POLITECHNIKA WARSZAWSKA
Wydział Elektroniki i Technik Informacyjnych

ROZPRAWA DOKTORSKA

mgr inż. Piotr Płoński

Zastosowanie wybranych metod przekształcania i selekcji danych oraz konstrukcji cech w zadaniach klasyfikacji i klasteryzacji

Promotor
prof. dr hab. inż. Krzysztof Zaremba

Warszawa, 2016

Podziękowania

Przygotowanie niniejszej rozprawy było możliwe dzięki pomocy wielu osób, którym gorąco dziękuję! Przede wszystkim dziękuję promotorowi rozprawy, prof. Krzysztofowi Zarembie, za jego pomoc merytoryczną oraz cierpliwość okazane mi podczas przygotowania rozprawy. Dziękuję Robertowi Sulejowi i Dorocie Stefan za pokazanie mi fizyki neutrin z bliska. Za przedstawienie mi fascynującego świata obrazowania mózgu dziękuję Wojciechowi Gradkowskiemu, Katarzynie Jednoróg i prof. Piotrowi Bogorodzkiemu. Dziękuję Janowi Radomskiemu za zaszczepienie we mnie instynktu badacza oraz za przybliżenie świata bioinformatyki.

Osobne podziękowania za wytrzymałość, cierpliwość i wyrozumiałość w czasie mojej pracy nad rozprawą doktorską pragnę złożyć mojej rodzinie i przyjaciołom, w szczególności dziękuję mojej żonie Aleksandrze.

Streszczenie

Proces analizy danych wymaga ich odpowiedniego przygotowania. W rozprawie opisano wybrane metody używane do przygotowania danych, takie jak ich wstępne przekształcenie, selekcja i konstrukcja cech. Pominięcie któregoś z wyżej wymienionych etapów w procesie przygotowania danych może prowadzić do błędnych wniosków lub braku możliwości zbudowania systemu analizującego dane. Opisane metody zastosowano w zadaniach klasyfikacji i klasteryzacji w różnych dziedzinach nauki. Proces wyznaczania ważności cech został wykorzystany w rekonstrukcji drzewa ewolucyjnego dla wirusa grypy i próbie wyjaśnienia powtarzalności mutacji E391K. Proces wstępnego przekształcenia danych i selekcji cech wykorzystano w klasyfikacji dzieci pod względem posiadania dysleksji rozwojowej na podstawie badań metodą strukturalnego rezonansu magnetycznego. Konstrukcja oraz selekcja cech zostały wykorzystane do zbudowania systemu segmentującego tory cząstek oraz systemu klasyfikującego neutrina elektronowe na podstawie obrazów z detektora ciekłoargonowego z komorą projekcji czasowej.

Summary

Algorithms used in data analysis tasks require properly prepared features. Herein, the selected methods for data preparation are described, namely: data pre-processing, features selection and construction. Described methods are used in data analysis from various scientific disciplines. The feature selection is used in reconstruction of evolutionary tree for influenza virus. As a result of analysis the repetition of E391K mutation is explained. The pre-processing and feature selection is used for classification of developmental dyslexia in children based on structural magnetic resonance imaging. Features construction and selection are used in tasks: tracks segmentation and electron neutrino classification based on images from liquid argon time projection chamber detector.

Spis treści

Wstęp

Wprowadzenie

Rozwój elektroniki znacząco wpłynął na obniżenie kosztu przechowywania danych. Pozwala to na gromadzenie coraz większych ilości danych, których liczba rośnie wykładniczo z czasem. Dane gromadzone w tak dużej ilości są nieczytelne dla człowieka. Dlatego, aby móc skorzystać z informacji zawartej w danych, potrzebne są algorytmy potrafiące je analizować i wydobywać z nich użyteczne informacje.

Proces analizowania danych w celu pozyskania wiedzy nazywa się eksploracją danych (ang. *data mining*) [19], [78]. W zadaniu tym wykorzystuje się różne algorytmy pozwalające analizować dane, takie jak: metody uczenia maszynowego (ang. *machine learning*) [10], sztuczna inteligencja (ang. *artificial intelligence*) [116] oraz rozpoznawanie wzorców (ang. *pattern recognition*) [10]. Podczas analizy danych porusza się między innymi takie zagadnienia jak:

- klasyfikacja - podział obserwacji na klasy na podstawie cech tych obserwacji [52],
- regresja - przypisanie do obserwacji nieznanych wielkości (ciągłych) na podstawie wartości znanych cech [52],
- klasteryzacja (analiza skupień) - grupowanie próbek ze względu na podobieństwo między nimi [65],
- detekcja anomalii - wykrywanie próbek posiadających wzorzec inny niż oczekiwany [16],
- reguły asocjacyjne - wyznaczanie zależności (reguł) pomiędzy atrybutami w bazie danych [1].

Współczesne algorytmy analizy danych są bardzo rozwinięte, czego przykładem może być system klasyfikujący obrazy opracowany przez firmę Microsoft, który osiąga mniejszy błąd klasyfikacji niż człowiek [54], podobnie jak system rozpoznający twarze, którego liczba pomyłek jest zbliżona do osiąganej przez człowieka [123]. Jednakowoż, aby algorytmy eksplorujące dane działały z wysoką wydajnością, niezbędne są odpowiednio przygotowane dane wejściowe. Wedle popularnej w informatyce maksymy "śmieci na wejściu - śmieci na wyjściu" (ang. *garbage in, garbage out (GIGO)*) - przetwarzanie przez komputer błędnych danych wejściowych zwróci błędny wynik przy poprawnej procedurze przetwarzania. Dane dobrze przygotowane opisują analizowane zjawisko za pomocą małej, ale wystarczającej liczby odpowiednio przedstawionych cech. W procesie przygotowania danych można wskazać następujące ważne etapy:

- wstępne przekształcenie - transformacja danych, ich skalowanie do odpowiednich wartości, uzupełnianie brakujących wartości [29],
- konstrukcja cech - tworzenie nowych cech opisujących zjawisko w sposób bardziej zrozumiały dla analizującego algorytmu; proces ten może być przeprowadzony przez eksperta, dobrze znającego badane zjawisko, bądź automatycznie, za pomocą metod uczących się reprezentacji [8],
- selekcja cech - sprawdzanie jak dobrze cechy opisują analizowane zjawisko, w celu wybrania najistotniejszych cech, przypisania im wag, lub stworzenia ich rankingu [63].

Rozprawa obejmuje opis wybranych zagadnień związanych ze wstępnym przekształceniem, konstrukcją i selekcją cech oraz ich zastosowaniem w analizie danych z różnych dziedzin nauki, takich jak: bioinformatyka, medycyna oraz fizyka.

Cel i zakres pracy

Celem pracy jest przedstawienie wybranych metod przygotowania cech oraz ocena efektywności ich zastosowania w analizie danych pochodzących z różnych dziedzin nauki. W pracy zostanie pokazane, że pominięcie kroku przygotowania cech może prowadzić do:

- błędnych wniosków wyciągniętych na podstawie obserwowanych danych,

- braku możliwości zbudowania systemu analizującego dane.

W rozprawie zostaną omówione wybrane metody przygotowania danych:

- metody wstępnego przekształcenia danych,
- metody konstrukcji cech w analizie obrazów,
- metody selekcji cech, stosowane w klasyfikacji lub klasteryzacji.

Przedstawione metody zostaną użyte do:

- wyznaczania ważności nukleotydów w łańcuchu RNA, w celu wytłumaczenia ewolucji wirusa grypy,
- ujednolicenia danych pomiarowych z kilku ośrodków badawczych oraz ich selekcji w analizie danych z obrazowania za pomocą rezonansu magnetycznego, użytych do klasyfikacji dzieci ze względu na posiadanie dysleksji rozwojowej,
- konstrukcji wektora cech do segmentacji torów cząstek w obrazach z detektora ciekło-argonowego,
- konstrukcji wektora cech do klasyfikacji neutrin elektronowych na podstawie obrazów z detektora ciekło-argonowego.

Teza pracy

W zadaniach analizy danych wykorzystujących algorytmy ich klasyfikacji lub klasteryzacji etap wstępnego przygotowania danych poprzez ich przekształcanie, selekcję lub konstrukcję cech pozwala na zwiększenie, często znaczne, efektywności analizy.

Organizacja rozprawy

Na początku rozprawy (w rozdziale 1) przedstawiono algorytmy uczenia maszynowego. Zademonstrowano wybrane algorytmy klasyfikacji oraz klasteryzacji. Rozdział 2 zawiera opis różnych metod selekcji cech. Następnie, w rozdziale 3, przedstawiono metody używane do przekształcania danych, po czym, w rozdziale 4, scharakteryzowano metody konstrukcji

cech używane w analizie obrazu. W dalszej części rozprawy opisano zastosowania wcześniej przedstawionych metod pozwalających na przygotowanie cech do analizy.

W rozdziale 5 opisano sposób wyznaczenia istotności nukleotydów w łańcuchu RNA wirusa grypy, w celu wyjaśnienia obserwowanych mutacji. Selekcja cech i ujednolicenie danych zostały wykorzystane w analizie danych pochodzących z badań metodą rezonansu magnetycznego dzieci z dysleksją (rozdział 6). Metody konstrukcji cech zostały wykorzystane do segmentacji torów oraz klasyfikacji neutrin elektronowych na podstawie danych pochodzących z detektora ciekło-argonowego (rozdział 7). Podsumowanie i wnioski zaprezentowano w rozdziale 8.

Rozdział 1

Algorytmy uczenia maszynowego

1.1 Algorytmy klasyfikacji

Klasyfikacja jest to zadanie przydziału do klasy próbki wejściowej, dla której klasa nie jest znana [19], [10], [52]. Przypisania do klasy dokonuje wcześniej nauczony model, nazywany klasyfikatorem, na podstawie znanych wartości cech. Klasyfikator uczony jest za pomocą próbek, dla których klasa oraz wartości cech są znane - jest to tak zwane uczenie z nadzorem (ang. *supervised learning*).

Istnieje wiele algorytmów klasyfikacji. Nie można wskazać wśród nich jednego, zawsze osiągającego najlepszą skuteczność [36]. W niniejszej pracy przedstawiono wybrane algorytmy klasyfikacji: regresję logistyczną, drzewa decyzyjne, lasy losowe, maszynę wektorów nośnych. Wybrane algorytmy charakteryzują się stosunkowo prostą budową, aczkolwiek zapewniają wystarczającą skuteczność w analizowanych problemach. Przedstawione zostały również techniki stosowane do oceny klasyfikatorów i opisany został problem nadmiernego dopasowania.

Oznaczmy zbiór danych $D = \{(\vec{x}_1, y_1), ..., (\vec{x}_N, y_N)\}$, $\vec{x} \in \mathbb{R}^M$, $y \in \{C_1, .., C_K\}$, gdzie N to liczba próbek w zbiorze danych, M to wymiar próbki, K liczba klas. Klasyfikator h wyznacza dla każdej próbki wejściowej $\vec{x}_i$ odwzorowanie w klasie y, $h : x \rightarrow y$. Każdą próbkę opisuje wektor $\vec{x}$, który składa się z M liczb rzeczywistych, z których każda opisuje jedną cechę próbki (atrybut). Zbiór danych można też przedstawić jako zbiór atrybutów $\mathbf{X}_i$, gdzie jeden atrybut opisuje wartości jednej cechy dla wszystkich próbek, $D = \{\mathbf{X}_1, ..., \mathbf{X}_M, \mathbf{Y}\}$, a wektor $\mathbf{Y}$ opisuje przynależność próbki do klasy. Wartość atrybutu i-tego dla j-tej próbki będzie oznaczona jako $\mathbf{X}_i(j)$.

1.1.1 Regresja logistyczna

Jedną z powszechnie używanych metod klasyfikacji jest regresja logistyczna (RL) [52]. Klasyfikacja ta może być użyta, gdy próbka przypisywana jest do jednej z dwóch klas (klasyfikacja binarna). Wzór, na podstawie którego wyznaczane jest prawdopodobieństwo przynależności próbki $\vec{x}_i$ do klasy y, przedstawiony jest poniżej:

$$p(y|\vec{x}_i) = \frac{1}{1 - exp(\vec{W}\vec{x}_i)}, \tag{1.1}$$

gdzie $\vec{W}$ są to wagi modelu. Wagi zazwyczaj zawierają wagę zerową w_0, dodawaną do iloczynu $\vec{W}\vec{x}_i$ (ang. *intercept* lub *bias*), zapewniającą przesunięcie hiperpłaszczyzny decyzyjnej klasyfikatora w przestrzeni wag. Wagi w RL wyznaczane są na podstawie danych uczących. Istnieje kilka metod wyznaczania współczynników w RL. Do najczęściej stosowanych należą:

- metoda najmniejszych kwadratów (ang. *ordinary least squares*) [52],
- metoda wykorzystująca dekompozycję QR lub rozkład Cholesky'ego [52],
- metoda gradientowa (ang. *stochastic gradient descent*) [128].

Regresja logistyczna uczona metodą gradientową może być realizowana przez sieć neuronową z tylko jednym neuronem. W regresji logistycznej łatwo zinterpretować uzyskany model, ponieważ do każdej cechy przypisana jest jedna waga - co stanowi dużą zaletę tego klasyfikatora.

1.1.2 Drzewa decyzyjne

Drzewo decyzyjne to klasyfikator, reprezentowany przeważnie przez drzewo binarne [107]. Węzły drzewa opisują podział zbioru danych ze względu na wartość wybranej cechy, natomiast w liściach drzewa odbywa się przypisanie próbek do klasy. W trakcie budowania drzewa tworzone są reguły decyzyjne, którym odpowiadają węzły. Drzewo decyzyjne zazwyczaj konstruowane jest zstępująco - zaczynając od głównego węzła i dodając kolejne węzły potomne (ang. *top-down induction of decision trees*) [19]. Do wyboru reguły decyzyjnej w węźle drzewa można użyć różnych kryteriów [73], na przykład:

- przyrostu informacji (ang. *information gain*),

- współczynnika przyrostu informacji (ang. *gain ratio*),
- indeks Gini'ego (ang. *Gini's index*).

W problemach analizowanych w rozprawie używane będzie kryterium przyrostu informacji. Przyrost informacji jest wyznaczany na podstawie wzoru:

$$IG(\mathbf{X}, t) = E(\mathbf{X}) - \sum_j p_j E(\mathbf{X}^j), \tag{1.2}$$

gdzie:

- $E(\mathbf{X})$ przedstawia entropię rozkładu prawdopodobieństwa przynależności do klas:

$$E(\mathbf{X}) = -\sum_i p_i log(p_i), \tag{1.3}$$

- p_i jest prawdopodobieństwem wystąpenia klasy y_i w zbiorze zdefinowanym przez $\mathbf{X}$:

$$p_i = \frac{|y_i|}{|X|}, \tag{1.4}$$

- t jest testem atrybutu i w przypadku atrybutu ciągłego opisany jest wzorem:

$$t = \begin{cases} 1 & \text{jeżeli } \mathbf{X}(l) < c, \\ 0 & \text{w przeciwnym wypadku,} \end{cases} \tag{1.5}$$

  l to indeks próbki, natomiast c - wartość progu decyzyjnego,

- test t wyznacza podział zbioru $\mathbf{X}$ na j rozłącznych podzbiorów, taki że $\mathbf{X} = \{\mathbf{X}^1 \bigcup \mathbf{X}^2 \bigcup ... \bigcup \mathbf{X}^j\}$.

Uczenie drzewa zatrzymywane jest w momencie spełnienia kryterium stopu lub występowania tylko jednej próbki w węźle drzewa (liściu) [19]. Istnieje wiele różnych kryteriów zatrzymania uczenia drzewa. W rozprawie będą używane dwa rodzaje kryterium stopu:

- zatrzymanie uczenia drzewa, gdy osiągnie ono określoną głębokość (liczbę węzłów); w rozprawie by obliczyć skuteczność klasyfikatora wykorzystującego tylko jedną zmienną używane będą drzewa o głębokości równej 1, to znaczy tylko z jednym węzłem, drzewa takie nazywa się *decision stump* [60],

- uczenie drzewa do momentu, gdy w liściu będą tylko próbki pochodzące z tej samej klasy; drzewa takie nazywa się drzewami pełnymi.

W trakcie wyznaczania przez drzewo etykiety klasy dla próbki, jest ona poddawana kolejnym testom w węzłach drzewa, zaczynając od jego korzenia. Na koniec propagacji próbki przez drzewo, próbce jest przypisywana klasa jaką posiada liść.

Drzewa decyzyjne to popularne narzędzie do klasyfikacji, ponieważ ich struktura jest zrozumiała dla człowieka - łatwo sprawdzić jakie atrybuty są używane przy podejmowaniu decyzji. Niestety, drzewa decyzyjne posiadają kilka wad [13]:

- niestabilność - nawet przy małej zmianie zbioru uczącego uzyskane reguły decyzyjne w węzłach mogą się zmienić,
- niska skuteczność - przy złożonych zbiorach pojedyncze drzewa decyzyjne mogą osiągać niską skuteczność klasyfikacji.

1.1.3 Las Losowy

Las losowy (ang. *random forest*) [13] to zbiór klasyfikatorów (ang. *ensemble classifier*), w którym każdy pojedynczy klasyfikator jest drzewem decyzyjnym uczonym bez zatrzymywania (do momentu aż w liściu będą tylko próbki z tej samej klasy). Każdy klasyfikator wchodzący w skład lasu losowego jest uczony na specjalnie wylosowanej dla niego próbce danych D', która powstaje przez wylosowanie n razy ze zwracaniem ze wszystkich N próbek uczących. Taka technika generowania danych nazywana jest *bagging* lub też *bootstrap aggregating* [14]. Dodatkowo, w trakcie budowania drzewa decyzyjnego nie wszystkie atrybuty są brane pod uwagę przy wyznaczaniu reguły decyzyjnej w węźle. Losowanych jest m atrybutów, gdzie $m < M$, na podstawie których wyznaczana jest reguła decyzyjna w węźle. Taka metoda nazywana jest *feature subspace*. Obie techniki: *bagging* oraz *feature subspace* stosowane są w celu zwiększenia stabilności odpowiedzi algorytmu i jego ochrony przed nadmiernym dopasowaniem do danych uczących, co przekłada się na lepszą skuteczność działania klasyfikatora na nowych danych. W trakcie przewidywania klasy dla próbki jest ona pokazywana wszystkim drzewom decyzyjnym, a jako końcowa decyzja o przynależności do klasy traktowane jest przypisanie do klasy najczęściej wskazywanej przez drzewa decyzyjne w lesie. W algorytmie lasu losowego każde drzewo może być wyznaczone niezależnie, dzięki czemu obliczenia potrzebne do wyznaczenia całego lasu mogą

być łatwo zrównoleglone. Las losowy ma też tę zaletę, że potrafi wyznaczyć ważność atrybutów wejściowych - algorytm ten opisano w rozdziale 2.2.

Las losowy zapewnia lepszą stabilność działania i zazwyczaj lepsze wyniki klasyfikacji niż pojedyncze drzewo decyzyjne, jednakże jednocześnie tracona jest łatwość interpretacji działania klasyfikatora. Autor rozprawy zaproponował metodę interpretacji działania lasu losowego za pomocą sieci samoorganizujących się Kohonena, uczących się z wykorzystaniem macierzy podobieństwa pomiędzy danymi a wagami neuronów, otrzymywanej z lasu losowego [102].

1.1.4 Maszyna wektorów nośnych

Algorytm Maszyny Wektorów Nośnych (ang. *Support Vector Machine*) [12] wyznacza hiperpłaszczyznę oddzielającą próbki z różnych klas z maksymalnym marginesem pomiędzy nimi. Dla zbioru danych $D = \{(\vec{x}_1, y_1), ..., (\vec{x}_N, y_N)\}$, gdzie $y \in \{-1, 1\}$, równanie funkcji decyzyjnej można zapisać następująco:

$$h(\vec{x}_i) = sign(\mathbf{w}\vec{x}_i + b). \tag{1.6}$$

Aby zapewnić maksymalną szerokość marginesu pomiędzy próbkami z różnych klas, należy zminimalizować wartość modułu wektora $||\mathbf{w}||$ przy zachowaniu separowalności klas. Jest to problem optymalizacyjny, który można rozwiązać za pomocą programowania kwadratowego, gdzie minimalizowana jest funkcja celu zapisana jako:

$$\mathcal{F}(\mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w}||^2, \tag{1.7}$$

przy zachowaniu warunku:

$$y_i(\mathbf{w}\vec{x}_i + b) \geqslant 1. \tag{1.8}$$

Niestety, w rzeczywistości analizowane zbiory danych zazwyczaj nie są separowalne liniowo. W związku z powyższym, warunek separowalności klas należy rozluźnić, wprowadzając do równania (1.8) zmienne pomocnicze ξ (ang. *slack variables*) kontrolujące poziom błędnie sklasyfikowanych próbek [22]. Wyrażenie (1.8) przyjmuje wtedy postać:

$$y_i(\mathbf{w}\vec{x}_i + b) \geqslant 1 - \xi_i, \text{ gdzie } \xi_i \geqslant 0. \tag{1.9}$$

Funkcja celu zapewniająca maksymalny margines pomiędzy próbkami z różnych klas przybiera postać:

$$\mathcal{F}(\mathbf{w}) = \underset{\mathbf{w},\xi}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i, \tag{1.10}$$

gdzie parametr C kontroluje balans pomiędzy dokładnością a złożonością modelu. Dodatkowym sposobem na analizę danych nieseparowalnych liniowo jest zastosowanie przekształcenia jądrowego (ang. *kernel trick*) [22], [18]. Transformacja ta zmienia przestrzeń próbki wejściowej na inną przestrzeń, zazwyczaj o większej wymiarowości, w której separacja próbek jest łatwiejsza. Do najczęściej stosowanych przekształceń należą funkcje wielomianowe, gaussowskie, sigmoidalne.

1.1.5 Klasyfikator zbiorowy

Czasami skuteczność pojedynczego klasyfikatora jest niewystarczająca. W takich przypadkach dokładność klasyfikacji można spróbować zwiększyć przez zastosowanie klasyfikatora zbiorowego (ang. *ensemble classifier*) [28], [52]. Klasyfikator ten składa się ze zbioru zróżnicowanych klasyfikatorów, a jego odpowiedź to zagregowana odpowiedź klasyfikatorów w zbiorze. W działaniu klasyfikatora zbiorowego kluczowe jest zróżnicowanie używanych klasyfikatorów [131]. W sytuacji gdyby nie były one zróżnicowane, wszystkie odpowiedzi klasyfikatorów w zbiorze byłyby podobne, a zatem wszystkie popełniałyby błąd dla tych samych próbek. Dzięki użyciu zróżnicowanych klasyfikatorów dla rozpatrywanej próbki, część z nich będzie popełniać błąd, a część będzie odpowiadać poprawnie i jeżeli większość klasyfikatorów odpowie dobrze, to cały system klasyfikatorów również zwróci poprawną odpowiedź. Zróżnicowanie klasyfikatorów można uzyskać poprzez:

- użycie różnych, wylosowanych podzbiorów danych uczących; popularną techniką jest*bagging* [14], gdzie dla każdego klasyfikatora losowany jest inny zbiór uczący;
- użycie różnych podzbiorów atrybutów dla każdego klasyfikatora [58]; podzbiory atrybutów można losować, tak jak to się dzieje w algorytmie Lasu Losowego [13]; podejście to nazywa się losowym cięciem przestrzeni (ang. *random feature partitioning*), lub podzbiory atrybutów dla kolejnych klasyfikatorów mogą być wyznaczane tak, by zapewnić różnorodność klasyfikatora w stosunku do wcześniej wybranych atrybutów [97];

- zastosowanie różnych algorytmów do zbudowania klasyfikatora [74]; w podejściu tym można używać różnych algorytmów, lub tego samego algorytmu klasyfikacji z różnymi parametrami;
- naukę klasyfikatora na błędach poprzednich klasyfikatorów (metoda *boosting*); popularnym algorytmem wykorzystującym ten sposób konstrukcji klasyfikatora zbiorowego jest *AdaBoost* [42].

Po zbudowaniu zbioru klasyfikatorów ich odpowiedź jako klasyfikatora zbiorowego może być wyznaczana w różny sposób [112], na przykład:

- jako średnia arytmetyczna wszystkich odpowiedzi,
- za pomocą głosowania, gdzie wybierana jest najczęściej wskazywana klasa,
- za pomocą następnego klasyfikatora, który uczony jest odpowiedziami uzyskanymi z klasyfikatorów [64].

Budowanie klasyfikatora zbiorowego jest kosztowne obliczeniowo, ponieważ budowanych jest jednocześnie kilka klasyfikatorów, jednakże często dzięki temu rozwiązaniu można zyskać na jakości systemu klasyfikacji, natomiast czas obliczeń można skrócić poprzez zrównoleglenie budowania klasyfikatorów.

1.1.6 Ocena klasyfikatora

Sprawdzian krzyżowy

Każdy wytrenowany klasyfikator należy ocenić, by poznać jego jakość. W tym celu niezbędne są dwa zbiory danych. Pierwszy, przeznaczony do nauki klasyfikatora, to tzw. zbiór uczący, lub inaczej - treningowy. Drugi zbiór używany jest do przetestowania klasyfikatora i nie jest wykorzystywany w trakcie nauki - jest on nazywany zbiorem testującym, bądź też walidacyjnym. W obu zbiorach konieczna jest znajomość przynależności próbek do klas.

Bardzo często podział danych na zbiory: uczący i testujący nie jest zadany z góry. W takim przypadku dokonuje się ich losowego podziału na dwa rozłączne zbiory danych, zwykle ze zbiorem testującym mniejszym od uczącego. Taki podział danych nazywany jest prostą walidacją. Dokładniejszą estymatę oceny klasyfikatora zapewnia k-krotna walidacja - nazywana k-krotnym sprawdzianem krzyżowym lub też kroswalidacją [72], [19]

(ang. *k-fold cross validation*). W metodzie tej oryginalny zbiór dzielony jest na K podzbiorów. Następnie, każdy z podzbiorów używany jest jako testujący, a klasyfikator uczony jest na $K-1$ pozostałych podzbiorach. W ten sposób walidacja powtórzona jest K razy, a końcowy wynik jest najczęściej średnią ze wszystkich powtórzeń. Przy małych zbiorach danych często stosuje się k-krotną walidację, gdzie liczba podzbiorów równa się liczbie wszystkich próbek, $K = N$. W tym przypadku w zbiorze testowym jest tylko jedna próbka. Taka kroswalidacja nazywana jest *leave-one-out* (LOO) [72].

Metryki do oceny klasyfikatora

Do oceny klasyfikatora stosowane są różne metryki [33]. W celu przedstawienia używanych w rozprawie metryk należy przyjąć oznaczenia jak w tabeli 1.1 dla różnych przypadków odpowiedzi klasyfikatora h w zależności od prawdziwej wartości klasy dla próbki. W klasyfikacji binarnej przyjęto, że klasa 0 jest klasą ujemną (negatywną), a próbki należące do klasy 1 nazywane są dodatnimi (lub pozytywnymi). Możliwe przypadki odpowiedzi klasyfikatora oznaczone są następująco:

- TP (ang. *true positive*) - prawdziwie dodatni - ozaczna przypadek poprawnie sklasyfikowany jako pozytywny (klasa 1),
- TN (ang. *true negative*) - prawdziwie ujemny - oznacza próbkę poprawnie klasyfikowaną jako negatywna (klasa 0),
- FP (ang. *false positive*) - fałszywie dodatni - oznacza próbkę fałszywie przypisaną do klasy pozytywnej (klasa 1),
- FN (ang. *false negative*) - fałszywie negatywny - oznacza przypadek sklasyfikowany błędnie jako negatywny (klasa 0).

		odpowiedź klasyfikatora h	
		klasa 0	klasa 1
prawdziwa wartość	klasa 0	TN	FP
	klasa 1	FN	TP

Tabela 1.1: Oznaczenia przypadków odpowiedzi klasyfikatora h w zależności od prawdziwych wartości klasy.

W rozprawie do oceny klasyfikatora używane są następujące metryki:

- dokładność klasyfikatora (ang. *accuracy*), przedstawiająca stosunek liczby wszystkich poprawnie sklasyfikowanych próbek do liczby próbek:

$$\text{dokładność} = \frac{TP + TN}{TP + TN + FP + FN} \ , \tag{1.11}$$

- czułość klasyfikatora (ang. *sensitivity* lub też *true positive rate* (TPR) albo *recall*), określająca stusunek liczby poprawnych odpowiedzi dla klasy 1 do wszystkich próbek należących rzeczywiście do klasy 1:

$$\text{TPR} = \frac{TP}{TP + FN} \ , \tag{1.12}$$

- *false positive rate* lub też *fall-out* określa stosunek liczby próbek sklasyfikowanych błędnie jako pozytywne do wszystkich próbek należących do klasy 0:

$$\text{FPR} = \frac{FP}{FP + TN} \ , \tag{1.13}$$

- precyzja (ang. *precision*) określa jaka część próbek wskazanych przez klasyfikator jako pozytywne należy rzeczywiście do klasy 1:

$$\text{precyzja} = \frac{TP}{TP + FP} \ , \tag{1.14}$$

- specyficzność (ang. *specificity* lub też *true negative rate*) określa stosunek próbek poprawnie sklasyfikowanych jako negatywne do liczby wszystkich próbek należących do klasy 0:

$$\text{specyficzność} = \frac{TN}{TN + FP} \ , \tag{1.15}$$

Odpowiedź klasyfikatora może być dyskretna - wtedy klasyfikator wskazuje klasę, bądź ciągła - gdy klasyfikator wskazuje prawdopodobieństwo wystąpienia klasy. W przypadku odpowiedzi ciągłej należy wskazać próg decyzyjny θ, powyżej którego przyjmowana jest

przynależność do klasy 1:

$$h(\mathbf{X}_i) = \begin{cases} \text{klasa 1}, & \text{jeżeli } p(y|\mathbf{X}_i) > \theta, \\ \text{klasa 0}, & \text{w przeciwnym wypadku.} \end{cases} \tag{1.16}$$

Sprawdzając różne wartości progu θ w przedziale $(0; 1)$ i wyznaczając wartości TPR i FPR, otrzymamy krzywą ROC (ang. *Receiver Operating Characteristic*). Wyznaczając pole powierzchni pod krzywą ROC otrzymamy metrykę służącą do oceny klasyfikatora, nazywaną AUC (ang. *Area Under Curve)*. Natomiast wyznaczając wartości dla precyzji i czułości otrzymamy wykres *precision-recall*.

Dla przypadku ciągłej odpowiedzi klasyfikatora używana jest metryka *LogLoss*:

$$LogLoss = -\frac{1}{N} \sum \Big[y_i log\hat{y}_i + (1 - y_i) log(1 - \hat{y}_i \Big], \tag{1.17}$$

gdzie y_i to prawdziwa wartość klasy, a $\hat{y}_i$ to estymowane prawdopodobieństwo przynależności do klasy.

Test permutacji

W przypadku, gdy analizowany problem jest bardzo złożony, a klasyfikator uzyskuje niską skuteczność, istnieje potrzeba sprawdzenia z jakim prawdopodobieństwem nauczony klasyfikator jest lepszy od klasyfikatora losowego, czyli takiego, który losowo zgadywałby klasy dla klasyfikowanych próbek. W tym celu wykonuje się tak zwany test permutacji (ang. *permutation test*) [84]. W teście tym k-krotnie wyznacza się zbiór danych D' poprzez losowe przemieszanie klas dla próbek z oryginalnego zbioru D. Następnie wyznacza się dla każdego zbioru D' błąd klasyfikatora $E(D')$. Prawdopodobieństwo, że użyty klasyfikator działa jak klasyfikator losowy na danych D wyznacza się następująco:

$$p = \frac{|D' \in \hat{D} : E(D'_k) \leqslant E(D)| + 1}{k + 1}, \tag{1.18}$$

gdzie $\hat{D}$ jest zbiorem wszystkich wylosowanych zbiorów D'. Klasyfikator, który jest lepszy niż losowy powinien mieć wartości p bliskie zero. Zazwyczaj wartość $p < 0,05$ uznawana jest za granicę istotności.

Nadmierne dopasowanie

Niezbędną cechą klasyfikatora jest umiejętność generalizowania informacji zawartych w zbiorze uczącym. Dlatego też istotne jest, żeby w trakcie nauki klasyfikator za bardzo nie dopasował się do danych uczących - jest to problem nadmiernego dopasowania (ang. *overfitting*). W przypadku nadmiernego dopasowania klasyfikator będzie działał bardzo dobrze na zbiorze uczącym, natomiast jego skuteczność na zbiorze testującym będzie słaba - nie będzie potrafił generalizować informacji. Aby uniknąć nadmiernego dopasowania w klasyfikatorze stosuje się różne techniki:

- zastosowanie dodatkowego zbioru walidacyjnego, który nie bierze udziału w nauce, ale jest używany do sprawdzania generalizacji; nauka klasyfikatora jest przerywana, jeżeli skuteczność na zbiorze walidacyjnym zaczyna maleć;
- zastosowanie modeli z małą liczbą parametrów, niepozwalającą na nadmierne dopasowanie,
- zastosowanie metod takich jak *bagging* [14] lub *random feature subspace* [58].

Używane algorytmy klasyfikacji, takie jak regresja logistyczna i jedno-poziomowe drzewa decyzyjne mają małą liczbę parametrów i przy złożonych problemach nie są w stanie nadmiernie dopasować się do danych uczących. Natomiast algorytm lasu losowego używa połączonych metod *bagging* oraz *random feature subspace* by zapewnić dobrą generalizację na nowych danych.

1.2 Algorytmy klasteryzacji

Algorytmy klasteryzacji (ang. *clustering*) [65] mają za zadanie przyporządkować dane do podobnych do siebie grup. W przypadku tych algorytmów informacja o klasie nie jest używana. Jest to tzw. uczenie bez nadzoru (ang.*unsupervised learning*), a próbki przyporządkowane są na podstawie podobieństwa pomiędzy nimi. Istnieje wiele algorytmów klasteryzujących, np. sieci samoorganizujące się Kohonena [106], [105], algorytm k-średnich [103], algorytm gazu neuronowego [104].

W procesie grupowania próbek rozwiązywany jest problem odkrywania struktury danych poprzez wyznaczenie grup lub zbudowanie hierarchii pomiędzy próbkami - zazwyczaj przedstawianej za pomocą dendrogramu. Algorytmy klasteryzacji mają wiele za-

stosowań, np. w segmentacji obrazu, grupowaniu dokumentów, rozpoznawaniu obiektów [65]. Szczególnym przypadkiem zastosowania jest omawiane w pracy użycie klasteryzacji w filogenetyce, gdzie za pomocą dendrogramu przedstawia on historię ewolucji pomiędzy jednostkami, którymi mogą być np. gatunki lub sekwencje łańcucha DNA (kwas deoksyrybonukleinowy) albo RNA (kwas rybonukleinowy). Uzyskany w wyniku klasteryzacji dendrogram nazywany jest filogramem lub drzewem filogenetycznym [95] i obrazuje pokrewieństwo pomiędzy jednostkami - im bliżej znajdują się one na drzewie, tym bardziej są spokrewnione [49]. Dodatkowo za pomocą filogramu można przedstawić historię powstawania zmienności pomiędzy jednostkami - wierzchołki drzewa przedstawiają analizowane jednostki, natomiast gałęzie pokazują zdarzenia zmian pomiędzy nimi [49]. Jednym z popularnych algorytmów do budowy drzewa filogenetycznego jest algorytm *Neighbor-Joining* (NJ) [117]. Jego schemat działania przedstawiono poniżej. Dodatkowo opisano modyfikację tego algorytmu ułatwiającą odczytanie z drzewa historii zmian pomiędzy jednostkami. Przedstawione rozszerzenie algorytmu NJ zostało również opisane w suplemencie do artykułu [108].

1.2.1 Algorytm Neighbor-Joining

Algorytm *Neighbor-Joining* (NJ) [117] na wejściu pobiera macierz G odległości pomiędzy sekwencjami, gdzie $G(i, j)$ oznacza odległość pomiędzy sekwencjami i-tą i j-tą. Na początku każda sekwencja jest reprezentowana przez węzeł drzewa (wierzchołek w grafie), aczkolwiek niepołączony z żadnym innym - graf bez żadnych krawędzi. Następnie algorytm zaczyna łączyć węzły (sekwencje) w pary do momentu zbudowania drzewa składającego się ze wszystkich sekwencji. Algorytm budowania drzewa można przedstawić za pomocą następujących kroków:

1. Korzystając z odległości pomiędzy węzłami G wyznacz macierz kosztu Q, zdefiniowaną jako:
$$Q(i, j) = (n - 2)G(i, j) - \sum_{k=1}^{n} G(i, k) - \sum_{k=1}^{n} G(j, k), \tag{1.19}$$
gdzie $G(i, j)$ oznacza odległość pomiędzy węzłami i, j, a n to liczba węzłów, które pozostały do połączenia;

2. Wyszukaj parę węzłów f, g taką, że $f \neq g$, którym odpowiada najmniejsza wartość w macierzy Q.

3. Połącz znalezioną parę f, g w nowy węzeł u.

4. Wyznacz odległość pomiędzy parami węzłów f, u oraz g, u:

$$G(f,u) = \frac{G(f,g)}{2} + \frac{\sum_{k=1}^{n} G(f,k) - \sum_{k=1}^{n} G(g,k)}{2(n-2)}, \tag{1.20}$$

$$G(g,u) = G(f,g) - G(f,u). \tag{1.21}$$

5. Wyznacz odległości pomiędzy pozostałymi węzłami k a nowo utworzonym węzłem u:

$$G(k,u) = \frac{G(f,k) + G(g,k) - G(f,g)}{2}. \tag{1.22}$$

6. Usuń węzły f, g z macierzy odległości G, w ich miejsce dodaj węzeł u.

7. Jeżeli liczba węzłów w macierzy G jest większa niż 2, wykonaj ponownie wszystkie kroki algorytmu, w przeciwnym wypadku zakończ.

