UNIWERSYTET RZESZOWSKI
Kolegium Nauk Przyrodniczych

Daniel Czyż

Nr albumu: 106530

Kierunek: Informatyka

Przewidywanie remodelingu oskrzeli w oparciu o dane mikromacierzowe

Praca magisterska

Praca wykonana pod kierunkiem

Dr hab. Jan Bazan, prof. UR

Rzeszów, 2023

STRESZCZENIE

Niniejsza praca prezentuje eksplorację danych mikromacierzowych DNA pod kątem zbadania, czy jest możliwe przewidywanie przypadłości remodelingu oskrzeli u osób cierpiących na astmę na podstawie ekspresji genów. W pracy zbadano sześć argumentów decyzyjnych, stanowiących cechy kliniczne powiązane z występowaniem remodelingu oskrzeli. Dla każdego argumentu decyzyjnego przetestowano dziesięć różnych algorytmów klasyfikacyjnych i trzy metody selekcji cech. Celem było utworzenie modelu lub wielu modeli dobrej jakości, poprawnie identyfikujących pacjentów pod kątem remodelingu oskrzeli. Rezultaty eksperymentów okazały się mieszane. Jednak udało się uzyskać model dobrze radzący sobie z zadanym problemem w przypadku jednej z kolumn decyzyjnych. W pracy zbadano również na przykładach drzew decyzyjnych, jakie „spoty" mikromacierzy decydowały o klasyfikacji obiektów do poszczególnych klas, zwracając uwagę na obiekty błędnie zidentyfikowane. Przyglądnięto się w pracy również przyszłym zastosowaniom pozyskanych informacji wskutek eksploracji danych mikromacierzowych i jakie problemy pomagałyby rozwiązywać.

Prediction of bronchial remodeling based on microarray data

This paper presents an exploration of DNA microarray data to investigate whether it is possible to predict the occurrence of bronchial remodeling in asthma patients based on gene expression. In this study, six decision arguments, representing clinical features associated with the occurrence of bronchial remodeling, were examined. Ten different classification algorithms and three feature selection methods were tested for each decision argument. The results of the experiments turned out to be mixed. However, it was possible to obtain a model that handled the problem well for one of the decision columns. In the paper, it was also examined, with the use of decision tree examples, what "spots" of microarrays decided the classification of objects into different classes paying attention to the objects that were misidentified. The paper also looked at future applications of the information gained as a result of microarray data exploration and what problems it would help solve.

Spis treści

1. Wstęp

Rozdział obejmuje wyjaśnienie problematyki rozpatrywanej w pracy, plan pracy oraz uzasadnienie aktualności i innowacyjności tematyki.

1.1. Cel i teza pracy

Celem pracy magisterskiej jest wykorzystanie możliwości uczenia maszynowego w kontekście klasyfikacji, operując na danych mikromacierzowych DNA i powiązanych z nimi danych o cechach klinicznych pacjentów z astmą. Cechy kliniczne, jakie wzięto pod uwagę, są powiązane z występowaniem remodelingu oskrzeli. Tezą w pracy jest twierdzenie, iż eksploracja danych mikromacierzowych DNA pozwoli na utworzenie modelu lub wielu modeli dobrej jakości, wykazując, że istnieje informacja płynąca z ekspresji genów zawarta w mikromacierzy DNA, która w dobrym stopniu wyjaśnia cechy kliniczne, wskazujące na występowanie remodelingu oskrzeli. Dowodziłoby to tym samym, że przewidywanie przypadłości remodelingu oskrzeli u osób chorych na astmę jest możliwe.

1.2. Uzasadnienie tematyki pracy

Rozpatrywany w pracy problem przewidywania konkretnej przypadłości chorobowej na podstawie ekspresji genów jest jak najbardziej aktualny i innowacyjny. Wraz z rozwojem uczenia maszynowego, istnieją coraz większe możliwości poszerzania wiedzy na wielu obszarach naukowych, między innymi w medycynie. Łącząc technologię pozyskiwania informacji na temat ekspresji genów wraz z możliwością zapisania i przechowywania ich w postaci danych cyfrowych, otwierają się drzwi do nowych odkryć.

Eksploracja danych mikromacierzowych DNA pozwoli na przewidywanie z dużym prawdopodobieństwem różnych chorób uwarunkowanych genetycznie dużo wcześniej, zanim pojawią się pierwsze objawy. Wiedza o zwiększonej możliwości wystąpienia konkretnych chorób pozwoliłaby pacjentom na lepsze przygotowanie się, zaniechanie choroby odpowiednią profilaktyką lub zastosowanie odpowiedniego leczenia w odpowiednim czasie.

1.3. Plan pracy

Pierwszy rozdział poruszony w pracy skupia się na głównych zagadnieniach teoretycznych powiązanych z rozpatrywanym problemem. Wyjaśnia mikromacierze DNA, czym jest remodeling oskrzeli oraz jak się dokonuje klasyfikacji danych z wykorzystaniem uczenia maszynowego. Drugi rozdział opisuje metodykę, jaką zastosowano w przeprowadzonych eksperymentach badawczych. Opisuje między innymi zastosowane czynności przygotowujące dane, metody selekcji cech i metody ewaluacji jakości powstałych klasyfikatorów. Trzeci rozdział prezentuje rezultaty przeprowadzonych eksperymentów wraz z komentarzem odnośnie obserwacji. Zawiera wyniki klasyfikacji dla wielu argumentów decyzyjnych powiązanych z problemem pracy, wykorzystując wiele algorytmów klasyfikacyjnych oraz metod selekcji cech z różnych rodzin. Obrazuje przykładowe obiekty błędnie rozpoznane na drzewach decyzyjnych w celu zidentyfikowania powodu ich błędnej klasyfikacji. Kolejny rozdział skupia się na podsumowaniu pozyskanych informacji w trakcie przeprowadzonych eksperymentów badawczych i ocenie uzyskanego efektu. Ostatni rozdział zawiera informacje o wykorzystanych technologiach w trakcie części praktycznej wykonanej na potrzeby opisywanej pracy magisterskiej.

2. Podstawy teoretyczne

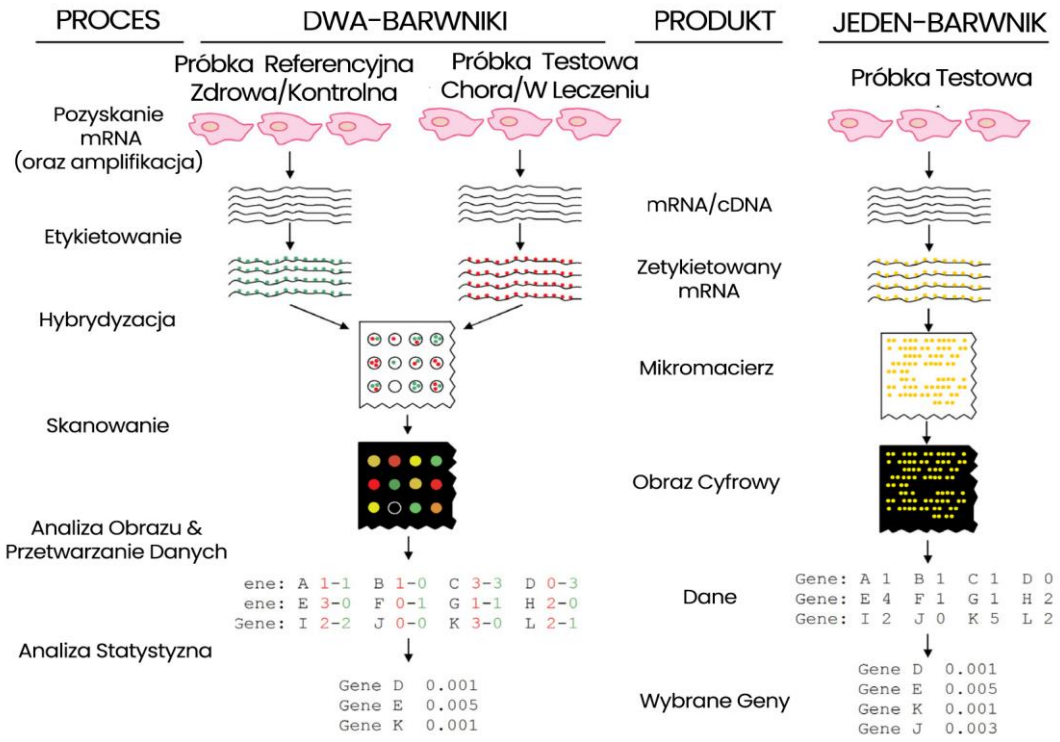
Rozdział obejmuje wyjaśnienie bazowych pojęć i zagadnień leżących u podstawy pisanej pracy magisterskiej.

2.1. Mikromacierz DNA

W 2001 roku dokonano sekwencjonowania ludzkiego genomu, co miało ogromny wpływ na świat medycyny [1,2]. Dzięki temu odkryciu możliwe stało się lepsze diagnozowanie chorób poprzez poznanie genów, które są z nimi związane, co z kolei pozwala na skuteczniejsze leczenie. Poznanie genomu człowieka dostarczyło naukowcom wiele informacji na temat ewolucji człowieka oraz przyniosło korzyści w dziedzinach takich jak biologia, genetyka czy biomedycyna. W artykule [3] opisano technologię mikromacierzy DNA oraz przedstawiono typowe etapy procedury laboratoryjnej wykorzystującej tą technikę.

Mikromacierz DNA to rodzaj pojemnika, który unieruchamia cząsteczkę DNA, komplementarne DNA lub oligonukleotydy do hybrydyzacji z cząsteczką DNA oznaczoną do analizy. Pojemnik taki wykonany jest z nylonu, szkła lub silikonu. Mikromacierze DNA umożliwiają pomiar poziomu ekspresji dziesiątek tysięcy genów jednocześnie i wyrazić je za pomocą ilościowych wartości, co jest przydatne w diagnostyce chorób, identyfikacji guzów, wyborze najlepszego leczenia i wykrywaniu mutacji. Aby uzyskać poziom ekspresji, należy porównać DNA testowej próbki z referencyjnym DNA kontrolnym, zwanym "kontrolą danych" (Rysunek 1) [3]. Po izolacji mRNA z obu tkanek, konieczne jest uzyskanie odpowiedniego cDNA, oznaczenie go fluoroforem Cy3 dla próbki eksperymentalnej (zielony kolor) i Cy5 dla próbki kontrolnej (czerwony kolor), oraz przeprowadzenie hybrydyzacji, co skutkuje powstaniem matrycy z wieloma kolorowymi plamami. Kolor czerwony wskazuje, że dany gen (plamka) ulega większej ekspresji w chorej próbce (eksperymentalnej). Kolor zielony oznacza, że dany gen jest bardziej wyrażony w próbce zdrowej (kontrolnej). Żółte plamki wskazują, że gen jest tak samo wyrażony w zdrowej i chorej próbce. Mikromacierze DNA to technologia, która, umożliwia szybkie wykrycie chorób oraz dobór najlepszych terapii dla pacjentów. Mikromacierze DNA pozwalają także na poznanie interakcji między genami w różnych warunkach, co może prowadzić do rewolucyjnych odkryć w dziedzinach takich jak

biologia, medycyna czy informatyka. Niestety, wadą tego typu danych jest ogromna ilość informacji, jaką trzeba przeanalizować, a także ograniczona liczba próbek z różnych chorób [4].



Rysunek 1. Technologia mikromacierzy DNA [3].

2.2. Remodeling oskrzeli

W większości publikacji naukowych, remodeling oskrzeli (dróg oddechowych) jest szeroko definiowany jako wszelkie zmiany w składzie, rozmieszczeniu, grubości, masie lub objętości oraz liczbie elementów strukturalnych w ścianie dróg oddechowych pacjentów z astmą w porównaniu z osobami zdrowymi. Zmiany te występują w różnych tkankach, takich jak nabłonek, tkanka międzykomórkowa, komórki mięśni gładkich, tkanka nerwowa i naczynia oskrzelowe. Mimo że remodeling dróg oddechowych może występować w innych przewlekłych chorobach płuc, takich jak przewlekła obturacyjna choroba płuc (POChP), pewne zmiany w strukturze dróg oddechowych wydają się odróżniać astmę od POChP. Istnieje też hipoteza, że remodeling dróg oddechowych jest pierwotnym procesem w patogenezie astmy, jeśli jest wynikiem zaburzonego rozwoju płuc [5].

Cechy kliniczne pacjentów, które posłużyły jako atrybuty decyzyjne w rozpatrywanym problemie klasyfikacji, tj. przewidywania remodelingu oskrzeli / dróg oddechowych u osób z astmą:

- Kolagen I % powierzchni oraz kolagen I siła – kolagen to podstawowy składnik macierzy zewnątrzkomórkowej dróg oddechowych i odgrywa kluczową rolę w ich właściwościach mechanicznych. Nieprawidłowe gromadzenie się kolagenu w drogach oddechowych związane jest z rozwojem i progresją chorób. W chorobach dróg oddechowych, zmiany w strukturze tkanki (remodeling) są związane z gromadzeniem się składników ECM, takich jak kolagen, fibronektyna i proteoglikany, w i wokół nabłonka i naczyń krwionośnych [6].
- Wall area ratio RB1, wall area ratio RB10 oraz wall thichness/airway diameter ratio RB1 – badania wykazały, że zgrubienie określonych oskrzeli w płucach (oskrzela koniuszkowego prawego górnego płata – RB1 i oskrzela podstawnego tylnego prawego dolnego płata - RB10) jest powiązane z różnymi problemami oddechowymi w astmie, takimi jak ograniczenie przepływu powietrza, nadreaktywność dróg oddechowych i uwięźnięcie powietrza podczas wydechu. Dodatkowo, stwierdzono, że remodeling oskrzeli RB1 koreluje z proksymalnymi drogami oddechowymi w ciężkiej astmie [7].
- Średnia harmoniczna liniowa – odnosi się do grubości błony podstawnej nabłonka dróg oddechowych. Remodeling dróg oddechowych w astmie charakteryzuje się pogrubieniem siateczkowej błony podstawnej (RBM), prawdopodobnie związanym ze zmianami strukturalnymi i funkcjonalnymi nabłonka [25].

2.3. Uczenie maszynowe

Uczenie maszynowe (ang. Machine learning, ML) to podkategoria sztucznej inteligencji, która pozwala maszynom na naśladowanie inteligentnych zachowań człowieka. Systemy sztucznej inteligencji wykorzystują złożone algorytmy do rozwiązywania problemów w sposób zbliżony do ludzkiego myślenia. Celem uczenia

maszynowego jest utworzenie modeli komputerowych, które wykazują inteligentne zachowania, takie jak ludzie, gdzie w tym celu stosuje się nauczanie poprzez doświadczenie. Proces ten zaczyna się od zbierania i przygotowywania danych treningowych, takich jak liczby, zdjęcia, teksty, transakcje bankowe, czy na przykład dane mikromacierzowe DNA wykorzystane w pisanej pracy. Im więcej danych, tym lepszy efekt finalny programu [8, 22].

Programiści wybierają model uczenia maszynowego i dostarczają dane, na których model będzie trenowany. W procesie uczenia, model komputerowy poszukuje wzorców lub dokonuje przewidywań. Niektóre dane są wyodrębniane z danych treningowych i wykorzystywane jako dane ewaluacyjne, które służą do oceny dokładności modelu w przypadku prezentowania mu nowych danych. W ten sposób powstaje model, który może być użyty w przyszłości z różnymi nowymi zestawami danych [8].

W uczeniu maszynowym wyróżnia się cztery podkategorie:

- Nadzorowane modele uczenia maszynowego, które są trenowane na etykietowanych danych. Na przykład, algorytm może być szkolony z oznakowanymi zdjęciami określonych obiektów, aby nauczyć się automatycznie rozpoznawać obiekty na zdjęciach.
- Uczenie maszynowe bez nadzoru, gdzie program szuka wzorców w nieoznakowanych danych. Ten rodzaj uczenia może pomóc w identyfikowaniu trendów i wzorców, których ludzie sami nie byliby w stanie zauważyć.
- Uczenie maszynowe z wykorzystaniem wzmocnień, które szkoli maszyny poprzez nagradzanie ich za podejmowanie dobrych decyzji. Na przykład, taki system może szkolić autonomiczne pojazdy, nagradzając maszynę za podejmowanie właściwych decyzji i pomagając jej nauczyć się, jakie decyzje podejmować w przyszłości [8].
- Uczenie maszynowe częściowo nadzorowane należy do dziedziny uczenia maszynowego, w której posiadane dane wejściowe, charakteryzują się tym, że tylko część z nich jest opatrzona etykietami. Uczenie częściowo nadzorowane łączy techniki uczenia nadzorowanego i nienadzorowanego. W takim podejściu trenowany jest początkowy model przy użyciu kilku oznaczonych próbek, a następnie jest wielokrotnie stosowany do większego zestawu nieoznaczonych

danych. W przeciwieństwie do uczenia bez nadzoru, uczenie częściowo nadzorowane ma zastosowanie do szeregu problemów, w tym klasyfikacji, regresji, grupowania i asocjacji. W przeciwieństwie do uczenia nadzorowanego, metoda ta wykorzystuje niewielką ilość oznaczonych danych wraz z dużą ilością nieoznaczonych danych, co zmniejsza potrzebę ręcznego etykietowania i oszczędza czas na przygotowanie danych [26].

2.4. Predykcja z wykorzystaniem uczenia maszynowego

Do predykcji odpowiedzi w kontekście określonego problemu wykorzystuje się algorytmy klasyfikacyjne oraz algorytmy regresyjne, które należą do grupy nadzorowanego uczenia maszynowego (ang. supervised ML). W pracy skupiono się na metodach klasyfikacyjnych, które stanowią podstawę wykonanych eksperymentów.

Klasyfikacja jest stosowana do problemów predykcji, wtedy, gdy zmienna decyzyjna jest dyskretna np. tak lub nie, 0 lub 1. W sytuacji, gdy zmienna decyzyjna jest ciągła, czyli przyjmuje wartość rzeczywistą z pewnego przedziału stosuje się regresję [22].

Oba podejścia, na ogół, wymagają wykonania tych samych kroków, lecz różnią się zwróconym wynikiem i wykorzystanymi metrykami jakości. Problem predykcji z wykorzystaniem uczenia maszynowego można opisać następującymi krokami:

1) Zebranie danych, ważne jest, aby dane pochodziły z rzetelnych źródeł, jakość danych bazowych będzie wpływała na jakość utworzonego modelu.
2) Przygotowanie danych. Może obejmować czynności takie jak: usuwanie obiektów z pustymi polami, usuwanie duplikatów, konwersja typu danych w kolumnach, progowanie, normalizacja, podział danych na zestaw trenujący, ewaluacyjny i testowy itp.
3) Dobranie odpowiedniego algorytmu klasyfikacyjnego tak, aby był kompatybilny z przygotowanymi danymi i był odpowiedni do wykonywanego zadania.
4) Uczenie / trenowanie modelu. Model dokonuje analizy przygotowanych danych trenujących tak, aby wykryć pewnego rodzaju wzorce, w sposób zależny od wybranego algorytmu i określonych hiperparametrów wejściowych.

5) Ewaluacja modelu, czyli dokonanie oceny jakości wytrenowanego modelu. Dokonywana jest za pomocą danych ewaluacyjnych (niebiorących udziału w procesie nauczania).
6) Dostosowywanie hiperparametrów. Powtarzanie punktu 4 i 5 z różnymi kombinacjami parametrów wejściowych tak, aby maksymalizować metryki jakości porównując z poprzednimi wersjami modelu. Należy jednak korzystać z odpowiedniej metodyki, aby uniknąć efektu przetrenowania (ang. overfitting).
7) Testowanie finalnej wersji modelu z nowymi zestawami danych, czyli danych testowych [9].

