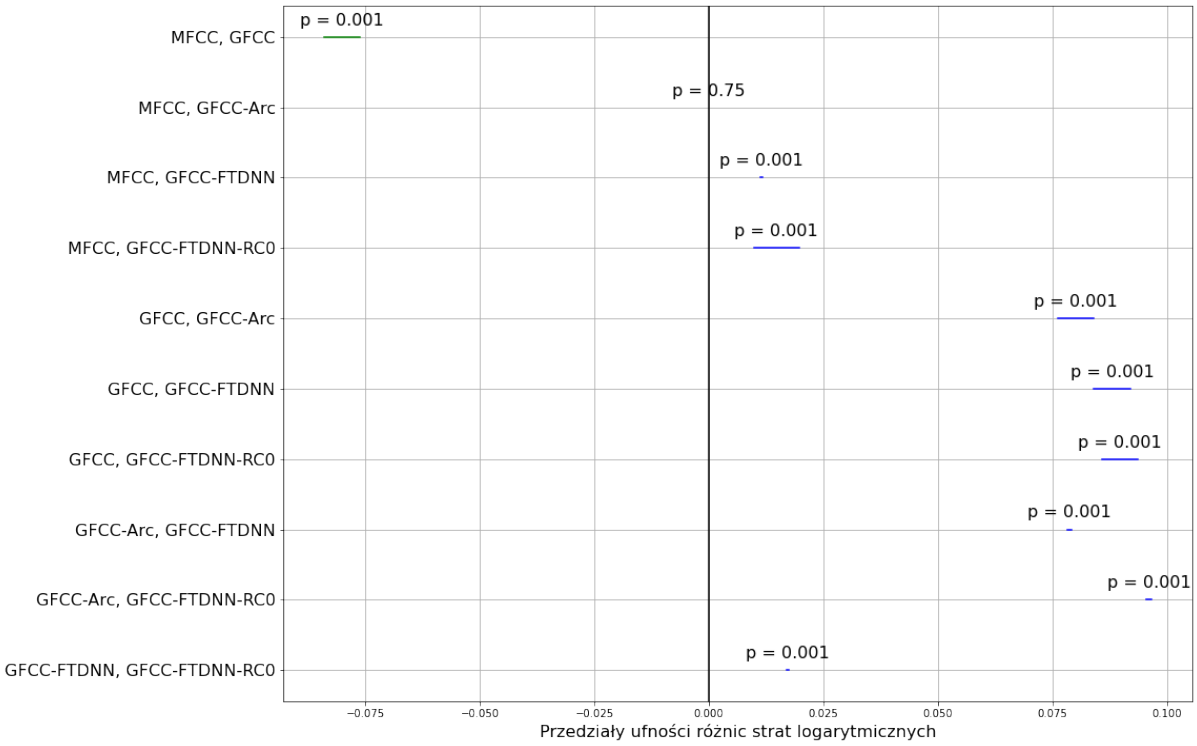
Tabela 5.27. Wartości metryki AUC ROC uzyskane na zbiorze testowym dla modeli wykorzystywanych do detekcji specyficznych zaburzeń językowych.

	AUC ROC				
Model	MFCC	GFCC	GFCC-Arc	GFCC-FTDNN	GFCC-FTDNN-RC0
AdaBoost	0.9751	0.9844	0.9806	0.9838	0.9838
Extra Trees	0.9658	0.9811	0.9528	0.9495	0.9758
KNN	0.9633	0.9831	0.9839	0.9706	0.9861
LDA	0.9138	0.9215	0.9117	0.9474	0.9510
LGB	0.9660	0.9734	0.9787	0.9684	0.9778
LR	0.9194	0.9115	0.9094	0.9288	0.9077
MLP	0.9922	**0.9947**	0.9891	**0.9929**	**0.9985**
NuSVM	0.9858	0.9940	0.9881	0.9840	0.9972
QDA	0.9426	0.8325	0.9614	0.8496	0.8999
RF	0.9281	0.9561	0.9590	0.9443	0.9588
RNN	0.9409	0.9466	0.5000	0.9397	0.5000
SGD	**0.9887**	0.9930	**0.9907**	0.9908	0.9959



Rys. 5.39. Przedziały ufności różnic oraz wartości p uzyskane w teście równości metryk AUC ROC najlepszych modeli wykorzystywanych do detekcji specyficznych zaburzeń językowych. Kolorem czerwonym oznaczono przedziały wskazujące na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość AUC ROC wariantu drugiego w parze, zaś kolorem niebieskim przedział wskazujący na istotnie większą wartość AUC ROC wariantu pierwszego w parze.

najlepsze modele trenowane na różnych wariantach x-wektorów okazały się nie być istotne statystycznie można podejrzewać, że nawet gdyby któryś z modeli uczonych na danych przepróbkowanych lub komitet modeli uzyskały AUC ROC większe niż MLP trenowany na wariancie GFCC-FTDNN-RC0, różnica pomiędzy tymi wartościami również byłaby nieistotna statystycznie. Ponadto na podstawie analiz przeprowadzonych dla schorzeń otolaryngologicznych można przypuszczać, że modele uczone na danych przepróbkowanych oraz komitety modeli uzyskałyby większe wartości straty logarytmicznej, niż model uczony na danych nieprzepróbkowanych. Można więc założyć, że trening kolejnych modeli w sposób analogiczny do postępowania podczas poszukiwania najlepszych modeli służących do wykrycia innych chorób w przypadku SLI nie doprowadziłby do poprawy rezultatów. Pominięcie tego etapu spowoduje natomiast dużą oszczędność zasobów obliczeniowych i czasowych.



Rys. 5.40. Przedziały ufności różnic oraz wartości p uzyskane w teście równości strat logarytmicznych modeli wykorzystywanych do detekcji specyficznych zaburzeń językowych, pomiędzy którymi nie stwierdzono istotnych różnic w uzyskiwanej metryce AUC ROC. Kolorem czerwonym oznaczono przedział wskaujący na brak istotnych różnic pomiędzy wariantami, kolorem zielonym przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu drugiego w parze, zaś kolorem niebieskim przedziały wskazujące na istotnie większą wartość straty logarytmicznej wariantu pierwszego w parze.

5.2.2.3. Zestawienie najlepszych wyników uzyskanych podczas klasyfikacji zaburzeń neurorozwojowych

Po wybraniu najlepszych modeli dokonujących detekcji ASD i SLI zbinaryzowano zwracane przez nie predykcje, otrzymując przynależność poszczególnych obiektów do jednej z dwóch klas. W oparciu o dychotomiczne predykcje wyznaczono wartości metryk sukcesu pozwalających na ocenę jakości klasyfikacji. Wartości otrzymane dla modelu służącego do detekcji SLI są bardzo obiecujące – czułość modelu wynosi 0,9765, zaś specyficzność 0,9929. Świadczy to o niewielkiej przewadze w klasyfikacji dzieci zdrowych, jednak różnica ta jest znikoma. Niestety w przypadku ASD uzyskane rezultaty nie są satysfakcjonujące – czułość modelu, a tym samym skuteczność detekcji ASD, wynosi zaledwie 0,6233 (tabela 5.28). Co więcej, model nie sprawdził się również w klasyfikowaniu osób zdrowych, o czym świadczy specyficzność wynosząca zaledwie 0,6350.

Dokonując prób detekcji zaburzeń neurorozwojowych z użyciem algorytmów uczenia maszynowego należy wziąć pod uwagę, że ich diagnostyka jest dużym wyzwaniem nawet dla doświadczonego specjalisty – wymaga przeprowadzenia wielu testów i długiej, wnikliwej obserwacji pacjenta. Złe wyniki uzyskane na podstawie krótkich segmentów nie są więc zaskakujące – zbyt krótkie wypowiedzi mogą nie zawierać w sobie wystarczającej ilości informacji o parametrach sygnału mowy osoby badanej. Ponieważ jednak podczas procesu diagnostycznego lekarze i psycholodzy opierają swoją decyzję m.in. na analizie wielu wypowiedzi tego samego pacjenta, postanowiono zastosować podobną strategię i sprawdzić, czy wpłynie ona na poprawę rezultatów automatycznej detekcji ASD. Przeprowadzono więc kolejną klasyfikację sygnałów z użyciem najlepszego modelu, jednak tym razem wykonaną ją w oparciu o wszystkie segmenty zawierające mowę wybranej osoby. W celu podjęcia ostatecznej decyzji o przynależności danego pacjenta do grupy osób zdrowych albo cierpiących na ASD przeanalizowano predykcje modelu dla pojedynczych segmentów zawierających mowę danej osoby – decyzja o przynależności analizowanego pacjenta do grupy osób cierpiących na ASD podejmowana była wówczas, gdy większość segmentów została zaklasyfikowana jako pochodząca od osoby ze zdiagnozowanym ASD. Takie podejście poskutkowało znaczącym wzrostem skuteczności detekcji zaburzenia i uzyskaniem czułości wynoszącej 0.7778 (tabela 5.28). Poprawę wyników zaobserwowano również w przypadku SLI – stosując taką strategię uzyskano stuprocentową dokładność klasyfikacji pacjentów.

Analizując metody redukcji wymiarowości zestawione w dodatku D (tabela D.10, tabela D.11) można zauważyć, że oba najlepsze modele wykorzystywane do detekcji zaburzeń neurorozwojowych zostały nauczone na danych zredukowanych za pomocą metod bazujących na dekompozycji macierzy. W przypadku ASD była to analiza czynnikowa (ang. *Factor Analysis*), zaś w przypadku SLI tSVD (algorytm rozkładu według wartości osobliwych).

Tabela 5.28. Wartości metryk sukcesu modeli o największej wartości pola pod krzywą ROC uzyskane na zbiorach testowych dla zaburzeń ze spektrum autyzmu i specyficznych zaburzeń językowych.

	Wyniki uzyskane dla pojedynczych segmentów		Wyniki uzyskane na podstawie równoczesnej analizy wszystkich segmentów	
Metryka	Zaburzenia ze spektrum autyzmu	Specyficzne zaburzenia językowe	Zaburzenia ze spektrum autyzmu	Specyficzne zaburzenia językowe
Dokładność	0.6280	0.9813	0.7778	1.0000
Czułość	0.6233	0.9765	0.7000	1.0000
Specyficzność	0.6350	0.9929	0.8750	1.0000
Model	NuSVM	MLP	NuSVM	MLP
Wariant	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0

6. Dyskusja

Przedstawione w pracy rezultaty można uznać za obiecujące - skuteczna detekcja zarówno zaburzeń neurorozwojowych, jak i wybranych schorzeń krtani na podstawie x-wektorów ekstrahowanych z mowy ciągłej jest możliwa. Oczywistym ograniczeniem prowadzonych badań powodującym ograniczoną możliwość potencjalnego wdrożenia proponowanej metody do praktyki klinicznej jest niewielki rozmiar wykorzystanych korpusów mowy. Przedstawione wyniki należy więc traktować jako badania wstępne, prezentujące możliwości zastosowania wektorów własnościowych mówcy do diagnostyki problemów medycznych należących do różnej grupy schorzeń mających wpływ na produkcję mowy. Koniecznym kolejnym krokiem jest przetestowanie proponowanego schematu postępowania na większej liczbie danych, w miarę możliwości zawierających mowę w różnych językach, co pozwoli lepiej oszacować stopień niezależności opisanej metody od języka i zdolność generalizacyjną zaproponowanych w pracy nowych wariantów x-wektorów.

6.1. Ocena modeli i ich porównanie z modelami opisanymi w literaturze

Istotnym elementem oceny nowych metod diagnostycznych jest ich porównanie z metodami już istniejącymi i opisanymi w literaturze. Zadanie to jest utrudnione ze względu na brak ogólnie przyjętego jednolitego sposobu oceny modeli oraz raportowania osiąganych przez nie rezultatów. Wyniki uzyskane przez opisywane w literaturze algorytmy służące do diagnostyki tych samych schorzeń, które analizowano w niniejszej pracy często oceniane były za pomocą sprawdzianu krzyżowego (rozdział 3). Jest to strategia powszechnie uważana za prawidłową i najlepszą w przypadku niewielkich zbiorów danych. Równocześnie dostępne są badania wskazujące na to, że takie podejście może doprowadzić do zaniżenia rzeczywistego błędu predykcji [139], [140]. Z tego względu w niniejszej pracy zdecydowano się prowadzić proces optymalizacji modeli z zastosowaniem sprawdzianu krzyżowego, jednak ostateczny wybór najlepszego modelu oraz ocenę możliwości proponowanego rozwiązania przeprowadzono w oparciu o wyodrębniony zbiór testowy. Miarodajność uzyskanych w ten sposób wyników zagwarantowano prowadząc obszerną analizę ich istotności statystycznej oraz stosując ściśle określony schemat oceny modeli, opracowany na podstawie zaleceń dotyczących oceny modeli predykcyjnych stosowanych w medycynie i dostępnych w literaturze. Jednakże ze względu na różnice w ocenie

modeli zastosowanej w niniejszej pracy oraz w badaniach opisanych w literaturze, porównanie proponowanej metody z metodami już istniejącymi jest utrudnione.

Kolejną problematyczną kwestią utrudniającą porównanie opisanych w pracy rezultatów z wynikami osiąganymi przez już istniejące i stosowane metody jest fakt, że część autorów oceniała proponowane przez siebie rozwiązania z użyciem metryki określanej jako *średnia czułość modelu*. Średnia czułość wyznaczana jest w praktyce jako średnia arytmetyczna czułości i specyficzności modelu – bez dostępu do obu tych wartości trudno jest więc stwierdzić, jaka jest faktyczna zdolność modelu do detekcji osób zdrowych i chorych. Co więcej, w niektórych przypadkach stosowana była średnia ważona zamiast arytmetycznej, co w przypadku dużego niezrównoważenia zbiorów danych może znacząco zaburzyć raportowane wyniki.

Aby uzasadnić przyjęte w pracy stanowisko, że zastosowanie sprawdzianu krzyżowego może mieć negatywny wpływ na wiarygodność wyników, postanowiono dodatkowo wyznaczyć i porównać wartości metryk sukcesu osiągnięte dla podzbiorów walidacyjnych podczas sprawdzianu krzyżowego w sposób analogiczny do opisywanego w literaturze. Wyniki te zestawiono w tabeli 6.1. Porównując te wartości z wartościami uzyskanymi z użyciem zbioru testowego, które przedstawiono w tabeli 5.28 można zauważyć, że zgodnie z przypuszczeniami autorki zastosowanie sprawdzianu krzyżowego spowodowało uzyskanie wyników nadmiernie optymistycznych. Szczególnie widoczne jest to w przypadku zaburzeń ze spektrum autyzmu – wartości uzyskane dla podzbiorów walidacyjnych są wyższe o ponad 20 punktów procentowych i w przypadku klasyfikacji opartej o wszystkie wypowiedzi danego pacjenta pozwalają uzyskać 100% skuteczność. Biorąc pod uwagę, jak trudnym zagadnieniem jest diagnostyka zaburzeń ze spektrum autyzmu oraz rozmiar i jakość użytego w tym celu korpusu mowy, wynik ten można uznać za pozbawiony wiarygodności.

Pomimo przyjęcia innego schematu oceny modeli, postanowiono jednak odnieść się do wyników prezentowanych w literaturze i porównać je z wynikami opisanymi w niniejszej pracy. W przypadku wielu rozwiązań opisanych w rozdziale 3, sekcji 3.1 wykorzystany był ten sam korpus mowy, którego użyto w pracy, co umożliwia przeprowadzenie dość wiarygodnej analizy porównawczej wyników.

Zaprezentowana w pracy metoda pozwoliła na uzyskanie większej dokładności klasyfikacji niż opisane w literaturze metody służące do detekcji dysfonii, ostrego zapalenia krtani oraz porażenia nerwu krtaniowego wstecznego (tabela 3.1, tabela 5.21, tabela 5.20). Lepsze rezultaty osiągnięto również w przypadku dysfonii spastycznej, jednak w tym przypadku należy zachować ostrożność podczas porównywania wyników – jak wspomniano w rozdziale 5, sekcji 5.2.1.3 dysfonia spastyczna może odnosić się do schorzeń o różnej patogenezie i nie jest jasne, w jakim znaczeniu użyto tego określenia w [100]. Istotną niedogodnością jest również fakt, że nie jest możliwe sprawdzenie istotności statystycznej różnic pomiędzy wartościami metryk sukcesu osiągniętymi przez modele opisywane w niniejszej pracy oraz modele opisywane w literaturze z powodu braku dostępu do tych modeli. Porównanie pojedynczych wartości metryk sugeruje przewagę modeli uczonych na x-wektorach zaproponowanych w pracy,

Tabela 6.1. Wartości metryk sukcesu modeli o największej wartości pola pod krzywą ROC uzyskane na podzbiorach walidacyjnych dla zaburzeń ze spektrum autyzmu i specyficznych zaburzeń językowych.