Algorytm NJ wymaga $N-2$ iteracji. W każdej iteracji jest przeszukiwana macierz Q, która początkowo ma rozmiar $N \times N$, a w kolejnych krokach ubywa z niej jeden węzeł. Wynika z tego, że algorytm NJ ma złożoność obliczeniową $\mathcal{O}(N^3)$.

1.2.2 Algorytm Neighbor-Joining Plus

Algorytm NJ buduje regularne drzewo binarne. Jeżeli w drzewie zaznaczymy korzeń, czyli najstarszą sekwencję, a sekwencje z niej wychodzące to jej dzieci, to w drzewie regularnym każdy węzeł będzie miał dwa lub zero węzłów pochodnych (dzieci). Jest to poważne ograniczenie, ponieważ w naturze sekwencja może mieć dowolną liczbę potomków [95]. Co więcej, algorytm NJ zakłada, że sekwencje mogą być przypisane tylko do liści drzewa (węzły bez potomstwa), z czego wynika, że w analizowanym zbiorze mogą być tylko te sekwencje które nie posiadają potomstwa obecnego jako sekwencje w tymże zbiorze. W przeciwnym wypadku wynikowe drzewo będzie błędne. Ograniczenia te bardzo utrudniają analizę przy użyciu algorytmu NJ zbiorów sekwencji szybko zmieniających się, np. takich jak sekwencje wirusa grypy.

Ograniczenia te w trakcie analizy drzewa pozwala ominąć zaproponowany przez autora rozprawy algorytm *Quick Path Finding* (QPF) [95], [96]. Algorytm QPF na wejściu pobiera drzewo wyprodukowane przez algorytm NJ (lub też inny algorytm), a zwraca

drzewo bez opisanych powyżej ograniczeń. Autor proponuje również bezpośrednią modyfikację algorytmu NJ. Modyfikacja algorytmu została przedstawiona i zastosowana do analizy sekwencji wirusa grypy w pracy [108]. Poniżej opisano zmodyfikowany algorytm, nazwany *Neighbor-Joining Plus* (NJ+).

Algorytm NJ+ rozszerza algorytm NJ, dodając do niego krok, w którym podejmowana jest decyzja o charakterze pary. Poniżej przedstawiono etapy postępowania przy zastosowaniu metody NJ+:

1. Korzystając z odległości pomiędzy węzłami G wyznacz macierz kosztu Q, używając równania (1.19).

2. Wyszukaj parę węzłów f, g taką, że $f \neq g$, dla których wartość w macierzy Q jest najmniejsza.

3. Stwórz nowy węzeł u, który nie jest jeszcze nigdzie przyłączony, jego charakter zostanie wyznaczony w kroku 6.

4. Wyznacz odległości pomiędzy węzłami f, u oraz g, u, używając wzorów (1.20), (1.21).

5. Zaokrągl odległości $G(g,u)$ i $G(f,u)$ do liczb przedstawiających pełne mutacje.

6. Wyznacz charakter węzłów f, g:

   (a) Jeżeli f i g są liśćmi:

      i. Połącz węzły f oraz g z węzłem u.

      ii. Wyznacz odległości pomiędzy pozostałymi węzłami k a nowo utworzonym węzłem u, korzystając z równania (1.22).

      iii. Usuń węzły f, g z macierzy odległości G, w ich miejsce dodaj węzeł u.

   (b) Jeżeli f jest węzłem wewnętrznym, a g jest liściem:

      i. Połącz węzeł g z f.

      ii. Usuń z macierzy G węzeł g.

   (c) Jeżeli g jest węzłem wewnętrznym, a f jest liściem:

      i. Połącz węzeł f z g.

      ii. Usuń z macierzy G węzeł f.

(d) Jeżeli f i g są węzłami wewnętrznymi:

i. Jeżeli f jest węzłem przedstawiającym sekwencję, a g nie przedstawia sekwencji lub jeżeli oba węzły nie przedstawiają sekwencji:

A. Do węzła f przyłącz wszystkie węzły, które są przyłączone do g.

B. Usuń węzeł g z macierzy G.

ii. Jeżeli g jest węzłem przedstawiającym sekwencję, a f nie przedstawia sekwencji:

A. Do węzła g przyłącz wszystkie węzły, które są przyłączone do f.

B. Usuń węzeł f z macierzy G.

iii. Jeżeli oba węzły przedstawiają sekwencje, postępuj według kroku 6a.

7. Jeżeli liczba węzłów w macierzy G jest większa niż 2, wykonaj ponownie wszystkie kroki algorytmu, w przeciwnym wypadku zakończ.

Zaokrąglenie do pełnych mutacji

W przypadku gdy odległość pomiędzy sekwencjami przedstawia liczbę mutacji pomiędzy nimi, długości gałęzi w drzewie powinny być wartościami całkowitymi. Tymczasem warunek ten nie jest zapewniony w równaniach (1.20), (1.21). W kroku 5 algorytmu NJ+ dochodzi do zaokrąglenia odległości $G(f,u)$ oraz $G(g,u)$ do liczb przedstawiających pełne mutacje. W proponowanej procedurze porównywana jest mantysa zaokrąglanych odległości. W przypadku gdy mantysa dla odległości $G(g,u)$ jest mniejsza niż dla $G(f,u)$, odległość $G(g,u)$ jest przybliżana do najbliższej liczby całkowitej. Odległość $G(f,u)$ natomiast wyznaczana jest jako różnica pomiędzy sumą odległości $G(f,u)$ oraz $G(g,u)$ przed zaokrągleniem i zaokrąglonej długości $G(g,u)$. W przypadku gdy mantysa dla odległości $G(f,u)$ jest mniejsza niż dla $G(g,u)$ postępowanie jest analogiczne.

Przykład: Rozważając odległości wejściowe o wartości: $G(f,u) = 3,4$, $G(g,u) = 3,2$, funkcja zaokrąglająca do pełnych mutacji zwróci wartości $G(f,u) = 4$, $G(g,u) = 3$.

Wyznaczenie charakteru węzła w drzewie

W czasie budowania drzewa sprawdzany jest charakter węzła. Węzeł drzewa może być liściem lub węzłem wewnętrznym. Podczas tego procesu stosuje się dwa następujące warunki:

1. Jeżeli wartość $G(f, u)$ jest mniejsza niż jeden (gdzie długość jeden odpowiada jednej mutacji), to węzeł f traktowany jest jako węzeł wewnętrzny, w przeciwnym wypadku jako liść. Analogicznie postępujemy w przypadku węzła g.

2. Jeżeli z pierwszego warunku wynika, że oba węzły f, g są liśćmi, używany jest dodatkowy warunek, sprawdzający czy na pewno nie występuje wśród f lub g węzeł wewnętrzny. Dodatkowy warunek jest potrzebny, ponieważ odległości $G(f, u)$ oraz $G(g, u)$ mogą być zaburzone przez wielokrotne mutacje na tej samej pozycji. Bazuje on na lokalnych odległościach, w których prawdopodobieństwo wielokrotnej mutacji na tej samej pozycji jest niskie. Wśród węzłów, które pozostały w macierzy G należy znaleźć taki węzeł h, który jest najbliżej węzła f lub g oraz posiada przypisaną sekwencję. W warunku sprawdzane jest zachowanie równości:

$$G(f, h) = G(f, g) + G(g, h). \tag{1.23}$$

   Jeżeli jest ona zachowana, należy przyjąć, że węzeł g jest węzłem wewnętrznym. Analogicznie postępuje się dla węzła f.

Rozdział 2

Selekcja cech

Zadanie selekcji cech polega na wskazaniu najważniejszych atrybutów w analizowanym problemie - często wystarczy analiza tylko wybranych cech, ponieważ pozostałe wnoszą bardzo mało do analizy [63]. Co więcej, duża liczba atrybutów nieistotnych może zakłócić lub też całkowicie uniemożliwić proces analizy danych [63]. Do zadań selekcji cech należy nie tylko wskazanie podzbioru najważniejszych cech i wykluczenie nieistotnych atrybutów, ale również uporządkowanie atrybutów w zależności od istotności w danym problemie, co jest równoznaczne z przypisaniem atrybutowi pozycji w rankingu lub też wagi [70]. Selekcja cech jest problemem NP trudnym (ang. *nondeterministic polynomial*), ponieważ rozważając M atrybutów można wskazać $2^M - 1$ niepustych możliwych podzbiorów atrybutów. Podobnie jak w przypadku algorytmów klasyfikacji i klasteryzacji, nie istnieje jeden uniwersalny algorytm, który zapewnia najlepsze działanie na danych pochodzących z różnych źródeł. Algorytmy służące do wyboru istotnych cech można podzielić na trzy grupy:

- Algorytmy filtrujące (ang. *filtering approach*) - potrafią określić, najczęściej za pomocą jednego równania, jak bardzo modelowany atrybut zależy od atrybutów wejściowych; z reguły są one najszybsze i nie zależą od używanej metody analizy danych [20].
- Algorytmy wbudowane (ang. *embedded approach*) - to techniki pozwalające pobrać informację o ważności atrybutów bezpośrednio z nauczonego klasyfikatora; szybkość tych metod zależy od szybkości nauki klasyfikatora [20], [63].
- Algorytmy opakowujące (ang. *wrapper approach*) - używają wybranej metody ana-

lizy danych i sprawdzają jej działanie na różnych podzbiorach atrybutów. Metody te potrafią sterować dobieranym podzbiorem atrybutów tak, by maksymalizować skuteczność działania wybranej metody [20], [63]. W związku z testowaniem wielu różnych podzbiorów atrybutów, algorytmy z tej grupy są najwolniejsze.

Oprócz wymienionych wyżej różnych typów algorytmów istnieją jeszcze takie podejścia, które łączą w swoim działaniu różne typy algorytmów. Popularną techniką selekcji jest stosowanie metod filtrujących, które są szybkie, ale mało dokładne, do wstępnego odrzucenia nieważnych atrybutów, a następnie zastosowanie wolniejszej, ale dokładniejszej, opakowującej metody selekcji dla pozostałych danych. Autor rozprawy posłużył się taką techniką w analizie danych pochodzących z raportów Państwowej Straży Pożarnej, której celem było wybranie atrybutów istotnych ze względu na bezpieczeństwo osób biorących udział w akcji ratunkowej [97].

Poniżej zostały przedstawione wybrane algorytmy selekcji cech z każdej grupy. Algorytmy te posłużyły do:

- wybrania wag określających ważność nukleotydów w łańcuchu RNA wirusa grypy (rozdział 5),
- selekcji cech w zadaniu klasyfikacji dzieci z dysleksją na podstawie obrazów z rezonansu magnetycznego (rozdział 6),
- sprawdzeniu istotości cech w zadaniach segmentacji torów oraz klasyfikacji neutrin elektronowych na podstawie obrazów z detektora ciekłoargonowego (rozdział 7).

Algorytmy selekcji cech zostały użyte w problemie doboru cech w klasyfikacji. Dodatkowo, w rozprawie przedstawiono sposób wykorzystania genetycznego algorytmu doboru wag dla atrybutów w zadaniu klasteryzacji. Omówiono także problem nadmiernego dopasowania i stabilności algorytmów selekcji cech.

2.1 Filtrujące algorytmy selekcji

Filtrujące algorytmy selekcji cech oceniają każdy atrybut osobno. Dzięki rozpatrywaniu tylko jednego atrybutu w danej chwili, metody filtrujące są szybkie, jednak mogą pominąć atrybuty, które są istotne dopiero w połączeniu z innymi [63].

2.1.1 Ocena atrybutów regułą Fishera

Metryka Fishera (ang. *Fisher score*) ocenia jak bardzo różnią się wartości rozpatrywanego atrybutu w różnych klasach [50]. Wartość metryki Fishera v_i dla i-tego atrybutu w przypadku klasyfikacji binarnej przedstawiona jest za pomocą równania:

$$v_i = \frac{(\mu_i^+ - \mu_i^-)^2}{(\sigma_i^+)^2 + (\sigma_i^-)^2}, \tag{2.1}$$

gdzie:

- μ_i^+ oznacza średnią wartość atrybutu $\mathbf{X}_i$ dla próbek należących do klasy 1:

$$\mu_i^+ = \frac{1}{N^+} \sum_{j=1}^{N^+} x_{ij}, \tag{2.2}$$

  wartość N^+ oznacza liczbę próbek należących do klasy 1,

- $(\sigma_i^+)^2$ oznacza wariancję wartości atrybutu $\mathbf{X}_i$ dla próbek należących do klasy 1:

$$(\sigma_i^+)^2 = \frac{1}{N^+ - 1} \sum_{j=1}^{N^+} (x_{ij} - \mu_i^+)^2, \tag{2.3}$$

- wartości μ_i^- oraz $(\sigma_i^-)^2$ dla próbek należących do klasy 0 wyznaczane są analogicznie.

Korzystając z metryki Fishera wyznaczana jest wartość v_i dla każdego atrybutu, a następnie jako najbardziej istotne wybierne są atrybuty z najwyższymi wartościami.

2.1.2 Ocena atrubutów za pomocą t-testu

Do oceny ważności atrybutu często używany jest t-test [53], który przy sprawdzaniu grup o rożnej liczebności oraz różnej wariancji nazywany jest również t-testem Welcha. Jest to test statystyczny, sprawdzający czy wartości i-tego atrybutu różnią się pomiędzy klasami. W teście tym wyznaczamy dwa parametry: wartość t oraz liczbę stopni swobody (ang. *degrees of freedom*) oznaczoną jako $d.f.$:

$$t = \frac{\mu_i^+ - \mu_i^-}{\sqrt{\frac{(\sigma_i^+)^2}{N^+} + \frac{(\sigma_i^-)^2}{N^-}}}, \tag{2.4}$$

$$d.f. = \frac{(\frac{(\sigma_i^+)^2}{N^+} + \frac{(\sigma_i^-)^2}{N^-})^2}{\frac{(\sigma_i^+)^4}{(N^+)^2(N^- - 1)} + \frac{(\sigma_i^-)^4}{(N^-)^2(N^+ - 1)}}. \tag{2.5}$$

Następnie korzystając z rozkładu t-Studenta wyznaczany jest poziom istotności (ang. *p-value*). Jeżeli wartość *p-value* jest mniejsza niż $0,05$, przyjmuje się, że wartości atrybutu różnią się statystycznie znacząco pomiędzy klasami. Przy selekcji cech za pomocą t-testu, dla każdego atrybutu wyznacza się wartość istotności p. Atrybuty charakteryzujące się najniższym poziomem istotności uważane są za najważniejsze w analizowanym zadaniu.

2.1.3 Ocena atrybutów na podstawie przyrostu informacji

Do oceny atrybutów można użyć również metryki mierzącej przyrost informacji (ang. *information gain*) [31], zwanej też czasami dywergencją Kullbacka–Leiblera i używanej często do budowy drzew decyzyjnych (równanie (1.2)). Wzór na przyrost informacji można zapisać następująco:

$$IG(\mathbf{Y}, \mathbf{X}_i) = E(\mathbf{Y}) - E(\mathbf{Y}|\mathbf{X}_i). \tag{2.6}$$

Szczegóły obliczania entropii zostały opisane w rozdziale 1.1.2. W przypadku atrybutów ciągłych należy dokonać dyskretyzacji ich wartości przed obliczeniem przyrostu informacji. Do dyskretyzacji stosowane są różne metody, które można podzielić na nadzorowane oraz nienadzorowane. Często używanymi metodami dyskretyzacji bez nadzoru są: podział na przedziały o równej długości lub liczebności [30]. Dobre rezultaty otrzymuje się nadzorowaną metodą zaproponowaną przez Fayyada i Irani'ego w [34], używającą entropii do wyznaczenia dyskretyzacji - metoda ta była używana w badaniach prezentowanych w rozprawie.

2.2 Wbudowane algorytmy selekcji

Wbudowane algorytmy selekcji potrafią określić jak bardzo użyteczne są atrybuty użyte do nauki klasyfikatora przy użyciu jego wewnętrznej struktury [63]. Popularną techniką stosowaną do uzyskania informacji o atrybutach jest wykorzystanie algorytmu lasu losowego (rozdział 1.1.3).

2.2.1 Ocena atrybutów za pomocą lasu losowego

W algorytmie lasu losowego przed rozpoczęciem budowania drzewa z oryginalnego zbioru D losowany ze zwracaniem jest zbiór D'. Próbki niewylosowane tworzą tak zwany zbiór *out of bag* (OOB) [13]. Jest on wykorzystywany do oszacowania ważności atrybutów. Algorytm postępowania przy wyznaczaniu ważności atrybutów przedstawiony jest poniżej:

1. Dla każdego i-tego drzewa:
   (a) Dla rozpatrywanego drzewa wyznacz liczbę poprawnych klasyfikacji k^i na jego zbiorze OOB.
   (b) Dla każdego atrybutu:
      i. Dla rozpatrywanego atrybutu j losowo zmień kolejność jego wartości.
      ii. Wyznacz liczbę poprawnie sklasyfiowanych próbek k^i_j na zbiorze ze zmienioną kolejnością wartości w j-tym atrybucie.
      iii. Przywróć oryginalną kolejność wartości w k-tym atrybucie.
2. Dla każdego atrybutu j wyznacz średnią różnicę pomiędzy liczbą poprawnych klasyfikacji na zbiorze oryginalnym i na zbiorze z przestawioną kolejnością wartości w atrybucie, uśredniając wynik po wszystkich T drzewach:

$$r_j = \frac{1}{T}\sum_{i=1}^{T} k^i - k^i_j. \tag{2.7}$$

   Średnia r_j wyliczona w ten sposób określa ważność j-tego atrybutu - im większa jej wartość, tym ważniejszy jest atrybut.

2.3 Opakowujące algorytmy selekcji

Algorytmy opakowujące należą do najwolniejszych metod selekcji cech, ponieważ sprawdzają wiele różnych podzbiorów atrybutów. Schemat działania tych algorytmów można przedstawić w następujących krokach:

1. Wybierz podzbiór atrybutów. W kroku tym może zostać wykorzystana informacja o wcześniej ocenionych podzbiorach atrybutów.
2. Sprawdź skuteczność działania algorytmu na wybranych atrybutach.

3. Jeżeli skuteczność działania algorytmu jest wystarczająca, zatrzymaj przeszukiwanie, w przeciwnym wypadku wróć do kroku 1.

W procedurze tej atrybuty oceniane są na podstawie skuteczności działania algorytmu z ich wykorzystaniem. W przypadku klasyfikacji do oceny zostanie użyta skuteczność klasyfikatora nauczonego na podzbiorze atrybutów, a w przypadku klasteryzacji zostanie oceniona jakość uzyskanego pogrupowania.

2.3.1 Algorytm selekcji w przód

Algorytm selekcji w przód [52] (ang. *forward feature selection*) zaczyna wyznaczanie podzbioru najważniejszych atrybutów od sprawdzenia skuteczności klasyfikatora na pojedynczych atrybutach i wybrania najlepszego spośród nich, czyli tego, który zapewniał najlepszą skuteczność działania. Atrybut ten stanowi początkowy zbiór wybranych atrybutów. Następnie algorytm zachłannie dodaje po jednym atrybucie do uzyskanego podzbioru i sprawdza działanie klasyfikatora. W ten sposób algorytm działa aż do uzyskania żądanej liczby atrybutów L w wybranym podzbiorze, lub sytuacji, w której nie będzie można dodać kolejnego atrybutu do podzbioru tak, by poprawić skuteczność działania klasyfikatora. Poniżej przedstawiono w krokach schemat działania algorytmu:

1. Zbiór wybranych atrybutów oznaczony jest jako S. Na początku jest on zbiorem pustym $S = \{\}$.
2. Dla każdego dostępnego i-tego atrybutu:
   (a) Wytrenuj model h_i używając atrybutów $S \bigcup \mathbf{X}_i$.
   (b) Oceń model h_i.
3. Wybierz j-ty atrybut, dla którego skuteczność klasyfikatora była najsłabsza.
4. Dodaj wybrany atrybut do zbioru S, $S = S \bigcup \mathbf{X}_j$, i nie rozpatruj go w kolejnych krokach.
5. Sprawdź warunek stopu:
   (a) Jeżeli warunek stopu jest spełniony, zatrzymaj selekcję,
   (b) W przciwnym wypadku wróć do kroku 2.

Algorytm selekcji cech w przód jest algorytmem zachłannym, ponieważ przy każdym dodaniu atrybutu do podzbioru dąży do maksymalizowania skuteczności klasyfikatora. Przy założeniu, że warunkiem zatrzymania algorytmu jest sprawdzenie wszystkich cech, $L = M$, algorytm ten wymaga $(M + 1)M/2$ iteracji. Jego złożoność jest kwadratowa, $\mathcal{O}(M^2)$. Takie podejście jest bardzo kosztowne obliczeniowo, ponieważ w każdej iteracji klasyfikator trenowany jest zazwyczaj od nowa innym zestawem danych.

2.3.2 Algorytm selekcji w tył

Algorytm selekcji cech w tył [52] (ang. *backward feature selection*) jest bardzo podobny w działaniu do algorytmu selekcji cech w przód, z tą różnicą, że algorytm zaczyna działanie z podzbiorem, w którym są wszystkie atrybuty, a następnie sekwencyjnie usuwa najsłabsze z nich. Kryterium zatrzymania przeszukiwania jest osiągniecie określonej liczby atrybutów L w podzbiorze wybranych atrybutów lub brak możliwości usunięcia atrybutu bez obniżenia skuteczności klasyfikatora. Poniżej przedstawione są kroki algorytmu:

1. Zbiór wybranych atrybutów oznaczony jest jako S. Na początku jest on zbiorem ze wszystkimi atrybutami $S = \{\mathbf{X}_1, ..., \mathbf{X}_M\}$.
2. Dla każdego i-tego atrybutu w zbiorze S:
   (a) Wytrenuj model h_i używając atrybutów $S \setminus \mathbf{X}_i$.
   (b) Oceń model h_i.
3. Wybierz j-ty atrybut, dla którego skuteczność klasyfikatora była najlepsza.
4. Usuń wybrany atrybut ze zbioru S, $S = S \setminus \mathbf{X}_j$.
5. Sprawdź warunek stopu:
   (a) Jeżeli warunek stopu jest spełniony, zatrzymaj selekcję,
   (b) W przeciwnym wypadku wróć do kroku 2.

Algorytm selekcji cech w tył również, podobnie jak algorytm selekcji cech w przód, jest algorytmem zachłannym - w każdym kroku usuwa atrybut najsłabszy oraz ma złożoność kwadratową. W przypadku, gdy liczba atrybutów jest bardzo duża i w końcowym zbiorze znajduje się mała liczba atrybutów $L < M$, podejście jest bardziej kosztowne niż selekcja atrybutów w przód, ponieważ klasyfikator jest uczony danymi o większej wymiarowości.

2.3.3 Algorytm genetyczny do selekcji lub ważenia atrybutów

Algorytm genetyczny naśladuje zachowanie ewolucji genetycznej w zadaniu poszukiwania rozwiązania dla problemu zależącego od wielu parametrów [87], [3]. W podejściu tym populacja, przedstawiona za pomocą zbioru chromosomów, z których każdy przedstawia sprawdzany zestaw parametrów, poddawana jest ewolucji w celu znalezienia najlepszego chromosomu. Schemat działania algorytmu genetycznego można przedstawić za pomocą następujących kroków:

1. Wylosuj populację początkową.
2. Oceń populację i usuń najsłabsze chromosomy.
3. Zastosuj dla pozostałych chromosomów operatory ewolucyjne, w wyniku czego powstaną nowe osobniki dodane do populacji.
4. Sprawdź kryterium stopu:
   (a) Jeżeli jest spełnione, zatrzymaj algorytm.
   (b) Jeżeli nie jest spełnione, wróć do kroku. 2.

By móc zastosować algorytm genetyczny do wybranego problemu, należy w nim zdefiniować następujące elementy:

- kodowanie chromosomu $\mathbf{V}$: zazwyczaj chromosom jest zakodowany jako wektor liczb całkowitych lub rzeczywistych, może być również zakodowany w postaci drzewiastych struktur danych [87],
- funkcję dopasowania (lub też funkcję celu), $\mathcal{F}(\mathbf{V})$, która używana będzie od oceny chromosomów,
- operatory ewolucyjne, które będą generować nowe chromosomy na podstawie już istniejących.

Kodowanie chromosomu oraz funkcja dopasowania muszą być dobrane indywidualnie do rozwiązywanego problemu. Ich dobór jest bardzo często kluczowy dla rozwiązywanego problemu - zła reprezentacja chromosomu lub źle zdefiniowana funkcja celu mogą znacznie utrudnić przeszukiwanie przestrzeni rozwiązań. Natomiast operatory ewolucyjne są bardziej niezależne od rozwiązywanego problemu. Wśród nich do najbardziej popularnych

należą operacja mutacji i krzyżowania. W trakcie mutacji wybiera się chromosom, który będzie podlegał zmianie, a następnie w losowych miejscach zmienia się jego wartości. O liczbie mutacji w populacji decyduje współczynnik określający szybkość mutowania, którego wartość najczęściej nie przekracza kilku procent. Zmutowane chromosomy przedstawiają nowe rozwiązanie w przeszukiwanej przestrzeni rozwiązań i są dodawane do populacji. Drugim, często używanym operatorem ewolucyjnym jest operacja krzyżowania. W operacji tej wybierane są losowo dwa chromosomy i każdy z nich zostaje podzielony na dwie części w tym samym wylosowanym miejscu. Następnie z połączenia krzyżowego powstają dwa nowe chromosomy, gdzie każdy nowy chromosom ma jedną część z pierwszego rodzica, a drugą z drugiego. Nowo powstałe chromosomy dodawane są do populacji.

Algorytm genetyczny może być użyty do wybrania podzbioru najważniejszych atrybutów [129]. W tym przypadku chromosom $\mathbf{V}$ zakodowany jest binarnie i ma długość odpowiadającą liczbie atrybutów M, $\mathbf{V} = \{v_1, ..., v_M\}$ oraz $v_i \in \{0, 1\}$. Jeżeli na i-tej pozycji w chromosomie występuje wartość $v_i = 1$, to znaczy, że atrybut został wybrany do podzbioru najważniejszych atrybutów. Jako funkcję dopasowania można przyjąć jakość klasyfikatora nauczonego atrybutami wskazanymi przez chromosom. Do operacji ewolucyjnych używa się najczęściej operatora mutacji i krzyżowania. Dodatkowo proces selekcji można usprawnić nie całkowicie losową inicjalizacją populacji. Jeżeli mamy jakieś przypuszczenia odnośnie liczebności wynikowego podzbioru atrybutów, informację taką można zawrzeć w trakcie inicjalizacji, zapewniając by początkowe chromosomy miały liczbę wylosowanych bitów z wartością 1 równą oczekiwanej liczbie atrybutów w wynikowym zbiorze.

Algorytm genetyczny może zostać również użyty do wyznaczenia wag określających ważność atrybutów. W problemie tym każdy i-ty atrybut ma przypisaną wagę v_i, gdzie wartości wag mogą być liczbami całkowitymi $v_i \in \mathcal{N}$ lub rzeczywistymi $v_i \in \mathcal{R}$. Jako funkcję dopasowania przyjmuje się jakość działania algorytmu z wykorzystaniem wag zakodowanych w chromosomie, natomiast jako operatorów ewolucyjnych używa się operatorów mutacji i krzyżowania. Również w tym przypadku optymalizacji, jeżeli istnieją jakieś przesłanki odnośnie wynikowego rozwiązania, to zawarcie tej informacji w procesie inicjalizacji może przyspieszyć przeszukiwanie.

2.4 Nadmierne dopasowanie algorytmów selekcji

Tak jak w przypadku klasyfikatorów, algorytmy selekcji mogą się nadmiernie dopasować do uczonego zbioru. Częstym błędem w analizie danych jest użycie całego dostępnego zbioru danych do wyselekcjonowania cech, a następnie użycie sprawdzianu krzyżowego na tym zbiorze do oceny pracy klasyfikatora [52]. W takim przypadku algorytm selekcji do wyznaczenia najważniejszych atrybutów bierze pod uwagę cały zbiór danych i do niego dobiera atrybuty. Także skuteczność klasyfikatora policzona na obciążonych atrybutach, nawet jeżeli policzona przy pomocy walidacji krzyżowej, będzie nieprawdziwa. Poprawnym podejściem jest wyznaczanie najważniejszych cech w każdym kroku walidacji, a następnie uczenie klasyfikatora i testowanie na odłożonych danych testujących. Proces selekcji cech oraz uczenia klasyfikatora musi być powtórzony w każdej iteracji sprawdzianu krzyżowego. Proces ten można opisać następującymi krokami:

1. Podziel zbior danych D na dwa rozłączne podzbiory: uczący U oraz testujący T.
2. Używając zbioru uczącego U wyznacz podzbiór najważniejszych atrybutów S_U.
3. Używając zbioru atrybutów S_U oraz zbioru uczącego U wyternuj klasyfikator h_U.
4. Sprawdź skuteczność działania klasyfikatora h_U na zbiorze testującym T.
5. Powtórz powyższe kroki k razy (w przypadku k-krotnej kroswalidacji krzyżowej).

2.5 Stabilność algorytmów selekcji

W przypadku, gdy liczba próbek w analizowanym zbiorze jest mała, a w szczególności gdy jest mniejsza niż liczba dostępnych atrybutów, $M \ll N$, wyselekcjonowane atrybuty w dużym stopniu mogą zależeć od dostępnej próbki danych. Jest to tak zwany problem stabilności algorytmów selekcji [70], [55], [75]. Problem ten dotyczy na przykład danych pochodzących z badań proteomicznych, genomicznych [57], [5], [55] lub obrazowania medycznego [130], [113], ponieważ w tych problemach liczba atrybutów jest znacznie większa niż liczba dostępnych próbek.

Istnieje wiele metryk stosowanych do oceny stabilności algorytmu selekcji [70], [55], [75], [118], [133]. Większość z nich bazuje na stosunku liczby cech wspólnych dwóch podzbiorów i liczby unikalnych atrybutów tych podzbiorów. Jest to tak zwany indeks Jaccarda

(lub też indeks Tanimoto) [70], który dla dwóch podzbiorów S_i oraz S_j jest określony następująco:

$$J(S_i, S_j) = \frac{|S_i \bigcap S_j|}{|S_i \bigcup S_j|}. \tag{2.8}$$

Indeks ten przyjmuje wartości z przedziału $[0; 1]$. Dla bardziej podobnych podzbiorów wartości przyjmowane przez indeks są wyższe. W czasie k-krotnej kroswalidacji krzyżowej, wyznaczanych jest k podzbiorów atrybutów. Używając indeksu Jaccarda stabilność ich selekcji opisuje wzór:

$$J = \frac{2}{k^2 - k} \sum_{i=1}^{k} \sum_{j=i+1}^{k-1} \frac{|S_i \bigcap S_j|}{|S_i \bigcup S_j|}. \tag{2.9}$$

We wzorze (2.9) wyznaczany jest średni indeks Jaccarda, porównujący wszystkie wybrane podzbiory atrybutów uzyskane w kroswalidacji. Po zbadaniu stabilności selekcji można wskazać te atrybuty, które najczęściej występowały we wszystkich zbiorach.

Jeżeli zastosowanym algorytmem selekcji cech nie udaje się osiągnąć zadowalającej stabilności, istnieje wiele metod jej poprawienia. Wśród popularnych metod poprawiających stabilność często stosuje się metody uczenia zbiorowego (ang. *ensemble learning*) [113], [55]. W podejściu tym wielokrotnie losuje się różne podzbiory próbek, a czasem również atrybutów, z oryginalnego zbioru uczącego, a następnie na każdym wylosowanym podzbiorze stosuje się algorytm selekcji cech. Jako wybrane cechy są wskazywane te, które były najczęściej wybierane na wylosowanych podzbiorach.

Rozdział 3

Wstępne przekształcenie cech

Zazwyczaj dane uzyskane bezpośrednio z pomiarów wymagają przed analizą wstępnego przekształcenia, w trakcie którego wykonuje się operacje na danych, umożliwiające ich poprawne przeanalizowanie. Często stosowanymi przekształceniami są np. skalowanie danych lub transformacje. W rozprawie zostaną opisane operacje normalizacji min-max oraz standaryzacji danych. Dodatkowo zostanie poruszony problem usunięcia z danych wpływu czynników zakłócających (ang. *confouding factor correction*). Problem ten jest bardzo częsty np. w przypadku obrazowania medycznego i badań dużej liczby pacjentów w wielu różnych ośrodkach badawczych, wyposażonych w różny sprzęt pomiarowy [69], [24], [98]. W takim przypadku rozkłady danych badanych osób mogą bardziej zależeć od miejsca, w którym było przeprowadzone badanie niż od schorzenia, które jest analizowane. W takim przypadku miejsce badania i parametry opisujące procedurę badania są czynnikami zakłócającymi, zatem przed analizą danych należy usunąć ich wpływ. Czynnikiem zakłócającym może być też parametr opisujący badaną grupę. Na przykład, w badaniu osób z chorobą Alzheimera czynnikiem zakłócającym jest wiek [69]. Wraz z wiekiem zmieniają się struktury w mózgu, które podobnie zmieniają się wskutek choroby. Aby możliwe było sprawdzenie jak bardzo zmiany w wybranych strukturach zależą od samej choroby, należy usunąć z danych wpływ czynnika zakłócającego - wieku. W rozprawie zostanie opisana metoda usuwania z danych czynników zakłócających.

3.1 Normalizacja min-max

Normalizacja min-max sprowadza wartości atrybutu $\mathbf{X}_i$ do przedziału $[0; 1]$. Dla wszystkich wartości atrybutu $\mathbf{X}_i$ wyliczana jest nowa wartość za pomocą równania:

$$X_i(j)' = \frac{X_i(j) - X_i^{min}}{X_i^{max} - X_i^{min}}, \tag{3.1}$$

gdzie X_i^{max} przedstawia maksymalną wartość atrybutu $\mathbf{X}_i$, a X_i^{min} wartość minimalną. Ważne jest, aby do procesu normalizacji wartości minimalne i maksymalne atrybutu wyznaczyć tylko na podstawie danych uczących. Następnie te same wartości $\mathbf{X}_i^{min}$ oraz $\mathbf{X}_i^{max}$ używane są do normalizacji danych uczących i testujących. Normalizacja min-max nie uwzględnia rozkładu danych. Jeżeli w rozkładzie występują wartości znacznie odstające od przeciętnych, to po operacji normalizacji może dojść do przypisania bardzo małych wartości normalizowanemu atrybutowi.

3.2 Standaryzacja danych

Proces standaryzacji danych przekształca atrybut $\mathbf{X}_i$ tak, by jego wartość oczekiwana wynosiła 0, a wariancja była równa 1. Najczęściej spotykanym typem standaryzacji jest standaryzacja Z, która przekształca wartości atrybutu w następujący sposób:

$$X_i(j)' = \frac{X_i(j) - \mu_i}{\sigma_i}, \tag{3.2}$$

gdzie μ_i to wartość średnia atrybutu $\mathbf{X}_i$, a σ_i to odchylenie standardowe tego atrybutu. Podobnie jak w przypadku normalizacji, wartości średniej i odchylenia należy wyznaczyć tylko na podstawie zbioru uczącego i następnie zastosować je do obu zbiorów: uczącego i testującego. Operacji tej nie należy stosować do atrybutów, które mają odchylenie standardowe bliskie zeru, ponieważ wprowadzi ona do danych duży szum.

3.3 Usunięcie wpływu czynników zakłócających

W przypadku gdy analizowane dane zawierają czynniki zakłócające, niezbędne jest ich usunięcie przed dalszą analizą. Czynnikiem zakłócającym jest atrybut, który zaburza analizę danych. W pracach [69], [24], [98] przedstawiono użycie metody usunięcia czynników

zakłócających (ang. *confounding factors correction*) w analizie danych pochodzących z obrazowania medycznego. Do czynników zakłócających należały parametry: wiek i płeć badanego pacjenta, typ skanera użytego do badania oraz jego konfiguracja. W pracach tych pokazano, że analiza danych po usunięciu czynników zakłócających pozwala na uzyskanie lepszych wyników.