3. Metodyka przeprowadzonych badań

Na potrzeby badań wykorzystano następujące dane:

- plik geny.csv zawierający dane mikromacierzowe DNA (wartości ekspresji genów na poszczególnych „spotach") osób cierpiących na astmę, kod identyfikujący pacjentów, oraz informacja o pochodzeniu próbki, tj. czy dane reprezentują wyniki badań z krwi lub szczotki pacjenta.
- plik cechy kliniczne.csv zawierający cechy kliniczne pacjentów o wartościach numerycznych, które posłużyły za atrybuty decyzyjne w przeprowadzonych badaniach. Plik zawiera również kody pacjentów odpowiadające tym samym kodom z pliku geny.csv, oraz informację o pochodzeniu próbki.

W dalszej części rozdziału zostanie opisana krok po kroku pełna metodyka, jaką zastosowano w przeprowadzonych eksperymentach. Badania skupiały się na utworzeniu i maksymalizacji jakości modeli klasyfikujących w kontekście przewidywania remodelingu oskrzeli u pacjentów z astmą na podstawie danych mikromacierzowych DNA (ekspresji genów).

3.1. Formatowanie danych

Wartości kolumny definiujące pochodzenie próbki przekonwertowano z wartości tekstowych na wartości numeryczne, tj. krew = 1, szczotka = 0. Takie dane wyseparowano do osobnych plików, tj. powstał plik geny_k.csv i cechy_k.csv (zawierające tylko próbki z krwi) oraz geny_s.csv i cechy_s.csv (zawierające tylko próbki z szczotki). Przygotowane pliki wczytywano do zmiennych w programie. W zależności od argumentu decyzyjnego, na którym prowadzono analizę, usuwano ze zbioru obiekty, które nie posiadały dla niego wartości. Jeżeli dany argument decyzyjny przyjmował zróżnicowane wartości (brak podziału na klasy), to dokonywano progowania na podstawie mediany, aby uzyskać zbalansowany podział klas. Gdy dany argument decyzyjny posiadał zdefiniowane klasy, lecz były niezbalansowane w dalszej części w procesie wstępnego przetwarzania danych stosowano metody balansujące klasy w zbiorze.

3.2. Walidacja krzyżowa

Walidacja krzyżowa (ang. Cross Validation, CV) w przypadku wykonywanych eksperymentów w pracy jest bardzo ważnym krokiem do uzyskania wiarygodności wyników. Dla małych zbiorów danych, takich jak dane mikromacierzowe DNA, sugerowanym rozwiązaniem jest zastosowanie walidacji krzyżowej Leave-one-out Cross Validation (LOOCV). Metoda LOOCV polega na iteracyjnym trenowaniu i testowaniu modelu na całym przekazanym zbiorze danych, przy czym w każdej iteracji wybierana jest tylko jedna obserwacja jako zbiór testowy (ang. leave-one-out), a pozostałe obserwacje są wykorzystywane jako zbiór uczący. Wynik każdej iteracji jest następnie uśredniany, aby obliczyć ogólny wynik [11]. LOOCV jest dosyć kosztowną metodą obliczeniowo, ponieważ wymaga trenowania i testowania modelu tyle razy, ile jest obserwacji w zbiorze danych. Jednakże, ma ona zaletę, zapewnia, że każda obserwacja zostanie użyta zarówno do trenowania, jak i testowania modelu. Dlatego LOOCV może być użyteczna w przypadkach, gdy zbiór danych jest bardzo mały lub gdy każda obserwacja jest cenna gdyż maksymalizujemy wykorzystanie dostępnych informacji, które są zawarte w badanym zbiorze danych [13][14].

Na potrzeby przeprowadzenia eksploracji z możliwością dostosowywania hiperparametrów testowanych w pracy metod selekcji cech pojawiła się konieczność zastosowania zagnieżdżonej pętli walidacji krzyżowej (zastosowano LOOCV). Taki mechanizm został zastosowany w celu uniknięcia sytuacji, wykorzystania tzw. „wiedzy z przyszłości" [27, 28].

Dla każdej iteracji pętli głównej (zewnętrznej) dla danych treningowych przeprowadzana była zagnieżdżona walidacja krzyżowa (pętla wewnętrzna), uruchamiana tyle razy, ile istnieje możliwych kombinacji różnych zestawień wartości hiperparametrów, jakie przekazano do ewaluacji. Przykładowo, jeżeli przeszukiwalibyśmy wartość tylko parametru '*k*' dla trzech wartości (np. 1,2 i 3), zagnieżdżona pętla wykonałaby się trzy razy. Jeżeli przeszukiwalibyśmy wartość parametru '*k*' dla trzech wartości (np. 1,2 i 3), oraz parametr '*score_func*' dla dwóch wartości (np. '*chi2*' i '*f_classif*'), zagnieżdżona pętla wykonałaby się sześć razy. Każde uruchomienie wewnętrznej pętli walidacji krzyżowej zwraca jako wynik uzyskaną jakość (np. dokładność klasyfikacji) i konfigurację wykorzystanych hiperparametrów. Spośród

wszystkich otrzymanych wyników wybierana jest ta konfiguracja, która otrzymała najkorzystniejszą jakość klasyfikacji. Wybrana „zwycięska” konfiguracja jest stosowana następnie w iteracji zewnętrznej pętli walidacji krzyżowej.

Zewnętrzna pętla walidacji krzyżowej została zaimplementowana ręcznie. Aby wdrożyć mechanizm wewnętrznej pętli walidacji krzyżowej, wykorzystano metodę GridSearchCV, zawartej w bibliotece scikit-learn. W pracy do metody GridSearchCV jako estymator przekazywano obiekt typu „Pipeline” z biblioteki imbalanced-learn, który stanowi zestaw wcześniej zdefiniowanych czynności wykonujących się kolejno po sobie (takich jak skalowanie danych, selekcja cech, równoważenie klas decyzyjnych, trenowanie i testowanie za pomocą wskazanego modelu klasyfikacyjnego). Zastosowanie obiektu „Pipeline” zapewnia poprawne wykonanie wszystkich wymaganych czynności wstępnego przetwarzania danych, przeprowadzonych osobno dla każdej pojedynczej iteracji walidacji krzyżowej [20].

Ponieważ w zewnętrznej pętli (LOOCV) zawsze istnieje tylko jeden obiekt w zbiorze testowym, nie jest możliwe obliczenie metryk jakości, takich jak precyzja czy czułość, osobno dla każdej iteracji z powodu błędu dzielenia przez zero. Dlatego zastosowano obliczenie metryk jakości w ujęciu całościowym po przeprowadzeniu wszystkich iteracji [20].

3.3. Skalowanie danych

Stosując uczenie maszynowe, często wykorzystywane są metody skalowania cech na etapie wstępnego przetwarzania danych. W wyniku skalowania wszystkie elementy w zbiorze danych są sprowadzone do jednej skali. Pozwala to zapobiec występowaniu wartości odstających, co na ogół polepsza jakość klasyfikacji. Ze względu na charakterystykę zbiorów mikromacierzowych DNA tj. wysoka wariancja cech, sugerowane jest zastosowanie skalowania cech przed ich eksploracją. W pracy w tym celu zastosowano metodę Min-Max. Dla każdej z cech w zbiorze danych, jej minimalna wartość z pośród wszystkich obiektów jest podmieniana na 0, a maksymalna wartość jest podmieniana na 1, zaś każda inna wartość jest przekształcana na liczbę z przedziału od 0 do 1 [10]. Metoda ta jest opisana następującym wzorem (1):

$$X = \frac{X_i - X_{min}}{X_{max} - X_{min}} \quad (1)$$

W podanym wzorze X_i reprezentuje oryginalną wartość cechy, X reprezentuje znormalizowaną cechę, X_{min} jest minimalną wartością, a X_{max} jest maksymalną wartością cechy w oryginalnym zbiorze przed skalowaniem [10].

Aby potwierdzić poprawność założenia, że przeskalowanie danych poprawi jakość predykcji w rozpatrywanym problemie, przeprowadzono testy, przekazując do wielu algorytmów klasyfikacyjnych najpierw oryginalne dane, a później znormalizowane dla tych samych hiperparametrów wejściowych. Testy powtórzono dla różnych argumentów decyzyjnych. Wyniki wskazywały jasno, że przy zastosowaniu normalizacji danych, metryki jakości klasyfikacji osiągały wyższe wartości.

3.4. Selekcja cech

Metody selekcji cech są stosowane w celu zmniejszenia wymiarowości zbioru danych wejściowych poprzez usunięcie niepotrzebnych i niepowiązanych cech. W przypadku danych mikromacierzowych DNA, gdzie przestrzeń wejściowa jest bardzo duża, selekcja cech jest konieczna. W obrębie przeprowadzonych badaniach wykorzystano trzy metody selekcji cech: SelectKBest, SelectFromModel oraz Recursive feature elimination (RFE). Wszystkie z wykorzystanych metod pochodzą z biblioteki scikit-learn [20].

3.4.1. Metoda SelectKBest

Metoda zachowuje cechy według hiperparametru '*k'* (czyli ilość cech jaką chcemy zachować) najwyżej ocenionych. Metoda ta pozwala na ustalenie parametru '*score_func'*, który stosuje określony sposób wyliczania wyniku cechom w zbiorze wejściowym. Wykorzystane w pracy sposoby obliczania wyniku cech to funkcja '*chi2'* oraz funkcja *'f_classif'* [20].

3.4.2. Metoda SelectFromModel

Metoda *SelectFromModel* to meta-transformator, który może być stosowany z dowolnym estymatorem, który przypisuje znaczenie do każdej cechy za pomocą

określonego atrybutu takiego jak *'coef_'*, *'feature_importances_'* (oznacza to, iż nie każdy estymator będzie z nią współpracował). Cechy uważane są za nieistotne i są usuwane, jeśli odpowiadające im wartości ważności cech są poniżej ustalonego progu. Parametr *'max_features'* może być użyty do ustawienia limitu na liczbę wybieranych cech. Istnieją również wbudowane metody do ustalenia progu, takie jak '*mean*' (średnia), '*median*' (mediana) lub ich zmiennoprzecinkowe wielokrotności, np. *'0,1 * mean'* [20].

W pracy metodę SelectFromModel stosowano z estymatorem RandomForestClassifier.

3.4.3. Rekurencyjna eliminacja cech (metoda RFE)

Metoda RFE (ang. Recursive Feature Elimination) to metoda selekcji cech, która polega na iteracyjnym usuwaniu coraz mniej ważnych cech. Na początku algorytm trenuje estymator na pełnym zbiorze cech, a następnie dla każdej cechy oblicza jej ważność przy użyciu konkretnego atrybutu, np. 'coef_' lub 'feature_importances_' (metoda ta tak samo jak SelectFromModel nie współpracuje z każdym estymatorem). Następnie, cechy o najmniejszej ważności są usuwane z aktualnego zbioru, a proces jest powtarzany rekurencyjnie na przyciętym zbiorze aż do uzyskania pożądanej liczby cech [20].

W pracy, jako estymator dla metody RFE wykorzystano LogisticRegression w dwóch wariantach. Pierwsza wersja wykorzystując metodę regularyzacji *'l1'*, czyli LASSO, oraz w drugiej wersji wykorzystując metodę regularyzacji *'l2'*, czyli RIDGE. Metody zostały wybrane, głównie ze względu na ich popularność przy selekcji cech, na danych mikromacierzowych, zwłaszcza metoda LASSO. Ponieważ biblioteka scikit-learn nie ma bezpośredniej implementacji tych metod dla problemu klasyfikacji, efekt można uzyskać łącząc metodę LogisticRegression (ustawiając w niej odpowiednie hiperparametry) z metodą selekcji cech jak RFE, czy SelectFromModel [29].

3.5. Równoważenie klas decyzyjnych

Równoważenie klas decyzyjnych było stosowane w przypadku tylko jednej z sześciu kolumn decyzyjnych tj. „Kolagen I siła". Kolumna ta zawiera trzy klasy decyzyjne,

gdzie w oryginalnym zbiorze danych wielokrotność wystąpień poszczególnych klas jest wysoce zróżnicowana. Dla pierwszej klasy istnieje 18 obiektów, dla drugiej klasy istnieje 22 obiektów i dla trzeciej istnieje tylko 6 obiektów. Przy klasyfikacji na oryginalnym zbiorze klasa trzecia (ze względu na znacząco mniejszą liczebność) praktycznie nigdy nie była przypisywana do testowanych obiektów. Aby rozwiązać problem braku zróżnicowanych klas zastosowano metodę SMOTE, która jest metodą z kategorii „oversampling". Metoda ta generuje nowe obiekty dla każdej z klas mniejszościowych tak, aby ilość obiektów zgadzała się z klasą o najwyższej liczebności. Nowe obiekty i ich wartości są generowane na podstawie wartości innych obiektów w tej samej klasie. Najpierw wybierany jest losowy obiekt z danej mniejszościowej klasy, następnie znajduje się *k* najbliższych sąsiadów dla tego obiektu. Wybierany jest losowy sąsiad z dostępnych i tworzony jest syntetyczny obiekt w losowo wybranym punkcie pomiędzy tymi dwoma obiektami w przestrzeni cech [23].

3.6. Algorytmy klasyfikujące

Lista algorytmów klasyfikujących (10), które zostały wykorzystane do rozwiązania problemu przewidywania remodelingu oskrzeli u osób chorych na astmę. W pracy wykorzystano ich implementacje pochodzącą z biblioteki scikit-learn.

3.6.1. Random Forest Classifier

Random Forest Classifier to algorytm zespołowy (ang. ensemble). Algorytm zaczyna pracę od utworzenia wielu drzew decyzyjnych. Każde drzewo decyzyjne w losowym lesie jest trenowane na innym podzbiorze danych treningowych. Osiąga się to poprzez proces zwany „bootstrapping" lub „sampling with replacement". Losowe próbki są pobierane z danych treningowych, a każde drzewo decyzyjne jest trenowane na jednej z tych próbek. Podczas budowy każdego drzewa decyzyjnego, wprowadzany jest element losowości. W każdym węźle drzewa, zamiast rozważać wszystkie cechy, wybierany jest losowy podzbiór cech. Pomaga to ograniczyć nadmierne dopasowanie i oddzielić drzewa od siebie. Drzewo jest budowane poprzez wielokrotne partycjonowanie danych w oparciu o wybrane cechy i ich odpowiednie punkty podziału. Kryterium podziału, takie jak „Gini impurity" lub entropia, jest używane do określenia najlepszej cechy i punktu podziału w każdym węźle. Po skonstruowaniu wszystkich

drzew decyzyjnych, prognozy są tworzone poprzez agregację indywidualnych prognoz każdego drzewa. W przypadku zadań klasyfikacyjnych klasa z większością głosów z drzew jest przypisywana jako ostateczna przewidywana klasa [20].

3.6.2. Multi-layer Perceptron

MLP jest rodzajem sieci neuronowej, która składa się z wielu warstw perceptronów. Perceptron jest podstawowym elementem sieci neuronowej, który przetwarza wejście, wykonując na nim operacje matematyczne i generując wyjście. W sieci MLP, perceptrony są zorganizowane w warstwy, zwykle z jedną warstwą wejściową, jedną lub więcej warstw ukrytych oraz jedną warstwą wyjściową. Podczas trenowania sieci MLP, algorytm propagacji wstecznej (backpropagation) jest używany do modyfikowania wag połączeń między perceptronami, aby zmniejszyć błąd prognozowania. Ten proces polega na przekazywaniu informacji o błędzie z warstwy wyjściowej do warstwy wejściowej, zmieniając wagi połączeń w taki sposób, aby zminimalizować błąd. Ostatnia warstwa perceptronów generuje wektor wyjściowy, który reprezentuje prawdopodobieństwa przynależności do poszczególnych klas. Klasa, do której przypisany jest największy prawdopodobieństwo, zostaje ostatecznie wybrana jako przewidywana klasa [20].

3.6.3. Decision Tree Classifier

Algorytm ten tworzy drzewo o strukturze decyzyjnej, które przetwarza przekazane na wejściu cechy tak, aby dokonać predykcji klasy decyzyjnej dla danego obiektu wejściowego. Budowanie drzewa decyzyjnego zaczyna się od korzenia drzewa, gdzie znajdują się wszystkie cechy wejściowe. Następnie, dla każdej cechy algorytm dokonuje podziału danych na podstawie wartości cechy, tworząc poddrzewa dla każdego z tych podziałów. Proces ten jest powtarzany dla każdego poddrzewa, aż do osiągnięcia liści, które reprezentują końcowe klasy decyzyjne [20].

3.6.4. KNeighbours Classifier

Algorytm oblicza odległość między nowymi obiektami przekazanymi na wejściu a każdym elementem w zbiorze danych treningowych. Następnie, wybiera *k* najbliższych

sąsiadów (najczęściej według odległości euklidesowej), i przypisuje im etykietę klasy. Ostateczna decyzja o przypisaniu nowych obiektów do klasy jest podejmowana na podstawie decyzji większościowej spośród *k* najbliższych sąsiadów. Jedną z zalet tego algorytmu jest to, że nie wymaga treningu [20].

3.6.5. GaussianNB

Działanie tego algorytmu polega na redukcji wymiarowości danych wejściowych i wyznaczeniu nowych zmiennych, które najlepiej oddzielają klasy. Algorytm LDA zakłada, że dane wejściowe pochodzą z rozkładu Gaussa, a każda z klas ma swój własny rozkład Gaussa. Na podstawie tych założeń, algorytm wyznacza parametry rozkładów Gaussa dla każdej z klas i oblicza funkcję dyskryminacyjną, która jest funkcją liniową nowych zmiennych wyznaczonych w procesie redukcji wymiarowości. Funkcja ta służy do klasyfikacji nowych danych na podstawie ich wartości w nowych zmiennych. W procesie klasyfikacji nowych danych, algorytm LDA oblicza wartości funkcji dyskryminacyjnej dla każdej z klas i klasyfikuje nowe dane do klasy z najwyższą wartością tej funkcji [20].

3.6.6. Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) to algorytm używany do znalezienia liniowej kombinacji cech, która najlepiej oddziela klasy w zbiorze danych. Metoda działa poprzez rzutowanie danych na przestrzeń o niższym wymiarze, której celem jest maksymalizacja separacji między klasami. Odbywa się to poprzez znalezienie kompletu liniowych dyskryminantów, które skupiają się na maksymalizacji stosunku wariancji międzyklasowej do wariancji wewnątrzklasowej, czyli znajduje kierunki w przestrzeni cech, które najlepiej oddzielają różne klasy danych [20].

3.6.7. Quadratic Discriminant Analysis

Tak jak w przypadku LDA, działanie algorytmu QDA polega na redukcji wymiarowości danych wejściowych. Algorytm QDA zakłada, że dane wejściowe dla każdej klasy pochodzą z rozkładu Gaussa, ale z różnymi macierzami kowariancji. Na podstawie tych założeń, algorytm wyznacza parametry rozkładów Gaussa dla każdej z

klas, czyli średnie wartości i macierze kowariancji i oblicza funkcję dyskryminacyjną, która jest funkcją kwadratową nowych zmiennych wyznaczonych w procesie redukcji wymiarowości. Funkcja ta służy do klasyfikacji nowych danych na podstawie ich wartości w nowych zmiennych. W procesie klasyfikacji nowych danych, algorytm QDA oblicza wartości funkcji dyskryminacyjnej dla każdej z klas i klasyfikuje nowe dane do klasy z najwyższą wartością tej funkcji [20].

3.6.8. Logistic Regression

Algorytm działa poprzez oszacowanie prawdopodobieństwa, z jakim dany obiekt należy do danej klasy. Model LogisticRegression łączy funkcję sigmoidalną z liniową regresją, aby uzyskać te oszacowania prawdopodobieństw. W modelu LogisticRegression, funkcja sigmoidalna służy do przetwarzania liniowej kombinacji cech wejściowych na wartości prawdopodobieństw i zwraca wartości między 0 a 1, co odpowiada prawdopodobieństwu należenia do jednej z klas. W liniowej regresji, modele są dopasowywane do danych poprzez minimalizowanie błędu kwadratowego pomiędzy przewidywaniami a rzeczywistymi wartościami. W przypadku modelu LogisticRegression, minimalizuje się błąd logistyczny, który bierze pod uwagę prawdopodobieństwa przyporządkowane do każdej klasy [20].