	Wyniki uzyskane dla pojedynczych segmentów		Wyniki uzyskane na podstawie równoczesnej analizy wszystkich segmentów	
Metryka	Zaburzenia ze spektrum autyzmu	Specyficzne zaburzenia językowe	Zaburzenia ze spektrum autyzmu	Specyficzne zaburzenia językowe
Dokładność	0.9120	1.0000	1.0000	1.0000
Czułość	0.8448	1.0000	1.0000	1.0000
Specyficzność	0.9987	1.0000	1.0000	1.0000
AUC ROC	0.8783	1.0000	1.0000	1.0000
Model	NuSVM	MLP	NuSVM	MLP
Wariant	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0	GFCC-FTDNN-RC0

jednak należy mieć na uwadze fakt, że przewaga ta mogłaby okazać się nieistotna z punktu widzenia statystyki.

Porównanie wyników uzyskanych podczas detekcji specyficznych zaburzeń językowych również wskazuje na przewagę metody opisanej w niniejszej pracy – pozwoliła ona na uzyskanie 100% dokładności klasyfikacji sygnałów (tabela 5.28). Większość rozwiązań opisanych w literaturze pozwoliła na uzyskanie dokładności nieprzekraczającej 86% pomimo wykorzystywania tego samego korpusu mowy (tabela 3.3). Wyniki takie same lub zbliżone do rezultatów przedstawionych w niniejszej pracy osiągnięto jedynie w przypadku dwóch metod, w tym jednej opracowanej przez autorkę niniejszej pracy.

Najbardziej niemiarodajne i budzące najwięcej wątpliwości wydaje się być porównanie wyników uzyskanych przez modele służące do detekcji zaburzeń ze spektrum autyzmu. Przede wszystkim ASD obejmuje bardzo szerokie spektrum objawów o różnym nasileniu – oczywistym wydaje się założenie, że detekcja zaburzenia u osób wysokofunkcjonujących, z dobrze rozwiniętą mową, wykazujących jedynie stosunkowo nisko rozwinięte kompetencje społeczne [197] będzie zagadnieniem dużo trudniejszym, niż w przypadku osób o bardziej nasilonych objawach. Zarówno badania opisane w literaturze, jak i przedstawione w niniejszej pracy wykorzystywały różne korpusy mowy – w przypadku rozwiązań opisanych w literaturze były to korpusy tworzone przez autorów badań, niedostępne dla innych badaczy. Ponadto wszystkie opisywane korpusy mowy dzieci ze zdiagnozowanym ASD cechują się bardzo małą liczebnością, co poddaje w wątpliwość istotność i wiarygodność uzyskanych na ich podstawie wyników [198]. Spośród sześciu opisywanych w literaturze metod automatycznej diagnostyki ASD tylko dwie pozwoliły na uzyskanie większej dokładności klasyfikacji niż przedstawiona w niniejszej pracy (tabela 3.2, tabela 5.28). Należy mieć jednak na uwadze fakt, że w badaniach tych użyto zrównoważonych zbiorów danych, co uczyniło rozwiązywany problem klasyfikacyjny łatwiejszym.

6.2. Strategia doboru wartości progowej używanej podczas binaryzacji predykcji ciągłych

Kwestią budzącą wątpliwości może być również metoda doboru wartości progowej używanej podczas binaryzacji predykcji ciągłych zwracanych przez model. Zastosowana w niniejszej pracy metoda oparta jest o maksymalizację iloczynu skalarnego czułości i specyficzności modelu. Tymczasem optymalizacja hiperparametrów modeli predykcyjnych stosowanych w medycynie często prowadzona jest w taki sposób, by maksymalizować czułość modelu kosztem wzrostu liczby predykcji fałszywie pozytywnych – ma to na celu zapewnienie jak największej wykrywalności chorób. Podejście takie wynika z założenia, że zasygnalizowanie zwiększonego ryzyka występowania danej choroby u pacjenta zwróci uwagę lekarza, który następnie zweryfikuje prawdziwość informacji zwracanej przez system. Predykcje fałszywie negatywne uznawane są za bardziej niepożądane i mogą nieść większe konsekwencje zdrowotne dla pacjenta. W wielu przypadkach jest to strategia uzasadniona – w przypadku schorzeń onkologicznych

szczególnie istotne jest wczesne wykrycie zmian nowotworowych, dzięki czemu leczenie może być bardziej skuteczne i mniej obciążające dla pacjenta. Badania opisywane w literaturze wykazały, że zarówno pacjenci, jak i lekarze wykazują tendencję do akceptacji znacznego wzrostu liczby diagnoz fałszywie pozytywnych, jeżeli wiąże się to z większą czułością stosowanej metody diagnostycznej [199]. Konieczność przejścia niekomfortowych lub bolesnych procedur medycznych jest uznawana za koszt wart poniesienia w sytuacji, gdy może to zminimalizować ryzyko śmierci lub trwałego okaleczenia w wyniku zbyt późnego rozpoznania choroby [200]. Jednakże nawet w przypadku chorób nowotworowych maksymalizacja czułości metod diagnostycznych nie jest jednogłośnie zalecana – jak doniesiono w [201], fałszywie pozytywny wynik badania mammograficznego wiązał się z wieloletnimi negatywnymi skutkami psychologicznymi pacjentek. Decyzja o wyborze maksymalizowanej metryki – czułości lub specyficzności – powinna być zatem przemyślana i oparta o analizę skutków zarówno predykcji fałszywie pozytywnych, jak i fałszywie negatywnych [202].

W przypadku zaburzeń neurorozwojowych, w szczególności zaburzeń ze spektrum autyzmu, fałszywie pozytywna diagnoza może mieć negatywne konsekwencje związane ze stygmatyzacją zarówno osób zaburzonych, jak i ich rodzin. W literaturze dostępne są badania wskazujące na istotnie niższy dobrostan subiektywny opiekunów osób ze zdiagnozowanym ASD niż osób sprawujących opiekę nad członkiem rodziny z niepełnosprawnością intelektualną lub fizyczną [203]. Ponadto spośród tych trzech grup tylko opiekunowie osób z ASD cechują się istotnie niższym raportowanym wskaźnikiem dobrostanu subiektywnego niż średni w populacji krajów zachodnich. Według autorów ma to związek z częstszym występowaniem autostygmatyzacji u opiekunów osób z ASD, niż ma to miejsce w przypadku pozostałych badanych grup. Zgodnie z tezą przedstawioną w [204] może to wynikać z przekonania rodziców, w szczególności matek, że są oni odpowiedzialni za wystąpienie u dziecka zaburzeń ze spektrum autyzmu. Tendencja do obwiniania rodziców za deficyty i zaburzenia dziecka jest szczególnie wyraźna w kulturze azjatyckiej – część społeczeństwa koreańskiego uznaje problemy rozwojowe dziecka za skutek zaniedbań matki już w trakcie ciąży [205]. Równocześnie ASD jako zaburzenie uwarunkowane genetycznie powoduje stygmatyzację całej rodziny w przypadku jego wystąpienia u jednego z jej członków [206]. Podobne podejście społeczeństwa do osób z diagnozą ASD i ich rodzin można zaobserwować w Japonii – zarówno rodzice, jak i nauczyciele dzieci ze zdiagnozowanymi zaburzeniami rozwojowymi często wyrażają wątpliwość, czy diagnoza dziecka nie spowoduje jego wykluczenia z grupy rówieśniczej, co może skutkować dalszym pogorszeniem jego umiejętności komunikacyjnych i społecznych. Co więcej, podobne reakcje względem osób z ASD i ich rodzin obserwowane są nie tylko w krajach Azji Wschodniej, ale też wśród imigrantów pochodzenia azjatyckiego w krajach europejskich czy Stanach Zjednoczonych [207]. Negatywny stosunek do diagnozy ASD można zauważyć też wśród samych zdiagnozowanych osób. Zmiany w klasyfikacji zaburzeń psychicznych Amerykańskiego Towarzystwa Psychiatrycznego DSM-5 łączące zaburzenie autystyczne, zespół Aspergera, dziecięce zaburzenie dezintegracyjne, zespół Retta oraz inne całościowe zaburzenia rozwojowe nieujęte w pozostałych

kategoriach diagnostycznych w jedną jednostkę diagnostyczną – zaburzenia ze spektrum autyzmu – spotkały się z negatywnym odbiorem niemal połowy ankietowanych osób. Było to spowodowane faktem, że zespół Aspergera miał mniej negatywne konotacje niż zaburzenia autystyczne, więc scalenie tych jednostek diagnostycznych wywołało obawy o utrudnienie procesu diagnostycznego związane z niechęcią rodziców do przyjęcia diagnozy zawierającej w sobie słowo *autyzm* [208]. Opisane problemy mogą przemawiać za koniecznością ograniczenia liczby diagnoz fałszywie pozytywnych.

Z drugiej strony w literaturze dostępne są również opracowania wskazujące na duże korzyści z wczesnej diagnostyki ASD. Jak wskazano w [209], uzyskanie diagnozy może mieć pozytywny wpływ na dobrostan psychiczny rodziców dzieci z ASD, ponieważ umożliwia lepsze zrozumienie mechanizmów kierujących zachowaniem ich dzieci oraz uzyskanie wsparcia, zarówno medycznego, jak i edukacyjnego oraz finansowego. Na uwagę zasługuje również fakt, że wczesne wykrycie ASD wiąże się z możliwością wczesnego wdrożenia odpowiedniej terapii, co znacząco poprawia jakość życia osób zdiagnozowanych [210]. Co więcej, rosnąca liczba osób zdiagnozowanych w społeczeństwie powoduje wzrost świadomości społecznej dotyczącej zaburzeń ze spektrum autyzmu, co rówież ma pozytywny wpływ na odbiór osób zdiagnozowanych i jakość ich życia [211]. Argumenty te przemawiają na korzyść maksymalizacji czułości narzędzi wykorzystywanych do diagnostyki ASD.

Ze względu na opisaną powyżej niejednoznaczność wytycznych dostępnych w literaturze fachowej oraz brak zgodności zarówno w środowisku lekarskim, jak i wśród osób z ASD oraz ich opiekunów w kwestii konsekwecji błędnej diagnozy oraz braku diagnozy, zdecydowano się na użycie metody uwzględniającej równocześnie czułość i specyficzność modelu. Uznano to za kompromis pozwalający zminimalizować negatywne skutki zarówno predykcji fałszywie negatywnych, jak i predykcji fałszywie pozytywnych.

7. Podsumowanie

Celem pracy było zbadanie możliwości przeprowadzenia detekcji wybranych schorzeń otolaryngologicznych oraz zaburzeń neurorozwojowych na podstawie x-wektorów – biometrycznych wektorów charakteryzujących mówcę ekstrahowanych z sygnału mowy. Cel pracy został osiągnięty – zbudowano modele klasyfikacyjne dokonujące detekcji wybranych schorzeń z dokładnością zbliżoną bądź przewyższającą wyniki opisywane w literaturze. Tym samym udowodniono tezę pracy mówiącą o tym, że x-wektory pozwalają na detekcję wybranych schorzeń krtani oraz zaburzeń neurorozwojowych z większą skutecznością, niż rozwiązania opisane w literaturze.

Przeprowadzone prace badawcze podzielone zostały na kilka etapów. W pierwszym etapie przeprowadzono klasyfikację sygnałów mowy ciągłej zawierającej mowę osób zdrowych oraz cierpiących na szereg chorób otolaryngologicznych, zawartych w korpusie mowy niemieckiej *Saarbrücken Voice Database*. Celem tego etapu było sprawdzenie, czy klasyfikacja przeprowadzona na podstawie x-wektorów będzie skuteczna oraz określenie, który rodzaj cech ekstrahowanych z sygnału mowy służących do wyznaczenia x-wektorów – współczynniki MFCC czy GFCC – pozwolą na uzyskanie większych wartości pola pod krzywą ROC (AUC ROC). Wykazano, że oba warianty x-wektorów umożliwiają skuteczną detekcję osób chorych, jednakże wariant ekstrahowany ze współczynników GFCC zapewnia lepsze rezultaty.

W kolejnym etapie pracy wprowadzono modyfikacje w sieci neuronowej wykorzystywanej do ekstrakcji x-wektorów na podstawie współczynników GFCC. Zaproponowano trzy warianty:

- wariant ekstrahowany z użyciem sieci, w której jako funkcję kosztu wykorzystano funkcję *ArcFace*, oznaczany jako GFCC-Arc,
- wariant ekstrahowany z użyciem sieci, w której warstwę z opóźnieniem czasowym (TDNN) zamieniono na warstwy faktoryzowane z opóźnieniem czasowym (FTDNN), oznaczany jako GFCC-FTDNN,
- wariant, w którym podczas treningu sieci prowadzono augmentację danych poprzez usunięcie losowego kanału współczynników GFCC, oznaczany jako GFCC-FTDNN-RC0.

Modyfikacje te wprowadzano w taki sposób, że każda kolejna modyfikacja dotyczyła sieci utworzonej z użyciem modyfikacji wcześniejszej, tzn. wariant GFCC-FTDNN wykorzystywał funkcję kosztu *ArcFace*, a wariant GFCC-FTDNN-RC0 składał się z warstw faktoryzowanych oraz wykorzystywał tę funkcję kosztu.

Zaproponowane pięć wariantów x-wektorów – x-wektory ekstrahowane ze współczynników MFCC, GFCC oraz wspomniane wyżej trzy warianty uzyskane poprzez modyfikacje sieci neuronowej ekstrahującej x-wektory – zostały wykorzystane do treningu dwunastu modeli klasyfikacyjnych należących do różnych grup klasyfikatorów. Ponadto w procesie optymalizacji zastosowano osiem technik redukcji wymiarowości, aby dobrać połączenie techniki redukcji wymiarowości oraz modelu klasyfikacyjnego gwarantujące uzyskanie najlepszych rezultatów. Uzyskane wyniki porównano przy użyciu odpowiednich testów statystycznych – testu DeLonga w przypadku porównywania metryki AUC ROC oraz testu Friedmana w przypadku porównywania strat logarytmicznych – w celu badania istotności statystycznej różnic pomiędzy wartościami uzyskanymi przez poszczególne modele oraz wyłonienia rozwiązania prowadzącego do uzyskania najlepszych wyników.

Następnie zastosowano opisane powyżej podejście do przeprowadzenia detekcji pięciu schorzeń otolaryngologicznych:

- dysfonii (bez podziału na typy oraz z podziałem na trzy typy: czynnościową, psychogenną oraz spastyczną),
- ostrego zapalenia krtani,
- porażenia nerwu krtaniowego wstecznego,
- obrzęku Reinkego,
- pachydermii kontaktowej.

Kolejnym krokiem było zastosowanie technik przepróbkowania zbioru danych w celu zmniejszenia niezrównoważenia użytego korpusu mowy. Zastosowano sześć metod nadpróbkowania zbioru danych oraz dziesięć metod podpróbkowania zbioru danych. W celu zmniejszenia kosztów obliczeniowych treningowi przy użyciu zbiorów przepróbkowanych poddano wyłącznie najlepsze modele wyłonione podczas poprzedniego etapu prac. Spośród uzyskanych modeli wyłoniono te dające najbardziej obiecujące wyniki, ponownie przeprowadzając testy DeLonga oraz Friedmana. Następnie dokonano porównania rezultatów uzyskiwanych przez modele uczone na zbiorach nieprzepróbkowanych oraz najlepsze modele uczone na zbiorach przepróbkowanych. Ich rezultaty porównano za pomocą testu DeLonga oraz testu Wilcoxona, które umożliwiają porównanie wyników uzyskiwanych przez dwa modele.

Następnie przeprowadzono trening komitetów klasyfikatorów o architekturze optymalizowanej genetycznie. Uzyskane w ten sposób komitety porównano z najlepszymi modelami prostymi uczonymi na poszczególnych wariantach x-wektorów, które na drodze analiz prowadzonych w poprzednich etapach uznano za najlepsze. Porównania ponownie dokonano przy użyciu testu DeLonga oraz testu Wilcoxona.