Rozważmy i-ty atrybut, na który mają wpływ czynniki zakłócające. Zmierzoną wartość atrybutu $\mathbf{X}_i$ możemy przedstawić jako sumę jego rzeczywistej wartości $\mathbf{X}_i^{true}$ oraz wartości czynników zakłócających $\mathbf{\Gamma}_i$:

$$\mathbf{X}_i = \mathbf{X}_i^{true} + \mathbf{\Gamma}_i. \tag{3.3}$$

Czynników zakłócających może być wiele, oznaczmy ich liczbę jako J, a pojedynczy czynnik zakłócający jako $\mathbf{\Gamma}_i^j$. Ich całkowity wpływ na atrybut może zostać przybliżony ich liniową kombinacją:

$$\mathbf{\Gamma}_i = \sum_{j=1}^{J} \beta_i \mathbf{\Gamma}_i^j. \tag{3.4}$$

Współczynniki β mogą być wyznaczone za pomocą liniowej regresji po rozwiązaniu równania:

$$\mathbf{X}_i = \beta_0 + \sum_{j=1}^{J} \beta_i \mathbf{\Gamma}^j, \tag{3.5}$$

gdzie współczynnik β_0 przedstawia wartość oczekiwaną rzeczywistej wartości atrybutu, $\mathbf{X}_i^{true}$. Po wyznaczeniu współczynników w równaniu (3.5) można wyznaczyć estymatę rzeczywistej wartości atrybutu:

$$\mathbf{X}_i^{true} = \mathbf{X}_i - \sum_{j=1}^{J} \beta_i \mathbf{\Gamma}^j. \tag{3.6}$$

Przedstawiona procedura jest stosowana do wszystkich atrybutów w zbiorze danych i dalsza analiza przeprowadzana jest na skorygowanych wartościach $\mathbf{X}_i^{true}$. W rozdziale 6 zostanie omówiony wpływ usunięcia czynników zakłócających na selekcję atrybutów oraz na skuteczność klasyfikacji, na przykładzie klasyfikacji dzieci z dysleksją na podstawie wyników badań za pomocą rezonansu magnetycznego.

Wybór czynników zakłócających

Czynniki zakłócające mogą być wskazane przez eksperta z odpowiednią wiedzą. W przypadku gdy nie jest możliwe ich jednoznaczne wskazanie, można spróbować różnych kombinacji czynników by wybrać właściwe [24]. W rozprawie zaproponowano metodę, która spośród zbioru wszystkich potencjalnych czynników zakłócających wybiera zbiór czynników zakłócających o minimalnej wielkości $\mathbf{S}_{CFC}$; po ich usunięciu nie występują istotne zaburzenia. Procedura wyboru czynników zakłócających opisana jest następującymi krokami:

1. Zidentyfikuj wszystkie czynniki zakłócające w zbiorze, $\mathbf{S}_{\Gamma} = \{\mathbf{\Gamma}^1, ..., \mathbf{\Gamma}^L\}$, gdzie $\mathbf{S}_{\Gamma}$ to zbiór wszystkich L czynników zakłócających.

2. Wykonaj kopię zbioru danych $D' = D$.

3. Oceń czy istnieje istotna zależność pomiędzy czynnikami zakłócającymi a wszystkimi atrybutami pochodzącymi z analizowanych danych D'.

4. Jeżeli istnieje istotna zależność, wybierz czynnik zakłócający $\mathbf{\Gamma}^i$, od którego dowolny atrybut w zbiorze zależy najbardziej i dodaj go do zbioru $\mathbf{S}_{CFC}$; Jeżeli nie istnieje istotna zależność pomiędzy atrybutami a czynnikami zakłócającymi, zakończ algorytm.

5. Używając danych oryginalnych D, oblicz zbiór danych po korekcji czynników zakłócających $\mathbf{S}_{CFC}$, używając wzorów (3.5) i (3.6); zapisz go jako D'.

6. Wróć do kroku 3.

Do wykrywania czy pomiędzy czynnikami zakłócającymi i rozpatrywanymi atrybutami występuje zależność wykorzystywane będą w rozprawie:

- korelacja liniowa Pearsona - w przypadku gdy czynnik zakłócający oraz atrybuty są ciągłe; Jeżeli wartość korelacji będzie większa niż $0,5$, przyjmowane będzie, że zależność jest istotna.

- t-test Welcha - jeżeli czynnik zakłócający jest zmienną dyskretną, a atrybuty są ciągłe; Jeżeli obliczona p wartość będzie mniejsza niż $0,05$, przyjmowane będzie, że zależność jest istotna.

Rozdział 4

Konstrukcja cech w analizie obrazu

Analiza obrazów jest jednym z zadań, w którym przed zastosowaniem algorytmów uczenia maszynowego niezbędna jest konstrukcja wektora cech opisującego obraz. Częstym zadaniem w analizie obrazu jest klasyfikacja w zależności od jego zawartości. W przypadku, gdy poda się na wejściu klasyfikatora wektor cech zawierający wartości pikseli z obrazu, skuteczność klasyfikacji może być niska. Dlatego też niezbędne jest skonstruowanie wektora cech opisującego w sposób precyzyjny cechy charakterystyczne dla każdej rozróżnianej klasy. Wektor cech może zostać przygotowany przez eksperta znającego specyfikę obrazów będących przedmiotem analizy [88], [11]. Takie podejście jest jednak trudne, ponieważ często zapis matematyczny obserwowanych w obrazie obiektów jest skomplikowany. Drugim sposobem konstrukcji wektora cech opisującego obraz jest użycie metod, które same potrafią zbudować reprezentację opisującą obserwowane obiekty na obrazie, tzw. uczenie się reprezentacji [8], a często w takich zadaniach wykorzystywane są głębokie sieci neuronowe [54], [123]. Wadą tego podejścia jest wymagana bardzo duża moc obliczeniowa. W rozprawie zastosowano pierwsze podejście, gdzie na podstawie wiedzy eksperckiej skonstruowano zbiór cech opisujących obraz. Konstrukcja cech została wykorzystana w dwóch zadaniach:

- segmentacji - podziału obrazu na obszary jednorodne pod względem pewnej właściwości; segmentacja została wykorzystana do wskazania tych obszarów na obrazie z detektora ciekłoargonowego gdzie występują tory cząstek;
- klasyfikacji - rozróżniania obrazów ze względu na ich zawartość; została ona wykorzystana w klasyfikacji neutrin elektronowych na podstawie obrazu z detektora ciekłoargonowego.

W rozprawie do konstrukcji cech wykorzystano wybrane metody przetwarzania obrazu [122]. Metody przetwarzania obrazu można podzielić na następujące grupy:

- przekształcenia geometryczne obrazu, takie jak obroty, skalowanie, odbicia, transformacja współrzędnych;
- przekształcenia punktowe (bezkontekstowe), w których zmian w obrazie dokonuje się tylko na podstawie wartości piksela;
- przekształcenia kontekstowe (filtry konwolucyjne), gdzie zmian w obrazie dokonuje się na podstawie wartości piksela oraz jego sąsiedztwa;
- przekształcenia morfologiczne, czyli takie które wykorzystują filtry konwolucyjne, ale zmiana wartości piksela zostaje wykonana po spełnieniu określonego warunku;
- przekształcenia widmowe, wykorzystujące np. transformatę Fouriera.

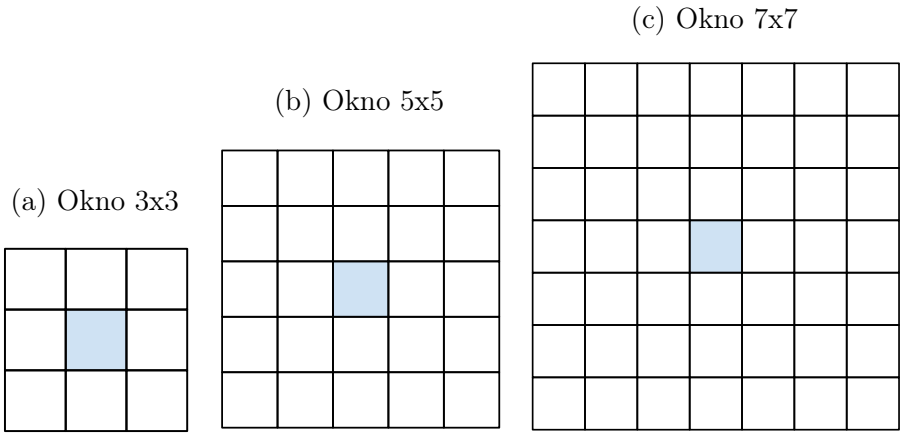
Poniżej zostały opisane wybrane metody przetwarzania obrazu, które zostały użyte w badaniach opisanych w rozprawie.

4.1 Filtry konwolucyjne

Filtracja obrazu jest metodą kontekstową, ponieważ w trakcie obliczania przekształcenia uwzględniane są wartości piksela oraz jego otoczenia. Filtracja ta wykorzystywana jest, by:

- pozyskać więcej informacji z oryginalnego obrazu, na przykład takich jak: gradient zmian w obrazie, położenie krawędzi lub rogów obiektów,
- usunąć szum z obrazu,
- dokonać przekształcenia obrazu, na przykład rozmycia.

Zastosowanie filtru konwolucyjnego w dziedzinie przestrzeni nazywa się splotem, natomiast zastosowanie go w dziedzinie częstotliwości - to iloczyn transformat obrazu i filtru. Filtr w dziedzinie przestrzeni zapisywany jest za pomocą okna, które zazwyczaj jest kwadratowe o nieparzystej liczbie pikseli. Przykład okien o różnej wielkości przedstawia rysunek 4.1.



Rysunek 4.1: Okna kwadratowe o różnej wielkości wykorzystywane do filtracji obrazów. W środku okna zaznaczono rozpatrywany piksel.

Wzór opisujący splot obrazu z filtrem można zapisać następująco:

$$I'[x,y] = 1/k \sum_i \sum_j F[i,j] I[x-i, y-j], \tag{4.1}$$

gdzie:

- oryginalny obraz przedstawiony jest jako macierz I, a wartość piksela o współrzędnych x, y wyrażona jest jako $I[x, y]$,
- obraz po przekształceniu opisany jest zmienną I',
- zmienna F przedstawia macierz filtra, a zmienne i, j są zmiennymi nawigującymi po macierzy,
- zmienna k przedstawia sumę wartości w filtrze.

W celu zdefiniowania przekształcenia kontekstowego konieczne jest podanie macierzy filtra.

4.1.1 Filtry dolnoprzepustowe

Filtry dolnoprzepustowe usuwają z obrazu elementy o wysokiej częstotliwości przestrzennej (szczegóły), czyli takie, które charakteryzują się szybkimi zmianami (np. krawędzie), a zostawiają elementy z niską częstotliwością przestrzenną (obszary stałe, o niskiej zmienności - ogólne). Filtry te stosowane są do usuwania szumu i zakłóceń z wysoką częstotliwością (np. typu pieprz i sól). Przykładami filtrów dolnoprzepustowych są:

- filtr uśredniający, który używany jest do obliczania średniej wartość amplitudy w pikselu oraz jego otoczeniu:

$$F_{avg} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix},$$

- filtr gaussowski, gdzie wartości wyliczane są za pomocą rozkładu Gaussa:

$$F_{gauss} = Ae^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)}, \tag{4.2}$$

gdzie σ to wartość odchylenia standardowego kontrolującego szerokość rozkładu, x to odległość od centrum okna liczona wzdłuż osi poziomej, a y - odległość od centrum okna liczona wzdłuż osi pionowej, zmienna A kontroluje wartość amplitud w filtrze. Przykładowy filtr gaussowski może wyglądać następująco:

$$F_{gauss} = \frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}.$$

4.1.2 Filtry górnoprzepustowe

Filtry górnoprzepustowe wzmacniają elementy o wysokiej częstotliwości przestrzennej, czyli szczegóły. Stosowane są do wykrywania takich elementów jak krawędzie, rogi elementów. W pracy użyto następujące filtry:

- operator Prewitta, używany do detekcji krawędzi i obliczania estymaty pochodnej kierunkowej, poniżej przedstawiono kolejno filtr do detekcji krawędzi poziomych oraz pionowych:

$$F_{Prewitt,x} = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix},$$

$$F_{Prewitt,y} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix},$$

- różnica gaussianów, gdzie obraz wynikowy powstaje przez odjęcie od obrazu rozmytego filtrem gaussowskim z odchyleniem standardowym σ_1 drugiego obrazu, bardziej rozmytego filtrem gaussowskim, z odchyleniem standardowym σ_2, gdzie $\sigma_2 > \sigma_1$; filtr ten wzmacnia elementy o dużej częstotliwości z wybranego pasma.

Operator Prewitta może zostać wykorzystany do wyznaczenia zmiennych opisujących zmiany o wysokiej częstotliwości przestrzennej: gradientu, hesjanu, tensora. Pochodne kierunkowe obrazu mogą zostać wyznaczone na podstawie wzoru:

$$\frac{\partial I}{\partial x} = I * F_{Prewitt,x}, \tag{4.3}$$

$$\frac{\partial I}{\partial y} = I * F_{Prewitt,y}. \tag{4.4}$$

Na ich podstawie możliwe jest wyznaczenie modułu gradientu M dla obrazu:

$$M = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{4.5}$$

Moduł gradientu opisuje szybkość zmian w obrazie - im większa wartość modułu, tym większa zmienność w obrazie. Do wyznaczenia gradientu mogą zostać również użyte inne operatory, takie jak: krzyż Robertsa lub operator Sobela. Mając obliczone pierwsze pochodne kierunkowe możliwe jest obliczenie hesjanu (macierzy drugich pochodnych):

$$H = \begin{pmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial xy} \\ \frac{\partial^2 I}{\partial xy} & \frac{\partial^2 I}{\partial y^2} \end{pmatrix},$$

które w praktyce wyznaczane są jako splot pochodnych kierunkowych z operatorem Prewitta. Z obliczonej macierzy hesjanu możliwe jest wyznaczenie wartości własnych macierzy: λ_1, λ_2, które opisują rodzaj zmiany w obrazie:

- jeżeli $\lambda > 0$, obraz staje się jaśniejszy,
- jeżeli $\lambda < 0$, obraz staje się ciemniejszy.

Wektory własne macierzy H przedstawiają kierunek tych zmian. Sumując wartości w macierzy hesjanu w pikselu i jego okolicy obliczony zostanie tensor obrazu T:

$$T[x, y] = 1/9 \sum_i \sum_j F_{avg}[i, j] H[x + i, y + j]. \tag{4.6}$$

Tensor opisuje dominujący kierunek w otoczeniu analizowanego piksela. Jest wykorzystywany do detekcji rogów obiektów na obrazie. Do tego celu używane są wartości własne tensora, λ_1^T oraz λ_2^T ($\lambda_1^T > \lambda_2^T$):

- jeżeli $\lambda_1^T \approx 0$, $\lambda_2^T \approx 0$, piksel znajduje się w otoczeniu, które jest stałe,
- jeżeli $\lambda_1^T > 0$, $\lambda_2^T \approx 0$, piksel przedstawia krawędź,
- jeżeli $\lambda_1^T > 0$, $\lambda_2^T > 0$, piksel przedstawia róg obiektu na obrazie.

4.1.3 Filtry statystyczne

W filtrach statystycznych [94] wartość piksela po przekształceniu obliczana jest za pomocą funkcji opisującej wybraną cechę zbioru pikseli znajdujących się w sąsiedztwie, które wyznaczane za pomocą okna kwadratowego, tak jak w przypadku filtrów konwolucyjnych. W rozprawie użyto następujących filtrów:

- filtr medianowy, w którym nowa wartość piksela jest medianą z wartości pikseli w oknie;
- filtr minimalny, w którym nowa wartość piksela to wartość minimalna spośród wszystkich pikseli w oknie; filtrowi temu towarzyszy zmniejszanie obiektów na obrazie (erozja);
- filtr maksymalny, w którym nowa wartość piksela to największa wartość spośród wszystkich pikseli w oknie; filtrowi temu towarzyszy zwiększanie obiektów na obrazie (dylatacja);
- filtr wyznaczający odchylenie standardowe na podstawie wartości pikseli w oknie.

4.2 Transformacje układu współrzędnych

Czasami by usprawnić proces analizy konieczne jest wykonanie transformacji układu współrzędnych obrazu. Przykładem takiej transformacji jest przekształcenie obrazu do biegunowego układu współrzędnych. We współrzędnych biegunowych położenie piksela zapisuje się za pomocą odległości od centrum r i kąta odchylenia θ. Transformacja

z układu współrzędnych kartezjańskiego (gdzie piksel opisany jest za pomocą zmiennych x, y) odbywa się za pomocą następujących równań:

$$r = \sqrt{x^2 + y^2}, \tag{4.7}$$

$$\theta = arccos(x/r)sign(y). \tag{4.8}$$

Korzystając z powyższych równań dla każdego piksela w układzie kartezjańskim wyznaczane są odpowiadające współrzędne r, θ w układzie biegunowym, do których zostaje przepisana jego wartość.

4.3 Metody histogramowe

W zadaniach rozpoznawania obrazu popularne są metody budujące histogramy cech (amplitudy, koloru, gradientu lub innych) [23], [81], [88], [11]. Uzyskane histogramy są informacją wejściową dla algorytmu klasteryzacji lub klasyfikacji. W rozprawie używana będzie metoda budująca histogram sumy wartości pikseli w różnych zakresach kątowych w obrazie przekształconym do biegunowego układu współrzędnych. W tym celu obraz I, opisany współrzędnymi r, θ, zostanie podzielony na osi θ na L obszarów, z których każdy będzie przedstawiał zakres kątowy $360^o/L$. Następnie w każdym przedziale sumowane są wartości zawierających się w nim pikseli. Otrzymane w ten sposób wartości będą opisywać rozkład sumy pikseli w wybranym przedziale kątowym.

Rozdział 5

Analiza danych opisujących wirusa grypy

W niniejszym rozdziale zastosowano algorytm selekcji cech do poprawienia jakości klasteryzacji sekwencji wirusa grypy. Dzięki uzyskanemu pogrupowaniu możliwe było wytłumaczenie obserwowanych powtórzeń mutacji.

5.1 Opis problemu

Wirus grypy (ang. *influenza virus*) jest zmieniającym się antygenicznie patogenem zdolnym do ciągłego unikania odpowiedzi immunologicznej. Mutacje gromadzone w części antygenicznej łańcucha RNA wirusa nazywa się "antygenicznym dryfem". Pośród obecnych w środowisku szczepów wirusa grypy, dryf antygeniczny jest głównym procesem powodującym różnorodność pomiędzy szczepami. W przypadku akumulacji dużej liczby mutacji w antygenicznej części wirusa, układ odpornościowy atakowanego gospodarza może nie być w stanie obronić się przed nim, ponieważ przez nagromadzone zmiany w sekwencji wirusa nie będzie w stanie go rozpoznać. Taki szczep z nagromadzoną dużą liczbą mutacji może wywołać epidemię wirusa grypy. Drugim rodzajem zmian wirusa, który może wywołać epidemię jest gwałtowna zmiana odpowiedzi receptora hemaglutyniny [62], [56], która powoduje, że nowo powstały wirus znacząco różni się od pozostałych, powszechnie występujących w środowisku, co czyni go bardziej groźnym. Aby zahamować rozwój wirusa grypy, Światowa Organizacja Zdrowia (ang. *World Health Organisation*) rekomenduje programy szczepień bazujące na najnowszych szczepach wirusa. Szczególnie ważne

w przypadku takich prewencyjnych programów wydaje się analizowanie danych opisujących zaobserwowane szczepy wirusa za pomocą narzędzi filogenetycznych. Pozwalają one na zrekonstruowanie procesu ewolucyjnego wirusa na podstawie wyizolowanych szczepów. Dzięki nim możliwe jest oszacowanie jak bardzo różnią się szczepy obserwowane w środowisku od użytych w szczepionce oraz stwierdzenie czy istnieje realne zagrożenie wybuchem epidemii.

W marcu 2009 r. wybuchła pandemia wirusa grypy H1N1 typu A. Na początku listopada 2009 roku było stwierdzonych laboratoryjnie ponad 440 000 przypadków zachorowań na pandemicznego wirusa grupy, przez którego zmarło 5 700 osób. Podczas trwania epidemii wiele szczepów wirusa zostało wyizolowanych i zsekwencjonowanych. Stwarza to unikalną okazję to dokładnego zbadania przyczyn powstawania epidemii. W pracy [90] badano jak rozwijała się pandemia we wczesnych fazach. Podczas analizy filogenetycznej stwierdzono występowanie 7 głównych grup szczepów (klastrów, lub inaczej kladów) w zrekonstruowanym drzewie filogenetycznym. Kluczową rolę w procesie formowania kladu mają sekwencje "zakładające" klad (ang. *founder strain*), od których wywodzą się wszystkie sekwencje w kladzie i zawierają powielony zestaw mutacji, tworzący sekwencję "zakładającą". Wykrywanie sekwencji "zakładających" jest szczególnie ważne dla przewidywania scenariuszy epidemii [79]. Do powstania sekwencji "zakładających" prowadzi pewien zestaw nagromadzonych mutacji. W przypadku pandemii obserwowanej w 2009 roku niektóre mutacje w populacji wirusa stały się obiektem szczególnego zainteresowania, ponieważ zostały zaobserwowane w różnych miejscach kilkakrotnie. Przykładem takiej mutacji jest zmiana E391K hemaglutyniny, czyli zmiana aminokwasu E (kwas glutaminowy) w K (lizyna) na pozycji 391 w łańcuchu kodującym hemaglutyninę (HA), odpowiadającą za przyłączenie wirusa do infekowanej komórki. Po raz pierwszy została ona opisana w pracy [86], gdzie zaobserwowano gwałtowne rozpowszechnienie się sekwencji niosących tę mutację w drugiej połowie 2009 roku, na świecie oraz lokalnie w Singapurze. Według pracy [86], sekwencje z mutacją HA-E391K najpierw pojawiły się w Nowym Jorku (w lipcu 2009 r.), a krótko po tym pojawiły się w Singapurze, gdzie zdominowały lokalną populację, ponieważ mutacja ta występowała w około 90% populacji wirusa grypy (w grudniu 2009 r.). Natomiast, mutacja ta występowała globalnie tylko w 35% populacji. Rozprzestrzenianie się mutacji E391K zaobserwowano również w innych krajach: Finlandii [61], [120], Australii i Nowej Zelandii [7], Brazylii [38], Wielkiej Brytanii [47], Kanadzie

[48], Włoszech[93], Malezji [6], Japonii [92], Hong Kongu [83], Tajwanie [71]. Ze względu na nietypowe właściwości oraz epidemiologiczny, globalny charakter, wirus H1N1 cieszył się dużym zainteresowaniem badaczy. Dzięki temu wyizolowano ponad 9 000 sekwencji tego wirusa z okresu pandemii, z czego połowa posiada dokładną datę oraz lokalizację izolacji.

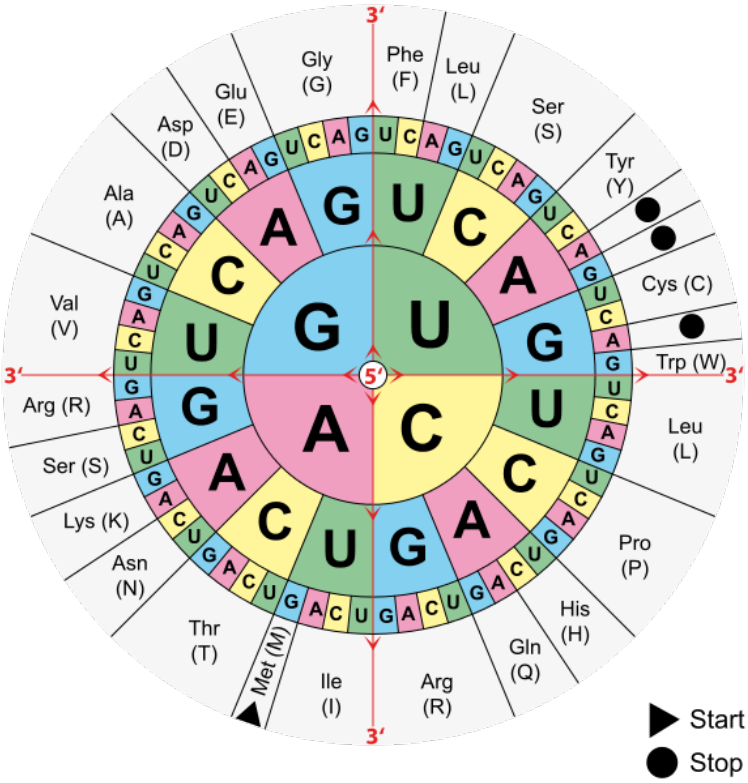
W niniejszym rozdziale ujęto i zbadano problem powtarzającej się mutacji E391K. W tym celu użyto metody selekcji cech do wyznaczenia istotności nukleotydów w sekwencji oraz nowej metody rekonstrukcji drzewa filogenetycznego, pozwalającej ją uwzględnić. Wagi nukleotydów wyznaczono za pomocą algorytmu genetycznego, który dobiera wartości wag tak, by jakość powstałych klastrów w uzyskanym drzewie była jak najlepsza. Analiza przedstawiona w tym rozdziale została również przedstawiona w artykule [108].

5.2 Opis analizowanych danych

Analizowany zbiór danych tworzą sekwencje RNA wirusa grypy. Każda sekwencja S opisana jest łańcuchem znaków - nukleotydów. W łańcuchu RNA występują 4 rodzaje nukleotydów:

- A - adenina,
- C - cytozyna,
- G - guanina,
- U - uracyl (występuje w RNA zamiast tyminy, która występuje w łańcuchach DNA).

Trójki nukleotydów tworzą kodon, który koduje aminokwas. Istnieje 64 różnych kombinacji kodonów, które kodują 20 różnych aminokwasów, kodon start (inicjujący translację) oraz kodon stop (koniec translacji). Sposób kodowania aminokwasów przedstawiony jest na rysunku 5.1. Zbiór analizowanych sekwencji można zapisać jako $D_S = \{S_1, S_2, ..., S_N\}$, gdzie N to liczba sekwencji, a każdą sekwencję jako zbiór M nukleotydów, gdzie i-ta pozycja na sekwencji przedstawiona jest jako $S(i) \in \{A, C, G, U\}$.



Rysunek 5.1: Kod genetyczny, czyli reguły kodowania aminokwasów za pomocą nukleotydow. Źródło: `https://upload.wikimedia.org/wikipedia/commons/7/70/Aminoacids_table.svg`

W bazie NCBI[1] w dniu 9 czerwca 2012 roku dostępnych było 9 216 sekwencji nukleotydowych hemaglutyniny dla wirusa grypy, z czego 6 118 było kompletnymi genami o długości 1 701 nukleotydów. Na ich podstawie został zbudowany zbiór 3 243, zawierający tylko unikalne sekwencje. W przypadku powtórzeń łańcucha RNA, do zbioru włączone były sekwencje z wcześniejszą datą izolacji. W utworzonym zbiorze dla 0,96% sekwencji jest dostępna informacja tylko o roku izolacji, dla 6,54% sekwencji jest dostępna informacja tylko o miesiącu i roku izolacji, natomiast dla pozostałych sekwencji jest dostępna pełna informacja o dacie izolacji (dzień, miesiąc, rok).

Na podstawie tego zbioru zrekonstruowane zostało drzewo filogenetyczne za pomocą algorytmu *Neighbor-Joining+* (rozdział 1.2.2). W drzewie tym stwierdzono 113 przypadków mutacji G1171A (mutacja nukleotydów guaniny w adeninę na pozycji 1171, przedstawia mutację E391K, tylko, że zapisaną za pomocą nukleotydów zamiast aminokwasów), oraz 24 przypadki mutacji A1171G. W drzewie tym można wydzielić 3 grupy sekwencji:

[1]NCBI - National Center for Biotechnology Information

dwie grupy z sekwencjami, które zawierają na pozycji 1171 nukleotyd typu adenina lub guanina oraz grupę, która zawiera sekwencje, pomiędzy którymi zaszła mutacja G1171A lub A1171G. W dalszej analizie użyto sekwencji z ostatniej grupy (203 sekwencje), co zmniejszyło liczbę analizowanych sekwencji, przez co analiza i wizualizacja stała się łatwiejsza. Na rysunku 5.2 przedstawiono drzewo filogenetyczne zrekonstruowane za pomocą algorytmu NJ+ na wybranych sekwencjach. Na drzewie różnymi kolorami oznaczono wierzchołki drzewa z sekwencjami, w zależności od posiadanego nukleotydu na pozycji 1171. W uzyskanym drzewie nie można wyróżnić jednoznacznego podziału sekwencji na grupy ze względu na wartość nukleotydu na tej pozycji - sekwencje są rozrzucone po całym drzewie. W dalszej części rozdziału zostanie przedstawiona metoda szukająca wag opisujących istotność nukleotydów w łańcuchu RNA tak, by zrekonstruowane drzewo było jak najlepiej pogrupowane ze względu na mutacje występujące w analizowanym zbiorze.

5.3 Dobór wag w łańcuchu RNA

Dobór wag został przeprowadzony za pomocą algorytmu genetycznego opisanego w rozdziale 2.3.3. Aby móc go zastosować, należy zdefiniować: kodowanie chromosomu, operatory ewolucyjne oraz funkcję dopasowania. Elementy te zostały opisane poniżej. Dodatkowo zaproponowano różne techniki inicjalizacji chromosomów.

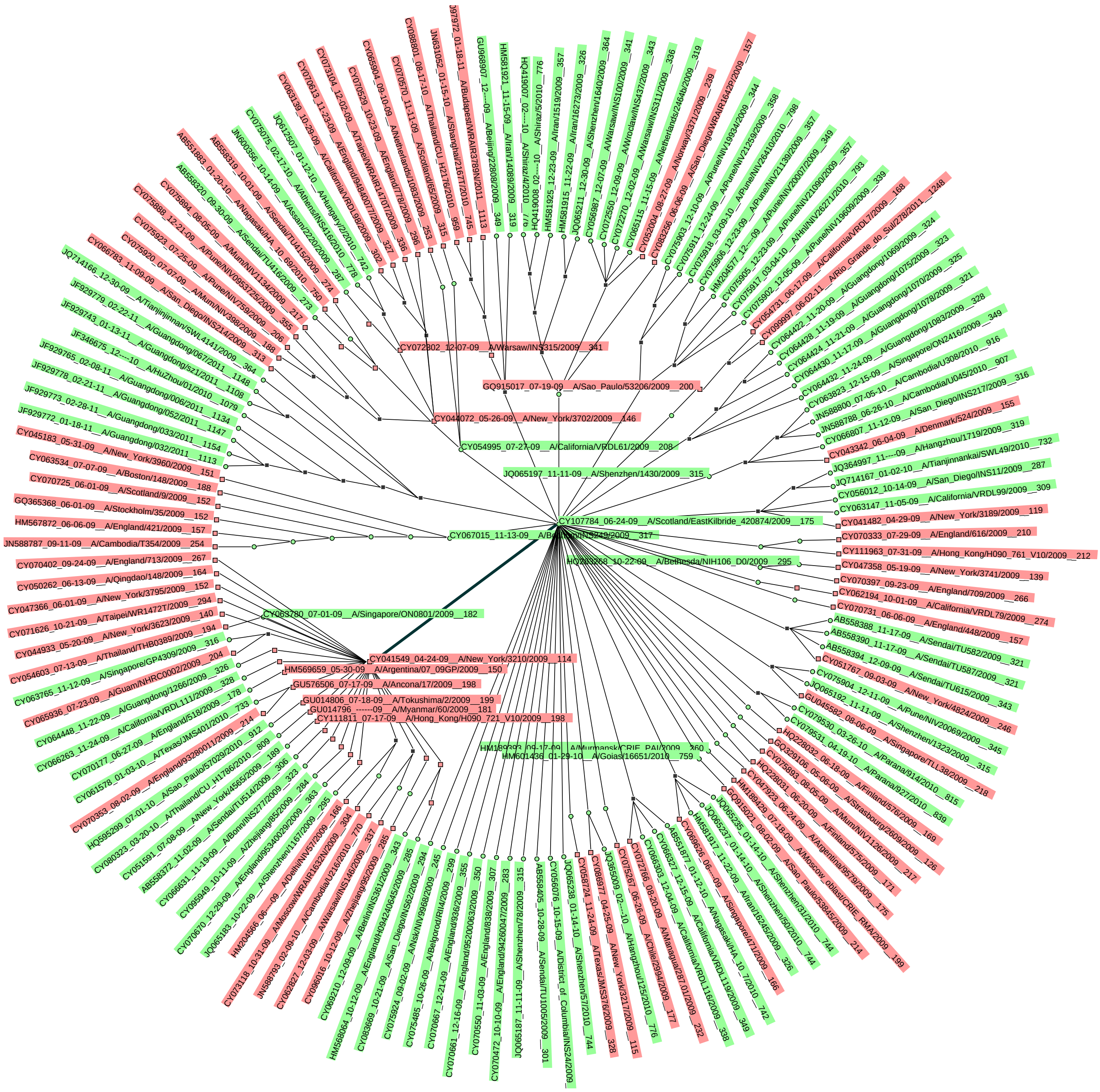
Kodowanie chromosomu

W rozwiązywanym problemie chromosom algorytmu genetycznego przedstawia ważność nukleotydów w łańcuchu RNA wirusa grypy. W chromosomie $\mathbf{V}$ każdy jego i-ty element jest liczbą naturalną i zawiera się w przedziale $[1; 100]$. Miejsce na łańcuchu RNA, które jest ważne podczas analizy i kluczowe dla prawidłowego zrekonstruowania drzewa, ma przypisane większe wartości, natomiast mało istotnym miejscom na łańcuchu RNA odpowiadają niskie wartości. Wartość istotności będzie wykorzystywana w procesie rekonstrukcji drzewa filogenetycznego. Odległość pomiędzy sekwencjami p, q z wykorzystaniem wag obliczana jest na podstawie wzoru:

$$d(p,q) = 1/M \sum_{i=1}^{M} \mathbf{V}(i)|S_p(i) - S_q(i)|, \tag{5.1}$$

który zlicza średnią liczbę różnic pomiędzy sekwencjami z uwzględnieniem wag.

Nie wszystkie miejsca w chromosomie wymagają doboru wagi. Dla miejsc na łańcuchu RNA, które mają ten sam nukleotyd w całym analizowanym zbiorze nie ma potrzeby doboru wag, ponieważ miejsca te nie wnoszą nic do analizy.



Rysunek 5.2: Drzewo filogenetyczne uzyskane za pomocą algorytmu NJ+ na zbiorze 203 sekwencji wirusa grypy H1N1. Kolorem czerwonym zaznaczone są sekwencje mające na pozycji 1171 nukleotyd typu G, natomiast kolorem zielonym zaznaczono sekwencje z nukleotydem typu A na pozycji 1171.

Inicjalizacja chromosomu

W ramach badań opisanych w niniejszej rozprawie sprawdzono działanie trzech sposobów inicjalizacji wag chromosomu:

- inicjalizacja losowa, gdzie każda waga losowana jest z przedziału $[1; 100]$,
- inicjalizacja losowa z uwzględnieniem pozycji na kodonie - pozycje, które odpowiadają pierwszej pozycji w kodonie mają wagi losowane z przedziału $[1; 100]$, nukleotydy z drugiej pozycji mają wagi wylosowane z przedziału $[1; 50]$, natomiast nukleotyd na trzecim miejscu w kodonie ma wagi losowane z przedziału $[1; 25]$,
- inicjalizacja wartościami wyznaczonymi przez eksperta.

Inicjalizacja losowa z uwzględnieniem pozycji na kodonie ma sens biologiczny. Nukleotydy, które są na trzecim miejscu w kodonie zmieniają się częściej, ale mimo ich zmiany nie musi dochodzić do zmiany aminokwasu (mutacja synonimiczna) - nie dochodzi do zmian właściwości wirusa. Dlatego ich zmiana powinna mieć mniejsze znaczenie niż zmiana nukleotydów na pierwszej pozycji, która niesie ze sobą mutację aminokwasu, a co za tym idzie - modyfikację w budowie wirusa.

Operatory ewolucyjne

W rozwiązywanym problemie zostały użyte standardowe operatory ewolucyjne: operator mutacji i krzyżowania. Przebieg operacji mutowania dla każdego chromosomu V_i można opisać poniższymi krokami:

1. Utwórz kopię chromosomu $V_i' = V_i$.
2. Dla każdej wagi j w chromosomie V_i' wylosuj liczbę l z rozkładu jednostajnego z przedziału $[0; 1]$,
3. Jeżeli wylosowana liczba jest mniejsza niż współczynnik μ, $l < \mu$, zmień wagę na pozycji j na losowo wybraną liczbę z przedziału $[1; 100]$, $V_i(j) = rand(1, 100)$.
4. Po rozpatrzeniu wszystkich wag chromosomu V_i', jeżeli różni się on od chromosomu oryginalnego V_i, $V_i' \neq V_i$, jest on dodany do populacji Pop.