3.6.9. C-Support Vector Classification

SVC jest algorytmem należącym do rodziny SVMs (Support Vector Machines). Opiera się na znajdowaniu optymalnej hiperpłaszczyzny separującej klasy. Hiperpłaszczyzna jest wyznaczana w taki sposób, aby odległość między hiperpłaszczyzną a najbliższymi punktami każdej z klas (nazywana marginesem) była maksymalna. Aby to osiągnąć, algorytm korzysta z optymalizacji funkcji celu, która minimalizuje błąd klasyfikacji i maksymalizuje margines. Funkcja celu składa się z dwóch składników - pierwszy składnik odpowiada za maksymalizację marginesu, a drugi składnik odpowiada za minimalizację błędu klasyfikacji [20].

3.6.10. Gradient Boosting Classifier

Metoda opiera się na zasadzie połączenia wielu słabych klasyfikatorów w silny klasyfikator. Na początku, tworzony jest pierwszy słaby klasyfikator, który przewiduje wynik na podstawie pojedynczej cechy wejściowej. Następnie, błędy tego klasyfikatora są analizowane i kolejny klasyfikator jest uczony, aby poprawić te błędy. Kolejne klasyfikatory są dodawane w procesie iteracyjnym, a każdy kolejny jest uczony w taki sposób, aby minimalizować błędy poprzednich klasyfikatorów. Każdy z klasyfikatorów, zwanych drzewami decyzyjnymi, jest uczony na podzbiorze danych treningowych. Warto zaznaczyć, że uczenie Gradient Boosting jest procesem czasochłonnym i złożonym, ponieważ każde kolejne drzewo jest uczone tak, aby dopasować się do błędów poprzednich drzew, co wymaga dostępu do całego zestawu danych treningowych podczas procesu uczenia. Ostateczna decyzja o klasyfikacji nowych danych jest dokonywana na podstawie decyzji większościowej spośród wszystkich drzew decyzyjnych. [20]

3.7. Weryfikacja jakości

Podrozdział ten opisuje, jakie metody ewaluacji jakości utworzonych modeli, zostały wykorzystane podczas przeprowadzonych eksperymentów.

3.7.1. Macierz pomyłek, Dokładność, Precyzja & Czułość

Macierz pomyłek (ang. confusion matrix) to narzędzie stosowane do oceny jakości modeli klasyfikacyjnych. Jest to tabela lub macierz, która przedstawia liczbę poprawnych i błędnych klasyfikacji modelu dla każdej z klas. W przypadku problemów klasyfikacyjnych z dwoma klasami, macierz pomyłek składa się z czterech pól:

- True Positive (TP) - liczba poprawnie zaklasyfikowanych próbek pozytywnych,
- False Positive (FP) - liczba próbek negatywnych błędnie zaklasyfikowanych jako pozytywne,
- False Negative (FN) - liczba próbek pozytywnych błędnie zaklasyfikowanych jako negatywne,
- True Negative (TN) - liczba poprawnie zaklasyfikowanych próbek negatywnych [24].

Tabela 1. Tabelaryczna reprezentacja macierzy pomyłek (confusion matrix).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	True Negative (TN)	False Positive (FP)
Rzeczywista klasa pozytywna	False Negative (FN)	True Positive (TP)

Dokładność (ang. accuracy) to procent poprawnie sklasyfikowanych obserwacji w stosunku do wszystkich obserwacji. Innymi słowy, jest to miara tego, jak dobrze model klasyfikuje dane [24]. Można ją obliczyć ze wzoru (2) wykorzystując macierz pomyłek:

$$(2) \qquad Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Precyzja (ang. precision) to miara tego, jak wiele obserwacji sklasyfikowanych jako pozytywne są rzeczywiście pozytywne [24]. Można ją obliczyć korzystając z macierzy pomyłek i wzoru (3):

$$(3) \qquad Precision = \frac{TP}{TP + FP}$$

Czułość (ang. recall) to miara tego, jak wiele pozytywnych obserwacji zostało sklasyfikowanych poprawnie [24]. Można ją obliczyć korzystając z macierzy pomyłek oraz wzoru (4):

$$(4) \qquad Recall = \frac{TP}{TP + FN}$$

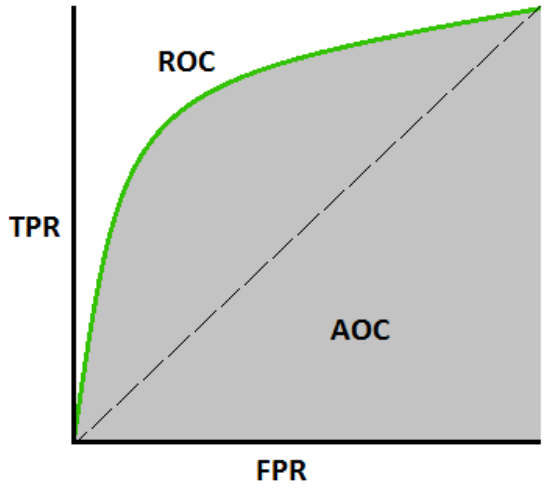
3.7.2. ROC & AUC

Receiver Operating Characteristic (ROC) i Area Under the Curve (AUC) to narzędzia stosowane do oceny jakości modeli klasyfikacji binarnej.

ROC to krzywa wykresu, która przedstawia stosunek liczby poprawnie sklasyfikowanych próbek pozytywnych (True Positive, TP) do liczby niepoprawnie sklasyfikowanych próbek negatywnych (False Positive, FP), dla różnych wartości progu

klasyfikacji. Krzywa ROC jest wykresem zależności miary TPR (True Positive Rate) od FPR (False Positive Rate). TPR to stosunek liczby poprawnie sklasyfikowanych próbek pozytywnych (TP) do łącznej liczby próbek pozytywnych (TP + FN), natomiast FPR to stosunek liczby niepoprawnie sklasyfikowanych próbek negatywnych (FP) do łącznej liczby próbek negatywnych (FP + TN) [12].

AUC to pole powierzchni pod krzywą ROC, a jego wartość oscyluje między 0 a 1. Im wyższa wartość AUC, tym lepsza jakość klasyfikacji modelu. Interpretacja ROC i AUC polega na tym, że im bardziej wypukła jest krzywa ROC, tym wyższa wartość AUC, a tym samym lepsza jakość klasyfikacji modelu. Przy wartości AUC równej 0.5, model klasyfikuje przypadki losowo, natomiast wartość AUC równe 1 oznacza doskonałą klasyfikację [12].



Rysunek 2. Wizualizacja reprezentująca przykładowy wykres ROC i AUC [12].

4. Część eksperymentalna

W rozdziale tym zostaną przedstawione rezultaty eksperymentów przeprowadzonych zgodnie z opisaną metodyką w poprzednim rozdziale. Przedstawionych zostanie dziesięć metod klasyfikacyjnych.

W pracy badania były przeprowadzone kolejno na sześciu różnych kolumnach decyzyjnych, których istota odnośnie rozpatrywanego problemu została opisana w podstawach teoretycznych. Ukazane eksperymenty są oparte na danych pochodzących z próbki z szczotki. Decyzyjnymi kolumnami wykorzystanymi do eksperymentów są:

- Kolagen I % powierzchni – klasy decyzyjne względem mediany,
- Kolagen I siła – posiada trzy klasy decyzyjne, są nimi wartości: 1,2 lub 3,
- Wall area ratio RB1 - klasy decyzyjne względem mediany,
- Wall area ratio RB10 - klasy decyzyjne względem mediany,
- Wall thichness/airway diameter ratio RB1 - klasy decyzyjne względem mediany,
- Średnia harmoniczna liniowa klasy decyzyjne względem mediany,

4.1. Wstępne uwagi

Eksperymenty na próbkach z szczotki

W pracy skupiono się na przeprowadzeniu eksperymentów na próbkach pochodzących z szczotki. Metoda służąca do poboru materiału DNA za pomocą szczotki w porównaniu do metody poboru krwi, charakteryzuje się brakiem inwazyjności, łatwością dostępu, przechowywania i transportu, a także relatywnie nie sprawia dyskomfortu pacjenta. Równie ważnym aspektem jest koszt pozyskania próbki, który w porównaniu do badań z krwi jest dużo niższy. Odkrycie możliwości przewidywania remodelingu oskrzeli na podstawie mikromacierzy, których próbki były pozyskane metodą szczotki stanowiłaby bardziej znaczącą informację w aspekcie praktycznym.

Poprawne działanie utworzonego skryptu

Po utworzeniu skryptu zgodnego z opisaną w poprzednim rozdziale metodyką, pozwalającym na eksplorację danych mikromacierzowych w kontekście klasyfikacji, w pierwszej kolejności zweryfikowano jego poprawne działanie. W tym celu skorzystano z

metody „make_classification” z biblioteki scikit-learn. Metoda ta pozwala wygenerować syntetyczny zbiór danych, którego łatwość lub trudność klasyfikacji można określić za pomocą jego parametrów. Po wygenerowaniu relatywnie łatwego zbioru danych do klasyfikacji i wykorzystaniu ich w utworzonym skrypcie, uśrednione metryki jakości wszystkich iteracji walidacji krzyżowej na danych testowych oscylowały w okolicach 80 - 90%. Drugi eksperyment polegał na utworzeniu zbioru skrajnie zaszumionych danych i wykorzystaniu ich z utworzonym skryptem. Wyniki metryk jakości oscylowały w okolicach 45 - 55%. Otrzymane rezultaty pokrywały się z oczekiwaniami, co pozwoliło przejść do eksploracji danych mikromacierzowych DNA.

Hiperparametry metod klasyfikacyjnych

Ponieważ dokonanie optymalizacji hiperparametrów metod klasyfikacyjnych wiązało by się z koniecznością ewaluacji najkorzystniejszej kombinacji za pomocą zagnieżdżonej pętli CV, zdecydowano się pozostawić je przy ustawieniach domyślnych dla wszystkich z przeprowadzonych eksperymentów. Dokonanie optymalizacji metody klasyfikacyjnej wraz z ewaluacją hiperparametrów metod selekcji cech wiązałoby się z ogromnym wzrostem kosztów obliczeniowych, co uniemożliwiłoby eksplorację w zaplanowanych ramach. Skupiono się więc na dostosowywaniu tylko hiperparametrów metod selekcji cech, które w wstępnych testach wpływały w znacznie większym stopniu na rezultat klasyfikacji oraz przy dostrajaniu metod klasyfikacyjnych bardzo często konfiguracje „zwycięskie” okazywały się ustawieniami domyślnymi.

Przestrzeń przeszukiwania metod selekcji cech

Każdą z metod selekcji cech potraktowano na równych zasadach. Przestrzeń przeszukiwania, jaką zdefiniowano, była taka sama w obrębie każdego z przeprowadzonych eksperymentów. Dotyczyła ona dostrajania hiperparametru metody selekcji cech, który określał ilość cech do wyselekcjonowania z oryginalnego zbioru. Wartości zawarte w przestrzeni przeszukiwania to: [1,2,3,4,5,6,7,8,9,10,12,15,20,25,30,35,40,45,50,60,70,80,90,100].

Prezentowane wyniki

Wyniki, jakie pojawią się w dalszej części rozdziału składają się na: uśrednione wyniki dokładności, precyzji, czułości oraz wartości AUC każdej z iteracji zewnętrznej pętli CV, stanowiących wynik eksperymentu na części testowej. W pracy zwrócono uwagę również na osiągane uśrednione wyniki na części walidacyjnej, tj. uśrednione wyniki dokładności (lub 'f_score weighted' w przypadku kolumny „Kolagen I siła") pochodzące z wewnętrznej pętli CV, która zwracała „zwycięskie" konfiguracje dla każdej z iteracji zewnętrznej pętli CV.

Aby uniknąć wykorzystania „wiedzy z przyszłości" lub tzw. „cherry picking`u" przy doborze ostatecznej konfiguracji dla danej kolumny decyzyjnej, ważnym aspektem jest kierowanie się wynikami z części walidacyjnej. Gdyby kierowano się doborem ostatecznego modelu patrząc na wynik z części testowej, dane zwarte w tej części nie były by już nieznane i nie byłoby możliwości zweryfikowania jak model poradzi sobie na nowych danych. Również gdyby zadecydowano o ostatecznej konfiguracji modelu z pośród wielu przeprowadzonych eksperymentów, kierując się tylko wynikiem na części testowej, wybierając ten przypadek, który poradził sobie najlepiej, mogłoby to oznaczać zaistnienie „overfitting`u" lub dany bardzo dobry wynik mógłby być bardziej szczęśliwym trafem aniżeli wynikiem dobrze przygotowanego modelu. Taka sytuacja jest możliwa gdyż z wielu przeprowadzonych prób naturalnie jedne przypadki będą wypadać lepiej od innych.

Konsensus wyselekcjonowanych cech

Ponieważ wyselekcjonowane cechy w ramach iteracji walidacji krzyżowej były zróżnicowane, decyzję o tym, jaki zestaw cech powinien być uważany za ostateczny, podejmowano na podstawie tzw. głosowania. W pierwszym kroku obliczano średnią ilość dobieranych cech, biorąc pod uwagę każdą z iteracji w ramach danej walidacji krzyżowej (zewnętrznej pętli). Uzyskana średnia stanowiła liczbę cech w ostatecznym zestawie, gdzie kolejno dobierano cechy, które pojawiały się największą ilość razy (w obrębie danej walidacji krzyżowej). Reguła ta była stosowana w dalszej części rozdziału na potrzebę wskazania konsensusu wyselekcjonowanych cech w obrębie danego eksperymentu.

4.2. Eksperymenty na atrybucie decyzyjnym „Kolagen I % powierzchni”

Charakterystyka kolumny decyzyjnej

Źródło danych: szczotka,

Ilość nie pustych obiektów/ wierszy w zestawie danych: 43,

Wartość minimalna w kolumnie: 0.0,

Wartość maksymalna w kolumnie: 100.0,

Wartość średnia w kolumnie: 44.53,

Wartość mediany w kolumnie: 40.0,

Podział klas w zbiorze: 20 do 23.

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectKBest:

Tabela 2. Wyniki eksperymentów dla „Kolagen I % powierzchni”, wykorzystując metodę „SelectKBest”.

Atrybut decyzyjny: **Kolagen I % powierzchni**											
Lp.	**Metoda klasyfikacyjna**	**SelecKBest**									
		score_func = chi2					**score_func = f_classif**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	69.21	44.19	48.00	52.17	43.59	74.20	62.79	66.67	60.87	62.93
2.	**MLP**	70.29	44.19	47.83	47.83	43.91	80.10	62.79	65.22	65.22	62.61
3.	**Random Forest**	67.34	39.53	44.83	56.52	38.26	71.54	53.49	56.52	56.52	53.26
4.	**SVC**	72.44	53.45	57.14	52.17	53.59	75.90	62.79	64.00	69.57	62.28
5.	**Quadratic Discriminant**	70.97	48.84	51.61	69.57	47.28	74.31	65.12	65.38	73.91	64.46
6.	**Linear Discriminant**	72.39	55.81	57.14	69.57	54.78	79.25	69.77	72.72	69.57	69.78
7.	**KNeighbors**	71.76	51.16	54.54	52.17	51.09	76.35	62.79	68.42	56.52	63.26
8.	**Decision Tree**	65.42	41.86	46.66	60.87	40.43	79.47	74.42	75.00	78.26	74.13
9.	**GaussianNB**	73.80	62.79	62.07	78.26	61.63	73.80	65.12	48.18	65.22	65.11
10.	**Gradient Boosting**	64.91	37.21	42.86	52.17	36.09	79.64	81.40	80.00	86.96	80.98

Eksperymenty na kolumnie „Kolagen I % powierzchni” z wykorzystaniem metody SelectKBest okazały się mieszane. W tej części dwa eksperymenty rzucają się w oczy, pierwszy (MLP, *‘f_classif’*) osiągający na części walidacyjnej 80.10% dokładności, lecz

wypadający bardzo słabo na części testowej oraz drugi (GradientBoostingClassifier, *'f_classif'*) bliski przebicia 80% na części walidacyjnej i przebijający tą barierę na części testowej.

Szczegóły dla eksperymentu MLP, SelectKBest, f_classif:

Tabela 3. Macierz pomyłek – MLP, SelectKBest, f_classif (Kolagen I % powierzchni).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	12 (TN)	8 (FP)
Rzeczywista klasa pozytywna	8 (FN)	15 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR09_S', 'AR12_S', 'AR16_S', 'AR23_S', 'AR24_S', 'AR25_S', 'AR26_S', 'AR27_S', 'AR36_S', 'AR43_S', 'AR47_S', 'AR49_S', 'AR52_S', 'AR55_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot6548', 'spot50306', 'spot40406', 'spot41219', 'spot40858', 'spot15445', 'spot16681', 'spot35192', 'spot43511', 'spot14948', 'spot4082', 'spot5991', 'spot39095', 'spot15223', 'spot5297', 'spot5467', 'spot1511']

Szczegóły dla eksperymentu GradientBoostingClassifier, SelectKBest, f_classif:

Tabela 4. Macierz pomyłek – GradientBoostingClassifier, SelectKBest, f_classif (Kolagen I % powierzchni).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	15 (TN)	5 (FP)
Rzeczywista klasa pozytywna	3 (FN)	20 (TP)

Obiekty błędnie rozpoznane:

['AR23_S', 'AR25_S', 'AR36_S', 'AR43_S', 'AR49_S', 'AR52_S', 'AR55_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot41219', 'spot6548', 'spot50306', 'spot16681', 'spot15445', 'spot40406', 'spot40858', 'spot5991', 'spot35192', 'spot15223', 'spot48214', 'spot5297', 'spot1511', 'spot4082', 'spot49677', 'spot32628', 'spot49366', 'spot39095', 'spot14948',

'spot34127', 'spot43511', 'spot5467', 'spot45343', 'spot15253', 'spot43088', 'spot28415', 'spot28370', 'spot5444', 'spot15079', 'spot41401', 'spot22065', 'spot41696', 'spot21770', 'spot7249', 'spot6367', 'spot7050', 'spot867', 'spot41498', 'spot6354', 'spot2964', 'spot35021', 'spot45626', 'spot39752', 'spot15622', 'spot11123', 'spot26890', 'spot48261', 'spot30759', 'spot5380', 'spot37498', 'spot41702', 'spot2603', 'spot41742', 'spot41336', 'spot39185', 'spot5389', 'spot12521', 'spot39958', 'spot10286', 'spot6948', 'spot12181', 'spot41397', 'spot44181', 'spot49936', 'spot1767', 'spot49915']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectFromModel:

Tabela 5. Wyniki eksperymentów dla „Kolagen I % powierzchni", wykorzystując metodę „SelectFromModel".

Atrybut decyzyjny: **Kolagen I % powierzchni**						
Lp.	**Metoda klasyfikacyjna**	**SelectFromModel**				
		estimator = RandomForest()				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec.(%)**	**Czuł.(%)**	**AUC (%)**
1.	**Logistic Regression**	61.05	58.14	59.26	69.57	57.28
2.	**MLP**	65.30	62.79	64.00	69.57	62.28
3.	**Random Forest**	62.41	46.51	50.00	43.48	46.74
4.	**SVC**	63.88	55.81	58.33	60.87	55.43
5.	**Quadratic Discriminant**	64.23	48.84	52.38	47.83	48.91
6.	**Linear Discriminant**	64.90	53.49	55.56	65.22	52.60
7.	**KNeighbors**	62.81	46.51	50.00	47.83	46.41
8.	**Decision Tree**	59.92	39.53	41.18	30.43	40.22
9.	**GaussianNB**	65.75	53.49	56.52	56.21	53.26
10.	**Gradient Boosting**	59.12	37.21	40.00	34.78	37.39

Wyniki eksperymentów na kolumnie „Kolagen I % powierzchni" z wykorzystaniem metody SelectFromModel okazały się słabe. Najkorzystniejszy przypadek patrząc po wynikach na części walidacyjnej (GaussianNB) pozwolił uzyskać tylko 65.75%, jednak po zweryfikowaniu kondycji modelu na części testowej, okazał się nieużyteczny.