W ostatnim etapie pracy przeprowadzono analizy analogiczne do opisanych powyżej dla sygnałów zawartych w dwóch korpusach mowy dziecięcej: korpusu mowy czeskiej zawierającego mowę dzieci zdrowych oraz ze zdiagnozowanymi specyficznymi zaburzeniami językowymi oraz korpusu mowy niderlandzkiej zawierającego mowę dzieci zdrowych oraz ze zdiagnozowanymi zaburzeniami ze spektrum autyzmu.

Przeprowadzone analizy pozwoliły wyłonić algorytmy wstępnego przetwarzania sygnału mowy, redukcji wymiarowości oraz klasyfikacji obiektów gwarantujące najlepsze rezultaty w detekcji wybranych schorzeń i zaburzeń ze względu na przyjęte kryteria: wartość pola pod krzywą ROC oraz wartość straty logarytmicznej. We wszystkich przypadkach lepsze rezultaty uzyskano przy użyciu wariantów x-wektorów bazujących na współczynnikach GFCC. W przypadku dysfonii czynnościowej oraz spastycznej był to wariant bazujący na GFCC, w przypadku dysfonii psychogennej oraz obrzęku Reinkego wariant GFCC-Arc, w przypadku pachydermii kontaktowej wariant GFCC-FTDNN, zaś w przypadku pozostałych schorzeń i zaburzeń wariant GFCC-FTDNN-RC0.

Przedstawione rozwiązania umożliwiają detekcję wybranych schorzeń krtani oraz zaburzeń neurorozwojowych z dokładnością zbliżoną do opisywanej w literaturze bądź ją przewyższającą. Pomimo znaczących różnic w metodach ewaluacji modeli klasyfikacyjnych zastosowanych w niniejszej pracy oraz w pracach innych badaczy można uznać, że przedstawione w niniejszej rozprawie wyniki są obiecujące i wskazują na przewagę proponowanego rozwiązania nad dotychczas stosowanymi metodami. Proponowane podejście pozwoliło na osiągnięcie dokładności klasyfikacji oraz metryki AUC ROC wynoszących odpowiednio:

- 82,37% i 0,8875 w przypadku detekcji osób chorych na schorzenie krtani,
- 79,84% i 0,8627 w przypadku detekcji dysfonii,
- 81,33% i 0,8819 w przypadku detekcji dysfonii czynnościowej,
- 80,69% i 0,8998 w przypadku detekcji dysfonii psychogennej,
- 97,14% i 0,9981 w przypadku detekcji dysfonii spastycznej,
- 87,50% i 0,9093 w przypadku detekcji ostrego zapalenia krtani,
- 87,43% i 0,9254 w przypadku detekcji porażenia nerwu krtaniowego wstecznego,
- 88,89% i 0,9774 w przypadku detekcji obrzęku Reinkego,
- 89,05% i 0,9087 w przypadku detekcji pachydermii kontaktowej,
- 77,78% i 0,8750 w przypadku detekcji zaburzeń ze spektrum autyzmu,
- 100,00% i 1,0000 w przypadku detekcji specyficznych zaburzeń językowych.

Warto podkreślić, że w literaturze nie zostały dotychczas opisane inne algorytmy wykorzystujące x-wektory do detekcji schorzeń krtani oraz zaburzeń neurorozwojowych – proponowane w niniejszej pracy rozwiązanie można więc uznać za innowacyjne. Ponadto przedstawiony w pracy model klasyfikacyjny dokonujący detekcji pachydermii kontaktowej jest pierwszym opisywanym rozwiązaniem umożliwiającym automatyczną detekcję tego schorzenia. Prowadzone analizy zostały udokumentowane w sposób umożliwiający odtworzenie uzyskanych wyników.

Opisane rezultaty należy uznać za obiecujące, jednakże na tym etapie badań wdrożenie proponowanego rozwiązania do praktyki klinicznej nie jest możliwe. Należy zwrócić uwagę, że wykorzystane korpusy mowy są niewielkie, co może wpływać na zmniejszoną reprezentatywność uzyskanych wyników. W celu potwierdzenia skuteczności proponowanego rozwiązania należy przeprowadzić dalsze badania, wykorzystujące większą liczbę sygnałów. Istotnym elementem dalszych prac badawczych będzie również zbadanie możliwości przeprowadzenia klasyfikacji wieloklasowej, która umożliwi nie tylko detekcję poszczególnych schorzeń i odróżnianie osób chorych od zdrowych, ale też różnicowanie stopnia zaawansowania choroby.

Bibliografia

[1] V. Berisha, P. Krantsevich Ch.and Hahn, S. Hahn i in., „Digital medicine and the curse of dimensionality," *NPJ Digital Medicine*, t. 4, 2021.

[2] I. Goodfellow, Y. Bengio i A. Courville, *Deep learning: systemy uczące się*, 1 wyd. Warszawa: PWN, 2018.

[3] A. Hassanat, M. Abbadi, G. Altarawneh i in., „Solving the problem of the K parameter in the KNN classifier using an ensemble learning approach," *International Journal of Computer Science and Information Security*, t. 12, s. 33–39, 2014.

[4] M. Muja i D. G. Lowe, „Scalable nearest neighbor algorithms for high dimensional data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 36, nr. 11, s. 2227–2240, 2014.

[5] H. Abu Alfeilat, A. Hassanat, O. Lasassmeh i in., „Effects of distance measure Choice on k-nearest neighbor classifier performance: a review," *Big Data*, t. 7, 2019.

[6] G.-X. Yuan, C.-H. Ho i C.-J. Lin, „Recent advances of large-scale linear classification," *Proceedings of the IEEE*, t. 100, nr. 9, s. 2584–2603, 2012.

[7] J. C. Stoltzfus, „Logistic regression: a brief primer," *Academic Emergency Medicine*, t. 18, nr. 10, s. 1099–1104, 2011.

[8] S. Ruder, „An overview of gradient descent optimization algorithms," *arXiv*, 2016.

[9] N. Ketkar, *Deep learning with Python: a hands-on introduction*. Berkeley: Apress, 2017, rozd. 8, s. 113–132.

[10] „Chapter 2 - Brain–computer interfaces and their applications," w *An industrial IoT approach for pharmaceutical industry growth*, V. E. Balas, V. K. Solanki i R. Kumar, red., Academic Press, 2020, s. 31–54.

[11] F. Pedregosa i in., „Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, t. 12, s. 2825–2830, 2011.

[12] S. Srivastava, M. Gupta i A. Frigyik, „Bayesian quadratic discriminant analysis," *Journal of Machine Learning Research*, t. 8, s. 1277–1305, 2007.

[13] A. Rabinowicz i S. Rosset, „Trees-based models for correlated data," *Journal of Machine Learning Research*, t. 23, s. 1–31, 2022.

[14] V. Kulkarni i P. Sinha, „Pruning of random forest classifiers: a survey and future directions," w *International Conference on Data Science Engineering (ICDSE)*, 2012, s. 64–68.

[15] M. Belgiu i L. Drăguţ, „Random forest in remote sensing: a review of applications and future directions," *ISPRS Journal of Photogrammetry and Remote Sensing*, t. 114, s. 24–31, 2016.

[16] G. Wang i J. Xu, „A decision tree algorithm based on coordination degree and Gini-coefficient," w *Proceedings of 2008 International Conference on MultiMedia and Information Technology, MMIT 2008*, 2008.

[17] S. Nowozin, „Improved information gain estimates for decision tree induction," *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, t. 1, 2012.

[18] P. Geurts, D. Ernst i L. Wehenkel, „Extremely randomized trees," *Machine Learning*, t. 63, s. 3–42, 2006.

[19] M. Götz, C. Weber, J. Blöcher i in., „Extremely randomized trees based brain tumor segmentation," w *Proceedings of MICCAI 2014 Brain Tumor Segmentation Challenge*, 2014.

[20] Y. Cao, Q.-G. Miao, J.-C. Liu i in., „Advance and prospects of AdaBoost algorithm," *Acta Automatica Sinica*, t. 39, 745–758, 2013.

[21] G. Ke, Q. Meng, T. Finley i in., „LightGBM: a highly efficient gradient boosting decision tree," w *Advances in neural information processing systems*, I. Guyon, U. V. Luxburg, S. Bengio i in., red., t. 30, Curran Associates, Inc., 2017.

[22] C. M. Bishop, *Pattern recognition and machine learning*. New York :Springer, 2006.

[23] A. Król-Nowak i K. Kotarba, *Podstawy uczenia maszynowego*. Wyd. AGH, 2022.

[24] A. Smola i B. Schölkopf, „A tutorial on support vector regression," *Statistics and Computing*, t. 14, s. 199–222, 2004.

[25] N. Guenther i M. Schonlau, „Support vector machines," *The Stata Journal*, t. 16, nr. 4, s. 917–937, 2016.

[26] Y. Liu, S. Liu, Y. Wang i in., „A stochastic computational multi-layer perceptron with backward propagation," *IEEE Transactions on Computers*, t. 67, nr. 9, s. 1273–1286, 2018.

[27] A. P. Bhatkar i G. Kharat, „Detection of diabetic retinopathy in retinal images using MLP classifier," w *2015 IEEE International Symposium on Nanoelectronic and Information Systems*, 2015, s. 331–335.

[28] A. Olgac i B. Karlik, „Performance analysis of various activation functions in generalized MLP architectures of neural networks," *International Journal of Artificial Intelligence And Expert Systems*, t. 1, s. 111–122, 2011.

[29] L. Yang i A. Shami, „On hyperparameter optimization of machine learning algorithms: theory and practice," *Neurocomputing*, t. 415, s. 295–316, 2020.

[30] E. Brochu, V. Cora i N. Freitas, „A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *Computing Research Repository*, t. abs/1012.2599, 2010.

[31] S. Watanabe, „Tree-structured Parzen estimator: understanding its algorithm components and their roles for better empirical performance," *arXiv*, 2023.

[32] J. Bergstra i in., „Algorithms for hyper-parameter optimization," w *Advances in Neural Information Processing Systems 24*, 2011.

[33] T. Akiba i in., „Optuna: a next-generation hyperparameter optimization framework," w *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[34] R. Sarker i C. Kamruzzaman J.and Newton, „Evolutionary optimization (Evopt): a brief review and analysis.," *International Journal of Computational Intelligence and Applications*, t. 3, s. 311–330, 2003.

[35] R. S. Olson i J. H. Moore, „TPOT: a tree-based pipeline optimization tool for automating machine learning," w *Automated Machine Learning: Methods, Systems, Challenges*. Springer International Publishing, 2019, s. 151–160.

[36] J. A. Vrugt i B. A. Robinson, „Improved evolutionary optimization from genetically adaptive multimethod search," *Proceedings of the National Academy of Sciences*, t. 104, nr. 3, s. 708–711, 2007.

[37] M. Bhattacharya, R. Islam i J. Abawajy, „Evolutionary optimization: a big data perspective," *Journal of Network and Computer Applications*, t. 59, s. 416–426, 2016.

[38] V. B. i in., „Digital medicine and the curse of dimensionality," *NPJ Digital Medicine*, t. 4, nr. 1, 2021.

[39] X. Huang, L. Wu i Y. Ye, „A review on dimensionality reduction techniques," *International Journal of Pattern Recognition and Artificial Intelligence*, t. 33, nr. 10, s. 1 950 017, 2019.

[40] N. Ahmad i A. Nassif, „Dimensionality Reduction: Challenges and Solutions," *ITM Web of Conferences*, t. 43, s. 01 017, 2022.

[41] J. Stone, „Independent component analysis: an introduction," *Trends in cognitive sciences*, t. 6, s. 59–64, 2002.

[42] N. Kwak i C. Kim, „Dimensionality reduction based on ICA for regression problems," *Neurocomputing*, t. 71, s. 1–10, 2006.

[43] A. Bell i T. Sejnowski, „An information-maximization approach to blind separation and blind deconvolution," *Neural computation*, t. 7, s. 1129–59, 1995.

[44] S. Tanwar, T. Ramani i S. Tyagi, „Dimensionality reduction using PCA and SVD in big data: a comparative case study," w *International Conference on Future Internet Technologies and Trends*, 2018, s. 116–125.

[45] Y. Wang i L. Zhu, „Research and implementation of SVD in machine learning," w *IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, 2017, s. 471–475.

[46] F. Anowar, S. Sadaoui i B. Selim, „Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE)," *Computer Science Review*, t. 40, 2021.

[47] L. Cao, K. Chua, W. Chong i in., „A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine," *Neurocomputing*, t. 55, s. 321–336, 2003.

[48] M. A. Belarbi, S. Mahmoudi i G. Belalem, „PCA as dimensionality reduction for large-scale image retrieval systems," *International Journal of Ambient Computing and Intelligence*, t. 8, nr. 2, s. 45–58, 2017.

[49] H. Hoffmann, „Kernel PCA for novelty detection," *Pattern Recognition*, t. 40, s. 863–874, 2007.

[50] M. Artač, M. Jogan i A. Leonardis, „Incremental PCA for on-line visual learning and recognition," w *International Conference on Pattern Recognition*, t. 3, 2002, s. 781 – 784.

[51] A. Tautan, A. Rossi, R. Francisco i in., „Dimensionality reduction for EEG-based sleep stage detection: comparison of autoencoders, principal component analysis and factor analysis," *Biomedizinische Technik. Biomedical engineering*, t. 66, s. 125–136, 2020.

[52] A. L. Comrey i H. B. Lee, *A first course in factor analysis (2nd ed.)* Psychology Press, 1992.

[53] I. Joliffe i B. Morgan, „Principal component analysis and exploratory factor analysis," *Statistical Methods in Medical Research*, t. 1, nr. 1, s. 69–95, 1992.

[54] J. Ford, R. MacCallum i M. Tait, „The application of exploratory factor analysis in applied psychology: a critical review and analysis," *Personnel Psychology*, t. 39, s. 291 –314, 1986.

[55] L. Le Magoarou i R. Gribonval, „Chasing butterflies: In search of efficient dictionaries," w *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[56] J. Mairal, F. Bach, J. Ponce i in., „Online dictionary learning for sparse coding," w *Proceedings of the 26th International Conference On Machine Learning, ICML 2009*, t. 382, 2009, s. 87.

[57] Y. Li, Y. Chai, H. Zhou i in., „A novel dimension reduction and dictionary learning framework for high-dimensional data classification," *Pattern Recognition*, t. 112, 2021.

[58] G. Lemaître, F. Nogueira i C. K. Aridas, „Imbalanced-learn: a Python toolbox to tackle the curse of imbalanced datasets in machine learning," *Journal of Machine Learning Research*, t. 18, nr. 17, s. 1–5, 2017.

[59] T. Wongvorachan, S. He i O. Bulut, „A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining," *Information*, t. 14, nr. 1, 2023.

[60] N. Chawla i in., „SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research (JAIR)*, t. 16, s. 321–357, 2002.

[61] F. Last, G. Douzas i F. Bação, „Oversampling for imbalanced learning based on k-means and SMOTE," *arXiv*, 2017.

[62] H. Han, W.-Y. Wang i B.-H. Mao, „Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," *Advances in Computational Intelligence*, t. 3644, s. 878–887, 2005.

[63] H. Nguyen, E. Cooper i K. Kamei, „Borderline over-sampling for imbalanced data classification," *International Journal of Knowledge Engineering and Soft Data Paradigms*, t. 3, s. 4–21, 2011.

[64] H. He i in., „ADASYN: adaptive synthetic sampling approach for imbalanced learning," *Proceedings of the International Joint Conference on Neural Networks*, s. 1322 –1328, 2008.

[65] Y.-P. Zhang, L.-N. Zhang i Y.-C. Wang, „Cluster-based majority under-sampling approaches for class imbalance learning," w *2nd IEEE International Conference on Information and Financial Engineering*, 2010.

[66] J. Zhang i I. Mani, „KNN Approach to unbalanced data distributions: a case study involving information extraction," w *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Datasets*, 2003.

[67] I. Tomek, „Two modifications of CNN," *IEEE Transactions on Systems, Man, and Cybernetics*, t. SMC-6, nr. 11, s. 769–772, 1976.