Szybkość mutowania chromosomów kontrolowana jest przez współczynnik μ. Dla dużych wartości współczynnika μ nowo powstały chromosom ma wiele zmienionych miejsc w porównaniu z chromosomem, z którego powstał. W operacji mutowania bierze udział jeden chromosom, natomiast w procesie krzyżowania biorą udział dwa chromosomy V_i oraz V_j, gdzie $i \neq j$. Procedurę krzyżowania można przedstawić w następujących krokach:

1. Sprawdź każdą parę chromosomów V_i oraz V_j, gdzie $i \neq j$.
2. Wylosuj liczbę l z rozkładu jednostajnego z przedziału $[0; 1]$,
3. Jeżeli wylosowana liczba jest mniejsza niż współczynnik γ, $l < \gamma$, wykonana zostanie operacja krzyżowania, w przeciwnym wypadku należy sprawdzić kolejną parę chromosomów (krok 1).
4. Wylosuj liczbę k z przedziału $[1, M]$,
5. Stwórz dwa nowe chromosomy V' i V'' przez podzielenie chromosomów V_i oraz V_j w miejscu k i krzyżowe złączenie, $V' = \{V_i(1 : k), V_j(k + 1 : M)\}$, $V'' = \{V_j(1 : k), V_i(k + 1 : M)\}$,
6. Dodaj nowo utworzone chromosomy V' i V'' do populacji Pop.

Współczynnik γ kontroluje częstość krzyżowania się chromosomów - im większe wartości przyjmuje γ, tym więcej powstaje nowych chromosomów w wyniku operacji krzyżowania.

Funkcja dopasowania

Za pomocą funkcji dopasowania $\mathcal{F}(V)$ oceniana jest jakość rekonstrukcji drzewa filogenetyczne przy użyciu informacji zakodowanych w chromosomie V. Jakość zrekonstruowanego drzewa można sprawdzić zliczając liczbę dobrze zrekonstruowanych klastrów w wynikowym drzewie. Klaster sekwencji to zbiór sekwencji pochodzących od tego samego przodka. Przed rekonstrukcją drzewa filogenetycznego trudno jest wskazać, które sekwencje powinny być przedstawione jako jeden klaster. Można na przykład zakładać, że sekwencje, które zostały wyizolowane w podobnym czasie i miejscu mają te same pochodzenie. Niestety, nie dla wszystkich sekwencji dostępne są takie informacje, a co więcej, czas wyizolowania sekwencji jest tylko przybliżoną datą powstania sekwencji.

W rozprawie by wyznaczyć oczekiwane klastry sekwencji w zrekonstruowanym drzewie przyjęto kryterium podobieństwa sekwencji na określonym miejscu w łańcuchu RNA. Procedura wyznaczająca oczekiwane klastry sekwencji sprawdza kolejno każdą i-tą pozycję w łańcuchu RNA. Algorytm szukający klastrów opisany jest w krokach poniżej:

1. Sprawdź każdą i-tą pozycję w łańcuchu RNA dla analizowanych sekwencji.
2. Rodzaj nukleotydu na i-tej pozycji dzieli zbiór sekwencji $D_S = D_A^i \bigcup D_C^i \bigcup D_T^i \bigcup D_G^i$ na 4 rozłączne zbiory, do których przynależność określona jest rodzajem nukleotydu,
3. Spośród podzbiorów wybierz taki, który jest najmniej liczny i jednocześnie niepusty. Będzie on oznaczony jako D_j^i.
4. Jeżeli $|D_j^i| \leqslant \kappa$, gdzie κ oznacza minimalną liczbę sekwencji w klastrze, wróć do punktu 1 i kontynuuj przeszukiwanie dla kolejnej pozycji na łańcuchu RNA.
5. Jeżeli $|D_j^i| > \kappa$, dodaj znaleziony zbiór do zbiorów klastrów $K' = K \bigcup D_j^i$. Dodatkowo, sprawdź czy dla dowolnej sekwencji S ze zbioru D_j^i istnieje sekwencja S' w zbiorze $D_S \setminus D_j^i$ taka, że sekwencje S oraz S' różnią się tylko jednym nukleotydem (jedną mutacją). Jeżeli tak, to zapamiętaj, że dla zbioru D_j^i istnieje taka para sekwencji S i S'.

W opisany powyżej sposób wyznaczane są oczekiwane klastry sekwencji, które powinny być zaobserwowane w zrekonstruowanym drzewie filogenetycznym. Klastry definiowane są ze względu na posiadanie tego samego nukleotydu na określonej pozycji. Dodatkowo, w ostatnim kroku algorytmu dla każdego klastra wyznaczana jest para sekwencji S, S', które różnią się tylko jednym nukleotydem (tzw. *Single Nucleotide Polymorphism* (SNP)) i nie są w tym samym podzbiorze w podziale na nukleotydy na i-tym miejscu. Krok ten ma na celu znalezienie pary sekwencji, która powinna łączyć wyznaczony klaster z drzewem.

W funkcji dopasowania sprawdzane jest na ile dobrze oczekiwane klastry są zrekonstruowane w drzewie z wykorzystaniem wag chromosomu V. Funkcję dopasowania można przedstawić jako sumę dokładności rekonstrukcji klastrów:

$$\mathcal{F}(V) = \sum_{i=|K|} R(K_i), \tag{5.2}$$

gdzie $R(K_i)$ określa dokładność rekonstrukcji klastra K_i. By móc określić dokładność rekonstrukcji klastra, należy znaleźć go w uzyskanym drzewie $T(P, E)$, gdzie P to wierzchołki drzewa, a E jego krawędzie. W drzewie T istnieją takie wierzchołki, które odpowiadają sekwencjom ze zbioru D_S, $|D_S| = |D_S \bigcap P|$, $|P| \geqslant |D_S|$. Każda krawędź E_i ze zbioru E definiuje podział wierzchołków na dwa rozłączne podzbiory: P_i oraz P_i'. Dokładność rekonstrukcji klastra K wyznaczana jest następująco:

1. Sprawdź każdą krawędź E_i w drzewie T.
2. Wyznacz podzbiory wierzchołków P_i oraz P_i' zdefiniowane krawędzią E_i.
3. Wyznacz maksymalne pokrycie z sekwencjami z klastra K_i dla każdego podziału wierzchołków:
$$r_i = max(\frac{2|K \bigcap P_i|}{|K| + |P_i|}, \frac{2|K \bigcap P_i'|}{|K| + |P_i'|}) \tag{5.3}$$
4. Spośród wszystkich obliczonych pokryć wybierz maksymalne,
$$r_{max} = argmax_i(r_i), \tag{5.4}$$
oraz zapamiętaj definiującą go krawędź E_i^{max}.
5. Oblicz dokładność rekonstrukcji klastra za pomocą wzoru:

$$R(K_i) = \begin{cases} (1 - r_{max}) \cdot |K|^2, & \text{jeżeli nie istnieje para z SNP ,} \\ (1 - r_{max}) \cdot |K|^2, & \text{jeżeli istnieje para z SNP i są to wierzchołki } E_i^{max}, \\ |K|^2, & \text{jeżeli istnieje para z SNP i nie są to wierzchołki } E_i^{max}. \end{cases} \tag{5.5}$$

Za pomocą opisanej funkcji dopasowania ocenia się drzewo zrekonstruowane z wykorzystaniem wag wyznaczonych algorytmem genetycznym. Przedstawiona funkcja dopasowania może zostać również użyta do oceny jakości rekonstrukcji drzewa za pomocą innego algorytmu, niekoniecznie wykorzystującego ważenie nukleotydów. Im większa wartość funkcji dopasowania, tym słabsze odwzorowanie szukanych klastrów w drzewie.

5.4 Wyniki analizy

Algorytm genetyczny został uruchomiony z następującymi parametrami:

- liczebność populacji chromosomów $|Pop| = 100$,
- liczba iteracji algorytmu = 500,
- minimalna wielkość rozpatrywanego klastra $\kappa = 5$,
- współczynnik krzyżowania $\gamma = 0,005$,
- współczynnik mutacji $\mu = 0,1$,
- trzy różne sposoby inicjalizacji: losowa, losowa z uwzględnieniem położenia w kodonie, wartościami wyznaczonymi przez eksperta,
- wszystkie obliczenia powtórzono 5 razy.

Uzyskane wartości funkcji dopasowania przedstawiono na rysunku 5.3. Najniższe wartości funkcji dopasowania zostały osiągnięte przy starcie algorytmu z wykorzystaniem wag wyznaczonych przez eksperta, aczkolwiek dwa przebiegi uzyskały wartości z podobnego zakresu przy użyciu inicjalizacji losowej z uwzględnieniem położenia nukleotydu w kodonie. Dodatkowo, warto zauważyć, iż algorytm genetyczny dla inicjalizacji losowej z uwzględnieniem położenia nukleotydu w kodonie szybciej minimalizuje funkcję dopasowania niż inicjalizacja całkowicie losowa - inicjalizacja algorytmu genetycznego potrafiąca zawrzeć "biologiczny sens" rozwiązywanego problemu jest bardziej skuteczna. Na rysunku 5.5 przedstawiono uzyskane drzewo dla najniższej wartości funkcji dopasowania. Podobnie jak na rysunku 5.2, węzły drzewa zostały pokolorowane w zależności od wartości nukleotydu na pozycji 1171. W uzyskanym drzewie sekwencje pogrupowane są w taki sposób, że ewolucję w obserwowanym zbiorze można wytłumaczyć za pomocą tylko jednej mutacji E391K pomiędzy sekwencjami o numerach CY041549 i CY107784. Dodatkowo, do oceny zrekonstruowanego drzewa mogą zostać wykorzystane daty izolacji sekwencji (sekwencje potomne powinny mieć datę izolacji późniejszą niż sekwencje matki, z których powstały). W otrzymanym zrekonstruowanym drzewie zostały wyznaczone wszystkie pary sekwencji (matka, córka), przy użyciu sekwencji CY041549 jako korzenia całego drzewa. Wśród nich chronologii nie zachowało 7 par, a średnia różnica w dniach pomiędzy pomylonymi parami

sekwencji wynosiła 21 dni. Natomiast dla drzewa uzyskanego bez użycia wag istotności nukleotydów, pomyłek czasowych było 41, ze średnią różnicą 91,85 dnia pomiędzy nimi.

Opisany algorytm został również zastosowany do analizy pełnego zbioru 3 243 sekwencji. Algorytm genetyczny został uruchomiony z tymi samymi parametrami, ale z mniejszą liczbą iteracji. Z uwagi na wysoki koszt obliczeniowy rekonstrukcji pojedynczego drzewa za pomocą algorytmu NJ+ dla dużego zbioru sekwencji przeprowadzono jedynie 30 iteracji algorytmu genetycznego. Do inicjalizacji chromosomów wykorzystano wartości wyznaczone arbitralnie przez eksperta. Wynik działania algorytmu genetycznego przedstawiono na rysunku 5.4. W uzyskanym drzewie na pozycji 1171 występuje tylko jedna mutacja, pomiędzy sekwencjami CY041549 i CY107784. Ta sama para sekwencji została wybrana podczas wyznaczania wag dla nukleotydów przy użyciu zbioru 203, który jest podzbiorem zbioru z 3 243 sekwencjami.

Uzyskane drzewa zostały porównane z drzewami uzyskanymi za pomocą algorytmu goeBURST [41], który wyznacza globalnie optymalne połączenia pomiędzy sekwencjami wykorzystując algorytm eBURST [35]. Algorytm ten jest dobrym przybliżeniem tego jak mogła wyglądać prawdziwa ewolucja wirusa grypy, aczkolwiek ma tę wadę, iż zakłada, że w analizowanym zbiorze występują wszystkie sekwencje, które wystąpiły w środowisku, co w przypadku analizy sekwencji wirusa grypy nie jest prawdą. Wyniki porównania dla obu zbiorów przedstawiono w tabeli 5.1. Najniższe wartości funkcji dopasowania uzyskiwane są dla metody używającej wektora wag wyznaczonego za pomocą algorytmu genetycznego, jednakże grupowanie sekwencji za pomocą algorytmu goeBURST osiąga niewiele gorsze wyniki. Drzewa zrekonstruowane przy pomocy samego algorytmu NJ+ uzyskują wartości funkcji dopasowania o rząd wielkości większe niż dla rekonstrukcji z uwzględnieniem wag.

5.5 Dyskusja

Podczas rekonstrukcji drzewa filogenetycznego na podstawie danych z epidemii z 2009 roku niektóre mutacje zwróciły na siebie większą uwagę badaczy, ponieważ powtórzyły się kilkadziesiąt razy w różnym czasie i miejscach. Taką mutacją jest zmiana aminokwasu E (kwas glutaminowy) w K (lizyna) na pozycji 391 w łańcuchu kodującym hemaglutyninę (HA) - mutacja adeniny w guaninę na pozycji 1171 na poziomie nukleotydowym.

Tabela 5.1: Zestawienie wartości funkcji dopasowania dla drzew zrekonstruowanych przy pomocy różnych metod.

Metoda rekonstrukcji	Wartość funkcji dopasowania
Zbiór 203 sekwencji	
NJ+	7 885,36
goeBURST	1 084,65
NJ+ z wykorzystaniem wag	729,77
Zbiór 3 243 sekwencji	
NJ+	1 163 430
goeBURST	144 715
NJ+ z wykorzystaniem wag	131 326

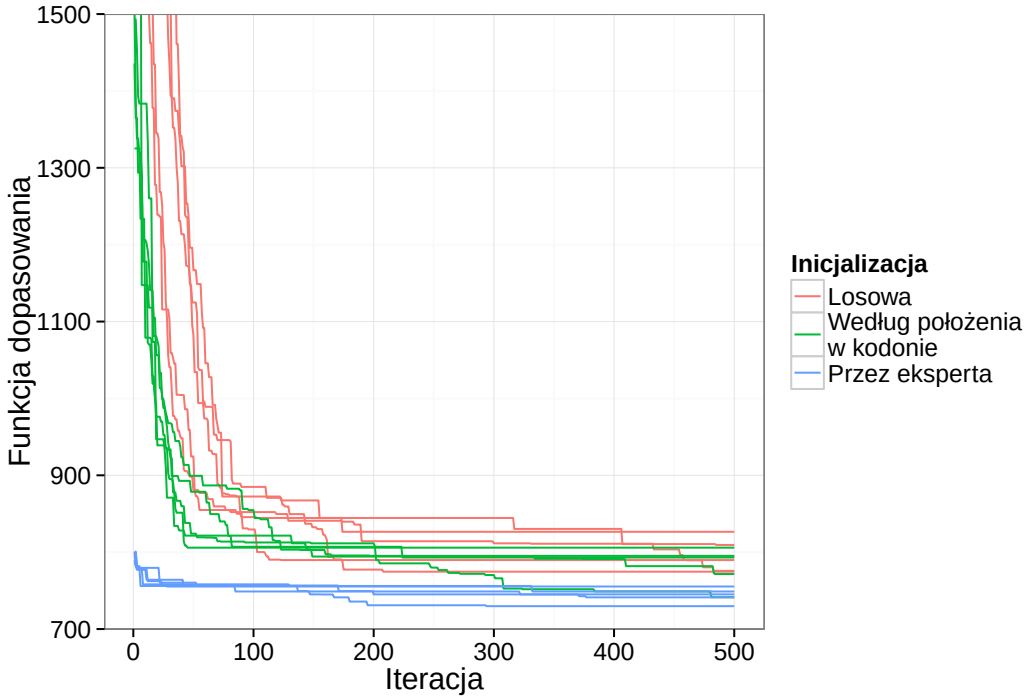
W drzewie zrekonstruowanym za pomocą algorytmu NJ+ mutacja ta jest powtórzona aż 113 razy. Co więcej, występują 24 przypadki mutacji odwrotnej - adeniny w guaninę na pozycji 1171. W rozprawie sprawdzono czy obserwowane mutacje są artefaktem algorytmu rekonstrukcji, czy rzeczywiście wirus mutował niezależnie kilkadziesiąt razy w ten sam sposób, w różnym czasie i różnych miejscach.

W analizie danych zaproponowano wprowadzenie wektora wag, który dla każdej pozycji w łańcuchu RNA wirusa przypisuje istotność - im większa waga, tym większa uwaga powinna być poświęcona danej pozycji w trakcie rekonstrukcji. W rozprawie została opisana metoda szukająca najlepszego zestawu wag za pomocą algorytmu genetycznego. Funkcja dopasowania w zaproponowanym algorytmie opisuje dokładność rozmieszczenia klastrów w wynikowym drzewie. Sprawdzane klastry zostały zdefiniowane ze względu na posiadanie tego samego nukleotydu na określonej pozycji. Do analizy użyto dwóch zbiorów danych o wielkości 203 i 3 243 sekwencji. Mniejszy zbiór jest podzbiorem większego i zawiera sekwencje, pomiędzy którymi w drzewie obliczonym dla 3243 sekwencji została powtórzona mutacja G1171A lub A1171G. Zaproponowaną metodą wyznaczono wektor wag dla nukleotydów dla obu zbiorów. W drzewie zrekonstruowanym przy użyciu wektora wag w obu rozpatrywanych zbiorach uzyskiwana jest tylko jedna mutacja G1171A pomiędzy sekwencjami CY041549 i CY107784. Dodatkowo, rekonstrukcja ewolucji została sprawdzona algorytmem goeBURST, w którym również uzyskano tylko jedną mutację E391K pomiędzy sekwencjami CY041549 i CY107784. Na podstawie uzyskanych wyników można stwierdzić, iż obserwacja powtarzających się mutacji E391K jest wynikiem błędnego działania algorytmu użytego do rekonstrukcji. Algorytm NJ+ może nie radzić

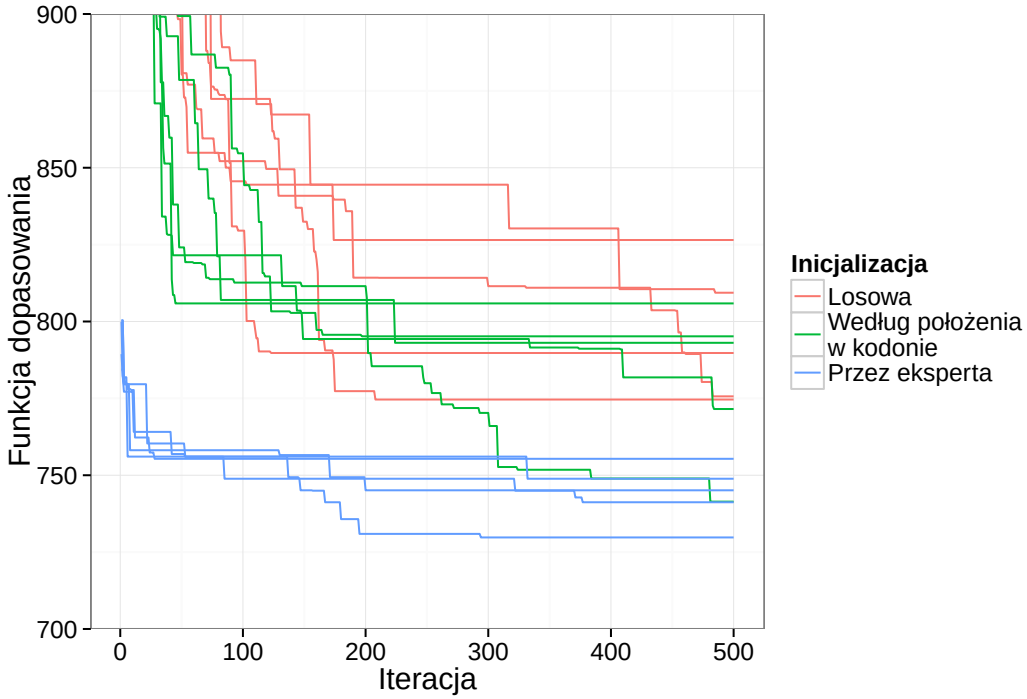
sobie z rekonstrukcją rozpatrywanego zbioru, ponieważ jest algorytmem zachłannym i utyka w minimum lokalnym. Wykorzystanie wektora wag opisującego istotność nukleotydów w trakcie rekonstrukcji drzewa filogenetycznego za pomocą algorytmu NJ+ pozwala zwiększyć przestrzeń hipotetycznych rozwiązań.

Na podstawie drzewa zrekonstruowanego przy użyciu wag znalezionych dla 3 243 sekwencji został oszacowany współczynnik mutacji wirusa grypy typu A i wynosi $3,84 \cdot 10^{-4}$ mutacji na każdy nukleotyd w ciągu roku (szczegóły w pracy [108]). W porównaniu z innymi oszacowaniami współczynników, na przykład $5,7 \cdot 10^{-3}$ z pracy [40] lub $1,56 \cdot 10^{-3}$ [48] jest on o rząd wielkości mniejszy, co sugeruje, że wirus mutuje wolniej. W pracy [108] wykazano, że współczynnik mutacji może zostać łatwo przeszacowany jeżeli zostanie użyty mały zbiór sekwencji oraz jeżeli mutacje w drzewie będą powtórzone w wyniku niepoprawnej rekonstrukcji drzewa.

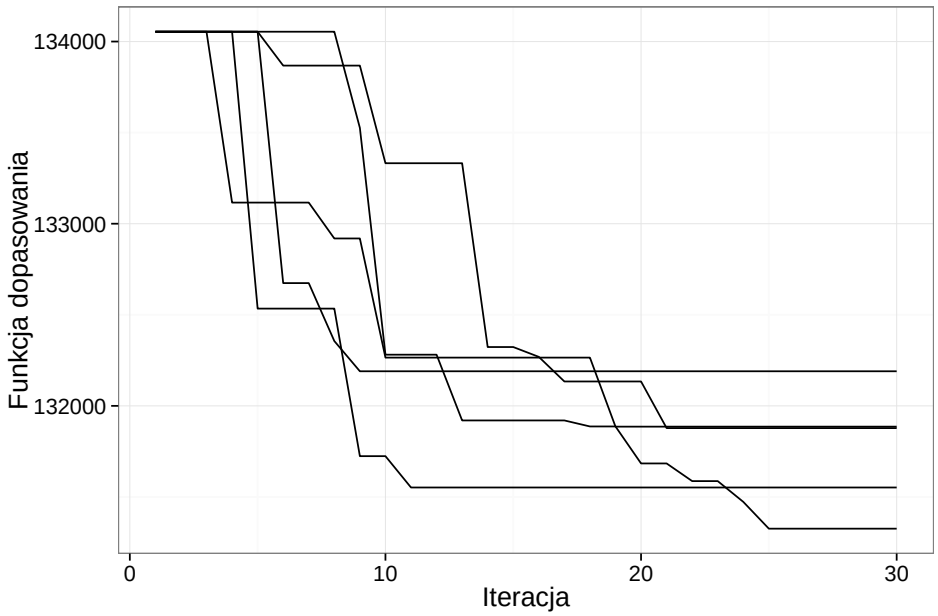
(a) Wartość funkcji dopasowania



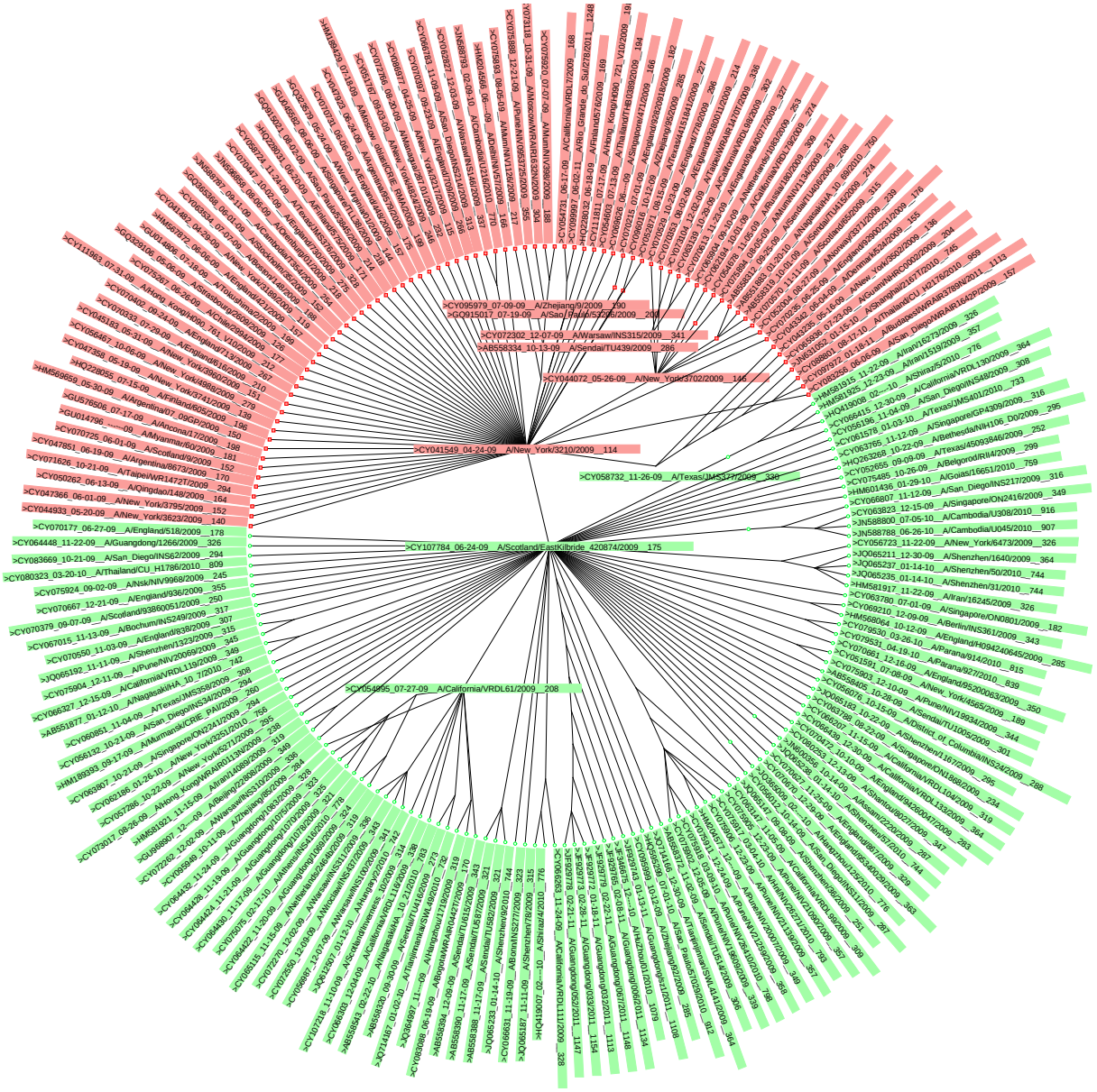
(b) Wartość funkcji dopasowania (w mniejszym zakresie wartości)



Rysunek 5.3: Wartość funkcji dopasowania dla najlepszego chromosomu w zależności od kroku iteracji wyznaczona w czasie optymalizacji wag przedstawiających istotność nukleotydów w rekonstrukcji drzewa filogenetycznego dla zbioru 203 sekwencji wirusa grypy typu H1N1. Na wykresie (b) przedstawiono te same przebiegi, tylko w innym zakresie wartości funkcji dopasowania.



Rysunek 5.4: Wartość funkcji dopasowania dla najlepszego chromosomu w zależności od kroku iteracji wyznaczona w czasie optymalizacji wag przedstawiających istotność nukleotydów w rekonstrukcji drzewa filogenetycznego dla zbioru 3 243 sekwencji wirusa grypy typu H1N1.



Rysunek 5.5: Zrekonstruowane drzewo uzyskane za pomocą algorytmu NJ+ oraz wag wyznaczonych algorytmem genetycznym uzyskane na zbiorze 203 sekwencji wirusa grypy H1N1. Kolorem czerwonym zaznaczone są sekwencje mające na pozycji 1171 nukleotyd typu G, natomiast kolorem zielonym zaznaczono sekwencje z nukleotydem typu A na pozycji 1171.

Rozdział 6

Klasyfikacja dzieci z dysleksją

W niniejszym rozdziale zastosowano metody przekształcenia i selekcji cech do zbudowania klasyfikatora rozróżniającego dzieci ze względu na posiadanie dysleksji rozwojowej na podstawie badań ze strukturalnego rezonansu magnetycznego. Dzięki uzyskanym wynikom możliwe było wskazanie różnic anatomicznych pomiędzy dziećmi zdrowymi a dziećmi z dysleksją rozwojową.

6.1 Opis problemu

Dysleksja rozwojowa (ang. *developmental dyslexia*) jest to zaburzenie w rozwoju, charakteryzujące się trudnością w nauce pisania i czytania przy stosowaniu standardowych metod nauczania oraz inteligencji na poziomie przynajmniej przeciętnym. Zaburzenia mogą dotyczyć percepcji wzrokowej, fonologicznej lub obu jednocześnie. Pomimo wielu lat badań, przyczyna tego zaburzenia na poziomie anatomicznym jak dotąd nie jest znana. Pierwsze wyniki badań histologicznych przeprowadzonych *post-mortem* w poszukiwaniu zmian w strukturze mózgu zostały opisane przez Galaburda et. al [44], [45] w latach 1979 i 1985. Stwierdzone zostały wtedy różnice w budowie mózgu w okolicy bruzdy Sylwiusza w lewej półkuli mózgu, które były tłumaczone zaburzeniem w migracji neuronów w fazie prenatalnej [45], [59]. Znalezione różnice zostały później potwierdzone w następnych badaniach [15], [17], [26]. Kolejną cechą anatomiczną zaburzenia dysleksji rozwojowej, stwierdzonej w badaniach post-mortem jest mniejsza liczba neuronów w następujących częściach mózgu: wzgórze, móżdżek oraz pierwszorzędowa kora wzrokowa [68], [39], [46].

Rozwój technik obrazowania z użyciem tomografii rezonansu magnetycznego ułatwił

analizę różnic w strukturze mózgu dzieci zdrowych i dotkniętych zaburzeniem dysleksji. Opublikowano dotychczas wiele prac (ponad 20) analizujących objętość istoty szarej w mózgu za pomocą metody *Voxel-Based Morphometry (VBM)*, aczkolwiek analiza wyników uzyskanych w tych pracach pokazuje, że ich wyniki są rozbieżne [80], [111], [67]. Alternatywą w badaniach metodą rezonansu magnetycznego zostają miary pozwalające przedstawić grubość oraz powierzchnię struktur w istocie szarej, które nie były dotąd tak intensywnie sprawdzane jak objętość istoty szarej [43], [82], [21].

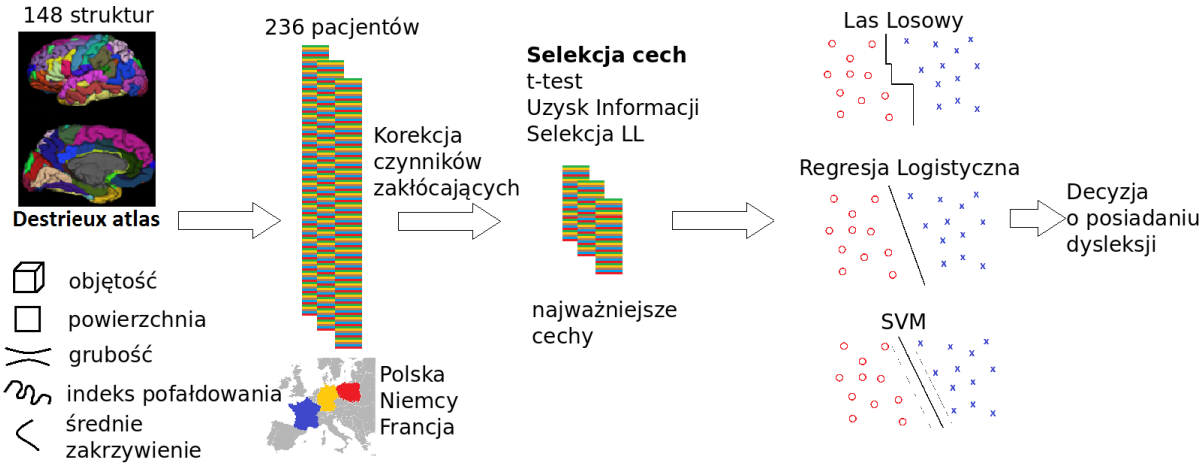
Wszystkie wymienione powyżej badania dysleksji rozwojowej wykorzystywały analizę za pomocą jednej zmiennej (ang. *univariate analysis*). Takie podejście może pomijać potencjalne zależności pomiędzy strukturami, jakie mogą występować w przypadku zaburzeń. W rozprawie przedstawiono podejście wykorzystujące analizę wielu zmiennych (ang.*multivariate analysis*) do wykrycia kluczowych struktur w klasyfikacji zaburzeń dysleksji rozwojowej u dzieci. Według wiedzy autora rozprawy, jest to pierwsze podejście tego typu w badaniu dysleksji u dzieci na podstawie strukturalnego rezonansu magnetycznego.

W badaniach przeprowadzonych w rozprawie wykorzystano dane użyte w pracy [67]. Dane pochodzą z obrazowania rezonansem magnetycznym i zostały zebrane w ośrodkach badawczych z Polski, Francji i Niemiec. W celu znalezienia struktur, które najbardziej różnicują dzieci ze względu na dysleksję rozwojową wykonano następujące kroki:

- segmentacja trójwymiarowego obrazu mózgu w celu wyodrębnienia w nim struktur według atlasu Destrieux [27] oraz ich cech, takich jak: objętość, powierzchnia, grubość, średnie zakrzywienie, wskaźnik pofałdowania,
- usunięcie czynników zakłócających,
- selekcja kluczowych cech,
- klasyfikacja z wykorzystaniem wybranych cech.

Przed zastosowaniem klasyfikatora dane uzyskane ze skanera są transformowane tak, by opisywały struktury występujące w mózgu. W procesie segmentacji otrzymywany jest zestaw kilkuset cech dla każdego badania (osoby). By wskazać te najważniejsze, wykonywana jest selekcja cech. Przed selekcją cech wykonywany jest proces usuwania czynników zakłócających, ponieważ analizowane dane pochodzą z kilku ośrodków badawczych, gdzie obrazowanie było przeprowadzone w różnych konfiguracjach skanera. Dodatkowo dane

zależą od takich czynników, jak płeć czy całkowita objętość wewnątrz czaszki (*Total Intracranial Volume* (TIV)). Na podstawie wybranych cech budowany jest klasyfikator. Schemat analizy użyty w rozprawie przedstawiono na rysunku 6.1. Opisane w rozprawie wyniki zostały również przedstawione w artykułach [98], [99].



Rysunek 6.1: Schemat analizy danych, zastosowany w klasyfikacji dzieci z dysleksją rozwojową, na podstawie obrazów uzyskanych ze strukturalnego rezonansu magnetycznego.

6.2 Opis danych

Analizowany zbiór danych składa się z 236 obrazów głów dzieci uzyskanych przy pomocy T1-zależnego rezonansu magnetycznego. Dane zostały zebrane w ośrodkach naukowych z 3 krajów:

- Polski - 81 dzieci, w tym 35 dzieci kontrolnych (22 dziewcząt) i 46 dzieci z dysleksją (20 dziewcząt),
- Francji - 84 dzieci, w tym 45 dzieci kontrolnych (23 dziewcząt) i 39 dzieci z dysleksją (14 dziewcząt),
- Niemiec - 71 dzieci, w tym 26 dzieci kontrolnych (10 dziewcząt) i 45 dzieci z dysleksją (22 dziewcząt).

Uczestnicy badania pochodzili ze zróżnicowanych społecznych środowisk; ukończyli co najmniej półtora roku nauki czytania - tak, by móc odróżnić trwałe problemy z czytaniem od początkowych, które są przejściowe. Wszystkie przebadane osoby spełniały następujące warunki:

- wiek pomiędzy 8,5 i 13,7 lat,
- iloraz inteligencji większy niż 85,
- brak zdiagnozowanej choroby ADHD,
- brak problemów z widzeniem, słyszeniem lub innych problemów neurologicznych.

Dysleksja u dzieci została zdiagnozowana w szkole lub na podstawie testu klinicznego. Wszystkie badania zostały zaakceptowane przez lokalne komitety etyczne: Uniwersytet Medyczny w Warszawie, CPP Bicêtre we Francji, Uniklinik RWTH Aachen w Niemczech. Dzieci oraz ich rodzice wyrazili pisemną zgodę na udział w badaniu.

Procedura obrazowania

Obrazy o wysokiej rozdzielczości zostały uzyskane przy pomocy T1-zależnego strukturalnego rezonansu magnetycznego. Badania zostały przeprowadzone w 3 różnych krajach. Ogólne informacje o uzyskanych danych znajdują się w tabeli 6.1. Natomiast poniżej opisano szczegóły akwizycji.

Tabela 6.1: Opis analizowanych danych ze względu na kraj badania, natężenie pola magnetycznego i posiadanie zaburzeń dysleksji rozwojowej.

	Francja	Niemcy		Polska
Natężenie pola	3 T	3 T	1,5 T	1,5 T
Dzieci zdrowe	45	11	15	35
Dzieci z dysleksją	39	35	10	46

Próbka z Polski

Dla wszystkich dzieci badanych w Polsce użyto skanera 1,5 T Siemens Avanto z 32-kanałową cewką. Uzyskane obrazy mają następującą specyfikację: macierz akwizycji $256 \times 256 \times 192$, TR=1720 ms, TE=2,92 ms, flip angle=9 deg, FOV=256 mm, rozmiar woksela $1 \times 1 \times 1$ mm.

Próbka z Francji

Dla 13 dzieci zdrowych i 11 z dysleksją użyto skanera 3 T Siemens Trio Tim MRI z 12-kanałową cewką oraz parametrami: macierz akwizycji $256 \times 256 \times 176$, TR=2300 ms,

TE=4,18 ms, flip angle=9 deg, FOV=256 mm, rozmiar woksela $1 \times 1 \times 1$ mm. Natomiast dla pozostałych 32 dzieci zdrowych i 28 z dysleksją użyto 32-kanałowej cewki z parametrami: macierz akwizycji $230 \times 230 \times 202$, TR=2300 ms, TE=3,05 ms, flip angle=9 deg, FOV=230 mm, rozmiar woksela $0,9 \times 0,9 \times 0,9$ mm.