Szczegóły dla eksperymentu GaussianNB, SelectfromModel:

Tabela 6. Macierz pomyłek – GaussianNB, SelectfromModel (Kolagen I % powierzchni).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	10 (TN)	10 (FP)
Rzeczywista klasa pozytywna	10 (FN)	13 (TP)

Obiekty błędnie rozpoznane:

['AR09_S', 'AR10_S', 'AR11_S', 'AR13_S', 'AR16_S', 'AR17_S', 'AR23_S', 'AR25_S', 'AR26_S', 'AR28_S', 'AR30_S', 'AR33_S', 'AR35_S', 'AR36_S', 'AR43_S', 'AR45_S', 'AR47_S', 'AR50_S', 'AR53_S', 'AR55_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot6630', 'spot30759', 'spot43513', 'spot41498', 'spot35271', 'spot49939', 'spot25064', 'spot41677', 'spot47145', 'spot46092', 'spot11354', 'spot38011', 'spot37144', 'spot37736', 'spot9626', 'spot42425', 'spot37046', 'spot39958', 'spot24170', 'spot2357', 'spot43165', 'spot48941', 'spot22903', 'spot40169', 'spot41950', 'spot44162', 'spot41810', 'spot19972', 'spot1879', 'spot1511', 'spot37017', 'spot5297', 'spot47977', 'spot10684', 'spot6655', 'spot37255', 'spot8520', 'spot4131', 'spot17347', 'spot6903', 'spot17404', 'spot10514', 'spot793', 'spot24184', 'spot6365', 'spot28254', 'spot9589', 'spot41194']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech RFE:

Tabela 7. Wyniki eksperymentów dla „Kolagen I % powierzchni", wykorzystując metodę „RFE".

Atrybut decyzyjny: **Kolagen I % powierzchni**											
Lp.	**Metoda klasyfikacyjna**	**Recursive Features Elimination**									
		estimator = LogisticRegression() LASSO					**estimator = LogisticRegression() RIDGE**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	74.54	65.12	65.38	73.91	64.46	66.27	51.16	54.17	56.52	50.76
2.	**MLP**	77.77	74.42	80.00	69.57	74.78	66.89	46.51	50.00	52.17	46.08
3.	**Random Forest**	76.58	67.44	69.57	69.57	67.28	67.22	58.14	61.90	56.52	58.26
4.	**SVC**	77.83	72.09	72.00	78.26	71.63	65.81	41.86	46.43	56.52	40.76

5.	**Quadratic Discriminant**	78.28	74.42	73.08	82.60	73.80	64.51	39.53	44.82	56.52	38.26
6.	**Linear Discriminant**	78.17	81.36	85.71	78.26	81.63	67.06	55.81	59.09	56.52	55.76
7.	**KNeighbors**	75.73	60.47	61.53	69.56	59.78	67.12	51.16	53.85	60.87	50.43
8.	**Decision Tree**	68.38	46.51	50.00	65.22	45.11	65.70	48.84	52.17	52.17	48.59
9.	**GaussianNB**	75.17	65.12	65.38	73.91	64.46	66.27	46.51	50.00	47.83	46.41
10.	**Gradient Boosting**	67.17	46.51	50.00	52.17	46.09	63.37	32.56	37.50	39.13	32.07

Uzyskane rezultaty z przeprowadzonych eksperymentów na kolumnie „Kolagen I % powierzchni" z wykorzystaniem metody RFE okazały się mieszane. Tylko metoda LASSO pozwoliła uzyskać korzystne klasyfikacje. Najlepiej wypadła metoda LinearDiscriminantAnalysis, która na części walidacyjnej uzyskała 78.17%, a na części testowej wypadła odrobinę lepiej przebijając nawet próg 80% dla dokładności, precyzji i AUC. Warto również zwrócić uwagę jak poradziła sobie metoda QDA, która osiągnęła odrobinę wyższy poziom dokładności na części walidacyjnej jednak, na części testowej poradziła sobie gorzej.

Szczegóły dla eksperymentu LDA, RFE, LASSO:

Tabela 8. Macierz pomyłek – LDA, RFE, LASSO (Kolagen I % powierzchni).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	17 (TN)	3 (FP)
Rzeczywista klasa pozytywna	5 (FN)	18 (TP)

Obiekty błędnie rozpoznane:

['AR23_S', 'AR25_S', 'AR26_S', 'AR33_S', 'AR36_S', 'AR43_S', 'AR47_S', 'AR50_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot50306', 'spot14948', 'spot15445', 'spot6548', 'spot20786', 'spot40406', 'spot41219', 'spot15253', 'spot40858']

Szczegóły dla eksperymentu QDA, RFE, LASSO:

Tabela 9. Macierz pomyłek – QDA, RFE, LASSO (Kolagen I % powierzchni).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	13 (TN)	7 (FP)
Rzeczywista klasa pozytywna	4 (FN)	19 (TP)

Obiekty błędnie rozpoznane:

['AR13_S', 'AR25_S', 'AR26_S', 'AR28_S', 'AR30_S', 'AR33_S', 'AR36_S', 'AR43_S', 'AR47_S', 'AR50_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot50306', 'spot14948', 'spot15445', 'spot40406', 'spot6548', 'spot20786', 'spot40858', 'spot41219']

4.3. Eksperymenty na atrybucie decyzyjnym „Kolagen I siła"

Charakterystyka kolumny decyzyjnej

Źródło danych: szczotka,

Ilość nie pustych obiektów/ wierszy w zestawie danych oryginalnie: 44,

Kolumna przyjmuje trzy wartości (posiada trzy klasy): 1, 2 lub 3,

Podział klas w zbiorze: 16 | 22 | 6.

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectKBest:

Tabela 10. Wyniki eksperymentów dla „Kolagen I siła", wykorzystując metodę „SelectKBest".

Atrybut decyzyjny: **Kolagen I siła**											
Lp.	**Metoda klasyfikacyjna**	**SelecKBest**									
		score_func = chi2					**score_func = f_classif**				
		Wal. f_score weighted (%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**Bal. Dokł (%)**	**Wal. f_score weighted (%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**Bal. Dokł. (%)**
1.	**Logistic Regression**	55.27	34.09	27.79	34.09	30.81	59.33	34.09	33.09	34.09	24.43
2.	**MLP**	53.91	31.18	25.45	31.81	29.29	60.73	34.09	32.39	34.09	24.43
3.	**Random Forest**	56.94	47.73	43.89	47.72	35.80	57.86	36.36	30.77	36.36	25.38
4.	**SVC**	55.38	38.63	31.64	38.63	30.37	60.03	31.82	32.37	31.82	28.09

5.	**Quadratic Discriminant**	52.46	45.45	30.57	45.45	34.34	55.81	29.54	27.02	29.54	21.40
6.	**Linear Discriminant**	55.54	29.55	25.10	29.55	27.78	57.86	38.63	39.02	38.64	29.17
7.	**KNeighbors**	56.35	34.09	29.28	34.09	31.38	58.24	43.18	45.72	43.18	32.77
8.	**Decision Tree**	52.56	27.27	29.07	27.27	23.93	50.67	27.27	24.84	22.73	20.90
9.	**GaussianNB**	51.48	52.27	48.02	52.27	47.54	58.51	34.09	30.35	34.09	24.43
10.	**Gradient Boosting**	57.86	40.91	36.20	40.91	30.11	52.94	31.82	33.10	31.82	28.09

Uzyskane wyniki z przeprowadzonych eksperymentów na kolumnie „Kolagen I siła" z wykorzystaniem metody SelectKBest okazały się niesatysfakcjonujące. Argument decyzyjny brany w eksperymentach pod uwagę, w odróżnieniu od wszystkich innych, posiada trzy klasy decyzyjne. Poziomem czysto losowej klasyfikacji w tym przypadku jest więc poziom 33.33%. Z racji, że klasy są silnie niezbalansowane, w ocenie modelu nie należy kierować się zwykłą dokładnością. Ponieważ metryka ROC/AUC przeznaczona jest dla problemów binarnych, zamiast niej badano poziom zbalansowanej dokładności. Najlepiej na części testowej wypadła metoda klasyfikacyjna GaussianNB przy wykorzystaniu w procesie selekcji cech funkcji *'chi2'*. Gdyby patrzono na wyniki części walidacyjnej metoda MLP przy wykorzystaniu funkcji 'f_classif', wyglądała najbardziej obiecująco, jednak wynik na zbiorze testowym dla tej konfiguracji okazały się bardzo słabe.

Szczegóły dla eksperymentu GaussianNB, SelectKBest, chi2:

Tabela 11. Macierz pomyłek – GaussianNB, SelectKBest, chi2 (Kolagen I siła).

	Predykowana klasa '1'	**Predykowana klasa '2'**	**Predykowana klasa '3'**
Rzeczywista klasa '1'	1	15	0
Rzeczywista klasa '2'	2	19	1
Rzeczywista klasa '3'	0	3	3

Obiekty błędnie rozpoznane:

['AR04_S', 'AR11_S', 'AR15_S', 'AR16_S', 'AR18_S', 'AR19_S', 'AR21_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR27_S', 'AR28_S', 'AR33_S', 'AR43_S', 'AR45_S', 'AR49_S', 'AR50_S', 'AR51_S', 'AR55_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot49650', 'spot11945']

Szczegóły dla eksperymentu MLP, SelectKBest, f_classif:

Tabela 12. Macierz pomyłek – MLP, SelectKBest, f_classif (Kolagen I siła).

	Predykowana klasa '1'	**Predykowana klasa '2'**	**Predykowana klasa '3'**
Rzeczywista klasa '1'	3	9	4
Rzeczywista klasa '2'	8	12	2
Rzeczywista klasa '3'	2	4	0

Obiekty błędnie rozpoznane:

['AR09_S', 'AR10_S', 'AR11_S', 'AR12_S', 'AR13_S', 'AR16_S', 'AR17_S', 'AR18_S', 'AR19_S', 'AR21_S', 'AR24_S', 'AR25_S', 'AR26_S', 'AR27_S', 'AR28_S', 'AR29_S', 'AR33_S', 'AR36_S', 'AR41_S', 'AR42_S', 'AR43_S', 'AR45_S', 'AR47_S', 'AR49_S', 'AR50_S', 'AR52_S', 'AR55_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot33189', 'spot5713', 'spot33246', 'spot20686', 'spot36924', 'spot37121', 'spot30564', 'spot38153', 'spot33306', 'spot29127', 'spot38492', 'spot28913', 'spot1972', 'spot30397', 'spot18732', 'spot33304', 'spot45983', 'spot16726', 'spot36062', 'spot37011', 'spot37131', 'spot36869', 'spot37282', 'spot26983', 'spot28752', 'spot33426', 'spot9802']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectFromModel:

Tabela 13. Wyniki eksperymentów dla „Kolagen I siła", wykorzystując metodę „SelectFromModel".

Atrybut decyzyjny: **Kolagen I siła**						
Lp.	**Metoda klasyfikacyjna**	**SelectFromModel**				
		estimator = RandomForest()				
		Wal. f_score weighted (%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**Bal. Dokł. (%)**
1.	**Logistic Regression**	52.78	38.64	38.45	38.64	36.11
2.	**MLP**	52.56	40.91	39.37	40.91	37.06

3.	**Random Forest**	52.78	29.55	19.70	29.55	19.70
4.	**SVC**	51.81	40.91	37.06	40.91	30.11
5.	**Quadratic Discriminant**	52.46	45.45	24.39	45.45	30.30
6.	**Linear Discriminant**	51.97	29.55	32.13	29.55	25.44
7.	**KNeighbors**	47.70	29.55	32.17	29.55	26.58
8.	**Decision Tree**	50.19	40.91	42.70	40.91	34.72
9.	**GaussianNB**	49.64	27.27	25.22	27.27	20.45
10.	**Gradient Boosting**	53.21	40.91	36.77	40.91	28.98

Eksperymenty na kolumnie decyzyjnej „Kolagen I siła” z wykorzystaniem metody SelectFromModel zwróciły słabe wyniki. Biorąc pod uwagę wynik z części walidacyjnej najlepiej wypadła metoda GradientBoostingClassifier, wynik jaki metoda uzyskała dla części testowej jest jednak dużo gorszy.

Szczegóły dla eksperymentu GradientBoostingClassifier, SelectFromModel:

Tabela 14. Macierz pomyłek – GradientBoostingClassifier, SelectFromModel (Kolagen I siła).

	Predykowana klasa ‘1’	**Predykowana klasa ‘2’**	**Predykowana klasa ‘3’**
Rzeczywista klasa ‘1’	3	10	3
Rzeczywista klasa ‘2’	5	15	2
Rzeczywista klasa ‘3’	2	4	0

Obiekty błędnie rozpoznane:

['AR08_S', 'AR13_S', 'AR15_S', 'AR16_S', 'AR18_S', 'AR19_S', 'AR21_S', 'AR24_S', 'AR25_S', 'AR26_S', 'AR27_S', 'AR28_S', 'AR33_S', 'AR39_S', 'AR41_S', 'AR42_S', 'AR43_S', 'AR45_S', 'AR47_S', 'AR49_S', 'AR50_S', 'AR53_S', 'AR54_S', 'AR55_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot2538', 'spot31106', 'spot40101', 'spot35205', 'spot35931', 'spot46945', 'spot30991', 'spot30237', 'spot49902', 'spot7382', 'spot21218', 'spot40901', 'spot23292', 'spot41397', 'spot17935', 'spot43264', 'spot11874', 'spot28598',

'spot42622', 'spot38153', 'spot7718', 'spot5937', 'spot21895', 'spot24296', 'spot6367', 'spot9166', 'spot12505', 'spot12839', 'spot37237', 'spot27429', 'spot25222', 'spot27403', 'spot29365', 'spot10650', 'spot28395', 'spot29709', 'spot30170', 'spot7747', 'spot34089']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech RFE:

Tabela 15. Wyniki eksperymentów dla „Kolagen I siła", wykorzystując metodę „RFE".

Atrybut decyzyjny: **Kolagen I siła**											
Lp.	**Metoda klasyfikacyjna**	**Recursive Features Elimination**									
		estimator = LogisticRegression() LASSO					**estimator = LogisticRegression() RIDGE**				
		Wal. f_score weighted (%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**Bal. Dokł (%)**	**Wal. f_score weighted (%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**Bal. Dokł. (%)**
1.	**Logistic Regression**	73.66	59.09	57.95	59.09	47.41	66.90	40.91	38.30	40.91	36.49
2.	**MLP**	75.50	63.64	64.03	63.64	51.01	68.30	59.09	56.46	59.09	53.79
3.	**Random Forest**	73.60	63.64	59.33	63.64	48.11	65.92	50.00	50.14	50.00	40.78
4.	**SVC**	74.74	61.36	55.29	61.36	44.89	65.06	52.27	54.11	52.27	45.77
5.	**Quadratic Discriminant**	66.95	56.82	49.85	56.82	41.29	60.24	45.45	35.35	45.45	31.44
6.	**Linear Discriminant**	72.09	59.09	62.05	59.09	47.41	67.98	61.36	61.24	61.36	52.40
7.	**KNeighbors**	68.09	54.55	57.61	54.54	44.38	57.32	43.18	44.45	43.18	38.57
8.	**Decision Tree**	58.32	40.91	41.49	40.91	37.63	60.46	54.54	56.86	54.54	45.52
9.	**GaussianNB**	73.28	70.45	62.55	70.45	53.79	62.25	47.73	48.11	47.73	39.27
10.	**Gradient Boosting**	61.06	47.73	45.92	47.73	34.66	59.38	40.91	42.62	40.91	41.10

Wyniki przeprowadzonych eksperymentów na kolumnie „Kolagen I siła" z wykorzystaniem metody RFE okazały się niesatysfakcjonujące. Najkorzystniejszy wynik na części testowej uzyskał klasyfikator GaussianNB w połączeniu z metodą selekcji cech RFE- LASSO. Gdyby spojrzeć na wyniki części walidacyjnej, metoda MLP wygląda najlepiej, jednak wynik zbalansowanej dokładności dla tej metody na zbiorze testowym to zaledwie 51%.

Szczegóły dla eksperymentu GaussianNB, RFE, LASSO:

Tabela 16. Macierz pomyłek – GaussianNB, RFE, LASSO (Kolagen I siła).

	Predykowana klasa '1'	Predykowana klasa '2'	Predykowana klasa '3'
Rzeczywista klasa '1'	12	3	1
Rzeczywista klasa '2'	3	19	0
Rzeczywista klasa '3'	4	2	0

Obiekty błędnie rozpoznane:

['AR09_S', 'AR11_S', 'AR12_S', 'AR27_S', 'AR39_S', 'AR41_S', 'AR42_S', 'AR45_S', 'AR47_S', 'AR49_S', 'AR50_S', 'AR55_S', 'AR56_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot38104', 'spot5713', 'spot18732', 'spot39680', 'spot35328', 'spot12793', 'spot9802', 'spot38153', 'spot19354', 'spot26983', 'spot30564']

4.4. Eksperymenty na atrybucie decyzyjnym „Wall area ratio RB1"

Charakterystyka kolumny decyzyjnej

Źródło danych: szczotka,

Ilość nie pustych obiektów/ wierszy w zestawie danych: 36,

Wartość minimalna w kolumnie: 59.0,

Wartość maksymalna w kolumnie: 86.5,

Wartość średnia w kolumnie: 73.06,

Wartość mediany w kolumnie: 73.2,

Podział klas w zbiorze: 18 do 18.

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectKBest:

Tabela 17. Wyniki eksperymentów dla „Wall area ratio RB1", wykorzystując metodę „SelectKBest".

Atrybut decyzyjny: **Wall area ratio RB1**											
Lp.	**Metoda klasyfikacyjna**	**SelecKBest**									
		score_func = chi2					**score_func = f_classif**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	51.51	11.11	11.11	11.11	11.11	46.77	16.67	16.67	16.67	16.67
2.	**MLP**	55.43	33.33	28.57	22.22	33.33	51.18	22.22	22.22	22.22	22.22
3.	**Random Forest**	54.85	22.22	18.75	16.67	22.22	42.85	13.89	15.79	16.66	13.89
4.	**SVC**	54.85	22.22	14.29	11.11	22.22	49.71	25.00	28.57	33.33	25.00
5.	**Quadratic Discriminant**	62.53	55.56	55.00	61.11	55.56	63.42	38.89	40.00	44.44	38.89
6.	**Linear Discriminant**	57.96	30.56	26.67	22.22	30.56	60.08	30.56	29.41	27.78	30.56
7.	**KNeighbors**	51.34	08.33	10.53	11.11	08.33	50.53	22.22	14.29	11.11	22.22
8.	**Decision Tree**	58.04	22.22	18.75	16.67	22.22	50.28	19.44	17.65	16.66	19.44
9.	**GaussianNB**	45.14	16.67	07.14	05.56	16.67	48.81	27.78	30.00	33.33	27.78
10.	**Gradient Boosting**	56.00	11.11	06.25	05.56	11.11	45.47	19.44	21.05	22.22	19.44

Wyniki eksperymentów na kolumnie „Wall area ratio RB1" z wykorzystaniem metody SelectKBest okazały się bardzo słabe. Kierując się wynikiem z części walidacyjnej, wybrano by konfiguracje z wykorzystaniem metody QDA oraz '*f_classif*'. Lecz metryki jakości na części testowej dla tego eksperymentu spadły poniżej 50%. Obserwując wyniki z części testowej najlepiej poradziła sobie również metoda QDA, ale z wykorzystaniem funkcji oceny cech '*chi2'*.