[68] D. L. Wilson, „Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man, and Cybernetics*, nr. 3, s. 408–421, 1972.

[69] R. Alejo, J. Sotoca, R. Valdovinos i in., „Edited nearest neighbor rule for improving neural networks classifications," w *7th International Symposium on Neural Networks*, 2010, s. 303–310.

[70] I. Tomek, „An experiment with the edited nearest-neighbor rule," *IEEE Transactions on Systems, Man, and Cybernetics*, t. SMC-6, nr. 6, s. 448–452, 1976.

[71] A. More, „Survey of resampling techniques for improving classification performance in unbalanced datasets," *arXiv*, 2016.

[72] P. Hart, „The condensed nearest neighbor rule," *IEEE Transactions on Information Theory*, t. 14, nr. 3, s. 515–516, 1968.

[73] M. Kubat, „Addressing the curse of imbalanced training sets: one-sided selection," w *14th International Conference on Machine Learning*, 2000.

[74] G. Batista, A. Carvalho i M.-C. Monard, „Applying one-sided selection to unbalanced datasets," w *Mexican International Conference on Artificial Intelligence*, 2006, s. 315–325.

[75] J. Laurikkala, „Improving identification of difficult small classes by balancing class distribution," w *Proceedings of 8th Conference on Artificial Intelligence in Medicine*, 2001, s. 63–66.

[76] D. Berrar i P. Flach, „Caveats and pitfalls of ROC analysis in clinical microarray research (and how to avoid them)," *Briefings in Bioinformatics*, t. 13, nr. 1, s. 83–97, 2011.

[77] B. Efron, „Better bootstrap confidence intervals," *Journal of the American Statistical Association*, t. 82, nr. 397, s. 171–185, 1987.

[78] K. Jung i in., „Comparison of bootstrap confidence interval methods for GSCA using a Monte Carlo simulation," *Frontiers in Psychology*, t. 10, 2019.

[79] B. Efron, „Bootstrap methods: another look at the jackknife," *The Annals of Statistics*, t. 7, nr. 1, s. 1–26, 1979.

[80] E. R. DeLong, D. DeLong i D. Clarke-Pearson, „Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach," *Biometrics*, t. 44, nr. 3, s. 837–45, 1988.

[81] H. Aguinis, M. Vassar i C. Wayant, „On reporting and interpreting statistical significance and p values in medical research," *BMJ Evidence-Based Medicine*, t. 26, 2019.

[82] N. Dehak, P. Kenny, R. Dehak i in., „Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, t. 19, s. 788 –798, 2011.

[83] K. Finnian, A. Alexander, O. Forth i in., „From i-vectors to x-vectors – a generational change in speaker recognition illustrated on the NFI-FRIDA database," w *31st International Association for Forensic Phonetics and Acoustics (IAFPA) Conference*, 2019.

[84] K.-A. Lee, Q. Wang i T. Koshinaka, „Xi-vector embedding for speaker recognition," *IEEE Signal Processing Letters*, t. 28, s. 1385–1389, 2021.

[85] D. Snyder, D. Garcia-Romero, G. Sell i in., „X-vectors: robust DNN embeddings for speaker recognition," w *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, s. 5329–5333.

[86] D. Snyder, D. Garcia-Romero, D. Povey i in., „Deep neural network embeddings for text-independent speaker verification," w *INTERSPEECH*, 2017, s. 999–1003.

[87] K. Kotarba i M. Kotarba, „Voice pathology assessment using x-vectors approach," *Vibrations in Physical Systems*, t. 32, nr. 1, s. 1–8, 2021.

[88] M. Farooq, F. Adeeba i S. Hussain, „X-vectors based Urdu speaker identification for short utterances," w *Conference of the Oriental COCOSDA International Committee for the Coordination and Standardisation of Speech Databases and Assessment Techniques*, 2019.

[89] R. Philips, N. Seim, L. Matrka i in., „Cost savings associated with an outpatient otolaryngology telemedicine clinic," *Laryngoscope Investigative Otolaryngology*, t. 4, 2019.

[90] A. Bokolo, „Use of telemedicine and virtual care for remote treatment in response to COVID-19 pandemic," *Journal of Medical Systems*, t. 44, s. 132, 2020.

[91] D. Hemmerling, A. Skalski i J. Gajda, „Voice data mining for laryngeal pathology assessment," *Computers in Biology and Medicine*, t. 69, 2015.

[92] S.-H. Fang, Y. Tsao, M.-J. Hsiao i in., „Detection of pathological voice using cepstrum vectors: a deep learning approach," *Journal of Voice*, t. 33, nr. 5, s. 634–641, 2019.

[93] A. Al-Nasheri, G. Muhammad, M. Alsulaiman i in., „An investigation of multidimensional voice program parameters in three different databases for voice pathology detection and classification," *Journal of Voice*, t. 31, 2016.

[94] M. Nicastri, G. Chiarella, L. Gallo i in., „Multidimensional Voice Program (MDVP) and amplitude variation parameters in euphonic adult subjects. Normative study," *Acta otorhinolaryngologica Italica : organo ufficiale della Società italiana di otorinolaringologia e chirurgia cervico-facciale*, t. 24, s. 337–41, 2005.

[95] G. Muhammad, G. Altuwaijri, M. Alsulaiman i in., „Automatic voice pathology detection and classification using vocal tract area irregularity," *Biocybernetics and Biomedical Engineering*, t. 36, 2016.

[96] H. Wu, J. Soraghan, A. Lowit i in., „Convolutional neural networks for pathological voice detection," w *40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, t. 2018, 2018, s. 1–4.

[97] M. A. Mohammed, K. H. Abdulkareem, S. A. Mostafa i in., „Voice pathology detection and classification using convolutional neural network model," *Applied Sciences*, t. 10, nr. 11, 2020.

[98] J. P. Teixeira, P. O. Fernandes i N. Alves, „Vocal acoustic analysis – classification of dysphonic voices with artificial neural networks," *Procedia Computer Science*, t. 121, s. 19–26, 2017.

[99] M. Madruga Escalona, Y. Campos-Roca i C. Pérez, „Robustness assessment of automatic Reinke's edema diagnosis systems," w *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, s. 891–895.

[100] Z. Ali, I. Elamvazuthi, M. Alsulaiman i in., „Automatic voice pathology detection with running speech by using estimation of auditory spectrum and cepstral coefficients based on the all-pole model," *Journal of Voice*, t. 30, nr. 6, 757.e7–757.e19, 2016.

[101] H. Cordeiro, C. Meneses i J. Fonseca, „Continuous speech classification systems for voice pathologies identification," w *IFIP Advances in Information and Communication Technology*, t. 450, 2015, s. 217–224.

[102] H. Cordeiro, J. Fonseca, I. Guimarães i in., „Hierarchical classification and system combination for automatically identifying physiological and neuromuscular laryngeal pathologies," *Journal of Voice*, t. 31, 2017.

[103] V. Guedes, F. Teixeira, A. Oliveira i in., „Transfer learning with AudioSet to voice pathologies identification in continuous speech," *Procedia Computer Science*, t. 164, s. 662–669, 2019.

[104] J. R. Orozco, J. Alonso, J. Vargas-Bonilla i in., „Voice pathology detection in continuous speech using nonlinear dynamics," *Electronic Letters*, 2011.

[105] N. Terbeh i M. Zrigui, „Vocal pathologies detection and mispronounced phonemes identification: case of Arabic continuous speech," w *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, European Language Resources Association (ELRA), 2016, s. 2108–2113.

[106] M. Vasilakis i Y. Stylianou, „Voice pathology detection based on short-term jitter estimations in running speech," *Folia phoniatrica et logopaedica : official organ of the International Association of Logopedics and Phoniatrics (IALP)*, t. 61, s. 153–70, 2009.

[107] K. Vicsi, V. Imre i K. Mészáros, „Voice disorder detection on the basis of continuous speech," w *5th European Conference of the International Federation for Medical and Biological Engineering*, 2012, s. 86–89.

[108] R. Fusaroli i in., „"Is voice a marker for autism spectrum disorder? A systematic review and meta-analysis": vocal production in ASD," *Autism Research*, t. 10, 2016.

[109] A. Wijesinghe i in., „Machine learning based automated speech dialog analysis Of autistic children," w *11th International Conference on Knowledge and Systems Engineering*, 2019, s. 1–5.

[110] E. Marchi i in., „Typicality and emotion in the voice of children with autism spectrum condition: evidence across three languages," w *INTERSPEECH*, 2015.

[111] F. Eyben, M. Wöllmer i B. Schuller, „openSMILE - the Munich versatile and fast open-source audio feature extractor," w *Proceedings of the ACM Multimedia International Conference*, 2010, s. 1459–1462.

[112] M. Schmitt i in., „Towards cross-lingual automatic diagnosis of autism spectrum condition in children's voices," w *Speech Communication; 12. ITG Symposium*, 2016.

[113] J. Deng i in., „Speech-based diagnosis of autism spectrum condition by generative adversarial network representations," w *7th International Digital Health Conference*, 2017.

[114] S. Cho, M. Liberman, N. Ryant i in., „Automatic detection of autism spectrum disorder in children using acoustic and text features from brief natural conversations," w *INTERSPEECH*, 2019, s. 2513–2517.

[115] F. Ringeval i in., „Automatic analysis of typical and atypical encoding of spontaneous emotion in the voice of children," w *INTERSPEECH*, 2016, s. 1210–1214.

[116] P. Grill i J. Tučková, „Speech databases of typical children and children with SLI," *PLOS One*, t. 11, nr. 3, s. 1–21, 2016.

[117] P. Grill, „Classification of children with SLI through their speech utterances," w *World Congress on Medical Physics and Biomedical Engineering*, L. Lhotska, L. Sukupova, I. Lacković i in., red., Springer Singapore, 2018, s. 441–447.

[118] D. Ramarao, C. Singh, S. Shahnawazuddin i in., „Detecting Developmental Dysphasia in Children using Speech Data," w *2018 International Conference on Signal Processing and Communications (SPCOM)*, 2018, s. 100–104.

[119] M. K. Reddy, P. Alku i K. S. Rao, „Detection of specific language impairment in children using glottal source features," *IEEE Access*, t. 8, s. 15 273–15 279, 2020.

[120] G. Sharma, X.-P. Zhang, K. Umapathy i in., „Audio texture and age-wise analysis of disordered speech in children having specific language impairment," *Biomedical Signal Processing and Control*, t. 66, s. 102 471, 2021.

[121] K. Kotarba i M. Kotarba, „Efficient detection of specific language impairment in children using ResNet classifier," w *2020 Signal Processing: Algorithms, Architectures, Arrangements, and Applications*, 2020, s. 169–173.

[122] R. Pappagari, J. Cho, S. Joshi i in., „Automatic detection and assessment of Alzheimer disease using speech and language technologies in low-resource scenarios," w *INTERSPEECH*, 2021.

[123] J. Devlin, M.-W. Chang, K. Lee i in., „BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv*, 2018.

[124] R. Pappagari, J. Cho, L. Moro-Velázquez i in., „Using state of the art speaker recognition and natural language processing technologies to detect Alzheimer's disease and assess its severity," w *INTERSPEECH*, 2020.

[125] L. Jeancolas, D. Petrovska-Delacrétaz, G. Mangone i in., „X-vectors: new quantitative biomarkers for early Parkinson's disease detection from speech," *Frontiers in Neuroinformatics*, t. 15, s. 4, 2021.

[126] K. Kotarba, „GFCC-based x-vectors for Reinke's edema detection," *Vibrations in Physical Systems*, t. 33, nr. 3, s. 1–7, 2022.

[127] S. S. Haj Zargarbashi i B. Babaali, „A multi-modal feature embedding approach to diagnose Alzheimer's disease from spoken language," *arXiv*, 2019.

[128] D. Jurafsky i J. H. Martin, *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*, 1st. Prentice Hall PTR, 2000.

[129] A. Pompili, T. Rolland i A. Abad, „The INESC-ID multi-modal system for the ADReSS 2020 Challenge," w *INTERSPEECH*, 2020.

[130] P. Perez-Toro, J. Vasquez, T. Arias-Vergara i in., „Acoustic and linguistic analyses to assess early-onset and genetic Alzheimer's disease," w *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, s. 8338–8342.

[131] L. Moro-Velázquez, J. Villalba i N. Dehak, „Using x-vectors to automatically detect Parkinson's disease from speech," w *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020.

[132] C. Botelho, F. Teixeira, T. Rolland i in., „Pathological speech detection using x-vector embeddings," *arXiv*, 2020.

[133] J. Perero-Codosero, F. M. Espinoza-Cuadros, J. Antón-Martín i in., „Modeling obstructive sleep apnea voices using deep neural network embeddings and domain-adversarial training," *IEEE Journal of Selected Topics in Signal Processing*, 2019.

[134] Y. Pan, V. Nallanthighal, D. Blackburn i in., „Multi-task estimation of age and cognitive decline from speech," w *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, s. 7258–7262.

[135] L. Moro-Velázquez, E. Hernández-García, J. Gómez-García i in., „Analysis of the effects of supraglottal tract surgical procedures in automatic speaker recognition performance," w *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2020.

[136] W. J. Barry i M. Pützer, *Saarbrücken Voice Databse*, Institute of Phonetics, University of Saarland, *http://www.stimmdatenbank.coli.uni-saarland.de* [Online; dostęp 05.01.2022].

[137] P. Grill i J. Tučková, „Speech databases of typical children and children with SLI," *PLOS One*, t. 11, nr. 3, s. 1–21, 2016.

[138] S. Kuijper, C. Hartman i P. Hendriks, „Who is he? Children with ASD and ADHD take the listener into account in their production of ambiguous pronouns," *PLOS One*, t. 10, 2015.

[139] R. Rao i G. Fung, „On the dangers of cross-validation. An experimental evaluation," w *SIAM International Conference on Data Mining*, 2008, s. 588–596.

[140] A. V. i in., „Machine learning algorithm validation with a limited sample size," *PLOS One*, t. 14, nr. 11, 2019.

[141] C. Dwork i in., „Preserving statistical validity in adaptive data analysis," *arXiv*, 2014.

[142] F. E. Harrell i J. C. Slaughter, *Biostatistics for biomedical research*, Department of Biostatistics Vanderbilt University School of Medicine, *https://hbiostat.org/doc/bbr.pdf* [Online; dostęp 14.01.2022].

[143] R. Dinga, B. Penninx, D. J. Veltman i in., „Beyond accuracy: measures for assessing machine learning models, pitfalls and guidelines," *Cold Spring Harbor Laboratory - bioRxiv*, 2019.

[144] T. Saito i M. Rehmsmeier, „The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PLOS One*, t. 10, nr. 3, 2015.

[145] F. Movahedi, R. Padman i J. Antaki, „Limitations of ROC on imbalanced data: evaluation of LVAD mortality risk scores," *arXiv*, 2020.

[146] J. Davis i M. Goadrich, „The relationship between precision-recall and ROC curves," w *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

[147] I. R. Titze, *Principles of voice production*, 2 wyd. National Center for Voice i Speech, 2000, rozd. 6, 149–184.

[148] G. Stegmann i in., „Repeatability of commonly used speech and language features for clinical applications," *Digital Biomarkers*, t. 4, s. 109–122, 2020.

[149] M. K. i in., „Designing neural speaker embeddings with meta learning," *arXiv*, 2020.

[150] D. Povey i in., „The Kaldi speech recognition toolkit," w *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 2011.

[151] J. S. Chung, A. Nagrani i A. Zisserman, „VoxCeleb2: Deep Speaker Recognition," w *INTERSPEECH*, 2018.

[152] Featxtra toolbox for Kaldi, *https://github.com/mvansegbroeck-zz/featxtra*, [Online; dostęp 22.01.2022],

[153] A. Paszke i in., „PyTorch: an imperative style, high-performance deep learning library," w *Advances in Neural Information Processing Systems 32*, Curran Associates, Inc., 2019, s. 8024–8035.

[154] G. K. i in., „Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, t. 30, s. 3146–3154, 2017.

[155] E. Debie i K. Shafi, „Implications of the curse of dimensionality for supervised learning classifier systems: theoretical and empirical analyses," *Pattern Analysis and Applications*, t. 22, 2019.

[156] W. L. i in., „SphereFace: deep hypersphere embedding for face recognition," w *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[157] J. D. i in., „ArcFace: additive angular margin loss for deep face recognition," w *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[158] F. W. i in., „Additive margin softmax for face verification," *IEEE Signal Processing Letters*, t. 25, nr. 7, s. 926–930, 2018.