Próbka z Niemiec

Dla 11 dzieci zdrowych i 35 z dysleksją użyto skanera 3 T Siemens Trio Tim z cewką typu *birdcage* oraz parametrami: macierz akwizycji: $256 \times 256 \times 176$, TR=1900 ms, TE=2,52 ms, flip angle=9 deg, FOV=256 mm, rozmiar woksela $1 \times 1 \times 1$ mm. Dla pozostałych 15 dzieci kontrolnych i 10 z dysleksją użyto skanera 1,5 T Siemens Avanto z cewką *birdcage* i parametrami: macierz akwizycji $256 \times 256 \times 170$, TR=2200 ms, TE=3,93ms, flip angle=15 deg, FOV=256 mm, rozmiar woksela $1 \times 1 \times 1$ mm.

6.3 Ekstrakcja danych z obrazów MRI

Każdy uczestnik badania opisany jest za pomocą:

- uzyskanego obrazu,
- klasy mówiącej czy posiada zaburzenie dysleksji,
- parametrów opisujących uczestnika (wiek, płeć, objętość mózgu),
- konfiguracji skanera użytej w badaniu.

W zadaniu klasyfikacji w opisywanym problemie wymagana jest metoda, która na podstawie wejściowego obrazu, parametrów skanera i cech osoby badanej będzie potrafiła wskazać prawdopodobieństwo dysleksji. Niestety, obraz uzyskany w badaniu nie nadaje się bezpośrednio do analizy. Wektor cech przedstawiający każdy woksel obrazu jako jedną cechę miał by ponad $1{,}1 \cdot 10^7$ zmiennych. Przy małej liczbie zebranych próbek (236 zbadanych pacjentów) taka liczba cech utrudnia zbudowanie klasyfikatora. Dlatego też z uzyskanych obrazów zostały wyekstrahowane cechy opisujące budowę mózgu mniejszą liczbą parametrów. Do tego celu zostało użyte oprogramowanie FreeSurfer[1] w wersji 5.1.0

[1]Pod adresem `http://surfer.nmr.mgh.harvard.edu/` dostępne jest bezpłatnie oprogramowanie FreeSurfer.

oraz automatyczna standardowa procedura przetwarzania [110]. Wykorzystano następujące kroki w procesie obróbki danych:

- normalizacja intensywności obrazu,
- usunięcie obrazu czaszki,
- segmentacja istoty białej, szarej oraz płynu mózgowo-rdzeniowego,
- dopasowanie rekonstruowanej struktury mózgu do szablonu za pomocą nieliniowych przekształceń [85] oraz rejestracja atlasu i map sferycznych z użyciem szablonu.

Użyta procedura przygotowania cech została sprawdzona przy pomocy analizy histologicznej [114] oraz pomiarów manualnych [76], [25]. Dodatkowo, stwierdzono dobrą powtarzalność wartości cech obliczanych za pomocą oprogramowania FreeSurfer, przy użyciu skanerów od różnych producentów i ich różnych konfiguracji [132], [110]. W przedstawionej analize został użyty atlas Destrieux [27], [109], który wyróżnia w każdej półkuli mózgu 74 rozłączne obszary. Każdy z nich został opisany za pomocą następujących miar:

- objętość,
- powierzchnia,
- średnia grubość kory mózgowej,
- średnia krzywizna regionu,
- średnie pofałdowanie regionu.

Ponadto, wyznaczono powierzchnię istoty białej dla każdej półkuli, co w rezultacie stworzyło wektor składający się z 742 cech. Krok ten zmniejszył początkowy wymiar próbki ponad $1,5 \cdot 10^4$ razy.

6.4 Opis metod

6.4.1 Usunięcie czynników zakłócających

W analizowanych danych dostępne są atrybuty opisujące cechy charakterystyczne dla pacjenta oraz aparatury użytej w badaniu - czynniki te mogą potencjalnie zakłócać analizę. Przedstawić je można za pomocą następujących parametrów:

- kraj, w którym wykonano badanie: $\mathbf{X}_{country}$,
- płeć, wiek, objętość mózgu dziecka - oznaczonych kolejno: $\mathbf{X}_{sex}$, $\mathbf{X}_{age}$, $\mathbf{X}_{TIV}$,
- natężenie pola magnetycznego skanera: $\mathbf{X}_{field}$,
- konfiguracja skanera, przedstawiająca rodzaj użytej w badaniu cewki w połączeniu z zastosowanym natężeniem pola magnetycznego - zakodowana binarnie za pomocą zmiennych: $\{\mathbf{X}_{3T.12}, \mathbf{X}_{3T.32}, \mathbf{X}_{3T.BC}, \mathbf{X}_{15T.32}, \mathbf{X}_{15T.BC}\}$.

Niektóre z wyżej wymienionych czynników nie mają wpływu na analizowana dane lub są redundantne. Dlatego, by wybrać minimalny zbiór parametrów zakłócających $\mathbf{S}_{CFC}$ zastosowano metodę opisaną w rozdziale 3.3. Metoda ta sprawdza zachłannie czy w analizowanych danych występuje atrybut, który zależy od czynników zakłócających, wybiera najbardziej zakłócający czynnik, dodając go do szukanego zbioru $\mathbf{S}_{CFC}$, czyści dane korzystając z wyznaczonego zbioru i powtarza przeszukiwanie, dotąd aż nie występuje żaden czynnik zakłócający. Za pomocą powyższej metody jako minimalny zbiór parametrów zakłócających został wybrany zbiór $\mathbf{S}_{CFC} = \{\mathbf{X}_{field}, \mathbf{X}_{15T.BC}, \mathbf{X}_{3T.12}, \mathbf{X}_{3T.BC}, \mathbf{X}_{sex}, \mathbf{X}_{TIV}\}$. Na rysunku 6.2 przedstawiono dwie przykładowe cechy, które zależały od czynników zakłócających oraz ich wartości po usunięciu wpływu czynników zakłócających.

6.4.2 System do selekcji cech i klasyfikacji

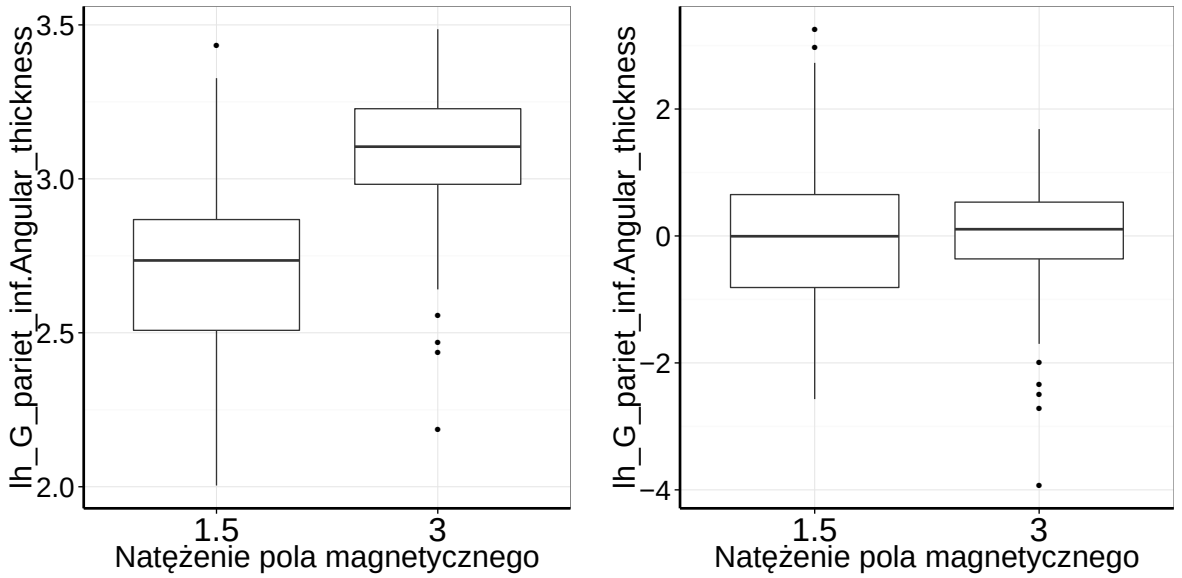
Do selekcji najważniejszych cech zostały użyte następujące algorytmy:

- selekcja za pomocą t-testu (rozdział 2.1.2),
- selekcja za pomocą przyrostu informacji (IG) (rozdział 2.1.3),
- selekcja za pomocą Lasu Losowego (LL) (rozdział 2.2.1).

Algorytmy te zostały wybrane, ponieważ są stosunkowo szybkie, zapewniają dobrą skuteczność selekcji oraz stabilność [53], [31], [13]. Wybrane algorytmy selekcji potrafią uporządkować cechy od najważniejszej do najmniej znaczącej. Niestety, same nie potrafią wskazać, jaka liczba cech powinna być wybrana aby klasyfikator działał najlepiej. Dlatego też wybrane algorytmy selekcji muszą być rozpatrywane w połączeniu z klasyfikatorem, który będzie użyty to wybrania liczby cech. Do klasyfikacji zostały użyte algorytmy:

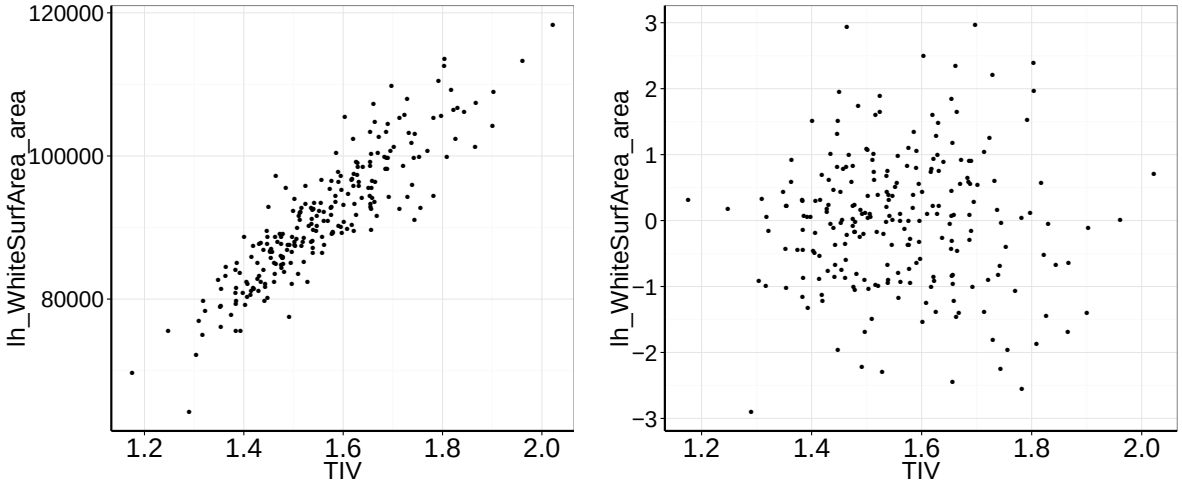
(a) Przed korekcją (p-wartość $< 0{,}001$)

(b) Po korekcji (p-wartość ≈ 1)



(c) Przed korekcją (korelacja $= 0{,}9$)

(d) Po korekcji (korelacja ≈ 0)



Rysunek 6.2: Przykład wybranych cech, które zależały od czynników zakłócających i ich wartości po korekcji (dane po korekcji zostały przeskalowane).

- Regresji Logistycznej (RL) (rozdział 1.1.1),
- Maszyny Wektorów Nośnych (SVM) (rozdział 1.1.4), parametr $C = 1$,
- Las Losowy (LL) (rozdział 1.1.3), liczba drzew wynosi 100.

Do oceny klasyfikacji użyta została kroswalidacja krzyżowa (rozdział 1.1.6) w dwóch wariantach: *leave-one-out* (LOO) oraz 10-krotna walidacja (*10-fold* CV). Dla każdego klasyfikatora zostały wyznaczone krzywe ROC, pole pod krzywymi ROC (AUC) oraz dokładność klasyfikacji (rozdział 1.1.6). W każdej iteracji kroswalidacji obliczane były współczynniki potrzebne do usunięcia czynników zakłócających tylko na podstawie zbioru uczącego. Po-

dobnie, w każdej iteracji kroswalidacji proces selekcji cech był wykonywany tylko na części uczącej zbioru. Do wyznaczenia liczby cech używany był klasyfikator oraz jego skuteczność klasyfikacji, wyznaczana za pomocą metryki *LogLoss*. Skuteczność klasyfikacji była sprawdzana na różnej liczbie cech, zaczynając od najważniejszej, i dokładając po jednej cesze do sprawdzanego zbioru według wyznaczonej ważności [32]. Klasyfikator wybierający liczbę cech oceniany był na zbiorze uczącym przy pomocy drugiej (zagnieżdżonej) pętli *leave-one-out*. Po wyznaczeniu zbioru najlepszych cech dla każdej metody wyznaczony został wskaźnik stabilności selekcji za pomocą indeksu Jaccarda (rozdział 2.5) oraz częstość wyboru cech w iteracjach sprawdzianu krzyżowego.

6.5 Wyniki

Wyniki klasyfikacji uzyskane dla różnych kombinacji metod selekcji oraz klasyfikatorów przedstawiono w tabeli 6.2. Skuteczność została policzona z wykorzystaniem walidacji typu LOO na danych przed i po korekcie czynników zakłócających. Ponieważ wyniki uzyskane przez klasyfikatory są niskie (poniżej 0,7), dla każdej metody sprawdzono z jakim prawdopodobieństwem przypominają klasyfikator losowy za pomocą testu permutacyjnego, w którym wykonano 100 powtórzeń (rozdział 1.1.6). Przede wszystkim należy zauważyć, iż po usunięciu czynników zakłócających klasyfikacja ma wyższą skuteczność. Dla danych bez korekty czynników zakłócających tylko dwa systemy klasyfikacji są istotnie różne od klasyfikatora losowego, natomiast dla danych po korekcie aż 7 systemów klasyfikacji znacząco różni się od systemu z losową klasyfikacją. Warto również zauważyć, że systemy klasyfikacji z selekcją cech uzyskały lepszą dokładność klasyfikacji niż system działający na wszystkich atrybutach.

Pośród systemów działających na danych z usuniętymi czynnikami zakłócającymi, wysoką skuteczność klasyfikacji uzyskały następujące metody:

- Uzysku Informacji do selekcji cech i Lasu Losowego jako klasyfikatora (AUC=0,68; dokładność=0,65),
- t-testu w połączeniu z Regresją Logistyczną (AUC=0,65; dokładność=0,64),
- selekcji i klasyfikacji za pomocą Lasu Losowego (AUC=0,67; dokładność=0,67).

Jako najlepsze systemy zostały wskazane trzy podejścia, ponieważ nie można wskazać

wśród nich jednego jednoznacznie lepszego od pozostałych tylko na podstawie skuteczności klasyfikacji. Dla wybranych najlepszych metod przedstawiono na rysunku 6.4 uzyskane rozkłady skuteczności działania klasyfikatorów na zbiorach z przestawionymi losowo klasami dla testu permutacji.

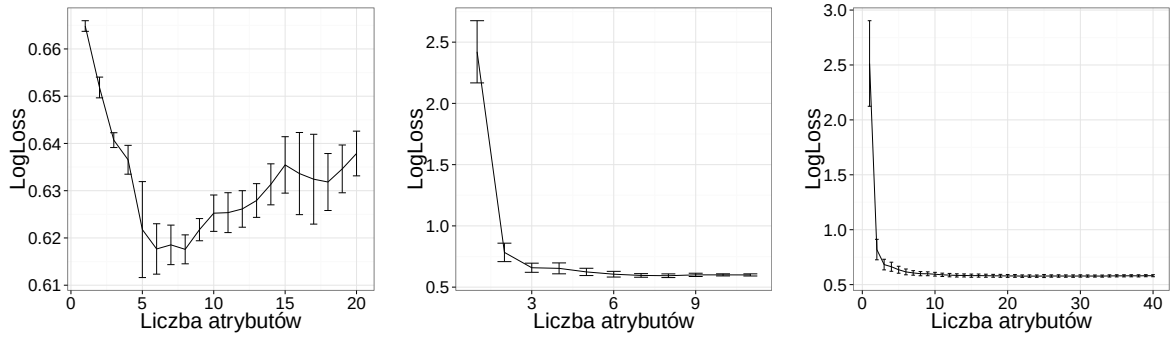
Tabela 6.2: Wyniki klasyfikacji dla różnej konfiguracji metod selekcji i klasyfikatorów, wyznaczone za pomocą walidacji LOO, policzone na danych przed i po korekcie czynników zakłócających. Do wyznaczenia p-wartości został użyty test permutacji powtórzony 100-krotnie. Stabilność została wyznaczona jako indeks Jaccarda pomiędzy selekcjami ze wszystkich iteracji sprawdzianu krzyżowego.

Metoda selekcji	Klasyfikator	AUC (p-wartość)	Dokładność (p-wartość)	Stabilność
Wyniki bez korekty czynników zakłócających				
Wszystkie cechy	Regresja Logistyczna	0,47 (0,82)	0,47 (0,13)	-
Wszystkie cechy	Las Losowy	0,60 (0,01)	0,58 (0,1)	-
Wszystkie cechy	SVM	0,52 (0,25)	0,56 (0,32)	-
t-test	Regresja Logistyczna	0,48 (0,79)	0,55 (0,83)	0,24
t-test	Las Losowy	0,59 (0,02)	0,61 (0,02)	0,03
t-test	SVM	0,51 (0,39)	0,57 (0,20)	0,10
Uzysk Informacji	Regresja Logistyczna	0,62 (0,01)	0,63 (0,01)	0,26
Uzysk Informacji	Las Losowy	0,52 (0,28)	0,56 (0,34)	0,14
Uzysk Informacji	SVM	0,48 (0,68)	0,57 (0,27)	0,19
Selekcja LL	Regresja Logistyczna	0,52 (0,31)	0,56 (0,61)	0,19
Selekcja LL	Las Losowy	0,51 (0,35)	0,56 (0,49)	0,05
Selekcja LL	SVM	0,50 (0,54)	0,56 (0,62)	0,11
Wyniki z korektą czynników zakłócających				
Wszystkie cechy	Regresja Logistyczna	0,51 (0,41)	0,47 (0,86)	-
Wszystkie cechy	Las Losowy	0,64 (0,01)	0,65 (0,01)	-
Wszystkie cechy	SVM	0,52 (0,34)	0,56 (0,13)	-
t-test	Regresja Logistyczna	0,65 (0,01)	0,64 (0,01)	0,93
t-test	Las Losowy	0,62 (0,01)	0,62 (0,01)	0,91
t-test	SVM	0,60 (0,02)	0,62 (0,01)	0,95
Uzysk Informacji	Regresja Logistyczna	0,39 (1,00)	0,52 (0,67)	0,78
Uzysk Informacji	Las Losowy	0,68 (0,01)	0,65 (0,01)	0,86
Uzysk Informacji	SVM	0,60 (0,01)	0,61 (0,01)	0,88
Selekcja LL	Regresja Logistyczna	0,53 (0,13)	0,57 (0,23)	0,25
Selekcja LL	Las Losowy	0,67 (0,01)	0,67 (0,01)	0,43
Selekcja LL	SVM	0,63 (0,01)	0,65 (0,01)	0,48

Na rysunku 6.3 przedstawiono wynik z procesu wybierania liczby cech dla najlepszych metod. Warto nadmienić, iż w każdej iteracji sprawdzianu krzyżowego mogła zostać wybrana inna liczba cech. Na podstawie wykresów można stwierdzić, że najmniejsza liczba cech była wybierana najczęściej dla metody t-test w połączeniu z Regresją Logistyczną - 6 atrybutów. Mała liczba atrybutów była też wskazywana dla metody Przy-

rostu Informacji połączonej z Lasem Losowym - 7. Natomiast selekcja i klasyfikacja za pomocą Lasu Losowego wymaga najwięcej atrybutów, dopiero od 15 wybranych atrybutów wartość funkcji LogLoss przestaje znacząco maleć. Co więcej, podejście to uzyskuje duże wartości funkcji kosztu dla małej liczby atrybutów.

(a) ttest, Regresja Logistyczna (b) IG, Las Losowy (c) Selekcja LL, Las Losowy

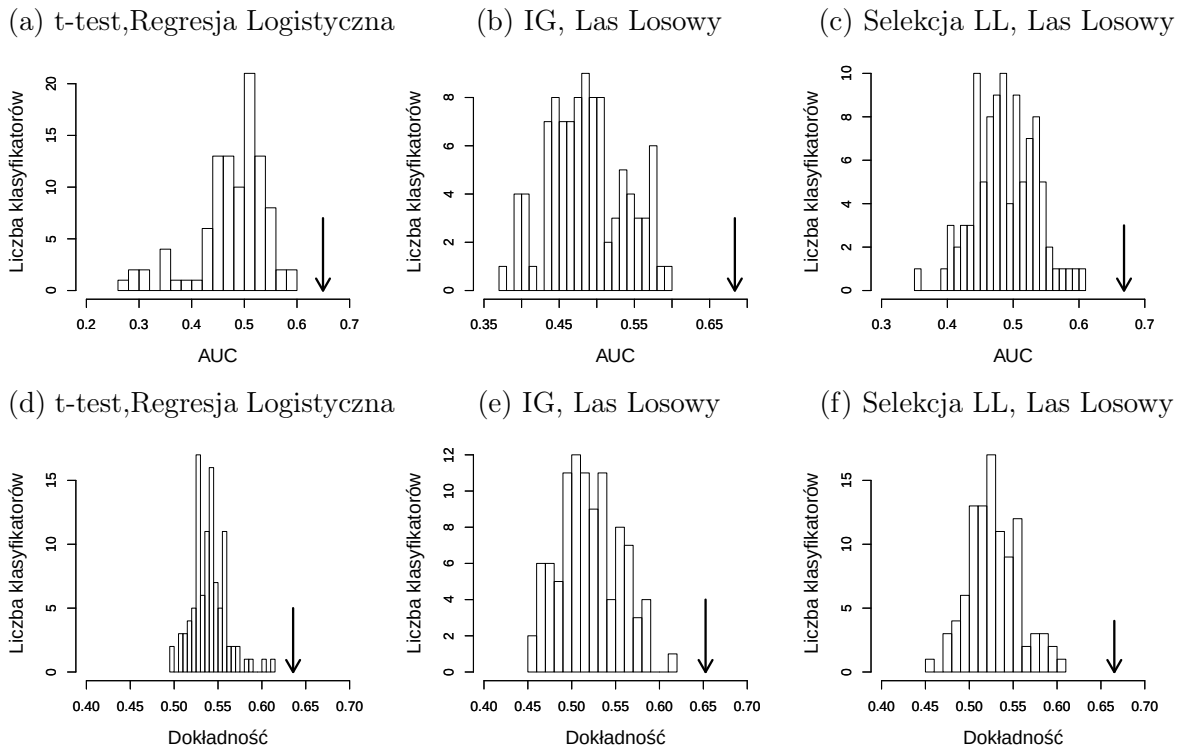


Rysunek 6.3: Wartości LogLoss w zależności od liczby wybranych atrybutów wyznaczone na częściach uczących zbioru w trakcie walidacji typu *leave-one-out* dla najlepszych metod.

Dodatkowo na rysunku 6.5 porównano działanie klasyfikacji za pomocą najlepszych metod z klasyfikatorami, które zostały zbudowane na podstawie losowo wybranych cech - dla każdej metody (selekcji+klasyfikacji) zostały wylosowane atrybuty w takiej samej liczbie jaka została wskazana w selekcji. Klasyfikatory działające na losowo wybranych cechach osiągają niższą skuteczność klasyfikacji niż klasyfikator wykorzystujący wyselekcjonowane cechy. Potwierdza to, że cechy wybrane w procesie selekcji niosą informację o dysleksji rozwojowej u dzieci oraz że system klasyfikacji zbudowany z ich wykorzystaniem różni się od systemu działającego na losowo wybranych cechach.

Cała procedura sprawdzająca metody selekcji i klasyfikacji dla danych po usunięciu czynników zakłócających została dodatkowo przeprowadzona przy 10-krotnej kroswalidacji krzyżowej powtórzonej 100 razy. Uzyskane wyniki przedstawiono w tabeli 6.3. Dla uzyskanych wyników został policzony przedział ufności ($\alpha = 0,05$), ponieważ przy małej próbce danych sam średni wynik AUC może być mylący i wymaga sprawdzenia czy dolny przedział granicy ufności nie znajduje się poniżej wartości 0,5 [51]. Tak jak w przypadku wyników dla LOO, najlepsze rezultaty zostały otrzymane za pomocą metod:

- t-test, Regresja Logistyczna (AUC=0,66, dokładność=0,65),
- Uzysk Informacji i Las Losowy (AUC=0,66, dokładność=0,65),



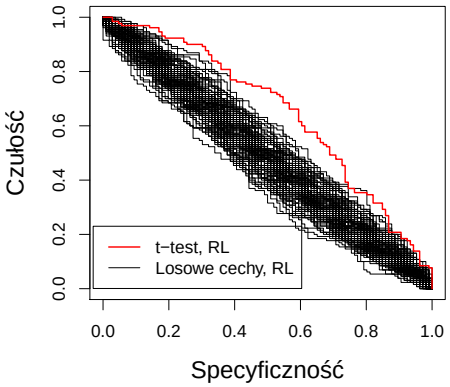
Rysunek 6.4: Wynik testu permutacji wykonanego dla kroswalidacji typu LOO. Test wyznaczony na podstawie 100 powtórzeń. Na wykresach strzałką zaznaczono wynik klasyfikacji otrzymany na zbiorze bez permutacji.

- Selekcja LL, Las Losowy (AUC=0,65, dokładność=0,64).

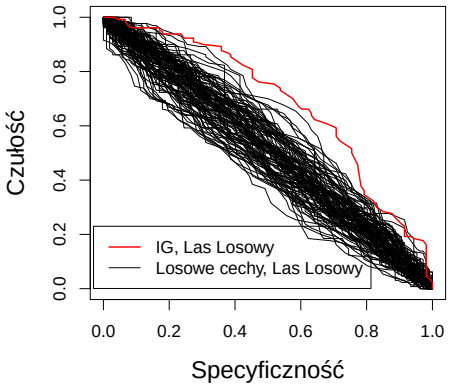
W porównaniu z wynikami uzyskanymi dla LOO, skuteczność klasyfikacji dla wszystkich metod jest niższa dla 10-krotnej kroswalidacji. Dzieje się tak, ponieważ w LOO więcej próbek jest używanych do nauki w każdej iteracji walidacyjnej, przez co klasyfikator może być dokładniejszy. Pomimo spadku skuteczności w stosunku do wyników z LOO, wszystkie trzy najlepsze metody mają dolną wartość przedziału ufności powyżej skuteczności klasyfikatora losowego (AUC=0,5).

W obu typach kroswalidacji została sprawdzona stabilność wybranych cech - wyniki są zamieszczone w tabelach 6.2, 6.3. Warto zwrócić uwagę na bardzo niskie wartości stabilności dla wyników uzyskanych bez usunięcia czynników zakłócających - najwyższa osiągnięta stabilność to 0,24. Natomiast w przypadku danych po korekcie czynników zakłócających, w obu typach kroswalidacji widać wyraźną przewagę w stabilności selekcji dla metody t-test, co może być zaskakujące, ponieważ t-test uważany jest za prosty i mało wyszukany algorytm selekcji cech. Podobne zachowanie zostało zaobserwowane w artykule [53], w przypadku analizy danych wielowymiarowych z mikromacierzy. Najmniej stabilny okazał się algorytm selekcji za pomocą Lasu Losowego. Niska stabilność

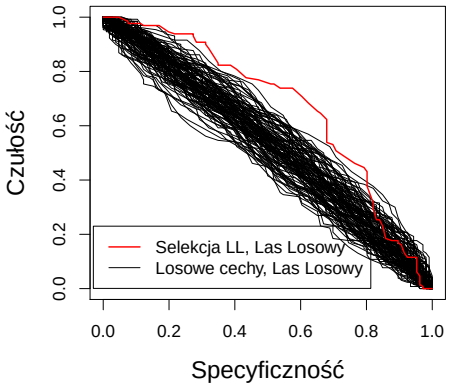
(a) t-test,Regresja Logistyczna



(b) IG, Las Losowy



(c) Selekcja LL, Las Losowy



Rysunek 6.5: Porównanie klasyfikatorów działających na cechach wybranych losowo z wybranymi za pomocą algorytmu selekcji. Proces losowania cech i budowy klasyfikatora został powtórzony 100 razy.

w tym przypadku może być spowodowana selekcją większej liczby cech za pomocą tego algorytmu niż w przypadku algorytmów selekcji t-test i Uzysku Informacji. Warto zwrócić również uwagę, iż wartości stabilności są niższe gdy obliczane są z wykorzystaniem 10-krotnej kroswalidacji krzyżowej w porównaniu z wartościami uzyskanymi za pomocą walidacji typu LOO. Zgodnie ze wskazówkami z artykułu [70], gdy kilka metod uzyska podobną skuteczność klasyfikacji, jako najlepszą należy wskazać tę, która osiąga najwyższą stabilność. W przypadku klasyfikacji dzieci z dysleksją rozwojową jako metodę, która wykazuje się dużą skutecznością klasyfikacji i stabilnością należy wskazać selekcję za pomocą t-test i klasyfikację Regresją Logistyczną.

Prócz stabilności, wyznaczona została również częstość selekcji cech w iteracjach sprawdzianu krzyżowego dla obu typów kroswalidacji. Zestawienie najczęściej wybieranych cech według najlepszych metod przedstawiono w tabeli 6.4. Należy zauważyć, że najczęściej

Tabela 6.3: Wyniki skuteczności klasyfikacji oraz stabilności dla różnej konfiguracji metod selekcji i klasyfikatorów wyznaczone za pomocą 10-krotnej kroswalidacji krzyżowej powtórzonej 100 razy. Wyniki zostały policzone na danych po korekcie czynników zakłócających.

Metoda selekcji i klasyfikator	AUC (p-wartość)	Dokładność (p-wartość)	Stabilność
Wszystkie cechy, RL	0,51 [0,42; 0,56] (0,43)	0,51 [0,42; 0,56] (0,40)	-
Wszystkie cechy, LL	0,62 [0,58; 0,66] (0,01)	0,62 [0,59; 0,66] (0,01)	-
Wszystkie cechy, SVM	0,47 [0,44; 0,51] (0,81)	0,54 [0,51; 0,56] (0,57)	-
t-test, RL	0,66 [0,63; 0,69] (0,01)	0,65 [0,61; 0,67] (0,01)	0,71
t-test, LL	0,61 [0,56; 0,66] (0,01)	0,61 [0,58; 0,64] (0,01)	0,68
t-test, SVM	0,61 [0,57; 0,66] (0,02)	0,62 [0,58; 0,65] (0,09)	0,70
Uzysk Informacji, RL	0,56 [0,50; 0,62] (0,05)	0,58 [0,56; 0,63] (0,14)	0,25
Uzysk Informacji, LL	0,66 [0,61; 0,70] (0,01)	0,65 [0,60; 0,69] (0,01)	0,43
Uzysk Informacji, SVM	0,63 [0,58; 0,68] (0,02)	0,62 [0,58; 0,67] (0,02)	0,46
Selekcja LL, RL	0,57 [0,51; 0,63] (0,03)	0,59 [0,56; 0,62] (0,06)	0,18
Selekcja LL, LL	0,65 [0,60; 0,69] (0,01)	0,64 [0,61; 0,68] (0,01)	0,34
Selekcja LL, SVM	0,61 [0,54; 0,67] (0,07)	0,61 [0,58; 0,65] (0,04)	0,34

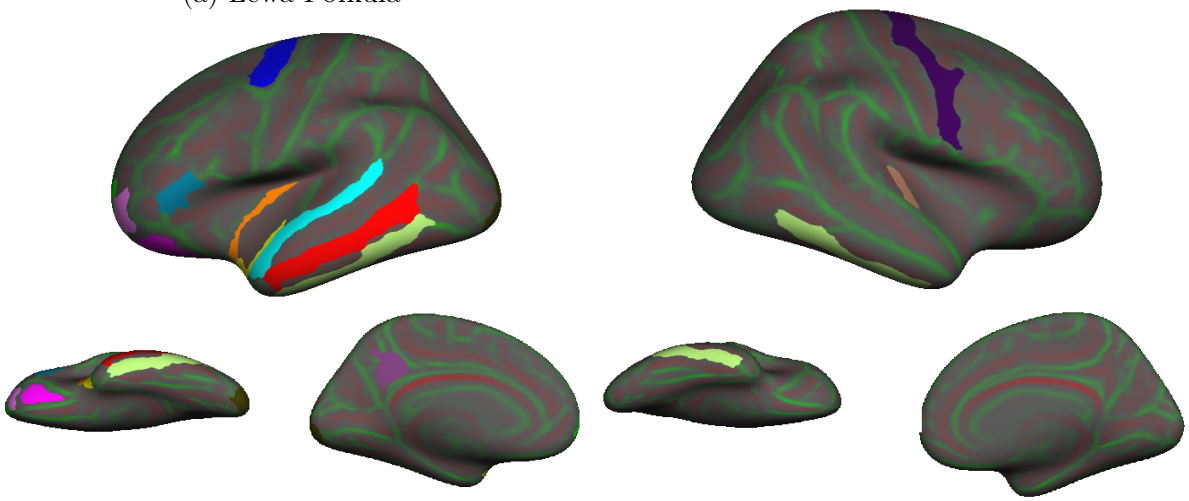
były wybierane regiony znajdujące się w lewej półkuli. Cechami opisującymi wybrane regiony przeważnie były: średnie pofałdowanie i krzywizna. Dodatkowo cechy, które były wybierane w walidacji typu LOO były również wybierane w 10-krotnej walidacji, pomimo iż zbiór uczący się zmniejszył. Na rysunku 6.6 przedstawiono lokalizację wskazanych regionów na mózgu oraz wykresy pudełkowe dla najczęściej wybieranych cech.

Skuteczność klasyfikacji została również wyznaczona dla zbioru podzielonego ze względu na płeć - podczas uczenia klasyfikatora wykorzystywane były próbki należące do obu płci, a testowanie odbywało się na próbkach pochodzących wyłącznie z wybranej płci. Wyniki dla trzech najlepszych metod zostały zaprezentowane w tabeli 6.5. Metoda łącząca t-test i Regresję Logistyczną z większą dokładnością klasyfikuje chłopców niż dziewczęta. Pozostałe metody działają lepiej w przypadku dziewcząt. Różnica w działaniu klasyfikatorów może wynikać z selekcji różnych zestawów cech. Dla poprawnej klasyfikacji chłopców to cechy wybrane za pomocą t-testu są ważniejsze niż cechy wybrane innymi metodami. Cechy, które zostały wybrane wyłącznie przez algorytm selekcji t-test to:

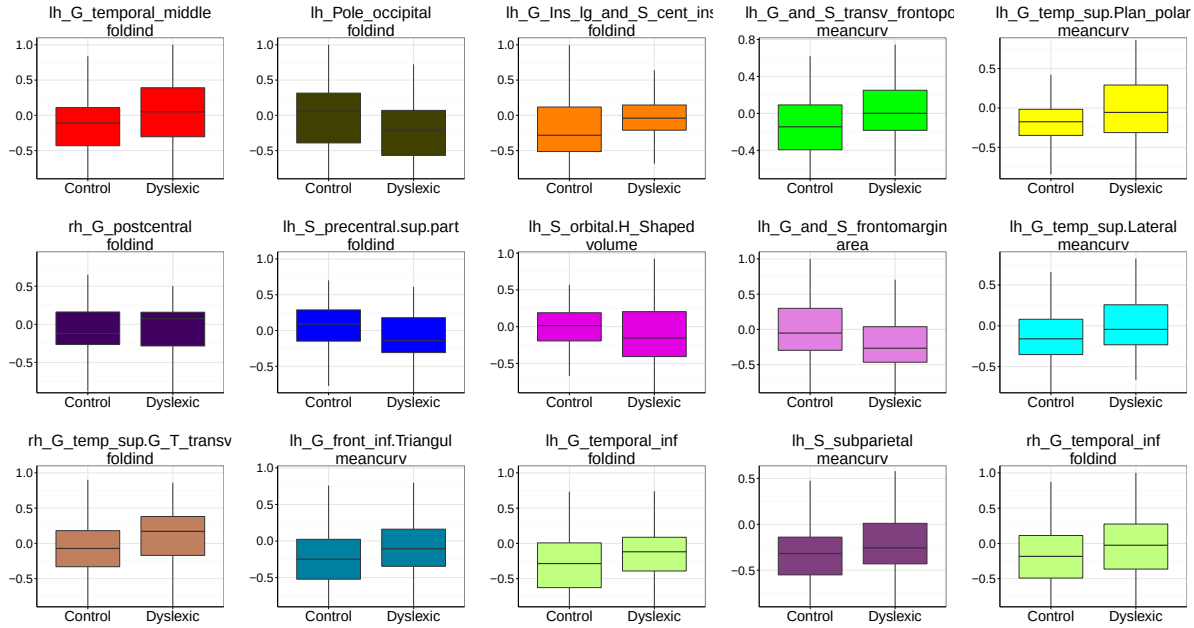
- powierzchnia regionu *fronto-marginal gyrus and sulcus* w lewej półkuli,
- średnie zakrzywienie regionu *lateral aspect of the superior temporal gyrus* w lewej półkuli,

(a) Lewa Półkula

(b) Prawa Półkula



(c) Najczęściej wybierane cechy



Rysunek 6.6: Wizualizacja najczęściej wybieranych cech. Kolor wykresów odpowiada obszarom zaznaczonym na lewej (a) oraz prawej półkuli mózgu (b).