Szczegóły dla eksperymentu QDA, SelectKBest, f_classif:

Tabela 18. Macierz pomyłek - QDA, SelectKBest, f_classif (Wall area ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	6 (TN)	12 (FP)
Rzeczywista klasa pozytywna	10 (FN)	8 (TP)

<u>Obiekty błędnie rozpoznane:</u>

['AR08_S', 'AR09_S', 'AR11_S', 'AR17_S', 'AR18_S', 'AR19_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR25_S', 'AR27_S', 'AR28_S', 'AR30_S', 'AR35_S', 'AR39_S', 'AR42_S', 'AR44_S', 'AR47_S', 'AR48_S', 'AR50_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot41202', 'spot29924', 'spot37957', 'spot29780', 'spot29601', 'spot37949', 'spot6725', 'spot38757', 'spot14447', 'spot19651', 'spot18572', 'spot15473', 'spot40493', 'spot4341', 'spot38279', 'spot42986', 'spot29992', 'spot40558', 'spot20900', 'spot31564', 'spot29879', 'spot31967', 'spot12679', 'spot31820', 'spot48111', 'spot39554', 'spot41925', 'spot16553', 'spot40164', 'spot5939', 'spot30338', 'spot39087', 'spot30584', 'spot48902', 'spot15440', 'spot38428', 'spot49214', 'spot8639', 'spot12074', 'spot19298', 'spot34197', 'spot30270', 'spot3880', 'spot26279', 'spot13668']

Szczegóły dla eksperymentu QDA, SelectKBest, chi2:

Tabela 19. Macierz pomyłek - QDA, SelectKBest, chi2 (Wall area ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	9 (TN)	9 (FP)
Rzeczywista klasa pozytywna	7 (FN)	11 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR10_S', 'AR11_S', 'AR12_S', 'AR18_S', 'AR19_S', 'AR21_S', 'AR22_S', 'AR23_S', 'AR28_S', 'AR30_S', 'AR43_S', 'AR45_S', 'AR51_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot46497', 'spot14447', 'spot29924', 'spot46585', 'spot15658', 'spot40164', 'spot31967', 'spot9163', 'spot33635', 'spot4647', 'spot20900', 'spot36166', 'spot36345', 'spot1572', 'spot12137', 'spot4633', 'spot7616', 'spot12089', 'spot46441', 'spot42703', 'spot9162', 'spot36065', 'spot7252', 'spot5130', 'spot26004', 'spot36489', 'spot37949', 'spot38757', 'spot30392', 'spot15234', 'spot2543', 'spot31564', 'spot1673', 'spot34440', 'spot1830', 'spot39767', 'spot36023', 'spot19651', 'spot46651', 'spot36615', 'spot248', 'spot28800', 'spot20794', 'spot38428', 'spot15148', 'spot41202']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectFromModel:

Tabela 20. Wyniki eksperymentów dla „Wall area ratio RB1", wykorzystując metodę „SelectFromModel".

Atrybut decyzyjny: **Wall area ratio RB1**						
Lp.	**Metoda klasyfikacyjna**	**SelectFromModel**				
		estimator = RandomForest()				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec.(%)**	**Czuł.(%)**	**AUC (%)**
1.	**Logistic Regression**	55.34	33.33	31.25	27.78	33.33
2.	**MLP**	59.02	36.11	38.10	44.44	36.11
3.	**Random Forest**	56.32	22.22	29.17	38.89	22.22
4.	**SVC**	58.85	25.00	23.52	22.22	25.00
5.	**Quadratic Discriminant**	60.16	44.44	44.44	44.44	44.44
6.	**Linear Discriminant**	60.49	44.44	43.76	38.89	44.44
7.	**KNeighbors**	57.22	16.67	12.50	11.11	16.67
8.	**Decision Tree**	57.22	30.55	31.58	33.33	30.56
9.	**GaussianNB**	53.71	27.78	30.00	33.33	27.78
10.	**Gradient Boosting**	57.47	30.56	33.33	38.89	30.56

Wyniki eksperymentów na kolumnie „Wall area ratio RB1" z wykorzystaniem metody SelectFromModel wyszły ponownie bardzo słabo. Ani jeden eksperyment nie pozwolił uzyskać interesującego wyniku. Metoda LDA pozwoliła uzyskań najlepszy wynik, jeżeli patrzono by tylko na część testową to by była na równi z metodą QDA.

Szczegóły dla eksperymentu LDA, SelectFromModel:

Tabela 21. Macierz pomyłek - LDA, SelectFromModel (Wall area ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	9 (TN)	9 (FP)
Rzeczywista klasa pozytywna	11 (FN)	7 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR12_S', 'AR13_S', 'AR17_S', 'AR19_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR27_S', 'AR35_S', 'AR39_S', 'AR42_S', 'AR43_S', 'AR44_S', 'AR47_S', 'AR48_S', 'AR51_S', 'AR53_S', 'AR55_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot7721', 'spot19909', 'spot18523', 'spot14177', 'spot37736', 'spot7508', 'spot40278', 'spot25864', 'spot20900', 'spot4788', 'spot1281', 'spot33284', 'spot2659', 'spot24103', 'spot31668', 'spot19180', 'spot32874', 'spot11138', 'spot44935', 'spot16764', 'spot13889', 'spot33840', 'spot37021', 'spot42619', 'spot26279', 'spot3978', 'spot11620', 'spot11392', 'spot45883', 'spot27214']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech RFE:

Tabela 22. Wyniki eksperymentów dla „Wall area ratio RB1", wykorzystując metodę „RFE".

Atrybut decyzyjny: **Wall area ratio RB1**											
Lp.	**Metoda klasyfikacyjna**	**Recursive Features Elimination**									
		estimator = LogisticRegression() LASSO					**estimator = LogisticRegression() RIDGE**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	45.38	22.22	22.22	22.22	22.22	57.38	25.00	23.53	22.22	25.00
2.	**MLP**	47.51	19.44	21.05	22.22	19.44	59.67	30.56	31.58	33.33	30.56
3.	**Random Forest**	49.06	38.89	38.89	38.89	38.89	59.34	36.11	35.29	33.33	36.11
4.	**SVC**	46.36	16.67	20.00	22.00	16.67	56.49	22.22	25.00	27.78	22.22
5.	**Quadratic Discriminant**	59.26	33.33	35.00	38.89	33.33	64.40	44.44	45.00	50.00	44.44
6.	**Linear Discriminant**	52.65	33.33	31.25	27.78	33.33	60.16	33.33	28.57	22.22	33.33
7.	**KNeighbors**	51.26	25.00	26.32	27.78	25.00	56.73	27.78	27.78	27.78	27.78
8.	**Decision Tree**	59.26	33.33	31.25	27.78	33.33	64.24	41.67	40.00	33.33	41.67
9.	**GaussianNB**	49.22	25.00	26.32	27.78	25.00	59.51	38.89	38.89	38.89	38.89
10.	**Gradient Boosting**	55.59	36.11	33.33	27.78	36.11	61.55	27.78	27.78	27.78	27.78

Wyniki eksperymentów na kolumnie „Wall area ratio RB1" z wykorzystaniem metody RFE wyszły bardzo słabo, podobnie jak dla dwóch poprzednich metod selekcji

cech. Wyniki dla metody regularyzacji RIDGE wypadły odrobinę lepiej, lecz i tak żaden eksperyment nie pozwolił uzyskać metryk jakości powyżej 50%, co stanowi ciekawą sytuację. Najkorzystniej wypadła metoda QDA z wykorzystaniem przy selekcji cech regularyzacji RIDGE.

Szczegóły dla eksperymentu QDA, RFE, RIDGE:

Tabela 23. Macierz pomyłek – QDA, RFE, RIDGE (Wall area ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	7 (TN)	11 (FP)
Rzeczywista klasa pozytywna	9 (FN)	9 (TP)

Obiekty błędnie rozpoznane:

['AR07_S', 'AR08_S', 'AR09_S', 'AR11_S', 'AR13_S', 'AR17_S', 'AR21_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR25_S', 'AR28_S', 'AR42_S', 'AR47_S', 'AR50_S', 'AR51_S', 'AR53_S', 'AR55_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot14447', 'spot38757', 'spot29924', 'spot19651', 'spot31967', 'spot39767', 'spot4341', 'spot20900', 'spot12074', 'spot8639', 'spot40164', 'spot37949', 'spot18572', 'spot23994', 'spot40493', 'spot30584', 'spot31564', 'spot30390', 'spot20794', 'spot26004', 'spot39837', 'spot35330', 'spot9163']

4.5. Eksperymenty na atrybucie decyzyjnym „Wall area ratio RB10”

Charakterystyka kolumny decyzyjnej

Źródło danych: szczotka,

Ilość nie pustych obiektów/ wierszy w zestawie danych: 37,

Wartość minimalna w kolumnie: 59.0,

Wartość maksymalna w kolumnie: 81.4,

Wartość średnia w kolumnie: 71.31,

Wartość mediany w kolumnie: 71.9,

Podział klas w zbiorze: 18 do 19.

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectKBest:

Tabela 24. Wyniki eksperymentów dla „Wall area ratio RB10", wykorzystując metodę „SelectKBest".

Atrybut decyzyjny: **Wall area ratio RB10**											
Lp.	**Metoda klasyfikacyjna**	**SelecKBest**									
		score_func = chi2					**score_func = f_classif**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	71.37	62.16	61.90	68.42	61.99	80.62	75.68	77.78	73.68	75.73
2.	**MLP**	78.46	70.27	70.00	73.68	70.18	83.41	72.97	76.47	68.42	73.10
3.	**Random Forest**	77.39	75.67	77.78	73.68	75.73	77.39	81.08	80.00	84.21	80.99
4.	**SVC**	73.92	70.27	68.18	78.95	70.03	79.39	70.27	78.57	57.89	70.61
5.	**Quadratic Discriminant**	73.45	64.86	60.71	89.47	64.18	84.49	75.68	77.78	73.68	75.73
6.	**Linear Discriminant**	75.77	67.57	65.22	78.95	67.25	84.80	78.38	82.35	73.68	78.51
7.	**KNeighbors**	77.77	75.68	75.00	78.95	75.58	87.96	83.78	84.21	84.21	83.77
8.	**Decision Tree**	81.63	75.68	75.00	78.95	75.58	85.95	78.38	78.95	78.95	78.36
9.	**GaussianNB**	75.38	70.27	66.67	84.21	69.88	84.56	75.68	77.78	73.68	75.73
10.	**Gradient Boosting**	80.40	70.27	68.18	78.95	70.03	86.26	75.68	75.00	78.95	75.58

Wyniki eksperymentów na kolumnie „Wall area ratio RB10" z wykorzystaniem metody SelectKBest okazały się relatywnie dobre. Udało się uzyskać konfiguracje (KNeighbors, '*f_classif*'), która pozwoliła uzyskać aż 87.96% dokładności na części walidacyjnej, i co najważniejsze, jakość pozostała na bardzo wysokich poziomach w części testowej, tj dokładność, precyzja, czułość i AUC oscylowały w granicach 84%. Taki model można już nazwać modelem dobrej jakości. Warto również wyróżnić takie metody jak GradientBoostingClassifier, DecisionTreeClassifier, RandomForestClassifier oraz LDA, które dla funkcji oceny cech '*f_classif',* również wypadły przyzwoicie.

Szczegóły dla eksperymentu KNeighbors, SelectKBest, f_classif:

Tabela 25. Macierz pomyłek - KNeighbors, SelectKBest, f_classif (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	15 (TN)	3 (FP)
Rzeczywista klasa pozytywna	3 (FN)	16 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR44_S', 'AR49_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Szczegóły dla eksperymentu GradientBoostingClassifier, SelectKBest, f_classif:

Tabela 26. Macierz pomyłek - GradientBoostingClassifier, SelectKBest, f_classif (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	13 (TN)	5 (FP)
Rzeczywista klasa pozytywna	4 (FN)	15 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR42_S', 'AR44_S', 'AR49_S', 'AR50_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857', 'spot26649', 'spot41227', 'spot47285']

Szczegóły dla eksperymentu DecisionTreeClassifier, SelectKBest, f_classif:

Tabela 27. Macierz pomyłek - DecisionTreeClassifier, SelectKBest, f_classif (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	14 (TN)	4 (FP)
Rzeczywista klasa pozytywna	4 (FN)	15 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR42_S', 'AR44_S', 'AR49_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857', 'spot47285']

Szczegóły dla eksperymentu LDA, SelectKBest, f_classif:

Tabela 28. Macierz pomyłek - LDA, SelectKBest, f_classif (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	15 (TN)	3 (FP)
Rzeczywista klasa pozytywna	5 (FN)	14 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR42_S', 'AR44_S', 'AR48_S', 'AR49_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Szczegóły dla eksperymentu RandomForestClassifier, SelectKBest, f_classif:

Tabela 29. Macierz pomyłek - RandomForestClassifier, SelectKBest, f_classif (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	14 (TN)	4 (FP)
Rzeczywista klasa pozytywna	3 (FN)	16 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR44_S', 'AR49_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectFromModel:

Tabela 30. Wyniki eksperymentów dla „Wall area ratio RB10", wykorzystując metodę „SelectFromModel".

Atrybut decyzyjny: **Wall area ratio RB10**						
Lp.	**Metoda klasyfikacyjna**	**SelectFromModel**				
		estimator = RandomForest()				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec.(%)**	**Czuł.(%)**	**AUC (%)**
1.	**Logistic Regression**	51.39	59.46	59.09	68.42	59.09
2.	**MLP**	55.71	51.35	52.38	57.89	51.17
3.	**Random Forest**	57.79	54.05	56.25	47.37	54.24
4.	**SVC**	56.25	62.16	61.90	68.42	61.99

5.	**Quadratic Discriminant**	63.19	43.24	44.44	42.11	43.27
6.	**Linear Discriminant**	59.10	64.87	66.67	63.16	64.91
7.	**KNeighbors**	55.40	56.76	57.89	57.89	56.73
8.	**Decision Tree**	70.44	54.05	56.25	47.37	54.24
9.	**GaussianNB**	51.62	54.05	55.56	52.63	54.09
10.	**Gradient Boosting**	71.76	67.57	66.67	73.68	67.40

Wyniki eksperymentów na kolumnie „Wall area ratio RB10" z wykorzystaniem metody SelectFromModel okazały się słabe. Najlepiej wypadła konfiguracja wykorzystująca metodę GradientBoostingClassifier.

Szczegóły dla eksperymentu GradientBoostingClassifier, SelectFromModel:

Tabela 31. Macierz pomyłek - GradientBoostingClassifier, SelectFromModel (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	11 (TN)	7 (FP)
Rzeczywista klasa pozytywna	5 (FN)	14 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR18_S', 'AR22_S', 'AR23_S', 'AR42_S', 'AR47_S', 'AR48_S', 'AR50_S', 'AR51_S', 'AR52_S', 'AR55_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot6052', 'spot7041', 'spot47285', 'spot13566', 'spot24194', 'spot34884', 'spot35375', 'spot11290', 'spot48949', 'spot4996', 'spot18731', 'spot29199', 'spot218', 'spot49357', 'spot46032', 'spot22966', 'spot33363', 'spot18190', 'spot44963', 'spot22979', 'spot48659', 'spot3693', 'spot30225', 'spot21124', 'spot20887', 'spot42192', 'spot47678', 'spot7367', 'spot20016', 'spot20534', 'spot22940', 'spot42754', 'spot47912', 'spot29475', 'spot37285', 'spot771', 'spot11363', 'spot1006', 'spot49943', 'spot13568', 'spot13684', 'spot2817', 'spot39531', 'spot5119', 'spot19886']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech RFE:

Tabela 32. Wyniki eksperymentów dla „Wall area ratio RB10", wykorzystując metodę „RFE".

Atrybut decyzyjny: **Wall area ratio RB10**											
Lp.	**Metoda klasyfikacyjna**	**Recursive Features Elimination**									
		estimator = LogisticRegression() LASSO					**estimator = LogisticRegression() RIDGE**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	79.47	81.08	83.33	78.95	81.14	70.29	72.97	73.68	73.68	72.95
2.	**MLP**	82.25	78.38	82.35	73.68	78.51	74.15	70.27	72.22	68.42	70.32
3.	**Random Forest**	84.56	83.78	80.95	89.47	83.63	77.39	70.27	70.00	73.68	70.18
4.	**SVC**	77.62	78.38	92.31	63.16	78.80	69.44	70.27	75.00	63.16	70.47
5.	**Quadratic Discriminant**	83.56	78.38	82.35	73.68	78.51	75.92	67.57	70.59	63.16	67.69
6.	**Linear Discriminant**	83.56	81.08	83.33	78.95	81.14	75.23	72.97	73.68	73.68	72.95
7.	**KNeighbors**	87.27	86.49	85.00	89.47	86.40	77.70	81.08	77.27	89.47	80.85
8.	**Decision Tree**	84.64	83.78	80.95	89.47	83.63	79.55	67.57	70.59	63.16	67.69
9.	**GaussianNB**	83.41	78.38	78.95	78.95	78.36	75.00	70.27	72.22	68.42	70.32
10.	**Gradient Boosting**	84.80	83.78	80.95	89.47	83.63	82.17	62.16	61.90	68.42	61.99

Wyniki eksperymentów na kolumnie „Wall area ratio RB10" z wykorzystaniem metody RFE wyszły optymistycznie. Tym razem aż pięć konfiguracje pozwoliły uzyskać rezultat, gdzie zarówno na części walidacyjnej jak i na testowej metryki jakości są na poziomie powyżej 80%. Wspomniane wyniki na dobrych poziomach pozwoliły uzyskać metody: KNeighbors, GradientBoostingClassifier, DecisionTreeClassifier, RandomForestClassifier oraz LDA, gdzie dla wszystkich zastosowano w procesie selekcji cech metodę regularyzacji LASSO.

Szczegóły dla eksperymentu KNeighbors, RFE, LASSO:

Tabela 33. Macierz pomyłek – KNeighbors, RFE, LASSO (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	15 (TN)	3 (FP)
Rzeczywista klasa pozytywna	2 (FN)	17 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR44_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Szczegóły dla eksperymentu GradientBoostingClassifier, RFE, LASSO:

Tabela 34. Macierz pomyłek – GradientBoostingClassifier, RFE, LASSO (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	14 (TN)	4 (FP)
Rzeczywista klasa pozytywna	2 (FN)	17 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR44_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Szczegóły dla eksperymentu DecisionTreeClassifier, RFE, LASSO:

Tabela 35. Macierz pomyłek – DecisionTreeClassifier, RFE, LASSO (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	14 (TN)	4 (FP)
Rzeczywista klasa pozytywna	2 (FN)	17 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR44_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Szczegóły dla eksperymentu RandomForestClassifier, RFE, LASSO:

Tabela 36. Macierz pomyłek – RandomForestClassifier, RFE, LASSO (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	14 (TN)	4 (FP)
Rzeczywista klasa pozytywna	2 (FN)	17 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR44_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

Szczegóły dla eksperymentu LDA, RFE, LASSO:

Tabela 37. Macierz pomyłek – LDA, RFE, LASSO (Wall area ratio RB10).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	15 (TN)	3 (FP)
Rzeczywista klasa pozytywna	4 (FN)	15 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR30_S', 'AR41_S', 'AR42_S', 'AR44_S', 'AR48_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot28857']

4.6. Eksperymenty na atrybucie decyzyjnym „Wall thichness/airway diameter ratio RB1”

Charakterystyka kolumny decyzyjnej

Źródło danych: szczotka,

Ilość nie pustych obiektów/ wierszy w zestawie danych: 36,

Wartość minimalna w kolumnie: 18.0,

Wartość maksymalna w kolumnie: 31.2,

Wartość średnia w kolumnie: 24.03,

Wartość mediany w kolumnie: 24.05,

Podział klas w zbiorze: 18 do 18.