[159] M. Rybicka i K. Kowalczyk, „On parameter adaptation in softmax-based cross-entropy loss for improved convergence speed and accuracy in DNN-based speaker recognition," w *INTERSPEECH*, 2020.

[160] H. W. i in., „CosFace: large margin cosine loss for deep face recognition," *arXiv*, 2018.

[161] A. W. i in., „Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, t. 37, nr. 3, 328–339, 1989.

[162] D. Povey i in., „Semi-orthogonal low-rank matrix factorization for deep neural networks," w *INTERSPEECH*, 2018, s. 3743–3747.

[163] D. Snyder, J. Villalba, N. Chen i in., „The JHU speaker recognition system for the VOiCES 2019 Challenge," w *INTERSPEECH*, 2019, s. 2468–2472.

[164] Z. Zhong, L. Zheng, G. Kang i in., „Random erasing data augmentation," w *Proceedings of the AAAI Conference on Artificial Intelligence*, t. 34, 2017.

[165] G. Hinton i in., „Improving neural networks by preventing co-adaptation of feature detectors," *arXiv*, 2012.

[166] L. Tosti i in., „Mapping transcription factor occupancy using minimal numbers of cells in vitro and in vivo," *Genome Research*, t. 28, 2018.

[167] F. O. Ruiz i F. Perez-Cruz, „A generative model for predicting outcomes in college basketball," *Journal of Quantitative Analysis in Sports*, 2015.

[168] A. H. i in., „The effect of novelty on reinforcement learning," w *Decision Making*, ser. Progress in Brain Research, V. C. Pammi i N. Srinivasan, red., t. 202, Elsevier, 2013, s. 415–439.

[169] S. W. Scheff, „Nonparametric statistics," w *Fundamental Statistical Principles for the Neurobiologist*, Academic Press, 2016, s. 157–182.

[170] R. H. Riffenburgh i D. L. Gillen, „Tests of location with continuous outcomes," w *Statistics in Medicine (Fourth Edition)*, Academic Press, 2020, s. 239–294.

[171] E. Guerrero Vázquez i in., „Repeated measures multiple comparison procedures applied to model selection in neural networks," w *Bio-Inspired Applications of Connectionism*, Springer, 2001, s. 88–95.

[172] M. Friedman, „The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, t. 32, nr. 200, s. 675–701, 1937.

[173] M. Friedman, „A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, t. 11, nr. 1, s. 86–92, 1940.

[174] P. Virtanen i in., „SciPy 1.0: fundamental algorithms for scientific computing in Python," *Nature Methods*, t. 17, s. 261–272, 2020.

[175] P. Nemenyi, „Distribution-free multiple comparisons", prac. dokt., Princeton University, 1963.

[176] J. Demsar, „Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, t. 7, 2006.

[177] M. A. Terpilowski, „scikit-posthocs: pairwise multiple comparison tests in Python," *Journal of Open Source Software*, t. 4, nr. 36, 2019.

[178] H. B. i in., *Choroby ucha, nosa i gardła z chirurgią głowy i szyi*, 2 wyd. Edra Urban & Partner, 2017.

[179] W. Arnold i U. Ganzer, *Otolaryngologia*. Medpharm, 2014.

[180] P. Z Schochet, „Guidelines for multiple testing in impact evaluations of educational interventions. Final report.," *Mathematica Policy Research, Inc.*, 2008.

[181] M. Galar, A. Fernández, E. Barrenechea i in., „A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, t. 42, s. 463 –484, 2012.

[182] B. Krawczyk, „Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, t. 5, 2016.

[183] J. Leevy, T. Khoshgoftaar, R. Bauder i in., „A survey on addressing high-class imbalance in big data," *Journal of Big Data*, t. 5, 2018.

[184] L. Yu i N. Zhou, „Survey of imbalanced data methodologies," *arXiv*, 2021.

[185] J. Tanha, Y. Abdi, N. Samadi i in., „Boosting methods for multi-class imbalanced data classification: an experimental review," *Journal of Big Data*, t. 7, s. 70, 2020.

[186] K. Hasib, M. Iqbal, F. Shah i in., „A survey of methods for managing the classification and solution of data imbalance problem," *Journal of Computer Science*, t. 16, s. 1546 –1557, 2020.

[187] R. S. Olson, N. Bartley, R. J. Urbanowicz i in., „Evaluation of a tree-based pipeline optimization tool for automating data science," w *Proceedings of the Genetic and Evolutionary Computation Conference*, 2016, s. 485–492.

[188] T. T. Le, W. Fu i J. H. Moore, „Scaling tree-based automated machine learning to biomedical big data with a feature set selector," *Bioinformatics*, t. 36, nr. 1, s. 250–256, 2020.

[189] D. J. Hand, „Classifier technology and the illusion of progress," *Statistical Science*, t. 21, nr. 1, 2006.

[190] J. O. i in., „Presenting artificial intelligence, deep learning, and machine learning studies to clinicians and healthcare stakeholders: an introductory reference with a guideline and a Clinical AI Research (CAIR) checklist proposal," *Acta Orthopaedica*, t. 92, nr. 5, s. 513–525, 2021.

[191] J. Sidey-Gibbons i C. Sidey-Gibbons, „Machine learning in medicine: a practical introduction," *BMC Medical Research Methodology*, t. 19, 2019.

[192] M. Yu i in., „Reporting on deep learning algorithms in health care," *The Lancet Digital Health*, t. 1, 2019.

[193] X. Liu, „Classification accuracy and cut point selection," *Statistics in medicine*, t. 31, s. 2676–86, 2012.

[194] I. Unal, „Defining an optimal cut-point value in ROC analysis: an alternative approach," *Computational and Mathematical Methods in Medicine*, t. 2017, s. 1–14, 2017.

[195] W. J. Youden, „Index for rating diagnostic tests," *Cancer*, t. 3, 1950.

[196] R. G. Miller i D. O. Siegmund, „Maximally selected chi square statistics," *Biometrics*, t. 38, 1982.

[197] C. Sung, A. Connor, J. Chen i in., „Development, feasibility, and preliminary efficacy of an employment-related social skills intervention for young adults with high-functioning autism," *Autism*, t. 23, nr. 6, s. 1542–1553, 2019.

[198] S. Cho i in., „Automatic detection of autism spectrum disorder in children using acoustic and text features from brief natural conversations," w *INTERSPEECH*, 2019, s. 2513–2517.

[199] L. Schwartz, S. Woloshin, H. Sox i in., „US women's attitudes to false positive mammography results and detection of ductal carcinoma in situ: cross sectional survey," *The Western journal of medicine*, t. 173, s. 307–12, 2000.

[200] D. B. i in., „Patients' & healthcare professionals' values regarding true- & false-positive diagnosis when colorectal cancer screening by CT colonography: discrete choice experiment," *PLOS One*, t. 8, nr. 12, 2013.

[201] J. Brodersen i V. Siersma, „Long-term psychosocial consequences of false-positive screening mammography," *Annals of family medicine*, t. 11, s. 106–115, 2013.

[202] P. Olliaro i E. Torreele, „Managing the risks of making the wrong diagnosis: first, do no harm," *International Journal of Infectious Diseases*, t. 106, s. 382–385, 2021.

[203] S. Werner i C. Shulman, „Subjective well-being among family caregivers of individuals with developmental disabilities: the role of affiliate stigma and psychosocial moderating variables," *Research in Developmental Disabilities*, t. 34, nr. 11, 2013.

[204] D. E. Gray, „Perceptions of stigma: the parents of autistic children," *Sociology of Health & Illness*, t. 15, nr. 1, s. 102–120, 1993.

[205] C. D. Kang-Yi, R. Grinker i D. Mandell, „Korean culture and autism spectrum disorders," *Journal of Autism and Developmental Disorders*, t. 43, 2013.

[206] Y. S. Kim, B. L. Leventhal, Y.-J. Koh i in., „Prevalence of autism spectrum disorders in a total population sample," *American Journal of Psychiatry*, t. 168, nr. 9, s. 904–912, 2011.

[207] M. Kayama i W. Haight, „Disability and stigma: how Japanese educators help parents accept their children's differences," *Social Work*, t. 59, nr. 1, s. 24–33, 2014.

[208] D. Kite, J. Gullifer i G. Tyson, „Views on the diagnostic labels of autism and Asperger's disorder and the proposed changes in the DSM," *Journal of Autism and Developmental Disorders*, t. 43, 2013.

[209] G. Russell i B. Norwich, „Dilemmas, diagnosis and de-stigmatization: parental perspectives on the diagnosis of autism spectrum disorders," *Clinical Child Psychology and Psychiatry*, t. 17, nr. 2, s. 229–245, 2012.

[210] L. Fusar-Poli, N. Brondino, P. Politi i in., „Missed diagnoses and misdiagnoses of adults with autism spectrum disorder," *European Archives of Psychiatry and Clinical Neuroscience*, t. 272, 2022.

[211] „Deficit, difference, or both? Autism and neurodiversity," *Developmental Psychology*, t. 49, nr. 1, 2013.

[212] *https://childes.talkbank.org/access/DutchAfrikaans/Asymmetries.html*, [Online; dostęp 05.01.2022].

[213] N. Mohd Razali i B. Yap, „Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests," *Journal of Statistical Modeling and Analytics*, t. 2, 2011.

Dodatek A

Rys. A.1. Historyjki obrazkowe użyte podczas tworzenia holenderskiego korpusu mowy wykorzystanego w celu detekcji zaburzeń ze spektrum autyzmu. Źródło: [212].

Dodatek B

Wyniki testów DeLonga sprawdzających równość wartości AUC ROC uzyskiwane przez prezentowane w pracy najlepsze modele uczone na poszczególnych wariantach x-wektorów z modelem uzyskującym AUC ROC = 0,5 (zwracającym predykcje losowe). Przyjęto poziom istotności statystycznej α = 0,05.

H_0: AUC ROC analizowanego modelu wynosi 0,5
H_1: AUC ROC analizowanego modelu nie wynosi 0,5

W przypadku, gdy p $\geqslant \alpha$ nie ma podstaw do odrzucenia hipotezy zerowej mówiącej o tym, że AUC ROC uzyskane przez analizowany model wynosi 0,5, co oznacza, że model ten nie jest lepszy od modelu zwracającego predykcje losowe.
W przypadku, gdy p < α istnieją podstawy do odrzucenia hipotezy zerowej i przyjęcia hipotezy alternatywnej mówiącej o tym, że analizowany model uzyskał AUC ROC istotnie różne od 0,5, co oznacza, że jest lepszy od modelu zwracającego predykcje losowe.

DETEKCJA NIEOKREŚLONEGO SCHORZENIA KRTANI

- MFCC: p = 0.0000 < α
- GFCC: p = 0.0000 < α
- GFCC-Arc: p = 0.0000 < α
- GFCC-FTDNN: p = 0.0000 < α
- GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA DYSFONII BEZ OKREŚLONEGO TYPU

- MFCC: p = 0.0000 < α
- GFCC: p = 0.0000 < α
- GFCC-Arc: p = 0.0000 < α
- GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA OSTREGO ZAPALENIA KRTANI

– MFCC: p = 2.7917e-10 < α

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA DYSFONII CZYNNOŚCIOWEJ

– MFCC: p = 8.3793e-02 $\geqslant \alpha$

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 1.1102e-15 < α

DETEKCJA DYSFONII PSYCHOGENNEJ

– MFCC: p = 0.0919 $\geqslant \alpha$

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA DYSFONII SPASTYCZNEJ

– MFCC: p = 0.1028 $\geqslant \alpha$

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA PORAŻENIA NERWU KRTANIOWEGO WSTECZNEGO

– MFCC: p = 0.0000 < α

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA OBRZĘKU REINKEGO

– MFCC: p = 4.5848e-10 < α

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA PACHYDERMII KONTAKTOWEJ

– MFCC: p = 2.8441e-06 < α

– GFCC: p = 1.7134e-12 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

DETEKCJA ZABURZEŃ ZE SPEKTRUM AUTYZMU

– MFCC: p = 1.6746e-04 < α

– GFCC: p = 1.0635e-09 < α

– GFCC-Arc: p = 1.3380e-05 < α

– GFCC-FTDNN: p = 2.5043e-05 < α

– GFCC-FTDNN-RC0: p = 8.5677e-12 < α

DETEKCJA SPECYFICZNYCH ZABURZEŃ JĘZYKOWYCH

– MFCC: p = 0.0000 < α

– GFCC: p = 0.0000 < α

– GFCC-Arc: p = 0.0000 < α

– GFCC-FTDNN: p = 0.0000 < α

– GFCC-FTDNN-RC0: p = 0.0000 < α

Dodatek C

Wyniki testów normalności prób bootstrapowych wygenerowanych w celu sprawdzenia hipotez o równości strat logarytmicznych uzyskanych przez poszczególne modele. Hipotezy o normalności rozkładu prób sprawdzone zostały testem Shapiro-Wilka, który cechuje się największą mocą przy analizowanej wielkości próby n = 500 [213]. Przyjęto poziom istotności statystycznej α = 0,05.

H_0: próba pochodzi z populacji opisanej dystrybuantą rozkładu normalnego
H_1: próba nie pochodzi z populacji opisanej dystrybuantą rozkładu normalnego

W przypadku, gdy p $\geqslant \alpha$ nie ma podstaw do odrzucenia hipotezy zerowej mówiącej o tym, że próba pochodzi z populacji opisanej dystrybuantą rozkładu normalnego.
W przypadku, gdy p < α istnieją podstawy do odrzucenia hipotezy zerowej i przyjęcia hipotezy alternatywnej mówiącej o tym, że próba nie pochodzi z populacji opisanej dystrybuantą rozkładu normalnego.

DETEKCJA NIEOKREŚLONEGO SCHORZENIA KRTANI

– GFCC: p = 0,0010 < α
– GFCC-FTDNN: p = 0,0010 < α
– GFCC-FTDNN-RC0: p = 0,0010 < α

DETEKCJA DYSFONII BEZ OKREŚLONEGO TYPU

– GFCC-Arc: p = 0,0189 < α
– GFCC-FTDNN: p = 0,6472 $\geqslant \alpha$
– GFCC-FTDNN-RC0: p = 0,4430 $\geqslant \alpha$