Tabela 6.4: Zestawienie najczęściej wybieranych cech dla metod z najwyższą skutecznością. Częstość wybierania w iteracjach kroswalidacji podano w procentach dla obu typów walidacji, wyniki dla 10-krotnej walidacji podane są w nawiasach obok częstości dla LOO. Cecha uznana za najczęściej wybieraną pojawiała się co najmniej w połowie iteracji w walidacji krzyżowej typu LOO.

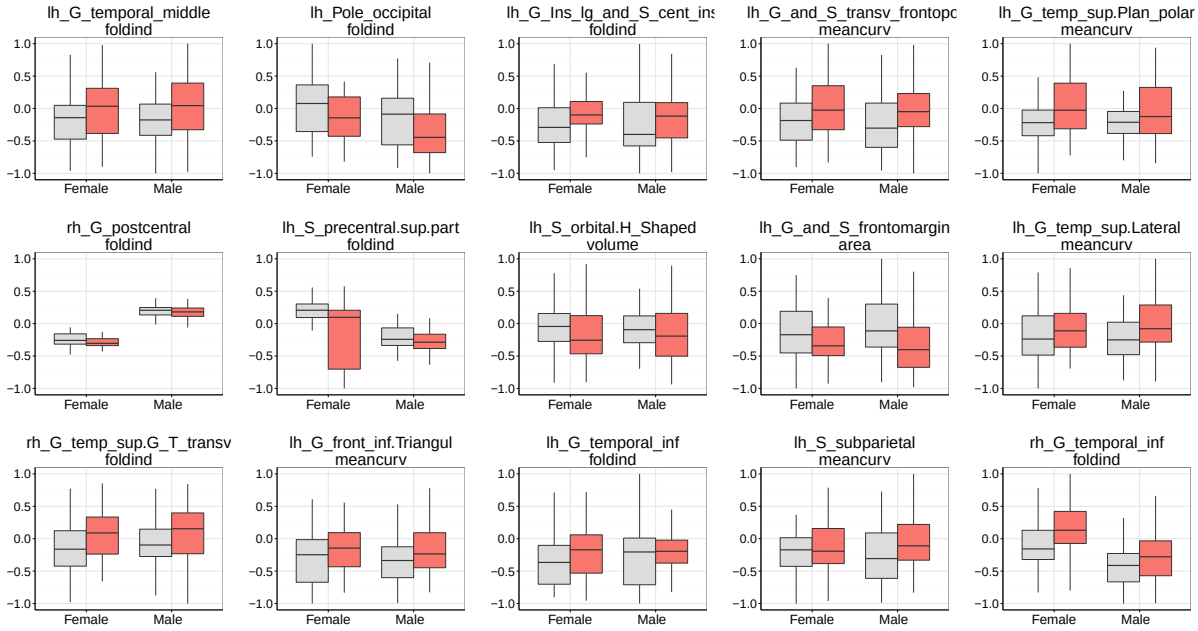
Region	Półkula	Cecha	t-test, Regresja Logistyczna (10-krotna KW)	Uzysk Informacji, Las Losowy (10-krotna KW)	Selekcja LL, Las Losowy (10-krotna KW)
Middle temporal gyrus	Lewa	indeks pofałdowania	100 (98,1)%	100 (89,2)%	95,3 (84) %
Planum polare of the superior temporal gyrus	Lewa	średnia krzywizna	100 (99,9)%	99,2 (72)%	78,8 (72)%
Transverse frontopolar gyri and sulci	Lewa	średnia krzywizna	100 (100)%	99,2 (53,4)%	94,9 (86,9)%
Fronto-marginal gyrus (of Wernicke) and sulcus	Lewa	powierzchnia	100 (100)%	0 (7)%	63,6 (67)%
Occipital pole	Lewa	indeks pofałdowania	0 (0) %	99,6 (79,9)%	98,7 (84,7)%
Orbital sulci (H-shaped sulci)	Lewa	objętość	0,4 (32,5)%	73,3 (59,1)%	77,1 (71,2)%
Subparietal sulcus	Lewa	średnia krzywizna	44,1 (98,3)%	0 (3,3)%	52,5 (44,8)%
Long insular gyrus and central sulcus of the insula	Lewa	indeks pofałdowania	0 (0) %	99,2 (73,1)%	89,3 (71,5)%
Lateral aspect of the superior temporal gyrus	Lewa	średnia krzywizna	100 (100)%	0 (1,7)%	32,6 (36,2)%
Superior part of the precentral sulcus	Lewa	indeks pofałdowania	0 (0) %	81,4 (59,5) %	94,1 (78,9)%
Postcentral gyrus	Prawa	indeks pofałdowania	0 (0)%	97,9 (81,6)%	34,3 (22,8)%
Anterior transverse temporal gyrus (of Heschl)	Prawa	indeks pofałdowania	0 (0,1)%	17,4 (26,3)%	64,4 (56,1)%
Triangular part of the inferior frontal gyrus	Lewa	średnia krzywizna	0 (8,3)%	26,4 (26,4)%	56,4 (49,5)%
Inferior temporal gyrus	Prawa	indeks pofałdowania	0 (5,2)%	0 (10,8)%	51,7 (51,8)%

Przy klasyfikacji dziewcząt większe znaczenie mają cechy (nie wybrane przez t-test), takie jak indeks pofałdowania dla regionów z lewej półkuli: *insula*, *occipital pole* i prawej: *post-central gyrus*. Na rysunku 6.7 przedstawiono wykresy pudełkowe najczęściej wybieranych cech z uwzględnieniem podziału na płeć oraz klasę.

Tabela 6.5: Skuteczność klasyfikacji dla najlepszych metod, wyznaczona z podziałem na płeć pacjentów. Wyniki przedstawiono dla obu typów kroswalidacji. Dodatkowo dla wyników z 10-krotnej kroswalidacji krzyżowej załączono przedział ufności.

Metoda	Podział danych	AUC (p-wartość)	dokładność (p-wartość)
		leave-one-out	
t-test,	Wszyscy	0,65 (0,01)	0,64 (0,01)
Regresja	Chłopcy	0,71 (0,01)	0,70 (0,01)
Logistyczna	Dziewczęta	0,59 (0,04)	0,59 (0,23)
Uzysk Informacji,	Wszyscy	0,68 (0,01)	0,65 (0,01)
Las Losowy	Chłopcy	0,62 (0,03)	0,64 (0,01)
	Dziewczęta	0,74 (0,01)	0,70 (0,01)
Selekcja LL,	Wszyscy	0,67 (0,01)	0,67 (0,01)
Las Losowy	Chłopcy	0,62 (0,03)	0,66 (0,02)
	Dziewczęta	0,71 (0,01)	0,69 (0,02)
		10-krotna kroswalidacja krzyżowa	
t-test,	Wszyscy	0,65 [0,63; 0,69] (0,01)	0,65 [0,61; 0,67] (0,01)
Regresja	Chłopcy	0,70 [0,65; 0,74] (0,01)	0,70[0,66; 0,74] (0,01)
Logistyczna	Dziewczęta	0,61 [0,55; 0,66] (0,05)	0,61[0,57; 0,66] (0,12)
Uzysk Informacji,	Wszyscy	0,66 [0,61; 0,70] (0,01)	0,65 [0,60; 0,69] (0,01)
Las Losowy	Chłopcy	0,62 [0,54; 0,67] (0,01)	0,65 [0,60; 0,70] (0,02)
	Dziewczęta	0,70 [0,64; 0,75] (0,01)	0,68 [0,63; 0,74] (0,01)
Selekcja LL,	Wszyscy	0,65 [0,60; 0,69] (0,01)	0,64 [0,61; 0,68] (0,01)
Las Losowy	Chłopcy	0,62 [0,56; 0,68] (0,01)	0,65 [0,62; 0,70] (0,01)
	Dziewczęta	0,68 [0,61; 0,75] (0,01)	0,67 [0,61; 0,73] (0,01)

Korzystając z faktu, iż wytrenowanych zostało kilka systemów klasyfikacji, możliwe było zbudowanie klasyfikatora zbiorowego złożonego z 3 pojedynczych klasyfikatorów działających na różnych zbiorach atrybutów. Końcowa odpowiedź systemu to średnia wartość odpowiedzi ze wszystkich klasyfikatorów. Skuteczność klasyfikatora zbiorowego została sprawdzona dla 10-krotnej kroswalidacji oraz LOO - wyniki przedstawiono w tabeli 6.6. Klasyfikator zbiorowy w każdym rodzaju klasyfikacji jest lepszy od pojedynczego klasyfikatora użytego do jego budowy, aczkolwiek różnica między pojedynczym klasyfikatorem a zbiorowym nie jest duża.



Rysunek 6.7: Wizualizacja najczęściej wybieranych cech z uwzględnieniem podziału na płeć oraz klasę. Kolorem szarym zaznaczono pacjentów zdrowych, natomiast kolorem czerwonym pacjentów z zaburzeniem dysleksji rozwojowej. Na wykresach pudełkowych wartości odstające zostały pominięte.

Tabela 6.6: Wyniki klasyfikacji dla klasyfikatora zbiorowego wyznaczone dla różnych typów kroswalidacji. Dodatkowo dla referencji w tabeli podano skuteczność działania pojedynczego klasyfikatora - metody t-test w połączeniu z Regresją Logistyczną.

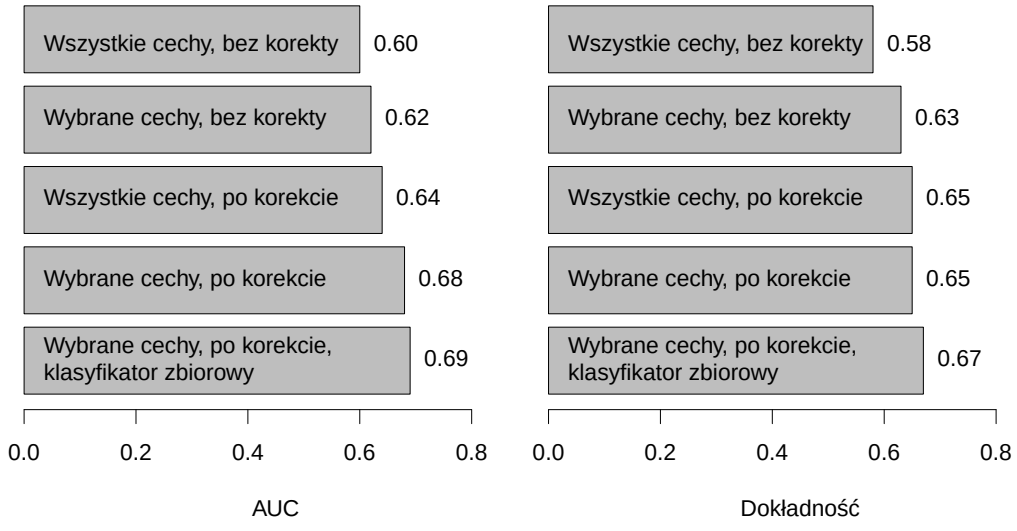
Rodzaj kroswalidacji	Metoda selekcji i klasyfikator	AUC	Dokładność
LOO	t-test, RL	0,65	0,64
LOO	Zbiorowy	0,69	0,67
10-krotna	t-test, RL	0,66 [0,63; 0,69]	0,65 [0,61; 0,67]
10-krotna	Zbiorowy	0,68 [0,63; 0,72]	0,66 [0,61; 0,70]

6.6 Dyskusja

Celem przeprowadzonych analiz było uzyskanie informacji o różnicach anatomicznych jakie występują pomiędzy dziećmi zdrowymi a dziećmi z dysleksją rozwojową. Analizy zostały przeprowadzone na podstawie danych zebranych z wykorzystaniem strukturalnego rezonansu magnetycznego z kilku różnych ośrodków badawczych. Do każdego badania został skonstruowany wektor cech opisujący obszary w mózgu. Podczas analizy usunięto czynniki zakłócające, wykonano selekcję cech oraz klasyfikację. Skuteczność systemu klasyfikacji została zbadana na całej próbce danych oraz z uwzględnieniem podziału na płeć. Zebrane wyniki umożliwiły również sprawdzenie wpływu usuwania czynników zakłócają-

cych oraz selekcji cech na dokładność działania klasyfikacji.

W zbudowanym systemie klasyfikacji przed zastosowaniem klasyfikatora przeprowadzone zostały dwa dodatkowe kroki przygotowujące dane: usunięcie czynników zakłócających oraz selekcja najbardziej istotnych cech. Na rysunku 6.8 przedstawiono skuteczność klasyfikacji w zależności od użycia kroków przygotowujących dane. W przypadku pojedynczego klasyfikatora pole pod krzywą ROC wzrasta z 0,6 do 0,68, a dokładność klasyfikacji wzrasta z 0,58 do 0,65 dzięki odpowiedniemu przygotowaniu danych. Uzyskane systemy klasyfikacji uzyskały niską skuteczność w porównaniu z wynikami klasyfikacji dla innych chorób neurologicznych, takich jak choroba Alzheimera lub Huntingtona [69],[24]. Niska skuteczność osiągana w klasyfikacji może być spowodowana istnieniem wielu typów dysleksji, które mają różną genezę [66]. Dlatego też zaobserwowano dużą rozbieżność pomiędzy wynikami z analiz wykorzystujących jedną zmienną [80], [111],[67]. Jednakże wybrane najlepsze systemy klasyfikacji są istotnie różne od klasyfikatora losowego, dlatego też możliwe jest wskazanie anatomicznych różnic w strukturze mózgu pomiędzy dziećmi zdrowymi a dziećmi z dysleksją rozwojową.



Rysunek 6.8: Skuteczność klasyfikacji wyznaczona za pomocą kroswalidacji krzyżowej typu LOO w zależności od etapu przygotowania danych.

Pośród wybranych cech anatomicznych, które rozróżniają dysleksję rozwojową wśród dzieci, większość cech została wskazana w lewej półkuli - z 15 najczęściej wybieranych cech aż 12 było z lewej półkuli. Co więcej, pośród wskazanych cech 13 opisuje wybrany region za pomocą średniej krzywizny lub indeksu pofałdowania, a po jednym razie została wybra-

na cecha opisująca powierzchnię i objętość. W wybranych cechach opisujących pofałdowanie regionów większe pofałdowanie występuje u dzieci z dysleksją rozwojową. Regiony najczęściej wskazywane znajdują się w okolicy bruzdy Sylwiusza - szczelina ta oddziela miejsce złączenia płata czołowego, ciemieniowego i skroniowego; jest najwcześniej formującą się bruzdą, ponieważ zaczyna się pojawiać w 14. tygodniu życia płodowego. Uzyskane wyniki mogą być rozpatrywane jako pierwsze potwierdzenie *in-vivo* badań przeprowadzonych przez Galaburda [44], [45].

Prócz bardzo ważnego rezultatu biologicznego przeprowadzonych analiz, warto jeszcze zwrócić uwagę na zachowanie się klasyfikatorów przed i po selekcji cech oraz na stabilność algorytmów selekcji przed i po korekcie czynników zakłócających. W uczeniu maszynowym zazwyczaj im więcej danych dostępnych jest dla klasyfikatora, tym dokładniej można go wytrenować, aczkolwiek, w rozpatrywanym przypadku nie ma potwierdzenia tej reguły. Sprawdzane algorytmy klasyfikacji działają lepiej przy zmniejszonej liczbie atrybutów w zbiorze uczącym. Dzieje się tak, ponieważ w wejściowym zbiorze jest wiele atrybutów, które nie niosą żadnej informacji. Wśród testowanych algorytmów klasyfikacji z wykorzystaniem wszystkich dostępnych cech najlepiej działał klasyfikator typu Las Losowy. Ma on w sobie wewnętrzny mechanizm, który w trakcie budowy klasyfikatora wybiera cechy niosące najwięcej informacji. Drugą ciekawą obserwacją jest bardzo niska stabilność algorytmów selekcji przed korektą czynników zakłócających - wszystkie metody mają stabilność poniżej 0,24. Te same metody po korekcie czynników zakłócających osiągają stabilność nawet do 0,95. Bardzo niska stabilność selekcji przed korektą może być spowodowana wybieraniem cech, które nie mają znaczenia biologicznego a jedynie opisują czynnik zakłócający.

Rozdział 7

Analiza danych z detektora ciekłoargonowego

W niniejszym rozdziale wykorzystano metody konstrukcji cech do budowy klasyfikatorów służących do: segmentacji torów cząstek w obrazie i klasyfikacji neutrin elektronowych. Opisane systemy klasyfikacji używają obrazów pochodzących z detektora ciekłoargonowego. Dodatkowo wykorzystano proces selekcji cech do sprawdzenia istotności zaproponowanych cech.

7.1 Opis zagadnienia

Neutrina to jedne z fundamentalnych cząstek, których struktura wewnętrzna nie jest znana, a jednocześnie jedne z najmniej dotychczas poznanych. Ich istnienie zostało przewidziane teoretycznie przez Wolfganga Pauliego w 1930 r., a następnie potwierdzone eksperymentalnie przez Frederika Reinesa i Clyde'a Cowana w 1956 roku. Neutrino jest cząstką podstawową w modelu standardowym, o zerowym ładunku i niewielkiej, bliskiej zeru, masie spoczynkowej. Występują one w 3 zapachach: elektronowym, mionowym i taonowym. Istnieje jednakże hipoteza o istnieniu czwartego zapachu, nazywanego sterylnym [121].

Neutrina są bardzo trudne do detekcji, ponieważ nie wchodzą w oddziaływania silne ani elektromagnetyczne. Dlatego by poznać ich właściwości, budowane są duże detektory, pośród których ciekłoargonowy detektor z komorą projekcji czasowej (ang. *Liquid Argon Time Projection Chamber Detector* (LArTPC)), zaproponowany przez Carlo Rubbię w 1977 r. [115], posiada bardzo dobre właściwości obrazujące, pozwalające na obserwowa-

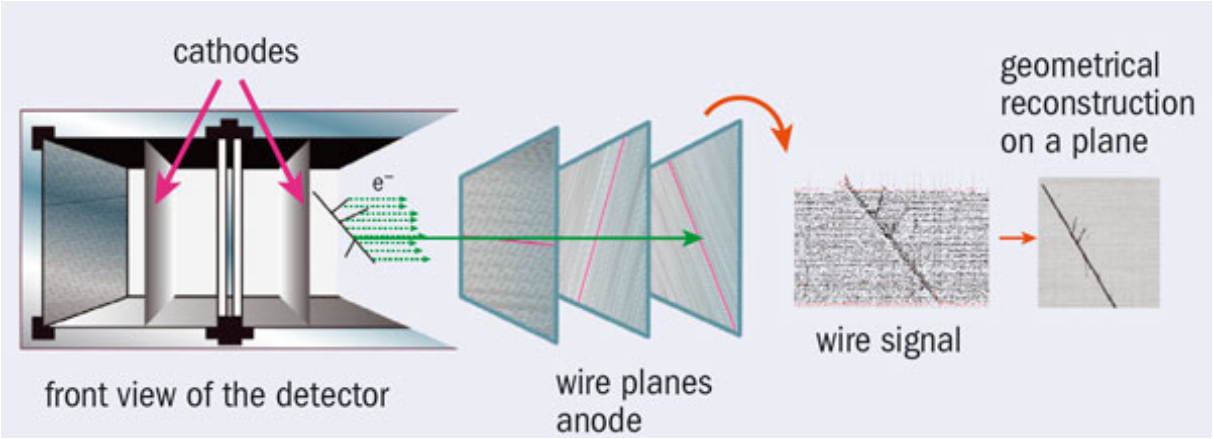
nie naładowanych cząstek. Technika obrazowania wykorzystywana w detektorze LArTPC stosowana jest w wielu eksperymentach na całych świecie: MicroBooNE [127], LAGUNA[4], ArgoNeuT [2], LBNE [126], ICARUS [124]. Pośród wymienionych eksperymentów detektor T-600 używany w eksperymencie ICARUS był największym zbudowanym dotychczas detektorem. Został on uruchomiony w 2010 r. i czynnie działał do 2014 r. we włoskim podziemnym laboratorium LNGS w Gran Sasso, gdzie operował na wiązce neutrin CNGS pochodzącej ze szwajcarskiego laboratorium CERN. Na rysunku 7.1 przedstawione jest zdjęcie detektora. W rozprawie użyto danych, które zostały wygenerowane przy użyciu parametrów detektora T-600 ICARUS. Obecnie używane lub planowane detektory typu LArTPC posiadają takie same lub bardzo przybliżone wartości parametrów.

Rysunek 7.1: Detektor ICARUS T-600 zlokalizowany w podziemnym laboratorium LNGS w Gran Sasso, Włochy. Źródło: http://physicsforme.com/2011/10/18/icarus-refutes-operas-superluminal-neutrinos/

Schemat działania detektora

Cząstka przelatująca przez detektor, w szczególności neutrino, może oddziaływać z nukleonami argonu. Dzięki dużej gęstości argonu w postaci ciekłej oraz jego dużej czystości możliwe jest uzyskanie wystarczającej liczby interakcji cząstek z detektorem do przeprowadzenia eksperymentu. W wyniku interakcji (zderzenia) przelatującej cząstki z detektorem powstają naładowane cząstki. Produkują one na swojej ścieżce światło scyntylacyjne oraz elektrony w wyniku jonizacji argonu. Światło to jest wykrywane przez fotopowielacze, które uruchamiają proces odczytu. Powstałe elektrony dryfują w kierunku anody w jed-

norodnym polu elektrycznym, wytworzonym pomiędzy ścianami detektora a umieszczoną w środku katodą. Prędkość dryfu, która wynosi 1,59 mm/μs, jest znacznie większa niż dyfuzja elektronów w argonie o dużej czystości (4,8 cm^2/s). Dlatego też elektrony mogą dryfować na makroskopowe dystanse. Anoda w detektorze składa się z trzech płaszczyzn drutów, które kolejno nazywają się: Induction1 (Ind1), Induction2 (Ind2), Collection (Coll). Sygnał jest indukowany na pierwszych dwóch płaszczyznach, które są transparentne dla dryfujących elektronów. W ostatniej płaszczyźnie sygnał powstaje przez zbieranie ładunku. W każdej płaszczyźnie druty są rozmieszczone co 3 mm. W kolejnych płaszczyznach druty ułożone są względem siebie pod różnymi kątami. Pozwala to na zlokalizowanie źródła obserwowanego sygnału w przestrzeni. Sygnał odczytywany przez druty jest wzmacniany i dyskretyzowany z częstotliwością próbkowania 2,5 MHz. Układając przebiegi odczytane z kolejnych drutów w jednej płaszczyźnie obok siebie otrzymany zostanie dwuwymiarowy obraz o rozdzielczości przestrzennej $0,64 \times 3$ mm, przedstawiający dwuwymiarową projekcję obserwowanego w detektorze zdarzenia. Na rysunku 7.2 przedstawiono schemat działania detektora.



Rysunek 7.2: Schemat działania detektora ciekłoargonowego z komorą projekcji czasowej. Źródło: http://cerncourier.com/cws/article/cern/46538

Rekonstrukcja obserwowanych zdarzeń

Odczytane dwuwymiarowe obrazy z trzech płaszczyzn drutów wykorzystywane są przez algorytmy służące do rekonstrukcji i identyfikacji obserwowanych w detektorze zdarzeń. W trakcie rekonstrukcji wyznaczane są wartości takie jak: pęd, rodzaj obserwowanych cząstek, energia oddziaływania. Na podstawie obliczonych wartości możliwe jest rozpoznanie rodzaju oddziaływania neutrina. Dzięki obliczonym informacjom możliwe jest badanie

następujących zagadnień:

- hierarchia mas neutrin,
- łamanie symetrii CP w sektorze leptonowym,
- weryfikacja istnienia neutrin sterylnych.

Obecnie używane algorytmy do analizy obrazów uzyskanych w detektorze bazują na porcjach sygnału, nazywanych *hitami* [2], [9], [119]. Jeden hit przedstawia porcję sygnału zarejestrowaną na pojedynczym drucie. Jego położenie wyznacza się przez dopasowywane złożenia odpowiedzi referencyjnych. Przygotowane zbiory hitów wykorzystywane są do rekonstrukcji torów [119]. Analiza bazująca na zbiorach hitów ma pewne wady:

- hity mogą być niedokładnie wyznaczone - do ich wyznaczenia potrzebne jest rozróżnienie miejsc w sygnale na drucie, gdzie sygnał pochodzi od tła, a gdzie od cząstki; Zazwyczaj decyzja ta jest podejmowana na podstawie wcześniej dobranej wartości progu, co może być zawodne w przypadku gdy stosunek sygnału do szumu jest niski;
- jeden hit może przedstawiać sygnał pochodzący od kilku cząstek, których projekcja torów wypadała w tym samym miejscu na drucie, w takim przypadku hit powinien być rozpatrywany razem ze swoim otoczeniem w przeciwnym wypadku tracona jest informacja o zdarzeniu.

W niniejszej rozprawie przedstawione zostaną dwie metody analizy danych z detektora bazujące na obrazie bez konieczności tworzenia pośrednich reprezentacji opisujących obraz:

- metoda segmentacji torów z obrazu - metoda ta pozwala na klasyfikację pikseli w obrazie w zależności czy przedstawiają one tor cząstki czy szum;
- metoda odróżniania zaobserwowanych zdarzeń pochodzących od neutrin elektronowych od tła.

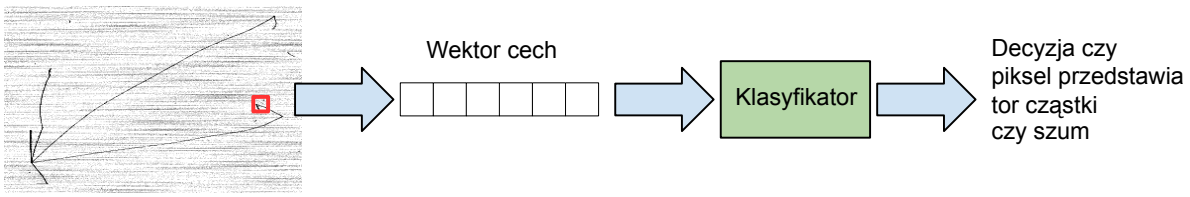
Opisane metody zostały przedstawione również w artykułach [100], [101].

7.2 Segmentacja obrazu z detektora

7.2.1 Opis problemu

Większość używanych obecnie algorytmów rekonstrukcji zdarzeń obserwowanych w detektorach ciekłoargonowych bazuje na zbiorach hitów. W procesie tworzenia hitów może dojść do utraty cennych informacji o zdarzeniu z dwóch przyczyn, wyżej wymienionych. W rozprawie proponowane jest podejście do analizy danych, które bazuje na całym obrazie uzyskanym w detektorze. Poniżej przedstawiona została metoda segmentacji torów z obrazu. Metoda ta klasyfikuje piksele obrazu ze względu na to czy przedstawiają tor cząstki czy szum. Algorytm ten może stanowić krok wstępny w analizie obrazów z detektora. W rozprawie metoda segmentacji została użyta do przygotowania danych w zadaniu klasyfikacji neutrin elektronowych. Schemat działania zaproponowanego rozwiązania do segmentacji torów został przedstawiony na rysunku 7.3. Składa się on z następujących kroków:

1. Konstrukcja wektora cech opisującego rozkład sygnału w rozpatrywanym pikselu oraz jego sąsiedztwie.
2. Nauka klasyfikatora z wykorzystaniem stworzonej reprezentacji opisującej piksel.
3. Klasyfikacja pikseli na przedstawiające tory cząstek lub szum.



Rysunek 7.3: Schemat działania algorytmu do segmentacji torów z obrazu.

7.2.2 Konstrukcja cech

Do opisania sygnału w pikselu oraz jego otoczeniu został użyty następujący zestaw cech:

- wartość sygnału w pikselu;
- statystyka sygnału w sąsiedztwie rozpatrywanego piksela, opisana jako: minimum, maksimum, mediana, średnia i odchylenie standardowe obliczone w oknach kwadratowych o wymiarach 3x3, 5x5, 7x7;

- różnica splotów gaussowskich obliczonych dla różnych wartości odchyleń standardowych: $\{0,5;2\}$, $\{0,5;3\}$, $\{0,5;4\}$, $\{0,75;2\}$, $\{0,75;3\}$, $\{0,75;4\}$, $\{1;2\}$, $\{1;3\}$, $\{1;4\}$;
- gradient obrazu - wyznaczony za pomocą operatora Prewitta;
- posortowane wartości własne hesjanu oraz ich suma i iloczyn;
- posortowane wartości własne tensora obrazu oraz ich suma i iloczyn policzone dla okien 3x3, 5x5, 7x7.

Wyznaczony w powyższy sposób zestaw 42 cech opisuje klasyfikowany piksel.

7.2.3 Opis danych

Do wygenerowania danych zostało wykorzystane oprogramowanie FLUKA [37] z wykorzystaniem parametrów detektora T-600. Wygenerowane zdarzenia miały energię 2 GeV. Z uzyskanych zdarzeń zostało wybranych niezależnie 50 zdarzeń dla płaszczyzny Induction2 oraz Collection. W rozprawie zostały wybrane dwie płaszczyzny, żeby pokazać działanie zaproponowanego systemu segmentacji torów na obrazach, które mają różne charakterystyki sygnału. Na wszystkich wybranych obrazach została przeprowadzona dekonwolucja z odpowiedzią impulsową elektronicznego układu odczytu. Dodatkowo obrazy zostały przeskalowane, by posiadać podobną rozdzielczość na obu osiach. Następnie wszystkie piksele w obrazie zostały przypisane do jednej z dwóch klas. Do klasy pozytywnej zaliczone zostały piksele przedstawiające sygnał pochodzący od torów cząstek, natomiast do klasy negatywnej przypisano piksele przedstawiające szum. Do pogrupowania pikseli została użyta dedykowana aplikacja z interfejsem graficznym stworzona przez autora rozprawy. Do każdego piksela oprócz klasy został dodany wektor cech opisujący rozkład sygnału w pikselu i jego okolicy (sposób wyznaczania wektora cech został opisany w rozdziale 7.2.2). Wektor cech razem z klasą piksela formują jedną próbkę w zbiorze danych. Zbiór danych został podzielony na podzbiór uczący (40 zdarzeń) i testujący (10 zdarzeń). Liczba próbek w każdym podzbiorze została przedstawiona w tabeli 7.1. Warto zwrócić uwagę, że analizowane dane są niezbalansowane. Stosunek próbek przedstawiających piksele odpowiadające torom cząstek do pozostałych wynosi 1:305 i 1:109, kolejno dla Induction2 i Collection.

Tabela 7.1: Liczba próbek negatywnych i pozytywnych w zbiorze uczącym i testującym w widokach Induction2 oraz Collection.

	Induction2		Collection	
Klasa	Negatywna	Pozytywna	Negatywna	Pozytywna
Zbiór uczący	15 648 122	51 300	7 290 844	66 749
Zbiór testujący	4 427 377	10 437	2 061 778	16 988

7.2.4 Wyniki analizy

W analizie danych sprawdzone zostało działanie trzech klasyfikatorów:

- Drzewo Decyzyjne jednostopniowe (*decision stump*) (rozdział 1.1.2),
- Regresja Logistyczna (rozdział 1.1.1),
- Las Losowy (rozdział 1.1.3).

Drzewo decyzyjne jednostopniowe zostało wybrane ponieważ działa tak jak próg decyzyjny w procedurze wyznaczania hitów używanej przez fizyków.

Analizowane dane są niezbalansowane [91], dlatego w celu ułatwienia uczenia klasyfikatora stosunek klas został zmieniony w zbiorze uczącym - jest to typowy zabieg w postępowaniu z danymi niezbalansowanymi, nazywany *downsampling* [91]. Klasyfikatory były uczone na danych z różnym stosunkiem klas: {1:1, 1:10, 1:20, 1:50, 1:100}. Natomiast w zbiorze testującym użyto wszystkich próbek z oryginalnym stosunkiem klas.

Wyniki klasyfikacji zostały przedstawione na rysunku 7.4. W klasyfikacji przy pomocy Lasu Losowego używanych było 100 drzew (w dalszej części rozdziału została przeprowadzona analiza wpływu liczby drzew na skuteczność klasyfikacji). Na podstawie uzyskanych rezultatów można zauważyć, że dokładność klasyfikacji dla Drzewa Decyzyjnego wzrasta ze zwiększaniem liczby próbek negatywnych w zbiorze. Wraz ze zwiększaniem liczby próbek w zbiorze uczącym reguła decyzyjna w drzewie jest dokładniej wyznaczana. Podobne zachowanie można zaobserwować dla pozostałych klasyfikatorów, w szczególności w płaszczyźnie Induction2 - im więcej danych jest dostępnych, tym dokładniej klasyfikator może być nauczony. W dalszych analizach używany będzie do nauki tylko stosunek klas 1:100.

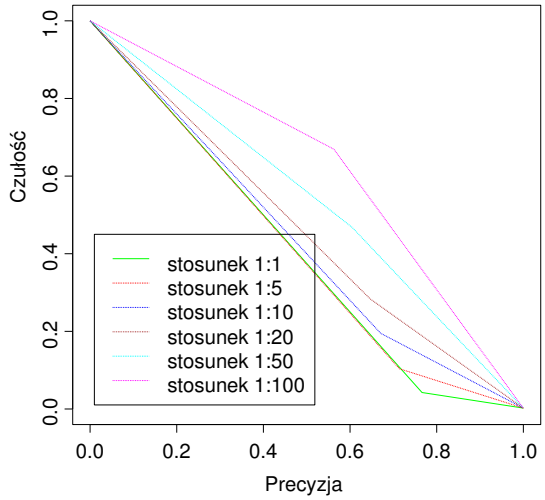
Na wykresie 7.5 przedstawione zostało porównanie klasyfikatorów. Las Losowy klasyfikuje najlepiej próbki z obu płaszczyzn. Fakt, że Las Losowy działa najlepiej wynika

z tego, że jest to w rzeczywistości klasyfikator zbiorowy złożony z większej liczby klasyfikatorów. W związku z tym posiada znacznie więcej parametrów niż pojedyncze Drzewo Decyzyjne lub Regresja Logistyczna, dzięki czemu jest w stanie nauczyć się więcej złożonych zależności występujących w danych. Różnica w skuteczności klasyfikacji pomiędzy Lasem Losowym a pozostałymi klasyfikatorami jest większa na zbiorze pochodzącym z płaszczyzny Induction2 niż Collection. Na tej podstawie można stwierdzić, że dane w zbiorze Induction2 są bardziej złożone.

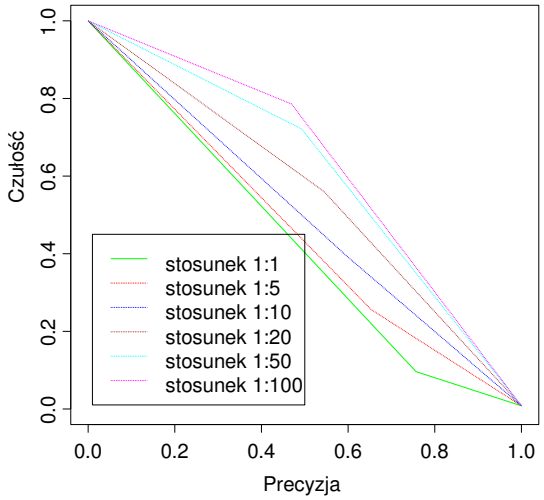
Skuteczność klasyfikacji w zależności od liczby drzew użytych w algorytmie Lasu Losowego została przedstawiona na wykresie 7.6. Sprawdzone zostały następujące liczby drzew w Lesie Losowym: {10, 20, 50, 100, 200, 500, 1000}. Dla danych pochodzących z płaszczyzny Induction2, skuteczność klasyfikacji jest najwyższa dla 100 drzew i nie zmienia się znacząco wraz z dodawaniem większej ich liczby. Natomiast dla danych z płaszczyzny Collection wysoka skuteczność jest już od 50 drzew w lesie. Większa liczba drzew potrzebna do analizy danych z płaszczyzny Induction2 potwierdza, że dane te są bardziej złożone niż z płaszczyzny Collection.

Po klasyfikacji wszystkich pikseli w obrazie, uzyskane odpowiedzi klasyfikatora można przedstawić w formie obrazu. Odpowiedzi sprawdzonych klasyfikatorów na przykładowym zdarzeniu testowym zostały pokazane na rysunku 7.7. Dla odpowiedzi uzyskanych za pomocą Drzewa Decyzyjnego lub Regresji Logistycznej jest więcej pikseli błędnie zaklasyfikowanych jako szum niż dla odpowiedzi uzyskanej za pomocą Lasu Losowego, przez co w wynikowym obrazie na torach cząstek można zaobserwować przerwy.