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectKBest:

Tabela 38. Wyniki eksperymentów dla „Wall thichness/airway diameter ratio RB1", wykorzystując metodę „SelectKBest".

Atrybut decyzyjny: **Wall thichness/airway diameter ratio RB1**											
Lp.	**Metoda klasyfikacyjna**	**SelecKBest**									
		score_func = chi2					**score_func = f_classif**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	51.51	11.11	11.11	11.11	11.11	46.77	16.67	16.67	16.67	16.67
2.	**MLP**	55.43	33.33	28.57	22.22	33.33	51.18	22.22	22.22	22.22	22.22
3.	**Random Forest**	54.85	22.22	18.75	16.67	22.22	42.85	13.89	15.79	16.66	13.89
4.	**SVC**	54.85	22.22	14.29	11.11	22.22	49.71	25.00	28.57	33.33	25.00
5.	**Quadratic Discriminant**	62.53	55.56	55.00	61.11	55.56	63.42	38.89	40.00	44.44	38.89
6.	**Linear Discriminant**	57.96	30.56	26.67	22.22	30.56	60.08	30.56	29.41	27.78	30.56
7.	**KNeighbors**	51.34	08.33	10.53	11.11	08.33	50.53	22.22	14.29	11.11	22.22
8.	**Decision Tree**	58.04	22.22	18.75	16.67	22.22	50.28	19.44	17.65	16.66	19.44
9.	**GaussianNB**	45.14	16.67	07.14	05.56	16.67	48.81	27.78	30.00	33.33	27.78
10.	**Gradient Boosting**	56.00	11.11	06.25	05.56	11.11	45.47	19.44	21.05	22.22	19.44

Ponieważ podział na klasy dla argumentu decyzyjnego „Wall thichness/airway diameter ratio RB1", jest dokładnie taki sam dla każdego obiektu w zbiorze, jak w przypadku kolumny „Wall area ratio RB1" uzyskane wyniki pomiędzy tymi dwoma argumentami są identyczne.

Przeprowadzone eksperymenty na kolumnie „Wall thichness/airway diameter ratio RB1" z wykorzystaniem metody SelectKBest wypadły bardzo słabo. Najlepiej się zaprezentowały tutaj dwa eksperymenty wykorzystujące metodę klasyfikacyjną QDA.

Szczegóły dla eksperymentu QDA, SelectKBest, f_classif:

Tabela 39. Macierz pomyłek - QDA, SelectKBest, f_classif (Wall thichness/airway diameter ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	6 (TN)	12 (FP)
Rzeczywista klasa pozytywna	10 (FN)	8 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR11_S', 'AR17_S', 'AR18_S', 'AR19_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR25_S', 'AR27_S', 'AR28_S', 'AR30_S', 'AR35_S', 'AR39_S', 'AR42_S', 'AR44_S', 'AR47_S', 'AR48_S', 'AR50_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot41202', 'spot29924', 'spot37957', 'spot29780', 'spot29601', 'spot37949', 'spot6725', 'spot38757', 'spot14447', 'spot19651', 'spot18572', 'spot15473', 'spot40493', 'spot4341', 'spot38279', 'spot42986', 'spot29992', 'spot40558', 'spot20900', 'spot31564', 'spot29879', 'spot31967', 'spot12679', 'spot31820', 'spot48111', 'spot39554', 'spot41925', 'spot16553', 'spot40164', 'spot5939', 'spot30338', 'spot39087', 'spot30584', 'spot48902', 'spot15440', 'spot38428', 'spot49214', 'spot8639', 'spot12074', 'spot19298', 'spot34197', 'spot30270', 'spot3880', 'spot26279', 'spot13668']

Szczegóły dla eksperymentu QDA, SelectKBest, chi2:

Tabela 40. Macierz pomyłek - QDA, SelectKBest, chi2 (Wall thichness/airway diameter ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	9 (TN)	9 (FP)
Rzeczywista klasa pozytywna	7 (FN)	11 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR09_S', 'AR10_S', 'AR11_S', 'AR12_S', 'AR18_S', 'AR19_S', 'AR21_S', 'AR22_S', 'AR23_S', 'AR28_S', 'AR30_S', 'AR43_S', 'AR45_S', 'AR51_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot46497', 'spot14447', 'spot29924', 'spot46585', 'spot15658', 'spot40164', 'spot31967', 'spot9163', 'spot33635', 'spot4647', 'spot20900', 'spot36166', 'spot36345', 'spot1572', 'spot12137', 'spot4633', 'spot7616', 'spot12089', 'spot46441', 'spot42703', 'spot9162', 'spot36065', 'spot7252', 'spot5130', 'spot26004', 'spot36489', 'spot37949', 'spot38757', 'spot30392', 'spot15234', 'spot2543', 'spot31564', 'spot1673', 'spot34440',

'spot1830', 'spot39767', 'spot36023', 'spot19651', 'spot46651', 'spot36615', 'spot248', 'spot28800', 'spot20794', 'spot38428', 'spot15148', 'spot41202']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectFromModel:

Tabela 41. Wyniki eksperymentów dla „Wall thichness/airway diameter ratio RB1", wykorzystując metodę „SelectFromModel".

Atrybut decyzyjny: **Wall thichness/airway diameter ratio RB1**						
Lp.	**Metoda klasyfikacyjna**	**SelectFromModel**				
		estimator = RandomForest()				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec.(%)**	**Czuł.(%)**	**AUC (%)**
1.	**Logistic Regression**	55.34	33.33	31.25	27.78	33.33
2.	**MLP**	59.02	36.11	38.10	44.44	36.11
3.	**Random Forest**	56.32	22.22	29.17	38.89	22.22
4.	**SVC**	58.85	25.00	23.52	22.22	25.00
5.	**Quadratic Discriminant**	60.16	44.44	44.44	44.44	44.44
6.	**Linear Discriminant**	60.49	44.44	43.76	38.89	44.44
7.	**KNeighbors**	57.22	16.67	12.50	11.11	16.67
8.	**Decision Tree**	57.22	30.55	31.58	33.33	30.56
9.	**GaussianNB**	53.71	27.78	30.00	33.33	27.78
10.	**Gradient Boosting**	57.47	30.56	33.33	38.89	30.56

Wyniki eksperymentów na kolumnie „Wall thichness/airway diameter ratio RB1" z wykorzystaniem metody SelectFromModel wypadły bardzo słabo. Ani jeden eksperyment nie pozwolił uzyskać interesującego wyniku. Najlepiej wypadła tu metoda LDA.

Szczegóły dla eksperymentu LDA, SelectFromModel:

Tabela 42. Macierz pomyłek - LDA, SelectFromModel (Wall thichness/airway diameter ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	9 (TN)	9 (FP)
Rzeczywista klasa pozytywna	11 (FN)	7 (TP)

Obiekty błędnie rozpoznane:

['AR08_S', 'AR12_S', 'AR13_S', 'AR17_S', 'AR19_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR27_S', 'AR35_S', 'AR39_S', 'AR42_S', 'AR43_S', 'AR44_S', 'AR47_S', 'AR48_S', 'AR51_S', 'AR53_S', 'AR55_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot7721', 'spot19909', 'spot18523', 'spot14177', 'spot37736', 'spot7508', 'spot40278', 'spot25864', 'spot20900', 'spot4788', 'spot1281', 'spot33284', 'spot2659', 'spot24103', 'spot31668', 'spot19180', 'spot32874', 'spot11138', 'spot44935', 'spot16764', 'spot13889', 'spot33840', 'spot37021', 'spot42619', 'spot26279', 'spot3978', 'spot11620', 'spot11392', 'spot45883', 'spot27214']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech RFE:

Tabela 43. Wyniki eksperymentów dla „Wall thichness/airway diameter ratio RB1", wykorzystując metodę „RFE".

Atrybut decyzyjny: **Wall thichness/airway diameter ratio RB1**											
Lp.	**Metoda klasyfikacyjna**	**Recursive Features Elimination**									
		estimator = LogisticRegression() LASSO					**estimator = LogisticRegression() RIDGE**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	45.38	22.22	22.22	22.22	22.22	57.38	25.00	23.53	22.22	25.00
2.	**MLP**	47.51	19.44	21.05	22.22	19.44	59.67	30.56	31.58	33.33	30.56
3.	**Random Forest**	49.06	38.89	38.89	38.89	38.89	59.34	36.11	35.29	33.33	36.11
4.	**SVC**	46.36	16.67	20.00	22.00	16.67	56.49	22.22	25.00	27.78	22.22
5.	**Quadratic Discriminant**	59.26	33.33	35.00	38.89	33.33	64.40	44.44	45.00	50.00	44.44
6.	**Linear Discriminant**	52.65	33.33	31.25	27.78	33.33	60.16	33.33	28.57	22.22	33.33
7.	**KNeighbors**	51.26	25.00	26.32	27.78	25.00	56.73	27.78	27.78	27.78	27.78
8.	**Decision Tree**	59.26	33.33	31.25	27.78	33.33	64.24	41.67	40.00	33.33	41.67
9.	**GaussianNB**	49.22	25.00	26.32	27.78	25.00	59.51	38.89	38.89	38.89	38.89
10.	**Gradient Boosting**	55.59	36.11	33.33	27.78	36.11	61.55	27.78	27.78	27.78	27.78

Wyniki eksperymentów na argumencie decyzyjnym „Wall thichness/airway diameter ratio RB1” z wykorzystaniem metody RFE wypadły ponownie bardzo słabo. Spośród przeprowadzonych eksperymentów najkorzystniej wypadła metoda QDA z wykorzystaniem metody regularyzacji RIDGE w procesie selekcji cech.

Szczegóły dla eksperymentu QDA, RFE, RIDGE:

Tabela 44. Macierz pomyłek – QDA, RFE, RIDGE (Wall thichness/airway diameter ratio RB1).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	7 (TN)	11 (FP)
Rzeczywista klasa pozytywna	9 (FN)	9 (TP)

Obiekty błędnie rozpoznane:

['AR07_S', 'AR08_S', 'AR09_S', 'AR17_S', 'AR13_S', 'AR11_S', 'AR21_S', 'AR22_S', 'AR23_S', 'AR24_S', 'AR25_S', 'AR50_S', 'AR42_S', 'AR47_S', 'AR28_S', 'AR51_S', 'AR53_S', 'AR55_S', 'AR56_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot14447', 'spot38757', 'spot29924', 'spot37949', 'spot31967', 'spot39767', 'spot4341', 'spot20900', 'spot12074', 'spot8639', 'spot40164', 'spot19651', 'spot18572', 'spot23994', 'spot40493', 'spot30584', 'spot31564', 'spot30390', 'spot20794', 'spot26004', 'spot39837', 'spot35330', 'spot9163']

4.7. Eksperymenty na atrybucie decyzyjnym „Średnia harmoniczna liniowa”

Charakterystyka kolumny decyzyjnej

Źródło danych: szczotka,

Ilość nie pustych obiektów/ wierszy w zestawie danych: 27,

Wartość minimalna w kolumnie: 4.05,

Wartość maksymalna w kolumnie: 9.9,

Wartość średnia w kolumnie: 6.18,

Wartość mediany w kolumnie: 5.82,

Podział klas w zbiorze: 13 do 14.

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectKBest:

Tabela 45. Wyniki eksperymentów dla „Średnia harmoniczna liniowa", wykorzystując metodę „SelectKBest".

Atrybut decyzyjny: **Średnia harmoniczna liniowa**											
Lp.	**Metoda klasyfikacyjna**	**SelecKBest**									
		score_func = chi2					**score_func = chi2**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	68.04	29.63	33.33	35.71	29.40	60.65	62.96	62.50	71.43	62.64
2.	**MLP**	69.22	62.96	64.29	64.29	62.91	66.12	25.93	31.25	35.71	25.55
3.	**Random Forest**	59.02	29.63	33.33	35.71	29.40	61.39	40.74	43.75	50.00	40.38
4.	**SVC**	66.27	62.96	62.50	71.43	62.64	61.09	29.63	33.33	35.71	29.40
5.	**Quadratic Discriminant**	72.04	51.85	53.33	57.14	51.65	71.00	25.93	28.57	28.57	25.82
6.	**Linear Discriminant**	72.33	59.26	58.82	71.43	58.79	67.75	29.63	33.33	35.71	29.40
7.	**KNeighbors**	71.44	48.15	50.00	71.43	47.25	65.23	37.04	40.00	42.86	36.81
8.	**Decision Tree**	60.06	29.63	33.33	35.71	29.40	74.11	62.96	62.50	71.43	62.64
9.	**GaussianNB**	68.19	70.37	68.75	78.57	70.05	63.01	29.63	33.33	35.71	29.40
10.	**Gradient Boosting**	60.50	37.04	38.46	35.14	37.09	71.00	51.85	53.33	57.14	51.65

Wyniki eksperymentów na kolumnie „Średnia harmoniczna liniowa" z wykorzystaniem metody SelectKBest wyszły słabo. Nie udało się uzyskać ani jednej konfiguracji zwracającej dobre wyniki. Patrząc zarówno na wyniki z części walidacyjnej jak i testowej najkorzystniej wypadła metoda GaussianNB, wykorzystując w procesie selekcji cech funkcję *'chi2'.* Na części walidacyjnej najlepiej wypadł DecisionTreeClassifier stosując funkcje *'f_classif'*, jednak wynik jaki metoda uzyskała w części testowej jest dużo gorszy.

Szczegóły dla eksperymentu GaussianNB, SelectKBest, chi2:

Tabela 46. Macierz pomyłek -GaussianNB, SelectKbest, chi2 (Średnia harmoniczna liniowa).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	8 (TN)	5 (FP)
Rzeczywista klasa pozytywna	2 (FN)	12 (TP)

Obiekty błędnie rozpoznane:

['AR16_S', 'AR21_S', 'AR26_S', 'AR30_S', 'AR35_S', 'AR48_S', 'AR50_S', 'AR53_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot32915']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech SelectFromModel:

Tabela 47. Wyniki eksperymentów dla „Średnia harmoniczna liniowa", wykorzystując metodę „SelectFromModel".

Atrybut decyzyjny: **Średnia harmoniczna liniowa**						
Lp.	**Metoda klasyfikacyjna**	**SelectFromModel**				
		estimator = RandomForest()				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec.(%)**	**Czuł.(%)**	**AUC (%)**
1.	**Logistic Regression**	56.80	48.15	50.00	71.42	47.25
2.	**MLP**	64.35	62.96	64.29	64.29	62.91
3.	**Random Forest**	66.86	51.85	53.33	57.14	51.64
4.	**SVC**	63.01	51.85	53.85	50.00	51.92
5.	**Quadratic Discriminant**	67.01	55.55	55.55	71.43	54.95
6.	**Linear Discriminant**	68.34	51.85	53.33	57.14	51.65
7.	**KNeighbors**	63.61	62.96	61.11	78.57	62.36
8.	**Decision Tree**	78.10	70.37	71.43	71.43	70.33
9.	**GaussianNB**	60.06	37.04	42.11	57.14	36.26
10.	**Gradient Boosting**	78.40	62.96	64.28	64.28	62.29

Wyniki eksperymentów na kolumnie „Średnia harmoniczna liniowa" z wykorzystaniem metody SelectFromModel wyglądają podobnie jak w przypadku metody selekcji cech SelectKBest. Konfiguracja wykorzystująca GradientBoostingClassifier pozwoliła uzyskać najlepszy wynik, jednak nie jest on dostatecznie dobry.

Szczegóły dla eksperymentu GradientBoostClassifier, SelectFromModel:

Tabela 48. Macierz pomyłek - GradientBoostClassifier, SelectFromModel (Średnia harmoniczna liniowa).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	8 (TN)	5 (FP)
Rzeczywista klasa pozytywna	5 (FN)	9 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR15_S', 'AR21_S', 'AR30_S', 'AR31_S', 'AR36_S', 'AR49_S', 'AR50_S', 'AR53_S', 'AR56_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot30365', 'spot38409', 'spot48593', 'spot11164', 'spot24948', 'spot19627', 'spot5754', 'spot15519', 'spot13535', 'spot11640', 'spot10259', 'spot38110', 'spot39229', 'spot1791', 'spot37443', 'spot1947', 'spot36876', 'spot10696', 'spot33405', 'spot14467', 'spot42118', 'spot956', 'spot44723', 'spot10598', 'spot32363', 'spot32931', 'spot24228', 'spot8318', 'spot18002', 'spot8966', 'spot940', 'spot22569', 'spot22692', 'spot42095', 'spot18184', 'spot26921', 'spot33213', 'spot14830', 'spot48725', 'spot10618', 'spot6783', 'spot14800', 'spot123', 'spot34289', 'spot25213', 'spot35897', 'spot37861', 'spot29356', 'spot34182', 'spot11874', 'spot24568', 'spot39554', 'spot47003', 'spot783', 'spot20910', 'spot7930', 'spot47811', 'spot30485']

Rezultaty eksperymentów z wykorzystaniem metody selekcji cech RFE:

Tabela 49. Wyniki eksperymentów dla „Średnia harmoniczna liniowa", wykorzystując metodę „RFE".

Atrybut decyzyjny: **Średnia harmoniczna liniowa**											
Lp.	**Metoda klasyfikacyjna**	**Recursive Features Elimination**									
		estimator = LogisticRegression() LASSO					**estimator = LogisticRegression() RIDGE**				
		Wal. Dokł.(%)	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**	**Wal. Dokł.(%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Logistic Regression**	55.62	29.63	27.27	21.43	29.95	58.58	22.22	29.41	35.71	21.70
2.	**MLP**	57.98	33.33	30.00	21.43	33.79	60.49	18.52	25.00	28.57	18.13
3.	**Random Forest**	58.13	37.04	40.00	42.86	36.13	58.87	25.93	28.57	28.57	25.82
4.	**SVC**	57.54	29.63	27.27	21.43	29.95	56.95	14.81	15.38	14.29	14.84
5.	**Quadratic Discriminant**	70.56	25.93	25.00	21.43	26.10	68.63	25.92	28.57	28.57	25.82

6.	**Linear Discriminant**	64.35	33.33	35.71	35.57	33.24	62.27	29.63	30.77	28.57	29.67
7.	**KNeighbors**	59.76	11.11	14.29	14.29	10.99	59.02	22.22	29.41	35.71	21.70
8.	**Decision Tree**	59.76	44.44	47.06	57.14	43.96	61.68	18.52	25.00	28.57	18.13
9.	**GaussianNB**	68.49	37.04	38.46	35.71	37.09	60.65	18.52	21.43	21.43	18.41
10.	**Gradient Boosting**	57.54	44.44	47.06	57.14	43.96	61.83	25.93	33.33	42.86	25.27

Wyniki eksperymentów na kolumnie decyzyjnej „Średnia harmoniczna liniowa" z wykorzystaniem metody RFE okazały się bardzo słabe. Można zaobserwować, że nie ma ani jednego eksperymentu, który pozwolił by osiągnąć metryki jakości na poziomie ponad 50% dla części testowej, a nawet tylko jeden z eksperymentów pozwolił uzyskać metryki na poziomie powyżej 40%. Patrząc na wyniki z części testowej najkorzystniej wypadł DecisionTreeClassifier. Ciekawym przypadkiem jest eksperyment wykorzystujący metodę QDA, który uzyskał około 70% dokładności w części walidacyjnej, lecz w części testowej osiągną zaledwie 26% dokładności.