DETEKCJA OSTREGO ZAPALENIA KRTANI

– GFCC-FTDNN: p = 0,0010 < α
– GFCC-FTDNN-RC0: p = 0,0010 < α

DETEKCJA DYSFONII CZYNNOŚCIOWEJ

– GFCC: p = 0,0010 < α

– GFCC-Arc: p = 0,4045 $\geqslant \alpha$

– GFCC-FTDNN: p = 0,001 < α

– GFCC-FTDNN-RC0: p = 0,4706 $\geqslant \alpha$

DETEKCJA DYSFONII PSYCHOGENNEJ

– GFCC: p = 0,0169 < α

– GFCC-Arc: p = 0,0010 < α

– GFCC-FTDNN: p = 0,4598 $\geqslant \alpha$

– GFCC-FTDNN-RC0: p = 0,0010 < α

DETEKCJA DYSFONII SPASTYCZNEJ

– GFCC: p = 0,0015 < α

– GFCC-Arc: p = 0,0010 < α

– GFCC-FTDNN: p = 0,9529 $\geqslant \alpha$

– GFCC-FTDNN-RC0: p = 0,0163 < α

DETEKCJA PORAŻENIA NERWU KRTANIOWEGO WSTECZNEGO

– GFCC: p = 0,3530 $\geqslant \alpha$

– GFCC-Arc: p = 0,7537 $\geqslant \alpha$

– GFCC-FTDNN: p = 0,0010 < α

– GFCC-FTDNN-RC0: p = 0,2157 $\geqslant \alpha$

DETEKCJA OBRZĘKU REINKEGO

– MFCC: p = 0,1530 $\geqslant \alpha$

– GFCC: p = 0,0010 < α

– GFCC-Arc: p = 0,0010 < α

– GFCC-FTDNN: p = 0,0010 < α

– GFCC-FTDNN-RC0: p = 0,0521 $\geqslant \alpha$

DETEKCJA PACHYDERMII KONTAKTOWEJ

– MFCC: p = 0,3515 $\geqslant \alpha$

– GFCC: p = 0,4206 $\geqslant \alpha$

– GFCC-Arc: p = 0,0010 < α

– GFCC-FTDNN: p = 0,0010 < α

– GFCC-FTDNN-RC0: p = 0,0632 $\geqslant \alpha$

DETEKCJA SPECYFICZNYCH ZABURZEŃ JĘZYKOWYCH

– MFCC: p = 0,1888 $\geqslant \alpha$

– GFCC: p = 0,0938 $\geqslant \alpha$

– GFCC-Arc: p = 0,5869 $\geqslant \alpha$

– GFCC-FTDNN: p = 0,0010 < α

– GFCC-FTDNN-RC0: p = 0,0174 < α

Dodatek D

Tabela D.1. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji osób chorych.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	234	FactorAnalysis	164	FactorAnalysis	229	SelectKBest	289	SelectKBest	274	SelectKBest
Extra Trees	260	SelectKBest	109	PCA	46	FactorAnalysis	10	FastICA	108	KernelPCA
KNN	32	PCA	131	PCA	165	SelectKBest	225	SelectKBest	271	SelectKBest
LDA	104	TruncatedSVD	61	FactorAnalysis	11	PCA	77	TruncatedSVD	40	FastICA
LGB	35	FastICA	30	TruncatedSVD	130	FactorAnalysis	10	FastICA	16	IncrementalPCA
LR	55	IncrementalPCA	53	FactorAnalysis	129	SelectKBest	42	KernelPCA	23	FactorAnalysis
MLP	60	PCA	298	PCA	30	IncrementalPCA	99	IncrementalPCA	281	TruncatedSVD
NuSVM	75	TruncatedSVD	283	TruncatedSVD	107	SelectKBest	52	TruncatedSVD	225	SelectKBest
QDA	81	FactorAnalysis	16	FactorAnalysis	15	TruncatedSVD	19	IncrementalPCA	25	SelectKBest
RF	30	FastICA	138	PCA	40	FactorAnalysis	82	IncrementalPCA	152	TruncatedSVD
RNN	190	IncrementalPCA	40	SelectKBest	103	SelectKBest	10	PCA	288	TruncatedSVD
SGD	55	FastICA	30	FactorAnalysis	275	PCA	62	FastICA	199	PCA

Tabela D.2. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji dysfonii.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	271	SelectKBest	103	KernelPCA	283	SelectKBest	287	SelectKBest	14	TruncatedSVD
Extra Trees	29	FastICA	253	IncrementalPCA	84	FactorAnalysis	100	PCA	129	PCA
KNN	38	KernelPCA	86	KernelPCA	220	SelectKBest	248	SelectKBest	228	SelectKBest
LDA	62	FastICA	10	PCA	19	FactorAnalysis	60	FastICA	155	SelectKBest
LGB	170	SelectKBest	53	Dictionary Learning	75	SelectKBest	68	PCA	10	TruncatedSVD
LR	28	PCA	42	KernelPCA	41	IncrementalPCA	119	KernelPCA	24	PCA
MLP	36	IncrementalPCA	241	PCA	15	FactorAnalysis	38	FastICA	99	TruncatedSVD
NuSVM	280	SelectKBest	25	TruncatedSVD	23	TruncatedSVD	36	TruncatedSVD	19	TruncatedSVD
QDA	33	IncrementalPCA	30	IncrementalPCA	65	KernelPCA	72	FastICA	157	SelectKBest
RF	229	FactorAnalysis	261	PCA	300	KernelPCA	132	Dictionary Learning	194	TruncatedSVD
RNN	15	FastICA	243	IncrementalPCA	197	IncrementalPCA	39	PCA	198	PCA
SGD	113	FactorAnalysis	11	PCA	86	IncrementalPCA	38	KernelPCA	46	IncrementalPCA

Tabela D.3. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji ostrego zapalenia krtani.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	147	FactorAnalysis	275	SelectKBest	139	TruncatedSVD	96	Incremental PCA	271	SelectKBest
Extra Trees	38	FactorAnalysis	249	Incremental PCA	275	PCA	88	TruncatedSVD	178	TruncatedSVD
KNN	99	FactorAnalysis	157	PCA	13	PCA	18	PCA	198	TruncatedSVD
LDA	55	FactorAnalysis	41	KernelPCA	79	KernelPCA	19	Incremental PCA	10	TruncatedSVD
LGB	99	FactorAnalysis	16	FastICA	295	SelectKBest	20	KernelPCA	197	TruncatedSVD
LR	42	TruncatedSVD	16	FactorAnalysis	40	SelectKBest	11	PCA	39	TruncatedSVD
MLP	41	FactorAnalysis	10	TruncatedSVD	10	PCA	26	FastICA	207	SelectKBest
NuSVM	139	FactorAnalysis	14	TruncatedSVD	153	TruncatedSVD	109	TruncatedSVD	173	TruncatedSVD
QDA	75	KernelPCA	285	KernelPCA	164	SelectKBest	71	SelectKBest	29	SelectKBest
RF	250	TruncatedSVD	228	SelectKBest	136	FactorAnalysis	276	SelectKBest	246	SelectKBest
RNN	280	TruncatedSVD	95	SelectKBest	10	FastICA	44	SelectKBest	77	Incremental PCA
SGD	39	FactorAnalysis	36	SelectKBest	18	PCA	181	TruncatedSVD	101	TruncatedSVD

Tabela D.4. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji porażenia nerwu krtaniowego wstecznego.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	69	Incremental PCA	211	SelectKBest	111	FactorAnalysis	130	PCA	113	KernelPCA
Extra Trees	70	TruncatedSVD	207	SelectKBest	78	FactorAnalysis	150	Incremental PCA	49	PCA
KNN	55	FastICA	84	PCA	79	SelectKBest	267	SelectKBest	10	TruncatedSVD
LDA	66	FactorAnalysis	79	TruncatedSVD	14	FastICA	27	TruncatedSVD	92	TruncatedSVD
LGB	62	PCA	208	SelectKBest	284	SelectKBest	23	KernelPCA	65	TruncatedSVD
LR	38	TruncatedSVD	26	FactorAnalysis	162	TruncatedSVD	18	Incremental PCA	11	Incremental PCA
MLP	69	KernelPCA	214	SelectKBest	72	SelectKBest	137	TruncatedSVD	239	SelectKBest
NuSVM	70	TruncatedSVD	233	SelectKBest	160	SelectKBest	184	TruncatedSVD	253	KernelPCA
QDA	61	FactorAnalysis	37	SelectKBest	74	SelectKBest	10	PCA	10	TruncatedSVD
RF	42	TruncatedSVD	232	SelectKBest	226	FactorAnalysis	283	SelectKBest	14	TruncatedSVD
RNN	39	FactorAnalysis	10	KernelPCA	10	KernelPCA	163	PCA	238	SelectKBest
SGD	69	PCA	278	SelectKBest	38	FastICA	147	PCA	161	KernelPCA

Tabela D.5. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji obrzęku Reinkego.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	300	SelectKBest	215	SelectKBest	52	FactorAnalysis	299	SelectKBest	30	IncrementalPCA
Extra Trees	137	SelectKBest	14	SelectKBest	87	SelectKBest	21	TruncatedSVD	20	TruncatedSVD
KNN	81	FastICA	157	PCA	17	PCA	30	TruncatedSVD	184	KernelPCA
LDA	129	Incremental PCA	51	FastICA	64	TruncatedSVD	56	PCA	29	KernelPCA
LGB	300	SelectKBest	120	Incremental PCA	267	SelectKBest	146	TruncatedSVD	238	SelectKBest
LR	185	SelectKBest	80	SelectKBest	48	Dictionary Learning	39	FactorAnalysis	32	TruncatedSVD
MLP	201	SelectKBest	29	SelectKBest	57	KernelPCA	25	IncrementalPCA	46	PCA
NuSVM	68	FastICA	74	SelectKBest	58	PCA	25	KernelPCA	65	FastICA
QDA	29	PCA	115	SelectKBest	48	SelectKBest	33	TruncatedSVD	32	KernelPCA
RF	136	SelectKBest	10	FastICA	165	SelectKBest	269	Dictionary Learning	161	SelectKBest
RNN	53	SelectKBest	26	SelectKBest	25	Incremental PCA	46	FactorAnalysis	159	TruncatedSVD
SGD	293	KernelPCA	159	TruncatedSVD	63	TruncatedSVD	44	KernelPCA	88	TruncatedSVD

Tabela D.6. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji pachydermii kontaktowej.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	94	FactorAnalysis	244	SelectKBest	189	SelectKBest	151	PCA	141	KernelPCA
Extra Trees	48	SelectKBest	48	TruncatedSVD	19	IncrementalPCA	102	IncrementalPCA	131	SelectKBest
KNN	163	Incremental PCA	161	SelectKBest	265	KernelPCA	160	Incremental PCA	26	Incremental PCA
LDA	33	Dictionary Learning	10	KernelPCA	60	TruncatedSVD	11	TruncatedSVD	10	PCA
LGB	42	SelectKBest	50	TruncatedSVD	28	PCA	14	FastICA	225	TruncatedSVD
LR	154	Dictionary Learning	14	SelectKBest	16	SelectKBest	23	FactorAnalysis	20	Dictionary Learning
MLP	98	SelectKBest	246	SelectKBest	16	TruncatedSVD	15	TruncatedSVD	10	PCA
NuSVM	76	FactorAnalysis	218	TruncatedSVD	98	PCA	196	TruncatedSVD	271	KernelPCA
QDA	292	Dictionary Learning	26	SelectKBest	129	SelectKBest	294	SelectKBest	160	FactorAnalysis
RF	59	SelectKBest	35	SelectKBest	56	SelectKBest	10	TruncatedSVD	100	SelectKBest
RNN	21	SelectKBest	114	TruncatedSVD	295	IncrementalPCA	36	FactorAnalysis	40	PCA
SGD	81	SelectKBest	199	PCA	277	PCA	297	KernelPCA	264	TruncatedSVD

Tabela D.7. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji dysfonii funkcjonalnej.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	280	KernelPCA	273	SelectKBest	48	SelectKBest	300	SelectKBest	42	TruncatedSVD
Extra Trees	66	FactorAnalysis	42	SelectKBest	10	IncrementalPCA	49	FastICA	13	FactorAnalysis
KNN	20	FastICA	176	Dictionary Learning	300	SelectKBest	295	Dictionary Learning	99	IncrementalPCA
LDA	12	IncrementalPCA	21	KernelPCA	25	PCA	37	TruncatedSVD	24	PCA
LGB	60	KernelPCA	20	SelectKBest	20	FastICA	22	FastICA	10	TruncatedSVD
LR	32	SelectKBest	290	Dictionary Learning	42	TruncatedSVD	15	FactorAnalysis	13	SelectKBest
MLP	50	Dictionary Learning	33	FactorAnalysis	65	SelectKBest	39	IncrementalPCA	10	TruncatedSVD
NuSVM	95	SelectKBest	141	SelectKBest	195	IncrementalPCA	195	SelectKBest	115	PCA
QDA	60	PCA	257	Dictionary Learning	111	PCA	159	FastICA	260	KernelPCA
RF	150	Dictionary Learning	28	SelectKBest	15	FactorAnalysis	266	SelectKBest	160	SelectKBest
RNN	201	SelectKBest	175	PCA	36	FastICA	10	PCA	250	TruncatedSVD
SGD	62	SelectKBest	12	KernelPCA	118	SelectKBest	74	PCA	10	FastICA

Tabela D.8. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji dysfonii psychogennej.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	10	TruncatedSVD	157	SelectKBest	27	IncrementalPCA	48	PCA	282	SelectKBest
Extra Trees	45	KernelPCA	10	PCA	33	KernelPCA	18	FastICA	21	KernelPCA
KNN	22	Dictionary Learning	141	TruncatedSVD	11	PCA	53	TruncatedSVD	79	TruncatedSVD
LDA	14	PCA	20	SelectKBest	36	PCA	37	SelectKBest	103	IncrementalPCA
LGB	10	FactorAnalysis	31	Dictionary Learning	10	PCA	107	SelectKBest	107	KernelPCA
LR	34	Dictionary Learning	32	SelectKBest	23	Dictionary Learning	41	FactorAnalysis	40	FactorAnalysis
MLP	10	FastICA	167	SelectKBest	11	FastICA	48	FastICA	10	KernelPCA
NuSVM	111	TruncatedSVD	184	FactorAnalysis	267	SelectKBest	149	TruncatedSVD	85	FactorAnalysis
QDA	239	IncrementalPCA	275	FastICA	29	FastICA	219	SelectKBest	201	SelectKBest
RF	15	TruncatedSVD	10	TruncatedSVD	38	KernelPCA	?	SelectKBest	20	IncrementalPCA
RNN	113	SelectKBest	20	TruncatedSVD	19	Dictionary Learning	245	PCA	161	TruncatedSVD
SGD	290	KernelPCA	26	IncrementalPCA	13	FastICA	64	FactorAnalysis	46	IncrementalPCA

Tabela D.9. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji dysfonii spastycznej.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	129	IncrementalPCA	293	SelectKBest	103	SelectKBest	74	FactorAnalysis	72	SelectKBest
Extra Trees	100	FactorAnalysis	51	FactorAnalysis	151	FastICA	28	FactorAnalysis	266	SelectKBest
KNN	237	Dictionary Learning	72	SelectKBest	96	TruncatedSVD	97	FactorAnalysis	41	IncrementalPCA
LDA	12	FastICA	202	FactorAnalysis	145	KernelPCA	123	FastICA	68	FactorAnalysis
LGB	277	KernelPCA	103	SelectKBest	241	SelectKBest	291	SelectKBest	279	SelectKBest
LR	155	Dictionary Learning	137	IncrementalPCA	272	FactorAnalysis	128	PCA	270	KernelPCA
MLP	14	SelectKBest	140	FastICA	289	PCA	186	FactorAnalysis	179	PCA
NuSVM	224	Dictionary Learning	99	IncrementalPCA	102	TruncatedSVD	293	SelectKBest	96	TruncatedSVD
QDA	228	SelectKBest	204	SelectKBest	16	FastICA	45	FactorAnalysis	48	FactorAnalysis
RF	192	KernelPCA	116	FastICA	300	SelectKBest	88	FastICA	25	SelectKBest
RNN	94	FastICA	35	PCA	28	FactorAnalysis	21	TruncatedSVD	25	SelectKBest
SGD	71	IncrementalPCA	126	FastICA	110	PCA	216	IncrementalPCA	125	FactorAnalysis