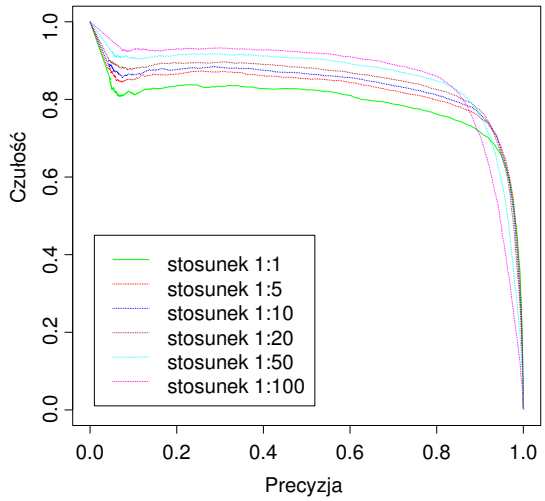
(a) Drzewo Decyzyjne, Induction2



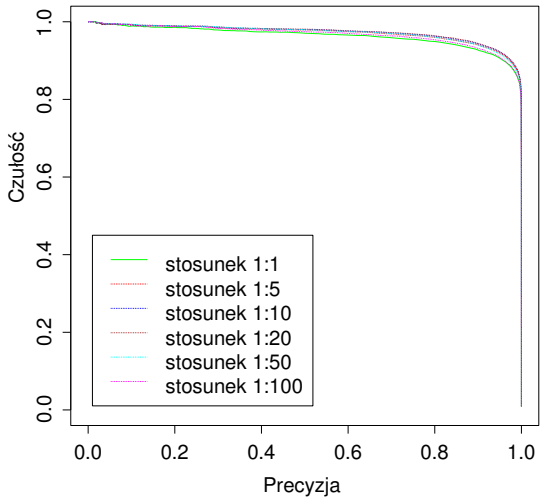
(b) Drzewo Decyzyjne, Collection



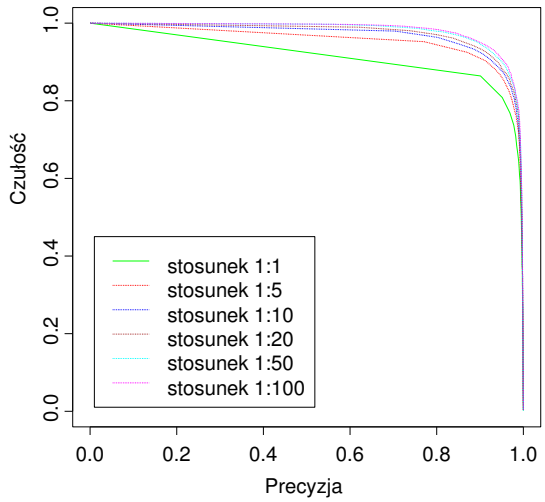
(c) Regresja Logistyczna, Induction2



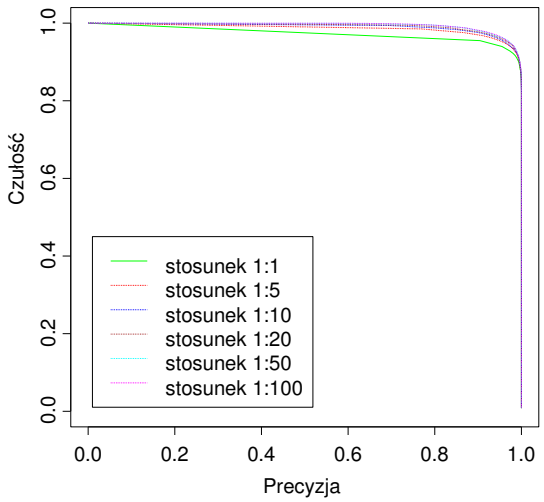
(d) Regresja Logistyczna, Collection



(e) Las Losowy, Induction2

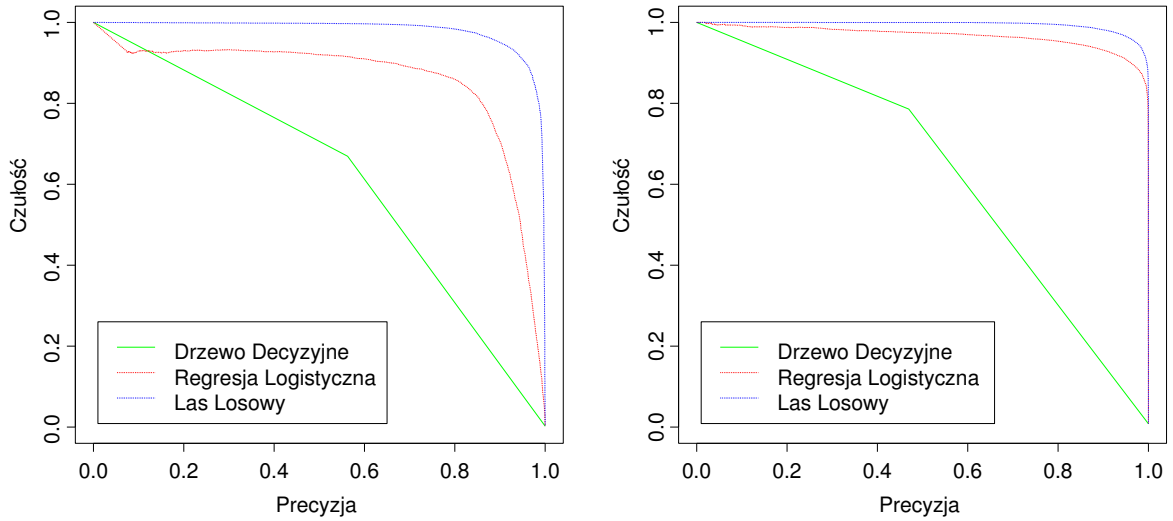


(f) Las Losowy, Collection



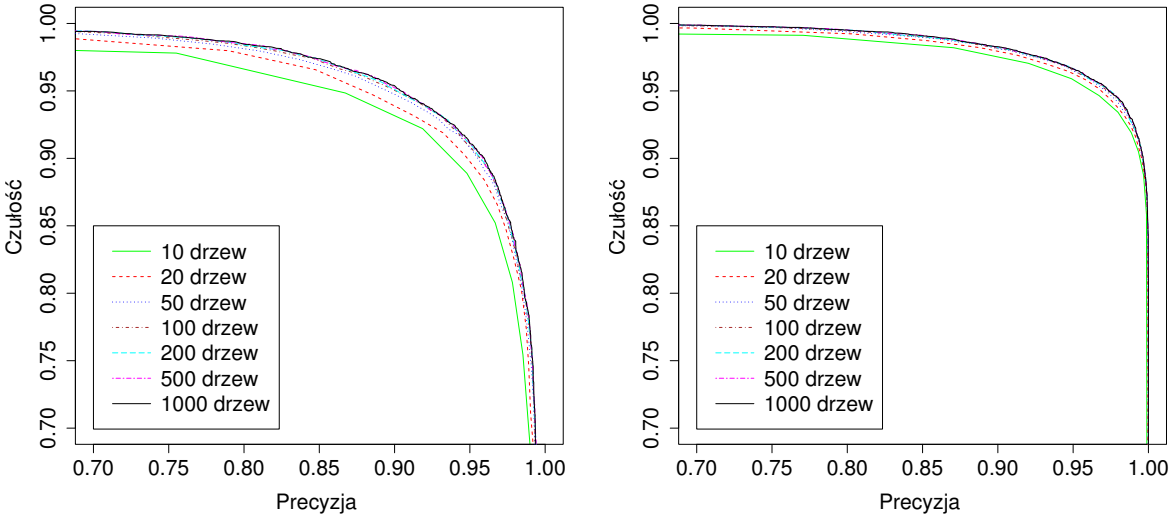
Rysunek 7.4: Wykresy precyzji i czułości dla używanych algorytmów klasyfikacji wyznaczone na zbiorach testowych. Klasyfikatory uczono na zbiorach danych z różnym stosunkiem klas.

(a) Induction2 (b) Collection



Rysunek 7.5: Porównanie skuteczności działania klasyfikatorów wyznaczone na zbiorze testowym. Klasyfikatory uczono na zbiorze danych ze stosunkiem klas 1:100.

(a) Induction2 (b) Collection



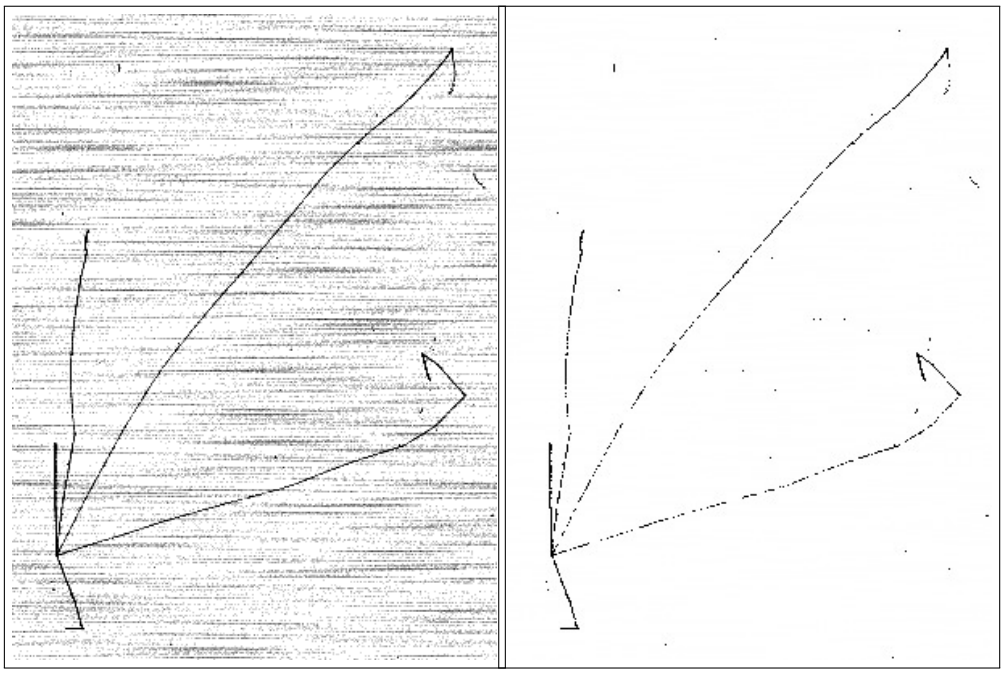
Rysunek 7.6: Skuteczność klasyfikacji Lasem Losowym w zależności od liczby drzew użytych w klasyfikatorze.

Wykorzystując klasyfikator nauczony algorytmem Lasu Losowego wyznaczono istotność cech użytych w analizie (rozdział 2.2.1). W tabeli 7.2 przedstawionych zostało 10 najważniejszych cech dla danych pochodzących z każdej płaszczyzny. Dla każdej z nich zostały wybrane różne cechy, ponieważ sygnał ma różną charakterystykę w płaszczyznach Induction2 i Collection. Warto zwrócić uwagę, że amplituda sygnału jest dopiero na 9. pozycji w rankingu dla klasyfikatora analizującego dane z płaszczyzny Induction2. Natomiast dla klasyfikatora działającego na danych z płaszczyzny Collection nie wystąpi-

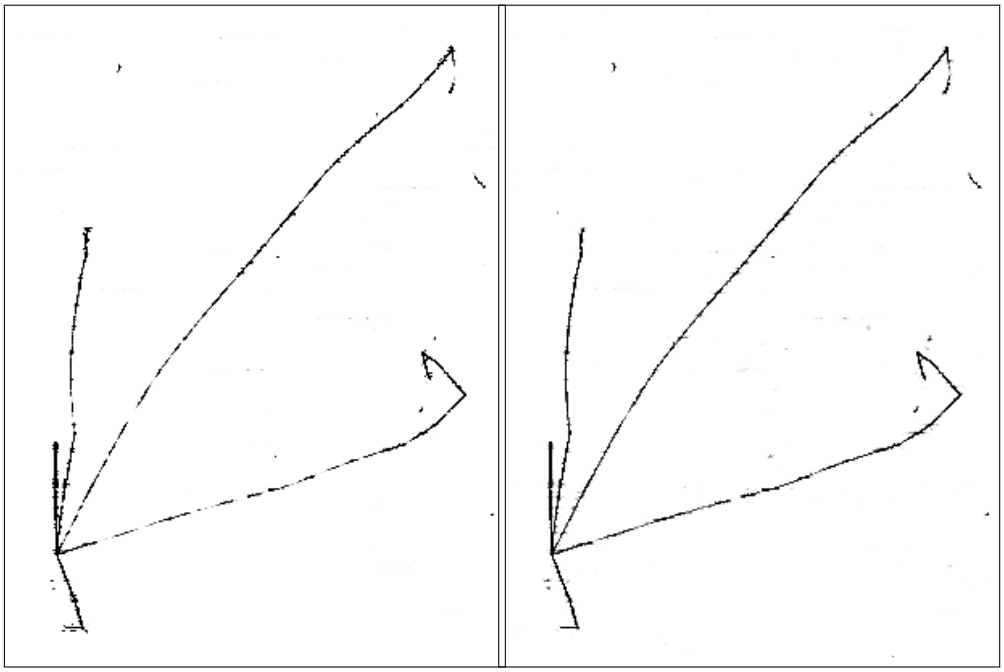
ła w pierwszej dziesiątce. Dzięki konstrukcji nowych cech, bazujących na wartości sygnału w pikselu i jego otoczeniu, uzyskano cechy, które są bardziej przydatne w klasyfikacji.

Wszystkie zaproponowane cechy w analizowanym problemie są istotne w klasyfikacji. Przy próbie eliminacji najsłabszej cechy algorytmem selekcji w tył (rozdział 2.3.2) skuteczność klasyfikacji dla obu płaszczyzn maleje.

(a) Odczytany sygnał (b) Odpowiedź z Drzewa Decyzyjnego



(c) Odpowiedź z Regresji Logistycznej (d) Odpowiedź z Lasu Losowego



Rysunek 7.7: (a) Odczytany sygnał dla przykładowego, testowego zdarzenia z płaszczyzny Induction2; (b,c,d) odpowiedź sprawdzanych klasyfikatorów.

Tabela 7.2: Najważniejsze cechy wybrane według algorytmu Lasu Losowego.

Ranking	Induction2	Collection
1	maksimum w oknie 5x5	średnia w oknie 3x3
2	maksimum w oknie 3x3	mediana w oknie 3x3
3	maksimum w oknie 7x7	średnia w oknie 5x5
4	odchylenie std. w oknie 5x5	pierwsza wartość własna tensora w oknie 3x3
5	odchylenie std. w oknie 3x3	mediana w oknie 5x5
6	pierwsza wartość własna tensora w oknie 3x3	maksimum w oknie 5x5
7	odchylenie std. w oknie 7x7	odchylenie std. w oknie 5x5
8	średnia w oknie 3x3	średnia w oknie 7x7
9	amplituda piksela	maksimum w oknie 3x3
10	mediana w oknie 3x3	maksimum w oknie 7x7

7.2.5 Dyskusja

Do segmentacji torów z obrazów uzyskanych w detektorze ciekłoargonowym zaproponowano nową metodę, która:

- konstruuje wektor cech przypisany do piksela,
- trenuje klasyfikator, który bazując na stworzonym wektorze cech podejmuje decyzję o przynależności sygnału do toru.

Zaproponowana metoda została porównana z dotychczas stosowaną metodą progowania na podstawie amplitudy sygnału w pikselu, która jest wykorzystywana do wyznaczania hitów w obrazie.

Przedstawiona metoda została przetestowana na obrazach pochodzących z dwóch różnych płaszczyzn, w których sygnał ma inną charakterystykę. Uzyskane wyniki pokazują, że metoda jest uniwersalna i może być stosowana niezależnie od typu sygnału. W przypadku użycia metody na danych pochodzących z nowego eksperymentu, który wykorzystuje technikę detekcji LArTPC, konieczne jest wytrenowanie nowego klasyfikatora na danych uzyskanych z nowego detektora (lub wygenerowanych z użyciem parametrów nowego detektora). Skonstruowane cechy użyte w proponowanej metodzie niosą więcej informacji niż amplituda piksela. Przedstawiona metoda stanowi krok przygotowujący obrazy z detektora do dalszej analizy. Opisana technika segmentacji torów może być ulepszana poprzez rozszerzanie wektora cech lub stosowanie bardziej złożonych klasyfikatorów. Ciekawym rozwiązaniem wydaje się również próba zastosowania głębokich sieci neuronowych do au-

tomatycznego wygenerowania reprezentacji atrybutów opisujących piksel.

7.3 Klasyfikacja neutrin elektronowych

7.3.1 Opis problemu

Jednym z celów eksperymentów neutrinowych jest zbadanie występowania neutrin elektronowych w wiązce neutrin mionowych. Do przeprowadzenia takich badań wymagana jest metoda pozwalająca na rozróżnianie zapachu obserwowanego neutrina, a w przypadku detektorów znajdujących się na powierzchni - metoda odrzucenia zdarzeń pochodzących od promieniowania kosmicznego. Metoda selekcji neutrin elektronowych powinna analizować zdarzenie w miejscu gdzie doszło do oddziaływania - w początkowym wierzchołku interakcji (ang. *primary interaction vertex* (PIV)). Taka analiza powinna wykryć pojedynczy elektron i istnienie hadronów. Pozwoli to na wykluczenie zdarzeń mogących przypominać neutrina elektronowe, takich jak:

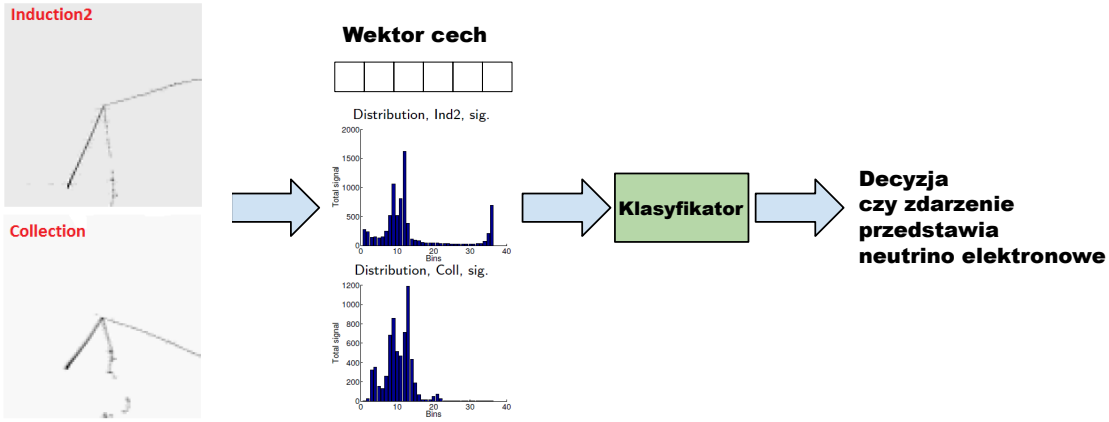
- oddziaływanie neutrina z cząstką π_0 w wierzchołku początkowej interakcji, wynikiem czego jest rozpad gamma, i konwersja w parę e+/e- w bliskim położeniu wierzchołka;
- cząstki gamma pochodzące z promieniowania kosmicznego, które konwertują w parę e+/e-.

W rozprawie przedstawiono metodę rozróżniania neutrin elektronowych od tła pochodzenia kosmicznego.

Każde zdarzenie zaobserwowane w detektorze opisane jest za pomocą 3 obrazów. W proponowanej metodzie, na podstawie dwóch obrazów (z płaszczyzn Induction2 oraz Collection) dla każdego zdarzenia konstruowany jest wektor cech. Stanowi on wejście dla klasyfikatora, którego odpowiedzią jest prawdopodobieństwo że obserwowane zdarzenie przedstawia neutrino elektronowe. Schemat działania zaproponowanej metody przedstawiono na rysunku 7.8.

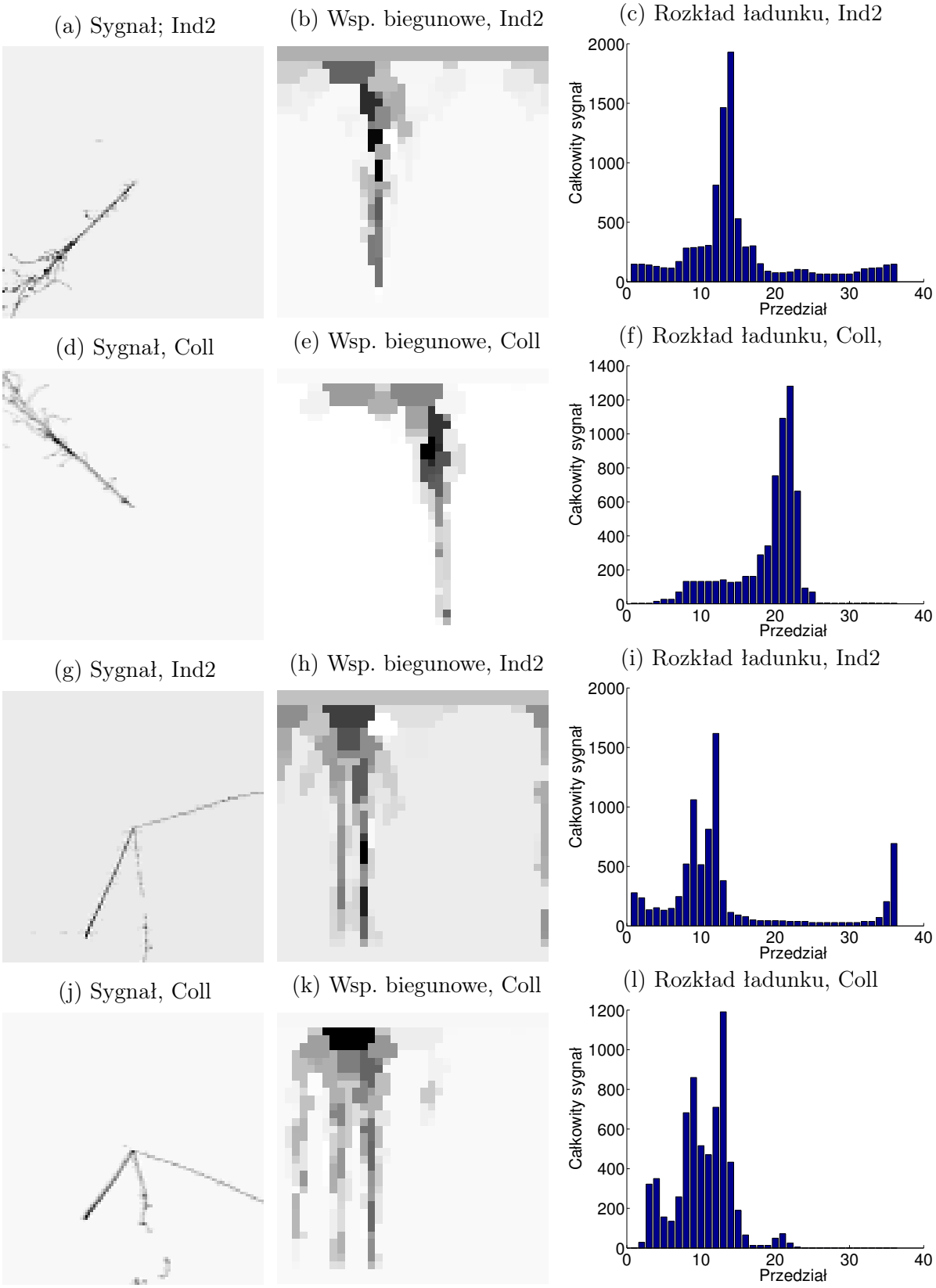
7.3.2 Konstrukcja cech

Kluczowym elementem zaproponowanej metody jest sposób tworzenia wektora cech, który opisuje zdarzenie. Wektor ten powinien uwzględniać fakt, iż w zdarzeniach pochodzących z różnych klas inaczej rozłożona jest amplituda sygnału, zaczynając od wierzchołka



Rysunek 7.8: Schemat działania algorytmu do klasyfikacji neutrin elektronowych na podstawie obrazu z detektora.

początkowej interakcji. Zdarzenie obserwowane w detektorze jest opisane 3 obrazami pochodzącymi z 3 płaszczyzn drutów. W metodzie wykorzystano dwa obrazy, pochodzące z płaszczyzn Induction2 oraz Collection. Z uwagi na słabą jakość obrazu z płaszczyzny Induction1 nie został on uwzględniony w analizie. Każdy rozpatrywany obraz transformowany jest do układu współrzędnych biegunowych ze środkiem w wierzchołku pierwotnej interakcji, długością promienia R oraz liczbą podziałów L - gdzie każdy podział przedstawia wycinek koła o szerokości $360/L$ stopni. W każdym wycinku koła sumowana jest wartość wszystkich pikseli w nim zawartych. W ten sposób uzyskany zostaje rozkład przedstawiający liczbę ładunku zarejestrowaną w różnych przedziałach kątowych. Dodatkowo, w każdej płaszczyźnie na podstawie wyznaczonego rozkładu obliczane są opisujące go zmienne statystyczne, takie jak: minimum, maksimum, odchylenie standardowe, średnia, całkowita wartość sygnału. Wyznaczone atrybuty z obu płaszczyzn tworzą wektor cech opisujący zdarzenie o rozmiarze $2(L + 5)$. Na rysunku 7.9 przedstawiono przykładowe zdarzenie zaobserwowane dla neutrina elektronowego oraz tła pochodzenia kosmicznego. Pokazano również odpowiadające im obrazy po transformacji do biegunowego układu współrzędnych i wyznaczony rozkład ładunku. Warto zauważyć, że na wykresach 7.9 c, f, przedstawiających rozkład ładunku dla klasy negatywnej, występuje jeden pik, natomiast dla rozkładów ładunku odpowiadających klasie pozytywnej (wykresy 7.9 i, l) występuje kilka pików. Jest to główna różnica pomiędzy zdarzeniami z obu klas. Co więcej, można zauważyć, że pik w rozkładach ładunku dla klasy negatywnej jest szerszy niż piki w rozkładach dla klasy pozytywnej.



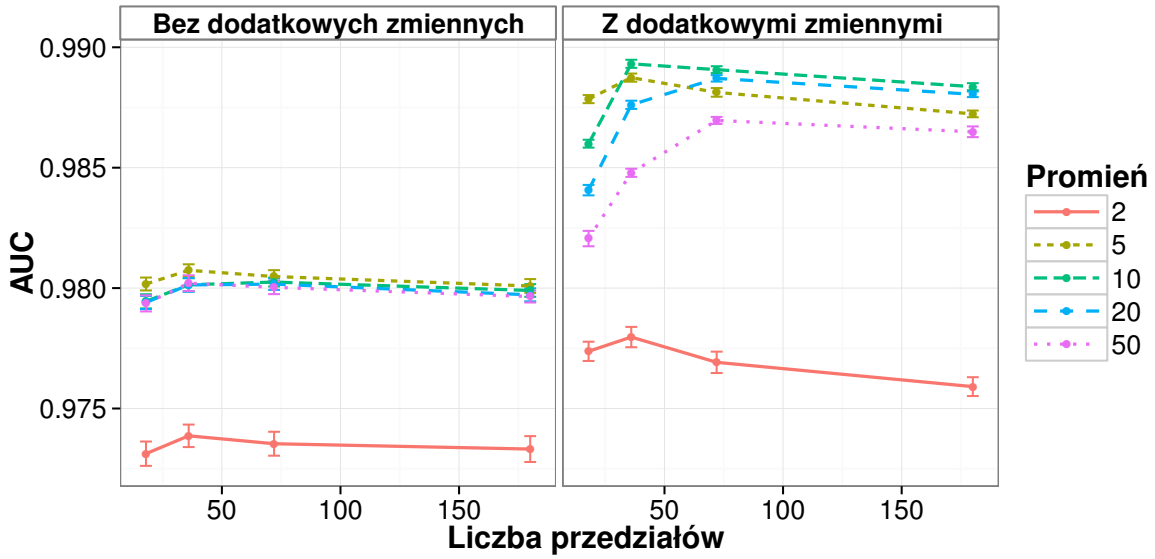
Rysunek 7.9: Przykład zdarzenia pochodzącego z klasy negatywnej (promieniowanie kosmiczne) (a, d) i pozytywnej (neutrino elektronowe) (g, j) zaprezentowanego jako obserwowany obraz i korespondujący obraz po transformacji do współrzędnych biegunowych i rozkład ładunku. W każdym wierszu zaprezentowano obserwację jednego zdarzenia z jednej płaszczyzny.

7.3.3 Opis danych

Analizowane dane zostały wygenerowane z wykorzystaniem pakietu FLUKA [37] oraz parametrów detektora T-600. Zostało wygenerowanych 7 090 zdarzeń z zakresu energii od 0,2 do 1 GeV. Wśród nich było 3 283 zdarzeń pochodzących od neutrina elektronowego i 3 807 zdarzeń pochodzących z promieniowania kosmicznego. Dla każdego zdarzenia przyjęto, że położenie lokalizacji wierzchołka interakcji początkowej jest znane. Wszystkie rozpatrywane zdarzenia mają wierzchołek początkowej interakcji zlokalizowany co najmniej 5 cm od anody lub katody. Wszystkie obrazy zostały poddane dekonwolucji z odpowiedzią impulsową układu elektroniki odpowiedzialnej za odczyt sygnału. Uzyskane obrazy poddane były procesowi segmentacji (opisanemu w rozdziale 7.2 oraz w artykule [100]). Dla każdego zdarzenia rozpatrywane były obrazy z płaszczyzn: Induction2 i Collection. Z każdego obrazu wycięty został kawałek o rozmiarach 101 drutów na 505 próbek i środku w wierzchołku początkowej interakcji. Obrazy zostały przeskalowane do rozmiaru 101x101, żeby posiadać podobną rozdzielczość na obu osiach. Rozpatrywana wielkość i rozdzielczość kawałków obrazu jest wystarczająca do analizy opisywanej w rozprawie.

7.3.4 Wyniki analizy

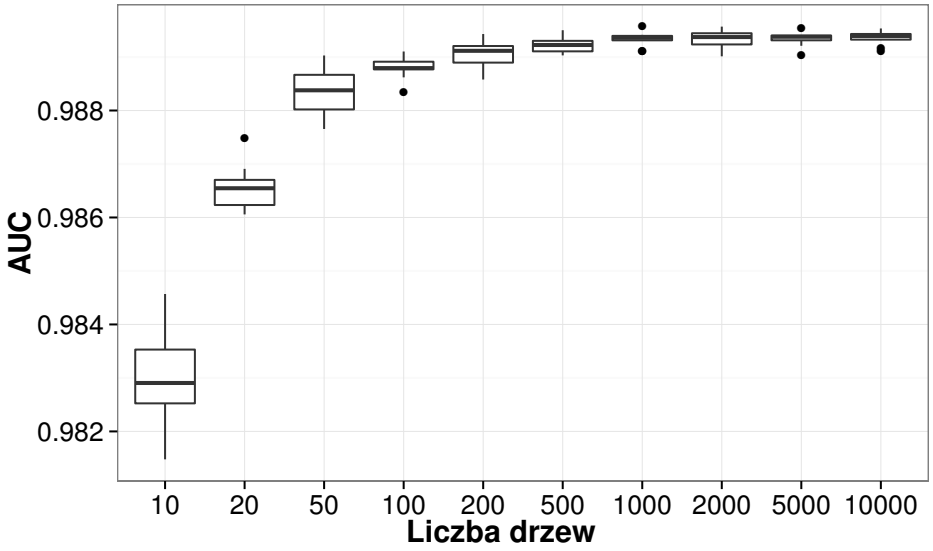
Rozmiar wektora cech zależy od liczby podziałów L użytego w transformacji do współrzędnych biegunowych. Wartości cech zależą od długości promienia R użytego w transformacji. W celu dobrania najlepszej kombinacji parametrów zostały sprawdzone następujące wartości: $L \in \{18, 36, 72, 180\}$ oraz $R \in \{2, 5, 10, 20, 50\}$, gdzie długość wyrażona jest w pikselach. Skuteczność klasyfikacji przy użyciu różnych kombinacji parametrów została obliczona przy użyciu 5-krotnej kroswalidacji krzyżowej powtórzonej 10 razy i klasyfikatora typu Las Losowy zawierającego 1000 drzew. Ponadto, sprawdzony został wpływ dodania zmiennych opisujących rozkład ładunku na skuteczność klasyfikacji. Jakość klasyfikacji była wyznaczona za pomocą pola pod krzywą ROC. Wyniki zostały przedstawione na wykresie 7.10. Skuteczność klasyfikacji jest wyższa, gdy wektor cech zawiera zmienne statystyczne opisujące rozkład ładunku. Najlepszą skuteczność klasyfikator osiąga, gdy przy transformacji do współrzędnych biegunowych użytych jest 36 podziałów, a długość promienia wynosi 10 pikseli oraz używane są dodatkowe zmienne opisujące rozkład ładunku - wartość AUC wynosi 0,9893±0,0001, natomiast dokładność klasyfikacji równa się 0,9535±0,0007. W dalszych analizach użyto najlepszej konfiguracji parametrów.



Rysunek 7.10: Skuteczność klasyfikacji obliczona za pomocą 5-krotnej kroswalidacji krzyżowej powtórzonej 10-krotnie dla różnych kombinacji wartości parametrów L i R oraz w zależności od użycia dodatkowych zmiennych opisujących rozkład ładunku.

W proponowanej metodzie jako klasyfikator został użyty Las Losowy. Na wykresie 7.11 przedstawiono jak skuteczność klasyfikacji zależy od liczby użytych drzew w klasyfikatorze. Działanie klasyfikatora zostało sprawdzone dla liczby drzew $T \in \{10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000\}$. Po zwiększeniu liczby drzew z 10 do 1000 skuteczność klasyfikacji wzrasta. Dla liczby drzew od 1000 do 10000 skuteczność się nie zmienia. W dalszych analizach używanych będzie 1000 drzew w klasyfikatorze.

W proponowanej metodzie przyjęto, że znana jest pozycja wierzchołka początkowej interakcji, aczkolwiek w przypadku rzeczywistych danych warunek ten nie jest spełniony, dlatego wymagany jest algorytm do jego wyznaczenia. W pracy [89] został przedstawiony algorytm do wyznaczania pozycji wierzchołków interakcji, który w połączeniu z dodatkowymi regułami decyzyjnymi może zostać użyty do wyznaczenia pozycji szukanego wierzchołka początkowego. W związku z tym, pozycja wierzchołka początkowej interakcji wyznaczona algorytmem będzie znana z błędem. Dlatego też została zbadana skuteczność klasyfikacji przy użyciu pozycji wierzchołka z szumem. Pozycja wierzchołka została zaburzona przez dodanie losowych liczb z określonego przedziału na obu osiach względem prawdziwej pozycji. Dodawany szum został wygenerowany w różnych przedziałach: $\{1, 2, 3, 4, 5\}$ pikseli. Skuteczność klasyfikacji przy różnych stopniach zaburzenia wierzchołka została przedstawiona w tabeli 7.3 oraz na wykresie 7.12a. Skuteczność działania metody malała wraz ze zwiększaniem poziomu zaburzenia w pozycji wierzchołka



Rysunek 7.11: Wynik działania klasyfikatora w zależności od liczby użytych drzew. Metryka została policzona przy pomocy 5-krotnej kroswalidacji krzyżowej powtórzonej 10 razy.

początkowej interakcji, jednakże przy poziomie zaburzenia równym 2 pikselom na każdej osi, co odpowiada zakłóceniom o ± 2 druty (oś X) i ± 10 próbek (oś Y) w wyznaczeniu pozycji wierzchołka, metoda ma wartość AUC równą 0,9360 ± 0,0006. Można oczekiwać, że algorytm do wyznaczenia pozycji wierzchołka początkowego nie powinien popełniać błędów na wyższym poziomie.