Szczegóły dla eksperymentu QDA, RFE, LASSO:

Tabela 50. Macierz pomyłek – QDA, RFE, LASSO (Średnia harmoniczna liniowa).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	4 (TN)	9 (FP)
Rzeczywista klasa pozytywna	11 (FN)	3 (TP)

Obiekty błędnie rozpoznane:

['AR04_S', 'AR08_S', 'AR11_S', 'AR15_S', 'AR18_S', 'AR21_S', 'AR22_S', 'AR26_S', 'AR31_S', 'AR35_S', 'AR36_S', 'AR39_S', 'AR45_S', 'AR48_S', 'AR49_S', 'AR50_S', 'AR52_S', 'AR53_S', 'AR55_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot31283', 'spot41733', 'spot1708', 'spot1707', 'spot14262', 'spot1709', 'spot1710', 'spot1706', 'spot1711', 'spot1712', 'spot1717', 'spot50371', 'spot14808', 'spot1715', 'spot1716', 'spot1714', 'spot1713', 'spot1718', 'spot1697', 'spot1699', 'spot1696', 'spot18858', 'spot16757', 'spot1698', 'spot1687', 'spot1686', 'spot1684', 'spot1681',

'spot1682', 'spot1719', 'spot1680', 'spot1685', 'spot31880', 'spot1690', 'spot47498', 'spot1689', 'spot1688']

Szczegóły dla eksperymentu DecisionTreeClassifier, RFE, LASSO:

Tabela 51. Macierz pomyłek – DecisionTreeClassifier, RFE, LASSO (Średnia harmoniczna liniowa).

	Predykowana klasa negatywna	**Predykowana klasa pozytywna**
Rzeczywista klasa negatywna	4 (TN)	9 (FP)
Rzeczywista klasa pozytywna	6 (FN)	8 (TP)

Obiekty błędnie rozpoznane:

['AR11_S', 'AR21_S', 'AR22_S', 'AR23_S', 'AR26_S', 'AR30_S', 'AR31_S', 'AR36_S', 'AR39_S', 'AR48_S', 'AR49_S', 'AR50_S', 'AR52_S', 'AR53_S', 'AR57_S']

Ustanowiony konsensus dla wyselekcjonowanych cech:

['spot31283', 'spot41733', 'spot14262']

4.8. Obiekty błędnie rozpoznane

Na przestrzeni wykonywanych eksperymentów z wykorzystaniem klasyfikatorów, wiele obiektów było błędnie rozpoznawanych. Aby zweryfikować, czy istnieją obiekty monotonnie źle klasyfikowane, śledzono i identyfikowano obiekty przypisywane do konkretnych klas w wyniku przeprowadzonych klasyfikacji.

Interesującą informacją również mogłyby się okazać powody, dla których dany obiekt został błędnie rozpoznany. Dlatego też, dzięki możliwości wizualizacji drzewa decyzyjnego, ukazano drzewa przykładowych iteracji LOOCV z przeprowadzonych wcześniej eksperymentów, które zwróciły błędne przypisanie obiektu do klasy przez klasyfikatory oparte na drzewie decyzyjnym. Wizualizacje te pozwolą dokonać obserwacji, które warunki na drzewach są spełniane i dlaczego dany obiekt jest przydzielany do danej klasy, a nie innej.

4.8.1. Przypadki charakterystyczne

Dane, jakie udało się zebrać odnośnie błędnej klasyfikacji poszczególnych obiektów w badanym zbiorze danych, zgromadzono w tabeli poniżej (Tabela 39) i

posortowano malejąco według sumarycznej wielokrotności błędnej klasyfikacji. Dane zawarte w tabeli pochodzą z sześćdziesięciu eksperymentów, po dziesięć dla każdego z argumentów decyzyjnych. Wybierano tą konfigurację dla danej metody klasyfikacyjnej, która uzyskała najwyższy wynik na części walidacyjnej.

Tabela 52. Zestawienie i podsumowanie obiektów błędnie sklasyfikowanych.

Lp.	Id obiektu	Wielokrotność błędnej klasyfikacji obiektu						
		Kolagen I % powierzchni	Kolagen I siła	Wall area ratio RB1	Wall area ratio RB10	Wall thichness/ airway	Średnia harmoniczna liniowa	Suma
1.	AR50_S	6	10	7	1	7	10	41
2.	AR23_S	9	3	10	-	10	4	36
3.	AR30_S	1	1	7	9	7	9	34
4.	AR42_S	-	8	9	8	9	-	34
5.	AR47_S	9	9	8	-	8	-	34
6.	AR52_S	5	2	6	4	6	8	31
7.	AR57_S	7	6	8	-	8	2	31
8.	AR48_S	3	1	6	6	6	8	30
9.	AR55_S	4	6	9	-	9	2	30
10.	AR11_S	2	7	7	-	7	5	28
11.	AR12_S	6	9	5	2	5	-	27
12.	AR49_S	6	8	-	9	-	4	27
13.	AR53_S	-	1	8	-	8	10	27
14.	AR24_S	5	3	9	-	9	-	26
15.	AR09_S	2	5	4	10	4	-	25
16.	AR33_S	7	7	5	1	5	-	25
17.	AR35_S	2	6	7	-	7	2	24
18.	AR44_S	-	-	7	10	7	-	24
19.	AR08_S	1	-	6	10	6	-	23
20.	AR25_S	6	1	8	-	8	-	23
21.	AR43_S	10	5	4	-	4	-	23
22.	AR19_S	-	6	8	-	8	-	22
23.	AR39_S	6	4	4	-	4	4	22
24.	AR41_S	-	8	7	10	7	-	22
25.	AR51_S	-	-	10	1	10	-	21
26.	AR13_S	3	5	6	-	6	-	20
27.	AR56_S	-	5	6	1	6	2	20
28.	AR26_S	9	2	-	-	-	8	19
29.	AR27_S	3	5	3	-	3	5	19
30.	AR21_S	-	-	5	-	5	8	18
31.	AR45_S	1	8	4	-	4	1	18

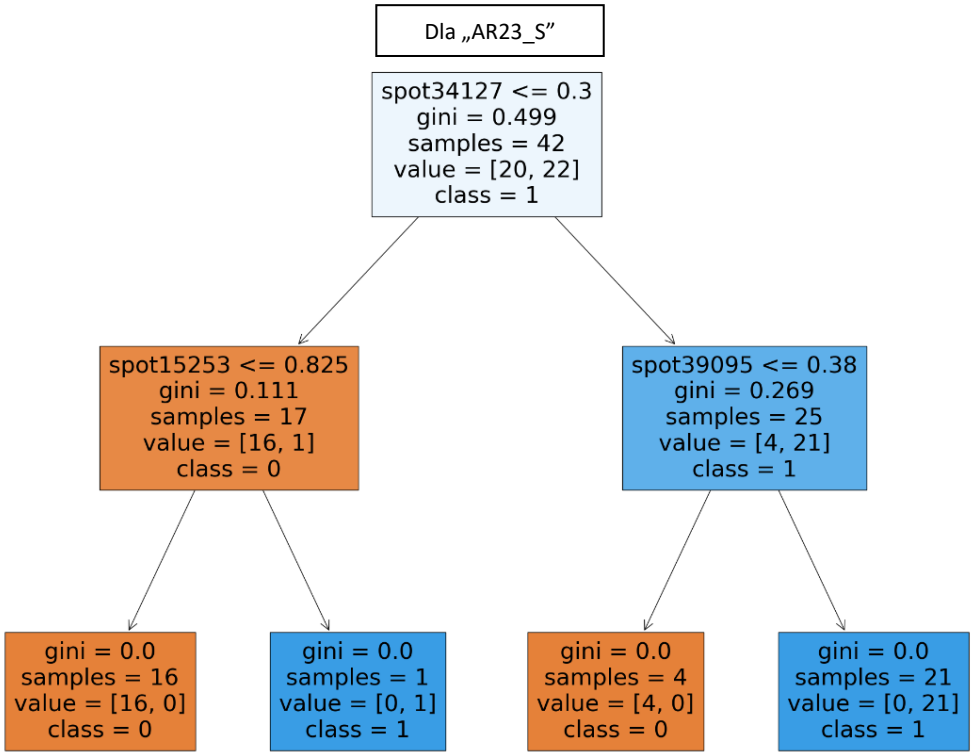
32.	AR04_S	2	5	2	3	2	3	17
33.	AR17_S	-	1	8	-	8	-	17
34.	AR28_S	2	7	4	-	4	-	17
35.	AR07_S	-	2	6	-	6	2	16
36.	AR22_S	-	1	7	-	7	1	16
37.	AR36_S	10	2	-	-	-	4	16
38.	AR18_S	-	4	3	-	3	2	12
39.	AR10_S	3	-	4	-	4	-	11
40.	AR16_S	1	1	-	-	-	7	9
41.	AR15_S	2	2	-	-	-	2	6
42.	AR31_S	-	1	-	-	-	5	6
43.	AR54_S	1	2	-	-	-	2	5
44.	AR29_S	-	2	-	-	-	-	2

Po bliższym przyjrzeniu się obiektom z największą liczbą błędnych klasyfikacji przez specjalistę mogłoby się okazać, że są one wyjątkami lub przypadkami znajdującymi się na pograniczu w kontekście rozpatrywanego w pracy problemu. Mogłoby się również okazać, że jest to mniejszy lub większy przypadek, gdyż próba badawcza rzędu 44 obiektów nie jest duża, a więc i do obserwacji wynikających z takiego badania należy podchodzić ostrożnie.

4.8.2. Powody błędnego rozpoznania na przykładach drzewa decyzyjnego

Wszystkie z przedstawionych poniżej drzew są dokładnym odtworzeniem poszczególnych iteracji LOOCV, wykorzystujących klasyfikator DecisionTree (drzewo decyzyjne), które błędnie sklasyfikowały obiekty w przeprowadzonych w badaniu eksperymentach.

Przykład drzewa decyzyjnego dla „Kolagen I % powierzchni”:

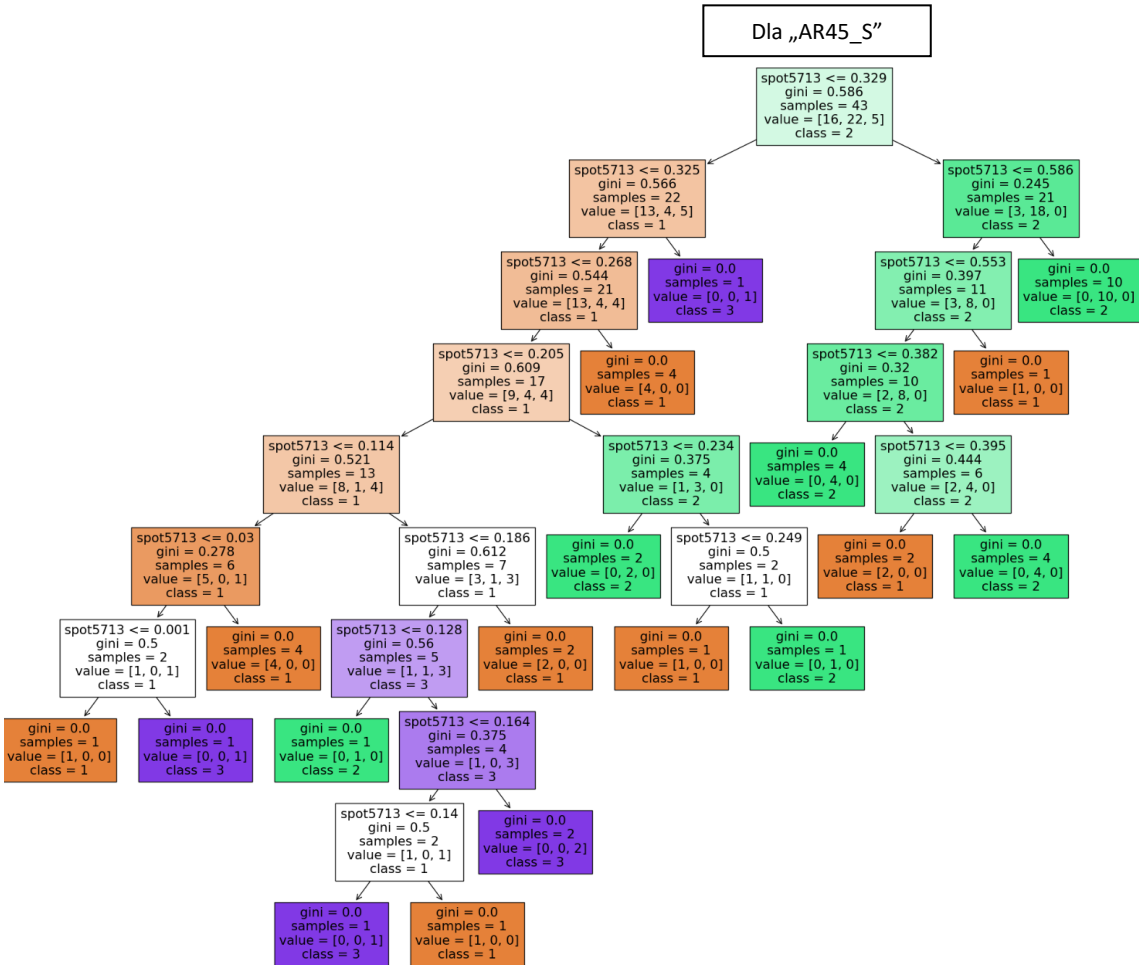


Rysunek 3. Drzewo decyzyjne (dla selektora SelectKBest f_classif) iteracji LOOCV z błędnym wynikiem obiektu „AR23_S” dla kolumny "Kolagen I % powierzchni".

Przykład ten obrazuje, dlaczego obiekt „AR23_S” został błędnie sklasyfikowany w przypadku klasyfikacji drzewem decyzyjnym na kolumnie decyzyjnej „Kolagen I % powierzchni”, gdzie wykorzystano metodę selekcji cech SelectKBest, *‘f_classif’*.

Wartość obiektu „AR23_S” dla cechy „spot15253” jest równa 0.09265822, dla cechy „spot34127” jest równa 0.15564831 oraz dla cechy „spot39095” jest równa 0.42852131. Pierwszy warunek na drzewie decyzyjnym został spełniony, więc obiekt trafia do lewego węzła. Drugi warunek również został spełniony, w wyniku czego obiekt został przypisany do klasy 0 (negatywnej), gdy w rzeczywistości obiekt należy do klasy 1 (pozytywnej).

Przykład drzewa decyzyjnego dla „Kolagen I siła”

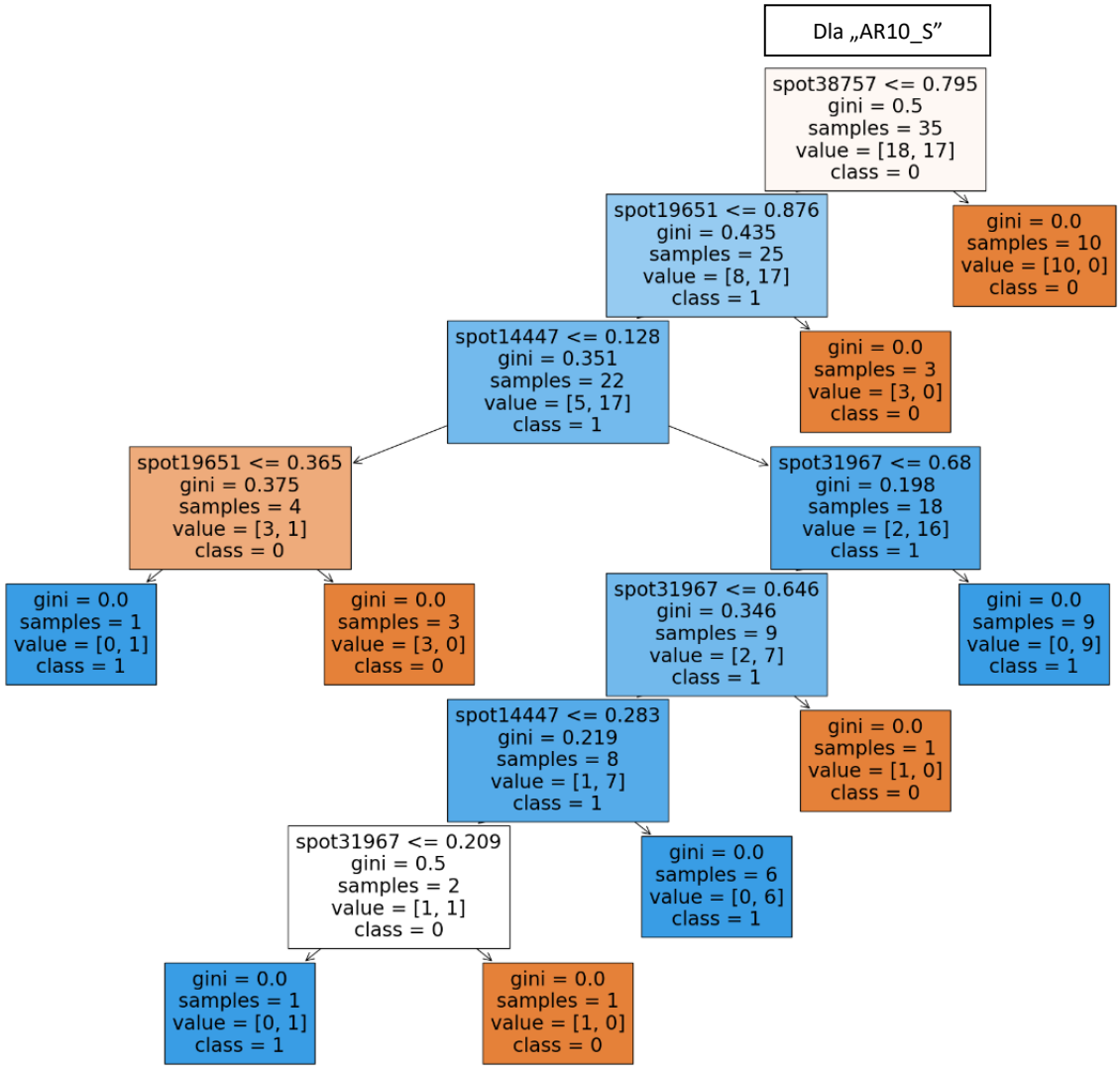


Rysunek 4. Drzewo decyzyjne (dla selektora RFE RIDGE) iteracji LOOCV z błędnym wynikiem obiektu „AR45_S” dla kolumny "Kolagen I siła".

Powyższy przykład obrazuje, dlaczego obiekt „AR45_S” został błędnie zidentyfikowany w wyniku klasyfikacji drzewem decyzyjnym na kolumnie decyzyjnej „Kolagen I siła”, wykorzystując metodę selekcji cech RFE RIDGE.

Wartość obiektu „AR45_S” dla cechy „spot5713” jest równa 0.39648455. Pierwszy warunek na drzewie decyzyjnym dla obiektu nie został spełniony, więc trafia z korzenia do prawego węzła. Drugi warunek został spełniony i obiekt ląduje w lewym węźle. Trzeci warunek jest również spełniony, więc obiekt trafia dalej do lewego węzła. Czwarty warunek nie został spełniony idąc głębiej do prawego węzła. Piąty warunek nie został spełniony, gdzie w rezultacie obiekt przypisano do klasy „2”. W rzeczywistości jednak obiekt należy do klasy „3”.