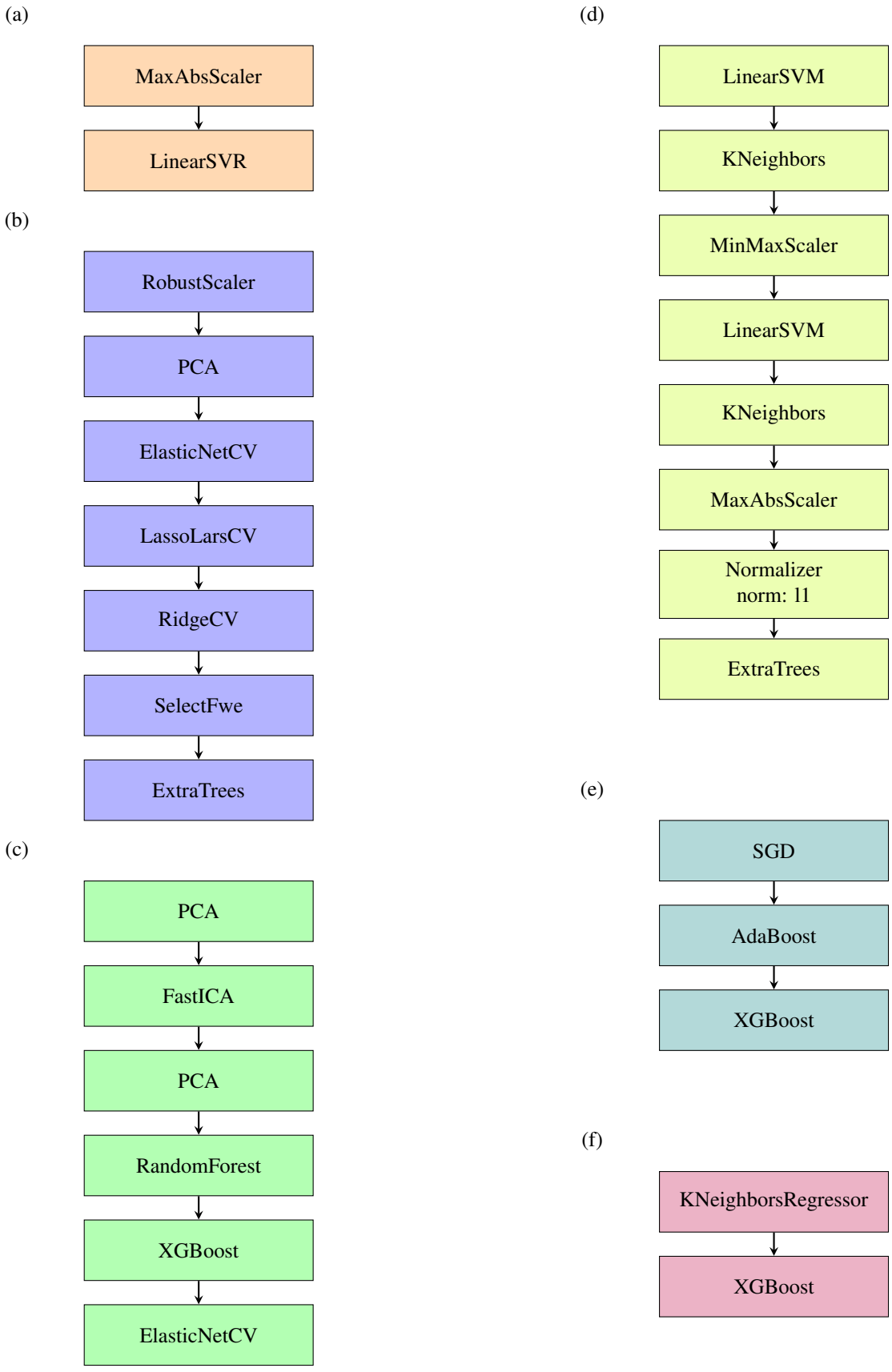
Tabela D.10. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji zaburzeń ze spektrum autyzmu.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	219	KernelPCA	186	FastICA	93	FastICA	10	IncrementalPCA	146	FastICA
Extra Trees	135	FactorAnalysis	186	FastICA	50	FastICA	31	FactorAnalysis	44	SelectKBest
KNN	299	KernelPCA	286	TruncatedSVD	229	KernelPCA	245	IncrementalPCA	278	PCA
LDA	96	KernelPCA	98	SelectKBest	12	FactorAnalysis	20	SelectKBest	102	FactorAnalysis
LGB	274	KernelPCA	298	KernelPCA	107	Dictionary Learning	19	Dictionary Learning	136	Dictionary Learning
LR	299	SelectKBest	283	KernelPCA	26	Dictionary Learning	85	FastICA	15	Dictionary Learning
MLP	18	Dictionary Learning	12	SelectKBest	228	FactorAnalysis	17	Dictionary Learning	30	Dictionary Learning
NuSVM	184	SelectKBest	278	FactorAnalysis	230	TruncatedSVD	174	SelectKBest	267	FactorAnalysis
QDA	151	FactorAnalysis	275	KernelPCA	136	FastICA	35	IncrementalPCA	226	KernelPCA
RF	268	KernelPCA	170	FastICA	10	FactorAnalysis	300	FactorAnalysis	50	SelectKBest
RNN	218	IncrementalPCA	65	Dictionary Learning	84	FastICA	106	KernelPCA	20	Dictionary Learning
SGD	63	SelectKBest	22	PCA	98	TruncatedSVD	12	FastICA	12	Dictionary Learning

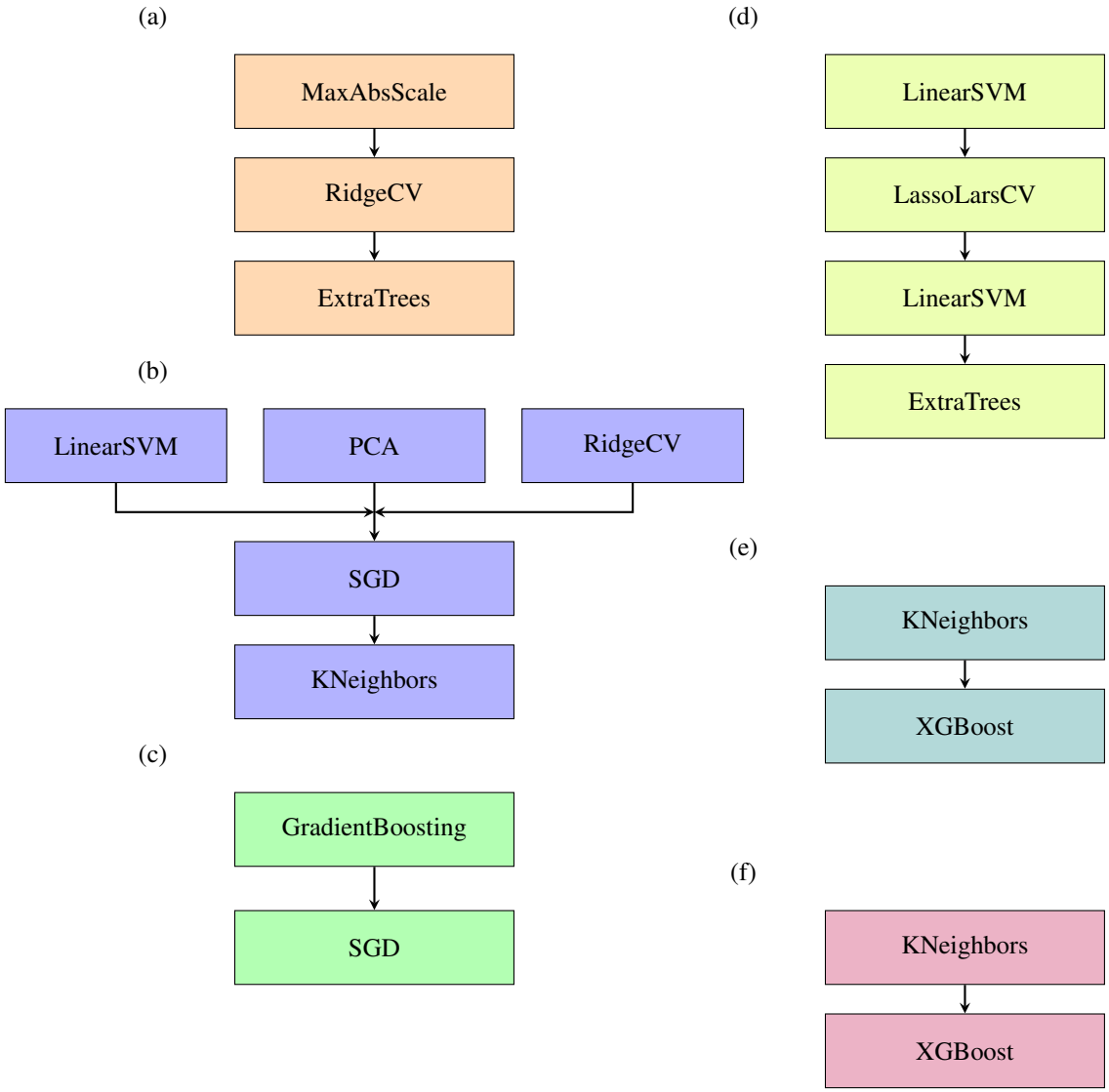
Tabela D.11. Techniki redukcji wymiarowości i liczba cech użyte przez poszczególne modele do detekcji specyficznych zaburzeń językowych.

	MFCC		GFCC		GFCC-Arc		GFCC-FTDNN		GFCC-FTDNN-RC0	
Model	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości	Liczba cech	Technika redukcji wymiarowości
AdaBoost	269	SelectKBest	234	KernelPCA	71	IncrementalPCA	287	SelectKBest	293	TruncatedSVD
Extra Trees	283	SelectKBest	284	SelectKBest	283	TruncatedSVD	80	TruncatedSVD	25	TruncatedSVD
KNN	45	PCA	252	TruncatedSVD	188	TruncatedSVD	66	PCA	215	SelectKBest
LDA	255	KernelPCA	19	PCA	26	FastICA	290	KernelPCA	245	FastICA
LGB	277	SelectKBest	262	KernelPCA	118	TruncatedSVD	300	TruncatedSVD	300	SelectKBest
LR	33	PCA	10	KernelPCA	11	PCA	269	TruncatedSVD	12	IncrementalPCA
MLP	242	FactorAnalysis	116	PCA	284	TruncatedSVD	277	TruncatedSVD	279	TruncatedSVD
NuSVM	242	FactorAnalysis	123	TruncatedSVD	292	PCA	261	TruncatedSVD	292	TruncatedSVD
QDA	159	FactorAnalysis	264	Dictionary Learning	185	KernelPCA	275	FactorAnalysis	185	SelectKBest
RF	195	SelectKBest	10	FastICA	173	TruncatedSVD	287	SelectKBest	206	SelectKBest
RNN	15	FastICA	10	TruncatedSVD	198	PCA	17	IncrementalPCA	145	PCA
SGD	283	FastICA	108	KernelPCA	287	FastICA	278	TruncatedSVD	222	KernelPCA

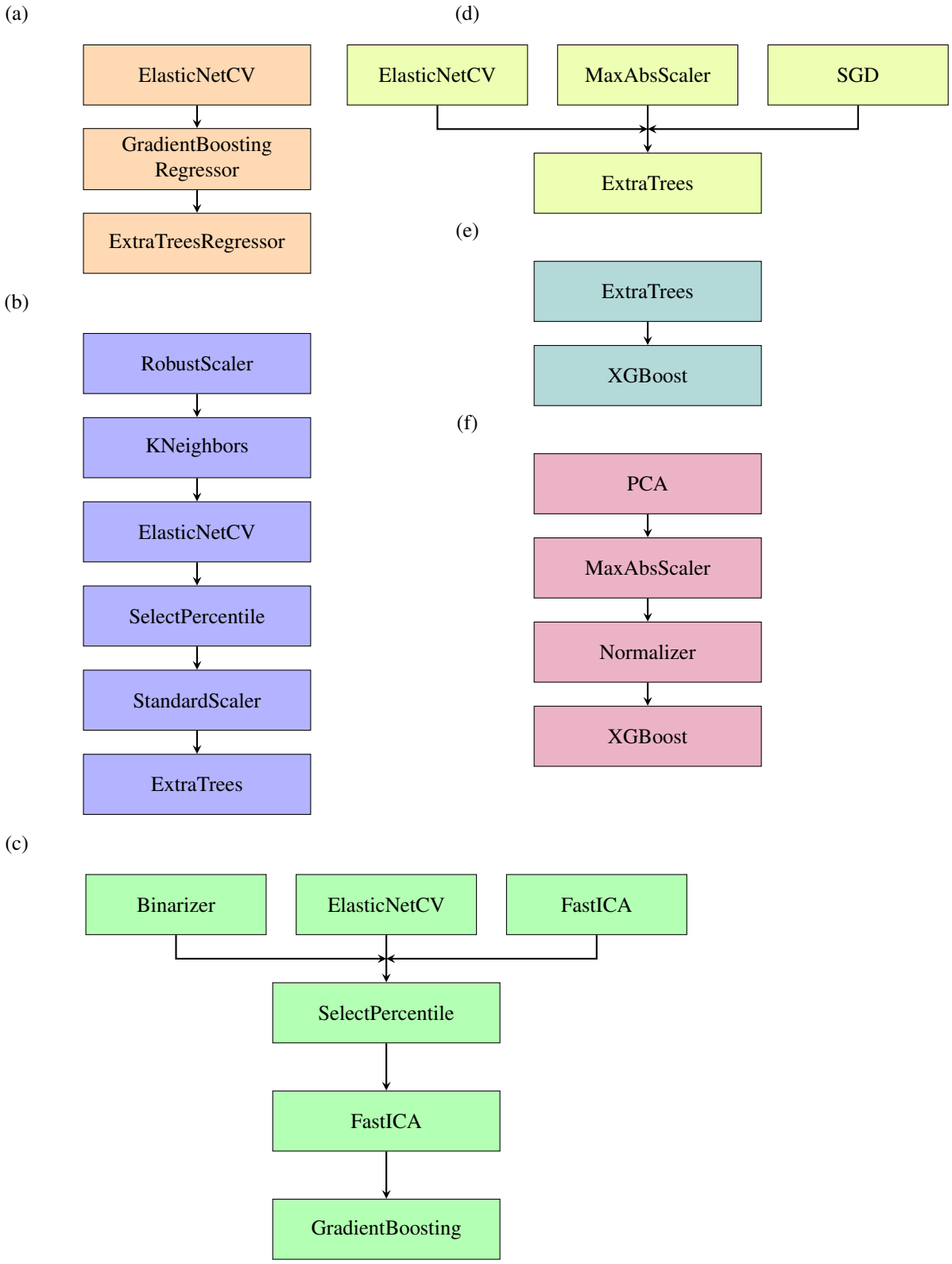
Dodatek E



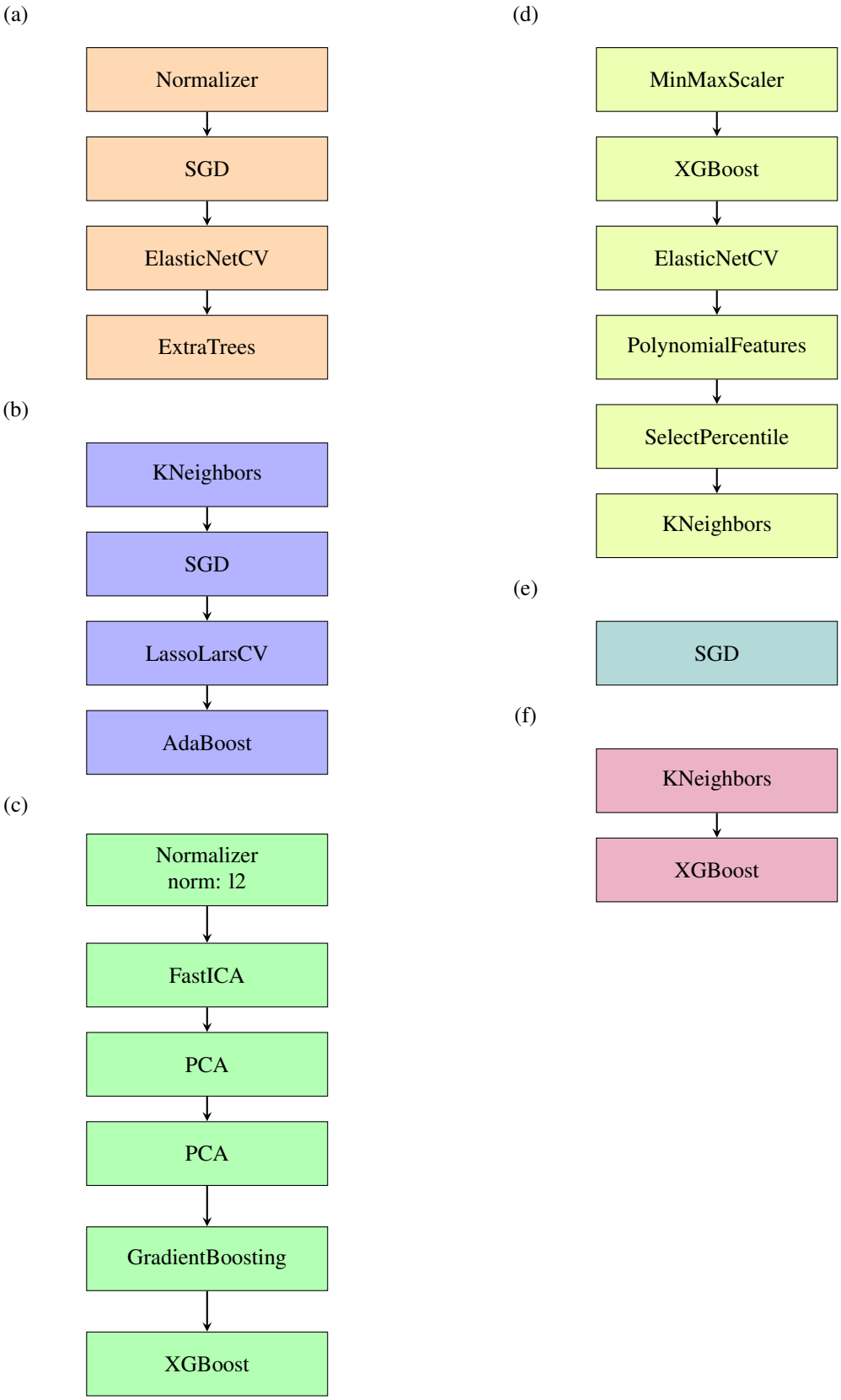
Rys. E.1. Schematy modeli o architekturze optymalizywanej genetycznie uczonych na cechach MFCC. Model wykorzystywany do detekcji: a) osób cierpiących na dowolne schorzenie krtani, b) ostrego zapalenia krtani, c) dysfonii, d) porażenia nerwu krtaniowego wstecznego, e) obrzęku Reinkego, f) pachydermii kontaktowej.