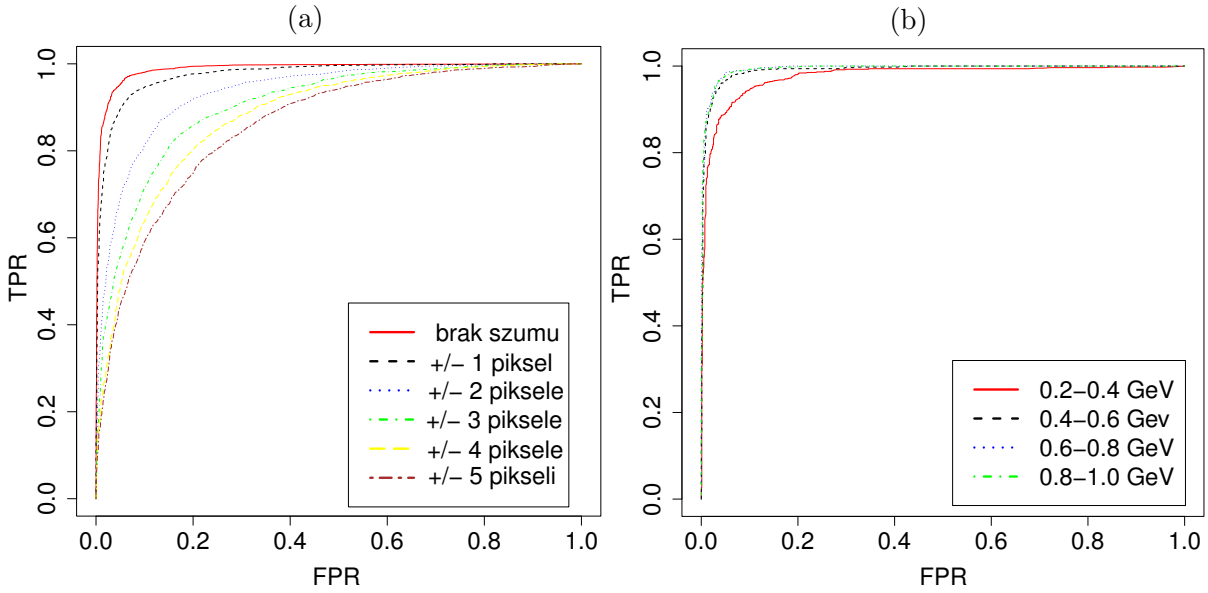
Obraz zarejestrowany w detektorze zależy od energii zdarzenia. Dlatego też zbadana została skuteczność klasyfikacji w zależności od energii zdarzenia. Rozpatrywane zostały następujące przedziały energii: 0,2 - 0,4; 0,4 - 0,6; 0,6 - 0,8; 0,8 - 1,0 GeV. Skuteczność klasyfikacji w zależności od energii została przedstawiona tabeli 7.4, a także przy pomocy krzywych ROC 7.12b. Z rezultatów wynika, iż skuteczność metody dla niskich energii, z zakresu (0,2 - 0,4) GeV nieznacznie spada. Natomiast dla większych energii skuteczność klasyfikacji jest prawie stała.

Tabela 7.3: Skuteczność klasyfikacji w zależności od poziomu zaburzenia w prawdziwej pozycji wierzchołka początkowej interakcji wyznaczona za pomocą 5-krotnej kroswalidacji krzyżowej powtórzonej 10 razy.

Poziom zaburzenia	AUC
brak zaburzenia	0,9893 ± 0,0001
± 1 piksel	0,9771 ± 0,0004
± 2 piksele	0,9360 ± 0,0006
± 3 piksele	0,9031 ± 0,0009
± 4 piksele	0,8800 ± 0,0009
± 5 pikseli	0,8601 ± 0,0014

Tabela 7.4: Skuteczność działania metody klasyfikacji neutrin elektronowych dla różnych zakresów energii zdarzeń obserwowanych w detektorze.

Zakres energii [GeV]	AUC
0,2 - 0,4	0,9770 ± 0,0005
0,4 - 0,6	0,9906 ± 0,0002
0,6 - 0,8	0,9944 ± 0,0003
0,8 - 1,0	0,9934 ± 0,0002



Rysunek 7.12: Skuteczność klasyfikacji w zależności od: (a) zaburzeń w położeniu wierzchołka interakcji początkowej, (b) energii obserwowanego zdarzenia.

7.3.5 Dyskusja

Przedstawiona metoda klasyfikacji neutrin elektronowych nie wymaga tworzenia reprezentacji pośrednich (hitów). W proponowanym podejściu dla każdego zaobserwowanego zdarzenia tworzony jest wektor cech. Na podstawie przygotowanych atrybutów trenowany jest klasyfikator, który potrafi wskazać prawdopodobieństwo zaobserwowania zdarzenia pochodzącego od neutrina elektronowego. Klasyfikacja proponowaną metodą osiąga pole pod krzywą ROC 0,9893±0,0001 oraz dokładność 0,9535±0,0007, wyznaczone na podstawie 5-krotnej kroswalidacji krzyżowej powtórzonej 10 razy. Opisana w rozprawie metoda zakłada, że pozycja wierzchołka początkowej interakcji jest znana. W analizie rzeczywistych przypadków założenie to nie będzie spełnione. Dlatego sprawdzono, jak szum w położeniu wierzchołka początkowego wpływa na skuteczność algorytmu. Przy losowym zakłócaniu położeniu wierzchołka na poziomie ±2 druty i ±10 próbek czasowych algorytm

osiąga pole pod krzywą ROC 0,9360 $\pm$ 0,0006.

Według wiedzy autora rozprawy jest to pierwsza metoda rozróżniania neutrin elektronowych od oddziaływania kosmicznego działająca na podstawie obrazów z detektora ciekłoargonowego.

Rozdział 8

Podsumowanie

W pracy zostały przedstawione wybrane metody służące do przekształcania, konstrukcji i selekcji cech, wraz z ich zastosowaniem w klasyfikacji i klasteryzacji w różnych dziedzinach nauki. Pokazano, iż pominięcie kroku przygotowania cech może prowadzić do błędnych wniosków wyciągniętych na podstawie analizy danych lub braku możliwości budowy systemu do klasyfikacji lub klasteryzacji.

W zadaniu klasteryzacji sekwencji wirusa grypy pochodzących z pandemii z 2009 roku zaproponowano metodę wyznaczającą wektor wag opisujący istotność pozycji nukleotydu w łańcuchu RNA w celu lepszej rekonstrukcji drzewa. Użyta metoda wykorzystywała algorytm genetyczny do wybrania takiego wektora wag, który zapewniał najlepsze rozmieszczenie klastrów w zrekonstruowanym drzewie. Dzięki zaproponowanej metodzie możliwe było zrekonstruowanie drzewa ewolucyjnego, w którym występuje tylko jedna mutacja E391K. Przy rekonstrukcji niewykorzystującej wektora wag obserwowanych było ponad 100 mutacji tego typu. Dodatkowo, współczynnik mutacji oszacowany na podstawie drzewa zrekonstruowanego z wykorzystaniem wag sugeruje, iż szybkość zmian wirusa jest o rząd wielkości mniejsza niż dotychczas sądzono.

W klasyfikacji dysleksji rozwojowej u dzieci na podstawie badań ze strukturalnego rezonansu magnetycznego, przed procesem budowy klasyfikatora przeprowadzono usunięcie czynników zakłócających oraz selekcję cech. Dzięki temu zwiększono stabilność algorytmów selekcji i skuteczność klasyfikacji. Dodatkowo, możliwe było wskazanie cech anatomicznych mózgu kluczowych w wykrywaniu dysleksji rozwojowej u dzieci. Uzyskane wyniki mogą być rozpatrywane jako pierwsze potwierdzenie *in-vivo* badań przeprowadzonych przez Galaburda w latach 70. i 80. ubiegłego wieku [44], [45].

W eksperymentach neutrinowych wykorzystujących detektory ciekłoargonowe bardzo ważnym elementem jest proces rekonstrukcji obserwowanych zdarzeń. W rozprawie wykorzystano metody konstrukcji cech w obrazie oraz selekcji cech w klasyfikacji do zbudowania systemów pozwalających na:

- segmentację torów z uzyskanych obrazów; jest to alternatywa do opisywania zdarzenia za pomocą zbioru *hitów* i może stanowić krok przygotowawczy dla metod działających na pikselach,
- klasyfikację neutrin elektronowych; metoda ta na podstawie obserwowanego obrazu potrafi wskazać, czy obserwowane zdarzenie pochodzi od neutrina elektronowego czy jest pochodzenia kosmicznego.

Opisane metody są pierwszymi metodami analizy zdarzeń z detektora ciekłoargonowego, które nie wykorzystują w rekonstrukcji reprezentacji pośredniej w postaci *hitów*. Dzięki temu do algorytmu analizującego trafia więcej informacji. W obu opisywanych rozwiązaniach został skonstruowany wektor cech opisujący obraz. Umożliwiło to zdefiniowanie problemu jako zadania klasyfikacji. Metoda klasyfikacji neutrin elektronowych jest pierwszą automatyczną metodą pozwalającą na rozróżnianie neutrin z zapachem elektronowym od tła kosmicznego w detektorze ciekłoargonowym.

Podsumowując, postawione w pracy cele zostały osiągnięte, a teza wykazana. Do największych i autorskich osiągnięć należą:

- użycie w trakcie rekonstrukcji filogenetycznej wektora wag opisującego ważność nukleotydów w łańcuchu RNA wirusa grypy oraz opracowanie metody jego wyznaczania; zrekonstruowanie drzewa filogenetycznego na podstawie sekwencji pochodzących z pandemii z 2009 roku, na podstawie którego możliwe było wyjaśnienie powtarzających się mutacji;
- zbudowanie systemu do klasyfikacji dzieci ze względu na posiadanie dysleksji rozwojowej na podstawie obrazu uzyskanego z badania rezonansem magnetycznym; w zaproponowanym systemie przed użyciem klasyfikatora wykonywana jest operacja usuwania czynników zakłócających poprawiająca stabilność selekcji cech oraz dokonywany jest wybór najważniejszych cech spośród dostępnych w celu zwiększenia skuteczności działania klasyfikatora; dzięki selekcji cech możliwe jest lepsze poznanie anatomicznych cech mózgu w dysleksji rozwojowej u dzieci;

- rozwój narzędzi do rekonstrukcji zdarzeń w detektorach ciekłoargonowych: zbudowanie systemu do segmentacji torów cząstek z obrazów oraz systemu klasyfikującego neutrina o zapachu elektronowym od cząstek pochodzenia kosmicznego; oba zaproponowane systemy wykorzystują klasyfikator działający na przygotowanym wektorze cech.

Według autora rozprawy uzyskane rezultaty stanowią dobry start do dalszej pracy. Możliwe przyszłe kierunki badań to:

- przyspieszenie działania algorytmów rekonstrukcji drzew filogenetycznych tak, by możliwa była analiza dużych zbiorów sekwencji - obecnie dostępnych jest ponad 6 500 sekwencji wirusa grypy typu H1N1,
- rozwój metod pozwalających na automatyczną analizę danych,
- rozwój algorytmów rekonstrukcji zdarzeń obserwowanych w detektorach ciekłoargonowych działających na podstawie uzyskanych obrazów z wykorzystaniem konwolucyjnych, głębokich sieci neuronowych.

Bibliografia

[1] Agrawal, R., Imieliński, T., Swami, A., Mining association rules between sets of items in large databases, Proceedings of the International Conference on Management of Data, 207-216, 1993

[2] Anderson, C., et. al., The ArgoNeuT Detector in the NuMI Low-Energy beam line at Fermilab, Journal of Instrumentation, vol. 7, 2012

[3] Arabas, J., Wykłady z algorytmów ewolucyjnych, Wydawnictwo Naukowo –Techniczne, 2004

[4] Autiero, D., et. al., Large underground, liquid based detectors for astro-particle physics in Europe: scientific case and prospects, Journal of Cosmology and Astroparticle Physics, vol. 11, 2007

[5] Azuaje, F., Devaux, Y., Wagner, D., Computational biology for cardiovascular biomarker discovery, Briefings in Bioinformatics, vol. 10, 367-77, 2009

[6] Balraj, P., Sidek, H., Suppiah, J., et al., Molecular analysis of 2009 pandemic influenza A (H1N1) in Malaysia associated with mild and severe infections, The Malaysian Journal of Pathology, vol. 33, 7–12, 2011

[7] Barr, I.G., Cui, L, Komadina, N., Lee, R.T., Lin, R.T., Deng, Y., Caldwell, N., Shaw, R., Maurer-Stroh, S., A new pandemic influenza A(H1N1) genetic variant predominated in the winter 2010 influenza season in Australia, New Zealand and Singapore, Euro Surveillance, vol. 15, 2010

[8] Bengio, Y., Courville, A., Vincent, P., Representation Learning: A Review and New Perspectives, IEEE Transactions on Pattern Analysis and Machine Intelligence, special issue Learning Deep Architectures, vol. 35, 1798-828, 2013

[9] Bennieston, A., J., Reconstruction techniques for fine-grained neutrino detectors, Doctoral dissertation, University of Warwick, 2013

[10] Bishop, C. M., Pattern Recognition and Machine Learning, Springer, 2006

[11] Błachnik, M., Laaksonen, J., Image Classification by Histogram Features Created with Learning Vector Quantization, Lecture Notes in Computer Science, vol. 5163, 827-836, 2008

[12] Boser, B. E., Guyon, I. M., Vapnik, V. N., A training algorithm for optimal margin classifiers, Proceedings of the fifth annual workshop on Computational learning theory, 1992

[13] Breiman, L.: Random Forests, Machine Learning, vol. 45, 5–32, 2001

[14] Breiman, L., Bagging Predictors, Machine Learning, vol. 26, 123-140, 1996

[15] Casanova, M.F., Araque, J., Giedd, J., Rumsey, J.M., Reduced brain size and gyrification in the brains of dyslexic patients, Journal of Child Neurology, vol. 19, 275-281, 2004

[16] Chandola, V., Banerjee, A., Kumar, V., Anomaly Detection: A Survey, ACM Computing Surveys, vol. 41, No. 3, 1-58, 2009

[17] Chang, B.S., Ly, J., Appignani, B., Bodell, A., Apse, K.A., Ravenscroft, R.S., Sheen, V.L., Doherty, M.J., Hackney, D.B., O'Connor, M., Galaburda, A.M., Walsh, C.A., Reading impairment in the neuronal migration disorder of periventricular nodular heterotopia, Neurology, vol. 64, 799-803, 2005

[18] Chudzian, P., Evaluation measures for kernel optimization. Pattern Recognition Letters, vol. 33, 1108-1116, 2012

[19] Cichosz, P., Systemy uczące się, Wydawnictwo Naukowo-Techniczne, 2000

[20] Cichosz, P., Data Mining Algorithms: Explained Using R, Wiley, 2015

[21] Clark, K.A., Helland, T., Specht, K., Narr, K.L., Manis, F.R., Toga, A.W., Hugdahl, K., Neuroanatomical precursors of dyslexia identified from pre-reading through to age 11, Brain, vol. 137, 3136-41, 2014

[22] Cortes, C.; Vapnik, V., Support-vector networks, Machine Learning, vol. 20, 1995

[23] Dalal, N., Triggs, B., Histograms of oriented gradients for human detection IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, 886 - 893, 2005

[24] Daniel Kostro, et al., Correction of inter-scanner and within-subject variance in structural MRI based automated diagnosing, NeuroImage vol. 98, 405–415, 2014

[25] David Salat, R.L. Buckner, A.Z. Snyder, Douglas N. Greve, R.S. Desikan, Evelina Busa, J.C. Morris, Anders Dale, and Bruce Fischl. Thinning of the cerebral cortex in aging. Cerebral Cortex, 14:721–730, 2004

[26] de Vasconcelos Hage, S.R., Cendes, F., Montenegro, M.A., Abramides, D.V., Guimarães, C.A., Guerreiro, M.M., Specific language impairment: linguistic annd neurobiological aspects, Arquivos de Neuro-Psiquiatria, vol. 64, 173–180, 2006

[27] Destrieux, C., Fischl, B., Dale, A., Halgren, E., Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature, Neuroimage, vol. 53, 1-15, 2010

[28] Dietterich, T.G., Ensemble methods in machine learning, Lecture Notes in Computer Science, vol. 1857, 1–15, 2001

[29] Donders, A. Rogier .T., van der Heijden, Geert J.M.G., Stijnen, T., Moons, Karel G.M., Review: A gentle introduction to imputation of missing values, Journal of Clinical Epidemiology, vol. 59, 1087 - 1091, 2006

[30] Dougherty, J., Kohavi, R., Sahami, M., Supervised and Unsupervised Discretization of Continuous Features, Proceedings of Twelfth International Conference on Machine Learning, 194–202, 1995

[31] Duch, W., Biesiada, J., Winiarski, T., Grudziński, K., Grąbczewski, K., Feature Selection Based on Information Theory Filters, Advances in Soft Computing, vol. 19, 173-178, 2003

[32] Eskildsen, S.,F., Coupé, P., Fonov, V., Pruessner, J.,C., Collins, D.,L., Structural imaging biomarkers of Alzheimer's disease: predicting disease progression, Neurobiology of Aging, vol. 36, S23–S31,2015

[33] Fawcett, T., An introduction to ROC analysis, Pattern Recognition Letters, vol. 27, 861–874, 2006

[34] Fayyad, U.M., Irani, K.B., Multi-interval discretization of continuousvalued attributes for classification learning. Thirteenth International Joint Conference on Articial Intelligence, 1022-1027, 1993

[35] Feil, E.J., Li, B.C., Aanensen, D.M., Hanage, W.P., Spratt, B.G., eBURST: Inferring Patterns of Evolutionary Descent among Clusters of Related Bacterial Genotypes from Multilocus Sequence Typing Data, Journal of Bacteriology, vol. 186, 1518–1530, 2004

[36] Fernandez-Delgado, M., Cernadas, E., Barro, S., Amorim, D., Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?, Journal of Machine Learning Research, vol. 15, 3133-3181, 2014

[37] Ferrari, A., Sala, P., R., Fasso, A., Ranft, J., FLUKA: a multi-particle transport code, CERN-2005-10 -2005 INFN TC 5 11, SLAC-R-773, 2005

[38] Ferreira, J.L., Borborema, S.E., Brigido, L.F., de Oliveira, M.I., de Paiva, T.M., dos Santos, C.L., Sequence analysis of the 2009 pandemic influenza A H1N1 virus haemagglutinin gene from 2009-2010 Brazilian clinical samples, Memorias do Instituto Oswaldo Cruz, vol. 106, 613-616, 2011

[39] Finch, A.J., Nicolson, R.I., Fawcett, A.J., Evidence for a neuroanatomical difference within the olivo-cerebellar pathway of adults with dyslexia, Cortex, vol. 38, 529–539, 2002

[40] Fitch, W.M., Bush, R.M., Bender, C.A., Cox, N.J., Long term trends in the evolution of H(3) HA1 human influenza type A, Proceedings of the National Academy of Sciences, vol. 94, 7712-7718, 1997

[41] Francisco, A.P., Bugalho, M., Ramirez, M., Carriço, J.A., Global optimal eBURST analysis of multilocus typing data using a graphic matroid approach, BMC Bioinformatics vol. 10, 2009

[42] Freund, Y., Schapire, R.E., Experiments with a New Boosting Algorithm, International Conference on Machine Learning, p. 148-156, 1996

[43] Frye, R.E., Liederman, J., Malmberg, B., McLean, J., Strickland, D., Beauchamp, M.S., Surface area accounts for the relation of gray matter volume to reading-related skills and history of dyslexia, Cerebral Cortex, vol. 20, 2625-2635, 2010

[44] Galaburda, A.M., Kemper, T.L., Cytoarchitectonic abnormalities in developmental dyslexia: a case study, Annals of Neurology, vol. 6, 94–100, 1979

[45] Galaburda, A.M., Sherman, G.F., Rosen, G.D., Aboitiz, F., Geschwind, N., Developmental dyslexia: four consecutive patients with cortical anomalies, Annals of Neurology, vol. 18, 222–233, 1985

[46] Galaburda, A.M., Menard, M.T., Rosen, G.D., Evidence for aberrant auditory anatomy in developmental dyslexia. Proceedings of the National Academy of Sciences USA, vol. 91, 8010-8013, 1994

[47] Galiano M., Agapow P.-M., Thompson C., Platt S., Underwood A., et al., Evolutionary Pathways of the Pandemic Influenza A (H1N1) 2009 in the UK. PLoS ONE, 2011

[48] Graham, M., Liang, B., Van Domselaar, G., Bastien, N., Beaudoin, C., et al., Nationwide Molecular Surveillance of Pandemic H1N1 Influenza A Virus Genomes: Canada, 2009, PLoS ONE 6, 2011

[49] Gregory, T.R., Understanding Evolutionary Trees, Evolution: Education and Outreach, vol. 1, 121-137, 2008

[50] Gu, Q., Li, Z., Han, J., Generalized Fisher Score for Feature Selection, In Proceedings of Conference on Uncertainty in Artificial Intelligence, 2011

[51] Hanczar, B., Hua, J., Sima, C., Weinstein, J., Bittner, M., Dougherty, E.R., Small-sample precision of ROC-related estimates, Bioinformatics, vol. 26, 822-830, 2010

[52] Hastie, T., Friedman, J., Tibshirani, R., The elements of statistical learning, Springer, 2009

[53] Haury, A.,C., Gestraud, P., Vert, J.,P., The Influence of Feature Selection Methods on Accuracy, Stability and Interpretability of Molecular Signatures, PLOS One, vol. 6, 2011

[54] He, K., Zhang, X., Ren, S., Sun, J., Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, Microsoft Technical report, arXiv:1502.01852, 2015

[55] Hea, Z., Yu, W., Stable feature selection for biomarker discovery, Computational Biology and Chemistry, vol. 34, 215–225, 2010

[56] Herfst, S., et al., Airborne Transmission of Influenza A/H5N1 Virus Between Ferrets, Science 336, 1534-1541, 2012

[57] Hilario, M., Kalousis, A., Approaches to dimensionality reduction in proteomic biomarker studies, Briefings in Bioinformatics, vol. 9, 102-118, 2008

[58] Ho, T., The random subspace method for constructing decision forests, Pattern Analysis and Machine Intelligence, vol. 20, 832-844, 1998

[59] Humphreys, P., Kaufmann, W.E., Galaburda, A.M., Developmental dyslexia in women: neuropathological findings in three patients, Annals of Neurology, vol. 28, 727–738, 1990

[60] Iba, W., Langley, P., Induction of One-Level Decision Trees, Proceedings of the Ninth International Conference on Machine Learning, 233–240, 1992

[61] Ikonen, N., Haanpaa, M., Ronkko, E., Lyytikainen, O., Kuusi, M., et al., Genetic Diversity of the 2009 Pandemic Influenza A(H1N1) Viruses in Finland, PloS ONE, 2010

[62] Imai, M., et al., Experimental adaptation of an influenza H5HA confers respiratory droplet transmission to a reassortant H5 HA/H1N1 virus in ferrets, Nature, vol. 486, 420-430, 2012

[63] Isabelle Guyon, Andre Elisseeff, An Introduction to Variable and Feature Selection, Journal of Machine Learning Research, vol. 3, 1157-1182, 2003

[64] Jahrer, M., Töscher, A., Legenstein, R., Combining predictions for accurate recommender systems, International Conference on Knowledge discovery and Data Mining, 693-702, 2010

[65] Jain, A. K., Murty, M. N., Flynn, P. J., Data clustering: a review, ACM Computing Surveys, vol. 31, 264-323, 1999

[66] Jednoróg, K., Gawron, N., Marchewka, A., Heim, S., Grabowska, A., Cognitive subtypes of dyslexia are characterized by distinct patterns of grey matter volume, Brain Structure and Function, vol. 219, 1697-707, 2014

[67] Jednoróg. K., Marchewka, A., Altarelli, I., Monzalvo, Lopez, A.K., van Ermingen-Marbach, M., Grande, M., Grabowska, A., Heim, S., Ramus, F., How reliable are gray matter disruptions in specific reading disability across multiple countries and languages? insights from a large-scale voxel-based morphometry study, Human Brain Mapping, vol. 36, 1741-54, 2015

[68] Jenner, A., Rosen, G.D., Galaburda, A.M., Neuronal asymmetries in the primary visual cortex of dyslexic and non-dyslexic brains, Annals of Neurology, vol. 46, 189–196, 1999

[69] Juergen Dukart, Matthias L. Schroeter, Karsten Mueller, Age Correction in Dementia – Matching to a Healthy Brain, PLOS ONE, vol. 6, 2011

[70] Kalousis, A., Prados, J., Hilario, M., Stability of feature selection algorithms: a study on high-dimensional spaces, Knowledge and Information Systems, vol. 12, 95-116, 2007

[71] Kao, C-L., Chan, T-C., Tsai, C-H., Chu, K-Y., Chuang, S-F., et al., Emerged HA and NA Mutants of the Pandemic Influenza H1N1 Viruses with Increasing Epidemiological Significance in Taipei and Kaohsiung, Taiwan, 2009–10, PLoS ONE, 2012

[72] Kohavi, R., A study of cross-validation and bootstrap for accuracy estimation and model selection, Proceedings of the International Joint Conference on Artificial Intelligence, vol. 2, 1137-1143, 1995

[73] Kotsiantis, S.B., Decision trees: a recent overview, Artificial Intelligence Review, vol. 39, 261-283, 2011

[74] Krawczyk, B., Schaefer, G., Effective multiple classifier systems for breast thermogram analysis, International Conference on Pattern Recognition, 3345-3348, 2012

[75] Kuncheva, L., A stability index for feature selection, Proceedings of 25th International Conference on Artificial Intelligence and Applications, 390-395, 2007

[76] Kuperberg, G.,R., M. Broome, P. K. McGuire, A. S. David, M. Eddy, F. Ozawa, D. Goff, W. C. West, S.C.R. Williams, Andre van der Kouwe, David Salat, Anders Dale, and Bruce Fischl. Regionally localized thinning of the cerebral cortex in schizophrenia. Archives of General Psychiatry, 60:878–888, 2003

[77] Kursa, M., B., Rudnicki, W., R., Feature Selection with the Boruta Package, Journal of Statistical Software, vol. 36, 2010

[78] Lee, S.J., Siau, K., A review of data mining techniques, Industrial Management and Data Systems, vol. 101, 41-46, 2001

[79] Lee, R.T., Santos, C.L., de Paiva, T.M., Cui L., Sirota, F.L., Eisenhaber, F., Maurer-Stroh, S., All that glitters is not gold – founder effects complicate associations of flu mutations to disease severity, Virology Journal, vol. 7, 2010

[80] Linkersdörfer, J., Lonnemann, J., Lindberg, S., Hasselhorn, M., Fiebach, C.J., Grey matter alterations co-localize with functional abnormalities in developmental dyslexia: an ALE meta-analysis, PLoS One, vol. 7, 2012

[81] Lowe, D., Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, vol. 60, 91-110, 2004

[82] Ma, Y., Koyama, M.S., Milham, M.P., Castellanos, F.X., Quinn, B.T., Pardoe, H., Wang, X., Kuzniecky, R., Devinsky, O., Thesen, T., Blackmon, K., Cortical thickness abnormalities associated with dyslexia, independent of remediation status. Neuroimage Clincal, vol. 7, 177-86, 2014

[83] Mak, G.C., Leung, C.K., Cheng, K.C., Wong, K.Y., Lim, W., Evolution of the haemagglutinin gene of the influenza A(H1N1)2009 virus isolated in Hong Kong 2009–2011, Euro Surveillance, 2011

[84] Markus Ojala, Gemma C. Garriga, Permutation Tests for Studying Classifier Performance, Journal of Machine Learning Research, vol. 11, 1833-1863, 2010

[85] Martin Reuter, Herminia Diana Rosas, and Bruce Fischl. Highly accurate inverse consistent registration: A robust approach. NeuroImage, vol. 53, 1181–1196, 2010

[86] Maurer-Stroh, S., Lee, R.T.C., Eisenhaber, F., Cui, L., Phuah, S.P., Lin, R.T., A new common mutation in the hemagglutinin of the 2009 (H1N1) influenza A virus, PLoS Currents Influenza, 2010

[87] Michalewicz, Z., Genetic Algorithms + Data Structures = Evolution Programs, Springer, 1996

[88] Mikołajczyk, K., Leibe, B., Schiele, B., Local features for object class recognition, IEEE International Conference on Computer Vision, vol. 2, 1792 - 1799, 2005

[89] Morgan, B., Interest Point Detection for Reconstruction in High Granularity Tracking Detectors, Journal of Instrumentation, vol. 5, 2010

[90] Nelson, M., Spiro, D., Wentworth, D., Fan, J., Beck, E., St. George, K., Ghedin, E., Halpin, R., Bera, J., Hine, E., Proudfoot, K., Stockwell, T., Lin, X., Griesemer, S., Bose, Jurgens, L., Kumar, S., Viboud, C., Holmes, E., Henrickson, K., The early diversification of influenza A/H1N1pdm. PLoS Currents Influenza, 2009

[91] Nitesh V. Chawla, N.V., Data Mining for Imbalanced Datasets: An Overview, Data Mining and Knowledge Discovery Handbook, 853-867, 2005

[92] Obuchi, M., Toda, S., Tsukagoshi, H., Oogane, T., et al., Molecular Analysis of Genome of the Pandemic Influenza A(H1N1) 2009 Virus Associated with Fatal Infections in Gunma, Tochigi, Yamagata, and Yamaguchi Prefectures in Japan during the First Pandemic Wave, Japanese Journal of Infectious Diseases, vol. 65, 363-367, 2012

[93] Piralla, A., Pariani, E., Rovida, F., Campanini, G., Muzzi, A., et al., Segregation of Virulent Influenza A(H1N1) Variants in the Lower Respiratory Tract of Critically Ill Patients during the 2010–2011 Seasonal Epidemic, PLoS ONE, vol. 6, 2011

[94] Pitas, I., Venetsanopoulos, A.N., Order statistics in digital image processing, Proceedings of IEEE, vol. 80, 1893-1921, 1992

[95] Płoński P., Radomski J.P., Quick Path Finding – Quick Algorithmic Solution for Unambiguous Labeling of Phylogenetic Tree Nodes, Computational Biology and Chemistry, vol. 34, 300–307, 2010

[96] Płoński P., Radomski J.P., Translacja drzew filogenetycznych – metoda QPF, Zeszyty Naukowe Wydziału Elektroniki Telekomunikacji i Informatyki Politechniki Gdańskiej, 2011

[97] Płoński P., Identification of Key Risk Factors for the Polish State Fire Service with Cascade Step Forward Feature Selection, IEEE Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, vol. 2, 369-373, 2014

[98] Płoński, P., Gradkowski,W., Jednoróg, K., Marchewka, A., Bogorodzki, P., Dealing with heterogeneous multi-site neuroimaging data sets: a study on discrimination of children dyslexia, In: Slezak, D., et al. (Eds.) Brain Informatics and Health, 2014, Lecture Notes in Artificial Intelligence, vol. 8609, 2014, 471-480

[99] Płoński, P., Gradkowski, W., Altarelli, I., Monzalvo, K., van Ermingen-Marbach, M., Grande, M., Heim, S., Marchewka, A., Bogorodzki, P.,Ramus, F., Jednoróg, K., Multiparameter classification approach to the neuroanatomical basis of developmental dyslexia, w przygotowaniu

[100] Płoński, P., Stefan, D., Sulej, R., Zaremba, K., Image Segmentation in Liquid Argon Time Projection Chamber Detector, Lecture Notes in Computer Science, vol. 9119, 606-615, 2015

[101] Płoński, P., Stefan, D., Sulej, R., Zaremba, K., Electron Neutrino Classification in Liquid Argon Time Projection Chamber Detector, Advances in Intelligent Systems and Computing, w druku,(dostępne również przez arXiv:1505.00424), 2015

[102] Płoński, P., Zaremba, K., Visualizing Random Forest with Self-Organising Map, Lecture Notes in Computer Science, vol. 8468, 63-71, 2014

[103] Płoński, P., Zaremba, K., Full and Semi-Supervised k-Means Clustering Optimised by Class Membership Hesitation, Lecture Notes in Computer Science, 7824, 218-225, 2013

[104] Płoński, P., Zaremba, K., Hesitant Neural Gas for Supervised and Semi-supervised Classification, Lecture Notes in Artificial Intelligence, 7894, 474-482, 2013

[105] Płoński, P., Zaremba, K., Self-Organising Maps for Classification with Metropolis-Hastings Algorithm for Supervision, Lecture Notes in Computer Science, 7665, 149-156, 2012

[106] Płoński, P., Zaremba, K., Improving Performance of Self-Organising Maps with Distance Metric Learning Method, Lecture Notes in Computer Science, 7267, 169-177, 2012

[107] Quinlan, J. R., Induction of Decision Trees, Machine Learning, vol. 1, 81-106, 1986

[108] Radomski, J.P., Płoński, P., Zagórski-Ostoja, W., The hemagglutinin mutation E391K of pandemic 2009 influenza revisited, Molecular phylogenetics and evolution, 70, 29-36, 2014

[109] Rahul S. Desikan, Florent Segonne, Bruce Fischl, Brian T. Quinn, Bradford C. Dickerson, Deborah Blacker, Randy L. Buckner, Anders M. Dale, R. Paul Maguire, and Bradley T. Hyman. An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. Neuroimage, 31(3):968–980, 2006

[110] Reuter, M., Schmansky, N.J., Rosas, H.D., Fischl, B.: Within-subject template estimation for unbiased longitudinal image analysis. NeuroImage, vol. 61, 1402–1418, 2012

[111] Richlan, F., Kronbichler, M., Wimmer, H., Structural abnormalities in the dyslexic brain: a meta-analysis of voxel-based morphometry studies, Human Brain Mapping, vol. 34, 3055-3065, 2013

[112] Rokach, L., Ensemble-based classifiers, Artificial Intelligence Review, vol. 33, 1-39, 2010

[113] Rondina, J.M., Hahn, T., de Oliveira, L., Marquand, A.F., Dresler, T., Leitner, T., Fallgatter, A.J., Shawe-Taylor, J., Mourao-Miranda, J., SCoRS—A Method Based on Stability for Feature Selection and Apping in Neuroimaging, IEEE Transactions on Medical Imaging, vol. 33, 85-98, 2014

[114] Rosas, H.D., Liu, A.K., Hersch, S., Glessner, M., Ferrante, R.J., Salat, D.H., van der Kouwe, A., Jenkins, B.G., Dale, A.M., Fischl, B.: Regional and progressive thinning of the cortical ribbon in huntington's disease, Neurology, vol. 58, 695–701, 2002

[115] Rubbia, C., The Liquid-Argon Time Projection Chamber: a new concept for neutrino detectors, CERN Report, 1977

[116] Rutkowski, L., Metody i techniki sztucznej inteligencji, Wydawnictwo Naukowe PWN, 2011

[117] Saitou N., Nei M., The neighbor-joining method: a new method for reconstructing phylogenetic trees, Molecular Biology and Evolution, vol. 4, 406-425, 1987

[118] Somol, P., Novovičová, J., Evaluating the stability of feature selectors that optimize feature subset cardinality, Proceedings of Structural, Syntactic and Statistical Pattern Recognition, 966-976, 2008

[119] Stefan, D., Sulej, R., et al., Precise 3D Track Reconstruction Algorithm for the ICARUS T600 Liquid Argon Time Projection Chamber Detector, Advances in High Energy Physics, vol. 2013

[120] Strengell, M., Ikonen, N., Ziegler, T., Julkunen, I., Minor Changes in the Hemagglutinin of Influenza A(H1N1)2009 Virus Alter Its Antigenic Properties, PLoS ONE, 2011

[121] Sulej, R., Sterile neutrino search with the ICARUS T600 in the CNGS beam, XV Workshop on Neutrino Telescopes, PoS Neutel, 2013

[122] Tadusiewicz, R., Korohoda, P., Komputerowa analiza i przetwarzanie obrazów, Wydawnictwo Fundacji Postępu Telekomunikacji 1997

[123] Taigman, Y.,Yang. M., Ranzato, M.'A., Wolf, L., DeepFace: Closing the Gap to Human-Level Performance in Face Verification, Conference on Computer Vision and Pattern Recognition (CVPR), 2014

[124] The ICARUS Collaboration, Design, construction and tests of the ICARUS T600 detector, Nuclear Instruments and Methods in Physics Research, vol. A527, 2004

[125] The ICARUS Collaboration, The trigger system of the ICARUS experiment for the CNGS beam, Journal of Instrumentation, vol. 9, 2014

[126] The LBNE Collaboration, The Long-Baseline Neutrino Experiment - Exploring Fundamental Symmetries of the Universe, FERMILAB-PUB-14-022, arXiv:1307.7335, 2014

[127] The MicroBooNE Collaboration, Proposal for a New Experiment Using the Booster and NuMI Neutrino Beamlines: MicroBooNE, FERMILAB-PROPOSAL-0974, 2007

[128] Thomas P. Minka, T.P., A comparison of numerical optimizers for logistic regression, Microsoft Technical Report, 2003

[129] Vafaie, H., De Jong, K., Genetic algorithms as a tool for feature selection in machine learning, Proceedings of International Conference on Tools with Artificial Intelligence, 200 - 203, 1992

[130] Venkataraman, A., Kubicki, M., Westin, C.-F., Golland, P., Robust Feature Selection in Resting-State fMRI Connectivity Based on Population Studies, Conference on Comput Vision and Pattern Recognition, 63-70, 2010

[131] Woźniak, M., Krawczyk, B., Combined classifier based on feature space partitioning, International Journal of Applied Mathematic and Computer Science, vol. 22, 855-866, 2012

[132] Xiao Han, Jorge Jovicich, David Salat, Andre van der Kouwe, Brian Quinn, Silvester Czanner, Evelina Busa, Jenni Pacheco, Marilyn Albert, Ronald Killiany, Paul Maguire, Diana Rosas, Nikos Makris, Anders Dale, Bradford Dickerson, and Bruce Fischl. Reliability of MRI-derived measurements of human cerebral cortical thickness: The effects of field strength, scanner upgrade and manufacturer. NeuroImage, vol. 32, 180–194, 2006

[133] Zucknick, M., Richardson, S., Stronach, E.A., Comparing the characteristics of gene expression profiles derived by univariate and multivariate classification methods, Statistical Applications in Genetics and Molecular, vol. 7, 2008