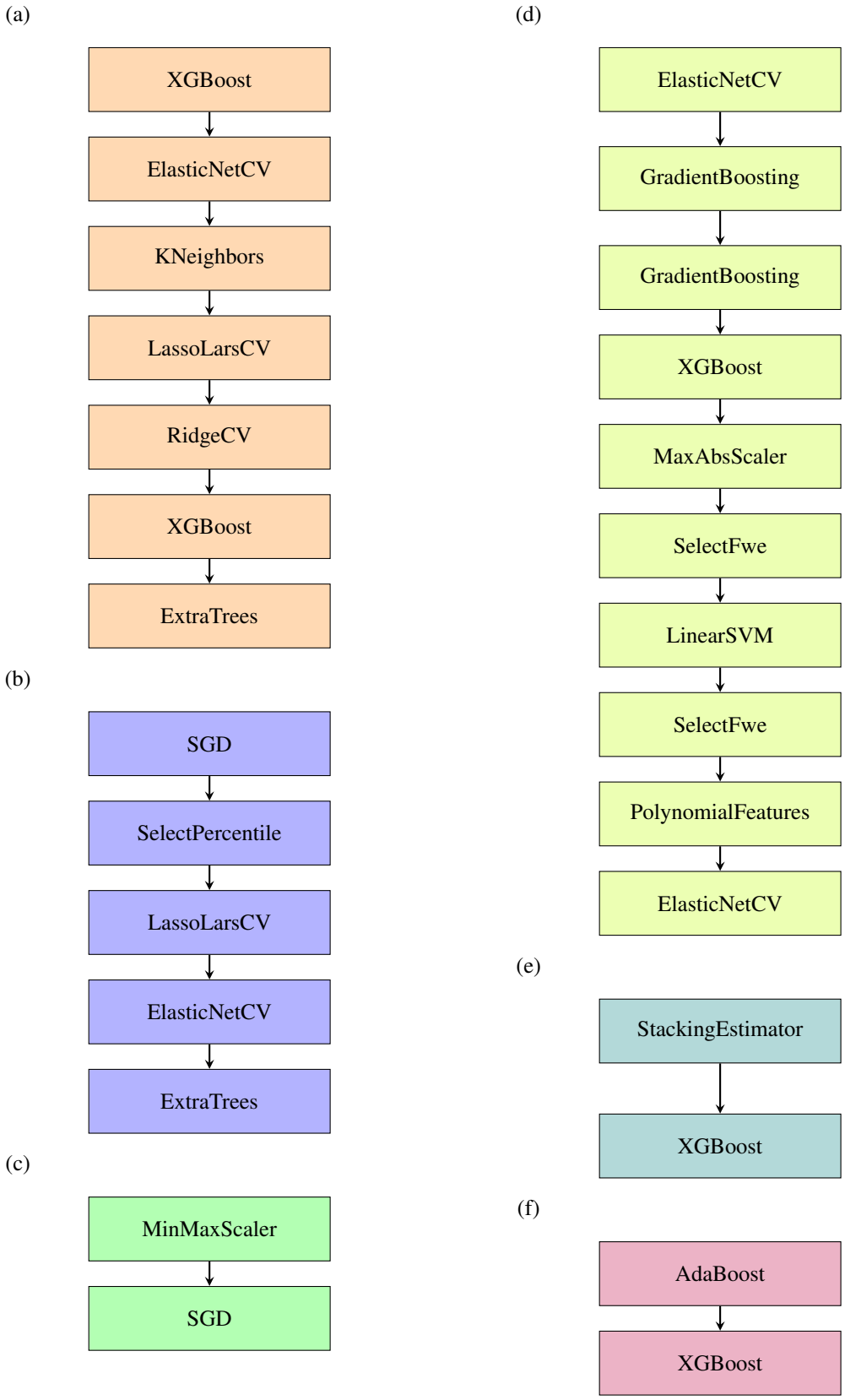
Rys. E.2. Schematy modeli o architekturze optymalizywanej genetycznie uczonych na cechach GFCC. Model wykorzystywany do detekcji: a) osób cierpiących na dowolne schorzenie krtani, b) ostrego zapalenia krtani, c) dysfonii, d) porażenia nerwu krtaniowego wstecznego, e) obrzęku Reinkego, f) pachydermii kontaktowej.



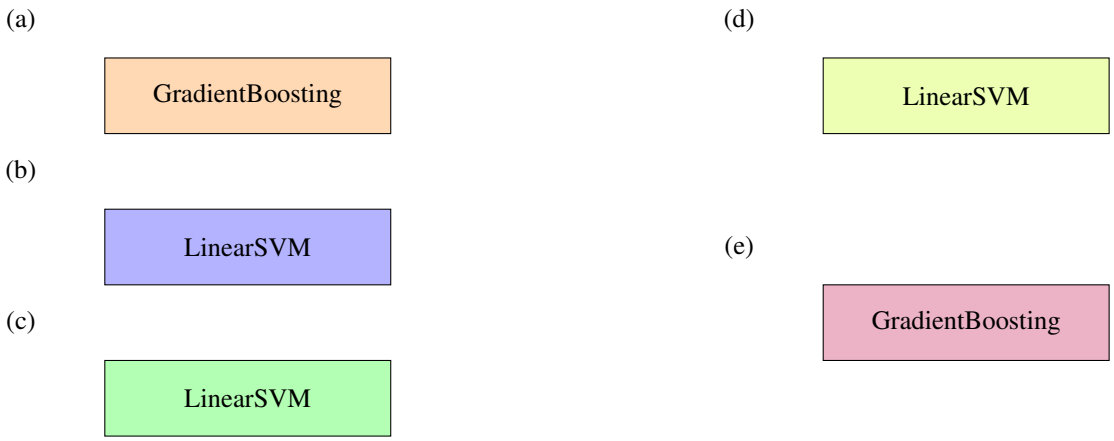
Rys. E.3. Schematy modeli o architekturze optymalizywanej genetycznie uczonych na cechach GFCC-Arc. Model wykorzystywany do detekcji: a) osób cierpiących na dowolne schorzenie krtani, b) ostrego zapalenia krtani, c) dysfonii, d) porażenia nerwu krtaniowego wstecznego, e) obrzęku Reinkego, f) pachydermii kontaktowej.



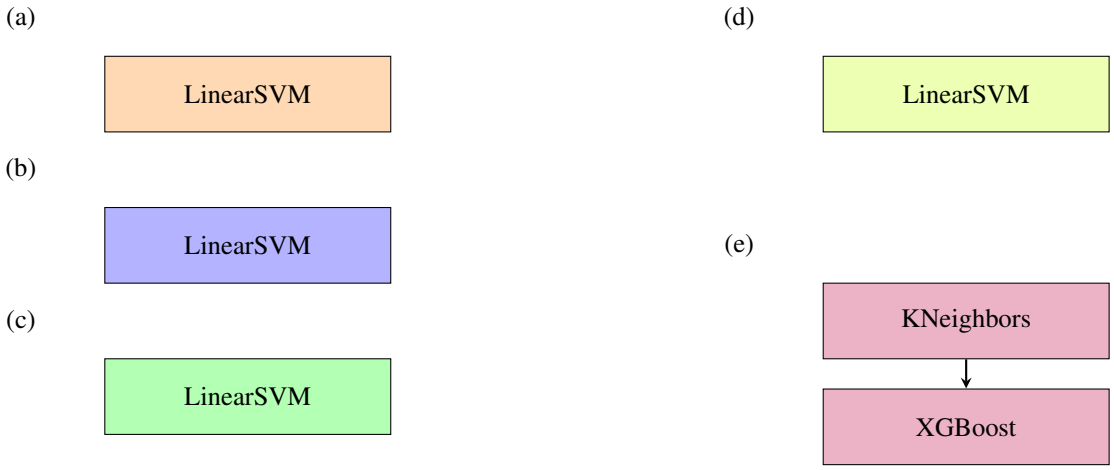
Rys. E.4. Schematy modeli o architekturze optymalizywanej genetycznie uczonych na cechach GFCC-FTDNN. Model wykorzystywany do detekcji: a) osób cierpiących na dowolne schorzenie krtani, b) ostrego zapalenia krtani, c) dysfonii, d) porażenia nerwu krtaniowego wstecznego, e) obrzęku Reinkego, f) pachydermii kontaktowej.



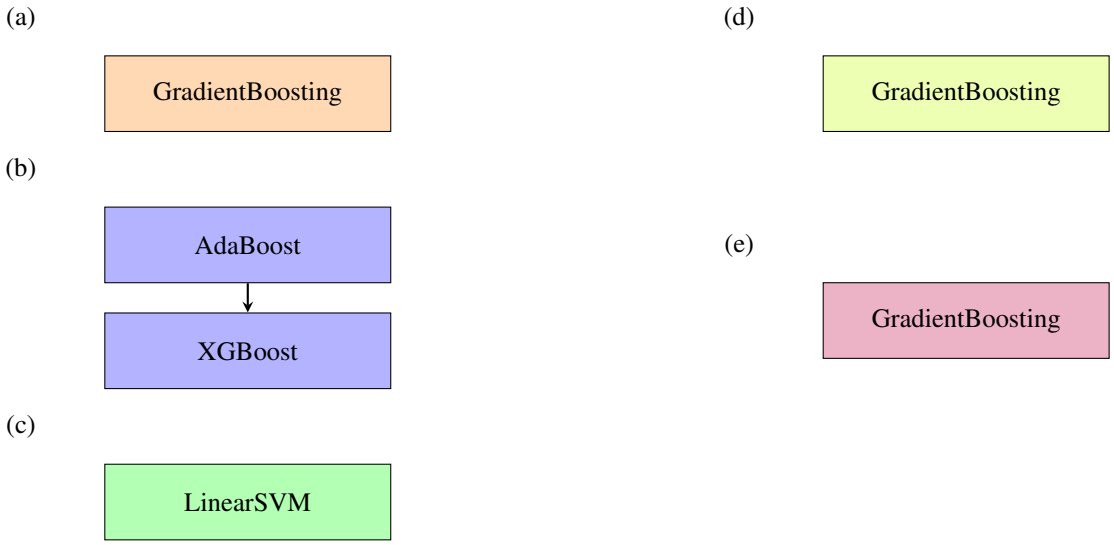
Rys. E.5. Schematy modeli o architekturze optymalizywanej genetycznie uczonych na cechach GFCC-FTDNN-RC0. Model wykorzystywany do detekcji: a) osób cierpiących na dowolne schorzenie krtani, b) ostrego zapalenia krtani, c) dysfonii, d) porażenia nerwu krtaniowego wstecznego, e) obrzęku Reinkego, f) pachydermii kontaktowej.



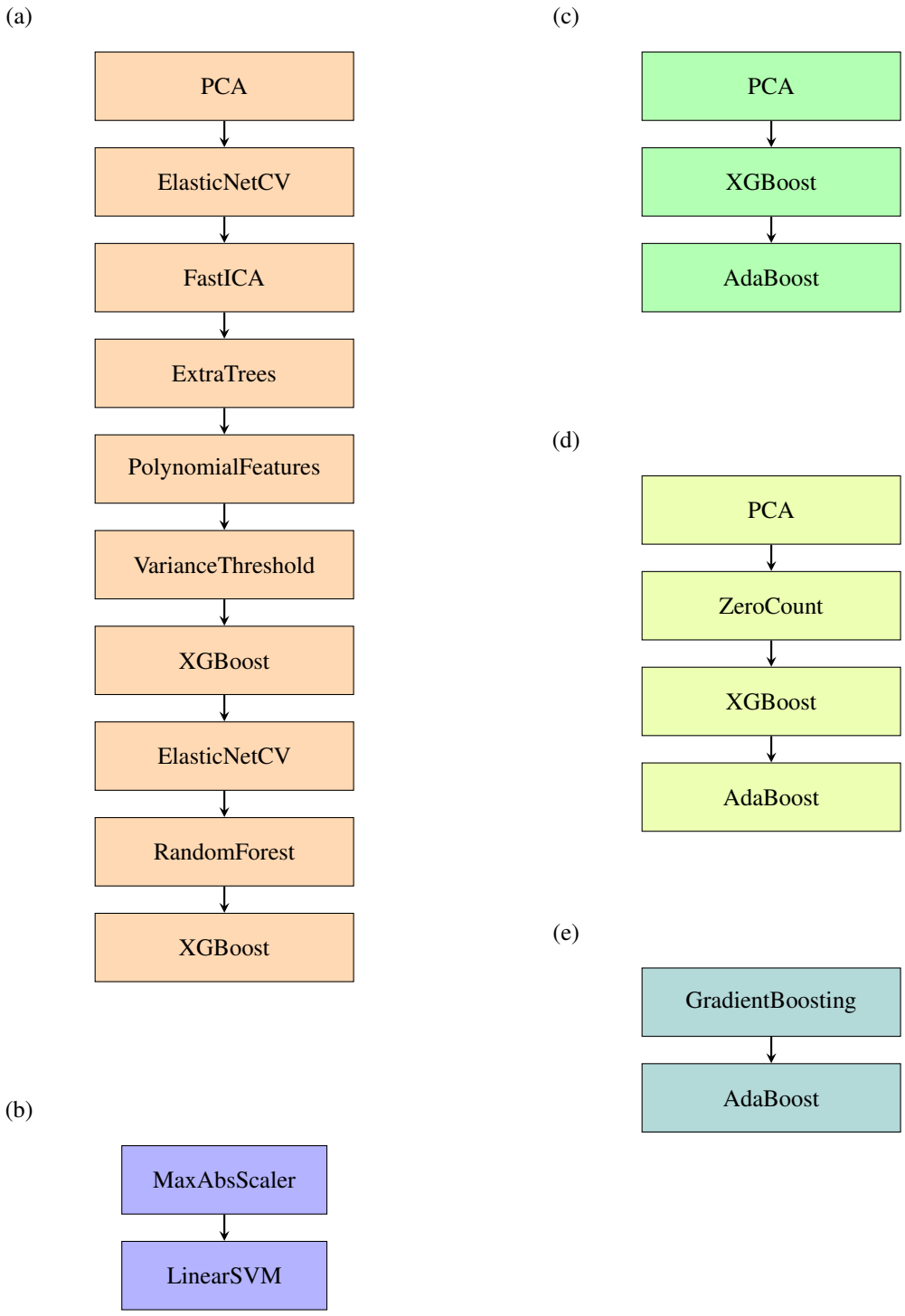
Rys. E.6. Schematy modeli o architekturze optymalizywanej genetycznie wykorzystywanych do detekcji dysfonii funkcjonalnej uczonych na wariancie: a) MFCC, b) GFCC, c) GFCC-Arc, d) GFCC-FTDNN, e) GFCC-FTDNN-RC0.



Rys. E.7. Schematy modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji dysfonii psychogennej uczonych na wariancie: a) MFCC, b) GFCC, c) GFCC-Arc, d) GFCC-FTDNN, e) GFCC-FTDNN-RC0.



Rys. E.8. Schematy modeli o architekturze optymalizowanej genetycznie wykorzystywanych do detekcji dysfonii spastycznej uczonych na wariancie: a) MFCC, b) GFCC, c) GFCC-Arc, d) GFCC-FTDNN, e) GFCC-FTDNN-RC0.



Rys. E.9. Schematy modeli o architekturze optymalizywanej genetycznie wykorzystywanych do detekcji zaburzeń ze spektrum autyzmu. Model uczony na cechach: a) MCC, b) GFCC, c) GFCC-Arc, d) GFCC-FTDNN, e) GFCC-FTDNN-RC0.