Przykład drzewa decyzyjnego dla „Wall area ratio RB1”

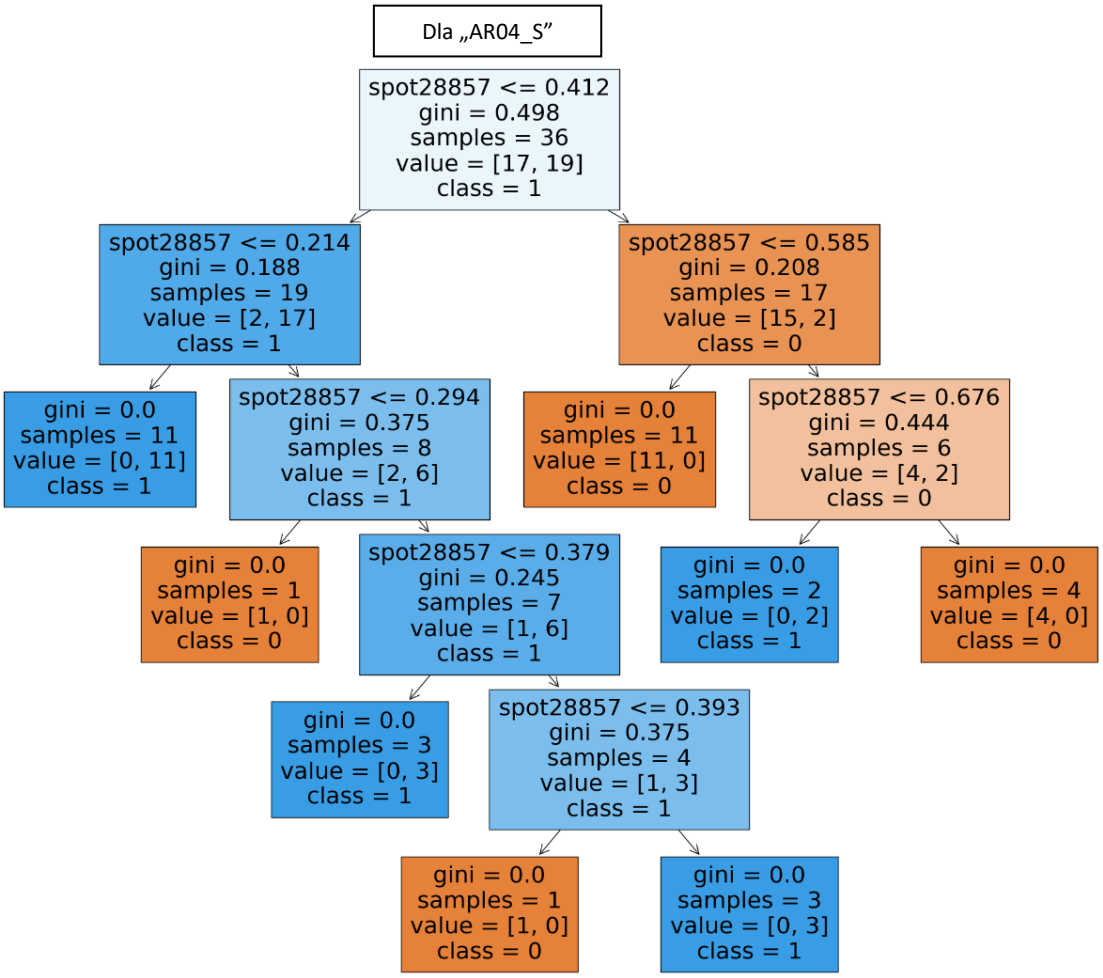


Rysunek 5. Drzewo decyzyjne (dla selektora RFE RIDGE) iteracji LOOCV z błędnym wynikiem obiektu „AR010_S” dla kolumny " Wall area ratio RB1".

Powyższy przykład obrazuje, dlaczego obiekt „AR10_S” został błędnie sklasyfikowany drzewem decyzyjnym na kolumnie decyzyjnej „Wall area ratio RB1”, gdzie wykorzystano metodę selekcji cech RFE RIDGE.

Wartość obiektu „AR010_S” dla cechy „spot14447” jest równa 0.62521524, dla „spot19651” jest równa 0.48231881, dla „spot31967” jest równa 0.48500506 oraz dla „spot38757” jest równa 0.79738251. Obiekt nie spełnia pierwszego warunku na drzewie decyzyjnym, w wyniku czego obiekt od razu zostaje przypisany do klasy 0 (negatywnej), a w rzeczywistości należy on do klasy 1 (pozytywnej).

Przykład drzewa decyzyjnego dla „Wall area ratio RB10”

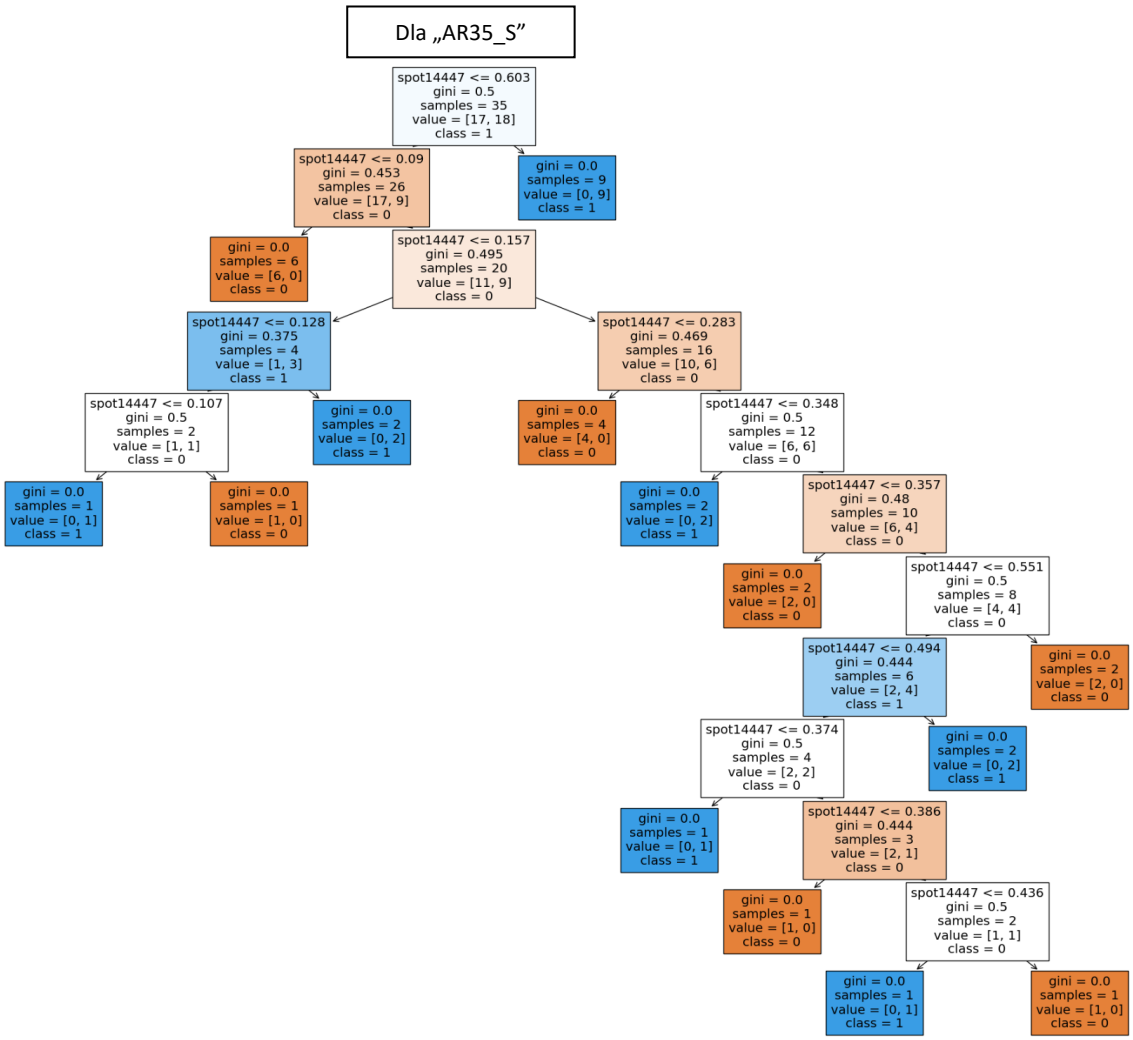


Rysunek 6. Drzewo decyzyjne (dla selektora SelectKBest f_classif) iteracji LOOCV z błędnym wynikiem obiektu „AR04_S” dla kolumny " Wall area ratio RB10".

Powyższe drzewo decyzyjne obrazuje, dlaczego obiekt „AR04_S” został błędnie sklasyfikowane na kolumnie decyzyjnej „Wall area ratio RB10”, gdzie wykorzystano metodę selekcji cech SelectKBest ‘*f_classif*’.

Wartość obiektu „AR04_S” dla cechy „spot28857” jest równa 0.64788336. Obiekt nie spełnia pierwszego warunku na drzewie decyzyjnym, więc ląduje w prawym węźle. Drugi warunek również nie jest spełniony, więc obiekt ląduje ponownie w prawym węźle. Trzeci warunek jest już spełniony, w wyniku czego obiekt jest przypisany do klasy 1 (pozytywnej), ale w rzeczywistości przynależy do klasy 0 (negatywnej).

Przykład drzewa decyzyjnego dla „Wall thichness/airway diameter ratio RB1”

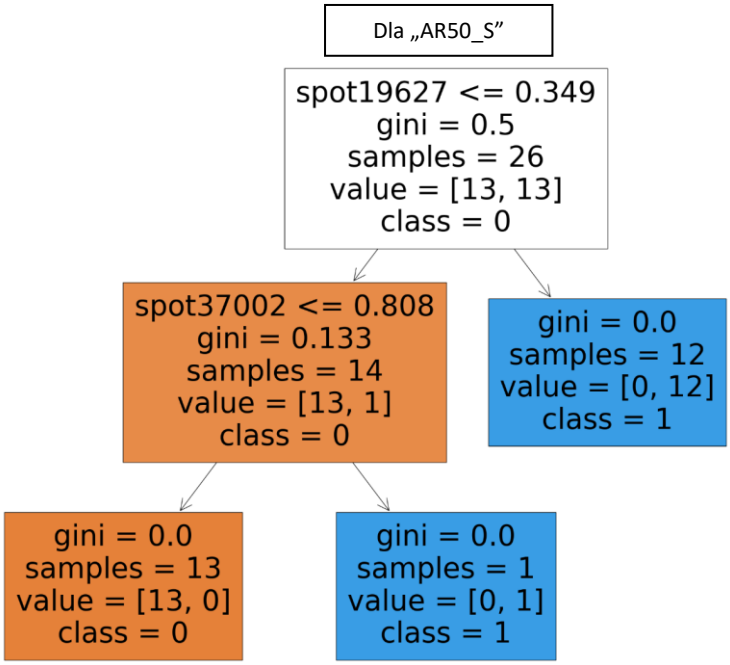


Rysunek 7. Drzewo decyzyjne (dla selektora RFE RIDGE) iteracji LOOCV z błędnym wynikiem obiektu „AR35_S” dla kolumny "Wall thichness/airway diameter ratio RB1".

Powyższe drzewo decyzyjne przedstawia, dlaczego obiekt „AR35_S” został błędnie sklasyfikowany na kolumnie decyzyjnej „Wall thichness/airway diameter ratio RB1”, gdzie zastosowano metodę selekcji cech RFE RIDGE.

Wartość obiektu „AR35_S” dla cechy „spot14447” jest równa 0.65342385. Obiekt nie spełnił pierwszego warunku i od razu został przypisany do klasy 1 (pozytywnej), ale w rzeczywistości przynależy do klasy 0 (negatywnej).

Przykład drzewa decyzyjnego dla „Średnia harmoniczna liniowa”



Rysunek 8. Drzewo decyzyjne (dla selektora SelectFromModel) iteracji LOOCV z błędnym wynikiem obiektu „AR50_S” dla kolumny "Średnia harmoniczna liniowa ".

Powyższe drzewo decyzyjne obrazuje, dlaczego obiekt „AR50_S” został błędnie sklasyfikowany na kolumnie decyzyjnej „Średnia harmoniczna liniowa”, gdzie zastosowano metodę selekcji cech SelectFromModel.

Wartość obiektu „AR50_S” dla cechy „spot19627” jest równa 0.1356942, a dla cechy „spot37002” jest równa 0.71220492. Obiekt spełnia zarówno pierwszy jak i drugi warunek, dlatego został przypisany do klasy 0 (negatywnej), ale w rzeczywistości należy do klasy 1 (pozytywnej).

5. Podsumowanie

Badaniom zostało poddanych sześć atrybutów decyzyjnych, a każdy z nich był eksplorowany dziesięcioma różnymi algorytmami klasyfikacyjnymi, wykorzystując w procesie trzy metody selekcji cech. Aby dobrać ostateczne konfiguracje do każdej z kolumny kierowano się najwyższym wynikiem z części walidacyjnej, który powinien być na poziomie wyższym niż 80%. Następnie weryfikowano patrząc na wyniki z części testowej, jak model poradził sobie z klasyfikacją nowych obiektów. Poziom wyników na części testowej również musi być wyższy niż 80% dla wszystkich z prezentowanych metryk jakości. W tabeli poniżej (Tabela 52) znajdują się zestawienie konfiguracji, które uznano za ostateczne dla każdej z kolumn decyzyjnych z osobna.

Tabela 53. Zestawienie finalnych konfiguracji modeli, dla każdego argumentu decyzyjnego.

Zestawienie ostatecznych konfiguracji modeli, dla każdego argumentu decyzyjnego (na podstawie danych z wszystkich opisanych w pracy eksperymentów z sześciu kolumn decyzyjnych)								
Lp.	**Argument decyzyjny**	**Metoda klasyfikacyjna**	**Metoda selekcji cech**	**Wal. Dokł. (%) / f_score weighted (%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Kolagen I % powierzchni**	**MLP**	**SelectKBest f_classif**	80.10	62.79	65.22	65.22	62.61
2.	**Kolagen I siła**	**MLP**	**RFE LASSO**	75.50	63.64	64.03	63.64	51.01
3.	**Wall area ratio RB1**	**Quadratic Discrminant Analysis**	**RFE RIDGE**	64.40	44.44	45.00	50.00	44.44
4.	**Wall area ratio RB10**	**KNeighbors**	**SelectKBest f_classif**	87.96	83.78	84.21	84.21	83.77
5.	**Wall thichness/ airway**	**Quadratic Discrminant Analysis**	**RFE RIDGE**	64.40	44.44	45.00	50.00	44.44
6.	**Średnia harmoniczna liniowa**	**Gradient Boosting Classifier**	**SelectFromModel (RandomForest)**	78.40	62.96	64.28	64.28	62.29

Z wszystkich argumentów decyzyjnych tylko kolumna „Wall area ratio RB10" pozwoliła uzyskać model dobrej jakości. Wynik jaki otrzymano dla tego argumentu nie jest idealny lecz pozwolił przebić próg jakości minimalnej (80%), aby uznać utworzony model za dobrej jakości.

Warto zwrócić jednak uwagę na argument decyzyjny „Kolagen I % powierzchni". Niewiele w tym przypadku brakowało, aby to metoda GradientBoostingClassifier znalazła się w ostatecznej konfiguracji, wykorzystując w procesie selekcji cech metodę SelectKBest (f_classif). Eksperyment ten prawie przebił próg 80% na części walidacyjnej, i udało się go przebić na części testowej. Gdyby dokonano głębszej eksploracji tej kolumny, istniałaby szansa utworzenia modelu dobrej jakości również dla niej.

Warto również zwrócić uwagę, iż dla argumentu decyzyjnego „Wall area ratio RB10", powstało łącznie pięć konfiguracji pozwalających otrzymać model dobej jakości. Pozostałymi czterema, na które warto zwrócić uwagę są:

Tabela 54. Zestawienie pozostałych konfiguracji dla „Wall area ratio RB10" pozwalających otrzymać dobre wyniki.

Pozostałe konfiguracje dla „Wall area ratio RB10" pozwalające otrzymać dobre wyniki								
Lp.	**Argument decyzyjny**	**Metoda klasyfikacyjna**	**Metoda selekcji cech**	**Wal. Dokł. (%)**	**Dokł. (%)**	**Prec. (%)**	**Czuł. (%)**	**AUC (%)**
1.	**Wall area ratio RB10**	**KNeighbors**	**RFE LASSO**	87.27	86.49	85.00	89.47	86.40
2.	**Wall area ratio RB10**	**RandomForest**	**RFE LASSO**	84.56	83.78	80.95	89.47	83.63
3.	**Wall area ratio RB10**	**DecisionTree Classifier**	**RFE LASSO**	84.64	83.78	80.95	80.47	83.63
4.	**Wall area ratio RB10**	**Gradient Boosting Classifier**	**RFE LASSO**	84.80	83.78	80.95	80.47	83.63

Podsumowując, próba udowodnienia, iż eksploracja danych mikromacierzowych DNA pozwoli na utworzenie modelu lub wielu modeli dobrej jakości, wykazując, że istnieje informacja płynąca z ekspresji genów zawarta w mikromacierzy DNA, która w dobrym stopniu wyjaśnia cechy kliniczne wskazujące na występowanie remodelingu oskrzeli, została zakończona powodzeniem lecz tylko w kontekście argumentu „Wall area ratio RB10".

Selektorem cech, który wybierał średnio najkorzystniejsze cechy z oryginalnego zbioru mikromacierzy DNA na przestrzeni wszystkich eksperymentów okazał się RecursiveFeaturesElimination (RFE), którego estymatorem była metoda LogisticRegression, wykorzystująca metodę regularyzacji LASSO. Drugi, który również radził sobie bardzo dobrze to SelectKBest – f_classif. Zapewne istnieją inne metody

selekcji cech, które poradziłyby sobie równie dobrze, albo i nawet lepiej, lecz ze względu na ograniczone zasoby czasu i mocy obliczeniowej, w pracy przetestowano tylko pięć konfiguracji metod selekcji cech, przedstawionych w części teoretycznej i później aplikowanych w eksperymentach.

W przypadku metod klasyfikacyjnych ciężko jest wskazać konkretną, która by radziła sobie z mikromacierzami najlepiej obejmując wszystkie przeprowadzone eksperymenty w pracy. Każda z wykorzystanych metod miała swoje wzloty i upadki. Tak samo jak w przypadku metod selekcji cech, mogą istnieć lepiej radzące sobie metody klasyfikacyjne, które nie pojawiły się w pracy i które mogłaby lepiej poradzić sobie z mikromacierzami DNA.

Wiele prac i publikacji naukowych prezentujących eksplorację danych mikromacierzowych DNA demonstrują, jak wiele pożytecznych informacji można pozyskać, łącząc uczenie maszynowe z technologią pozyskiwania ekspresji genów i możliwością przechowywania ich w formie cyfrowej. W przyszłości, gdy nauka będzie dysponowała większą próbą mikromacierzy DNA, pochodzących od chorych osób na różne przypadłości, tworzone modele klasyfikacyjne będą coraz dokładniejsze. Dobrze wyuczone modele mogłyby stać się częścią standardowej procedury diagnostycznej dla wielu osób, dla których informacja o zagrożeniu ciężką chorobą będzie się mogła okazać ocalającą życie. Warto więc badać mikromacierze DNA i poszerzać wiedzę na tym obszarze nauki.

6. Wykorzystane technologie

Python (wersja 3.9) - popularny język programowania wysokiego poziomu. Jest chwalony za swoją przejrzystość, czytelność i zdolność adaptacji. Python został opracowany przez Guido van Rossum i pierwotnie udostępniony w 1991 roku. Od tego czasu stał się jednym z najczęściej używanych języków programowania w różnych branżach, w tym w tworzeniu stron internetowych, analizie danych, sztucznej inteligencji, obliczeniach naukowych i automatyzacji. Według rankingu TIOBE w obecnej chwili (maj 2023r.) Python pozycjonuje się na pierwszym miejscu jako najpopularniejszy język programowania [15][16].

NumPy – jest jednym z podstawowych pakietów Pythona wykorzystywanym do obliczeń numerycznych. Oferuje on obsługę dużych, wielowymiarowych tablic i macierzy, a także szereg operacji matematycznych umożliwiających efektywne korzystanie z tych tablic. Biblioteka NumPy Pythona jest kluczowym elementem do przeprowadzania analizy danych i obliczeń naukowych [17].

Pandas - pakiet Pandas w Pythonie to elastyczne i rozwinięte narzędzie do manipulowania i analizowania danych. Jego bogate funkcje oraz przejrzysta i wyrazista składnia sprawiają, że jest to kluczowe narzędzie dla analityków danych, analityków i badaczy pracujących z danymi strukturalnymi. Najczęściej wykorzystywana do tworzenia obiektów typu DataFrame [18].

Pyplot - podmoduł biblioteki Pythona matplotlib. Składa się z wielu funkcji i metod do kreślenia podstawowych wykresów i wizualizacji [19].

ScikitLearn – dobrze znana biblioteka, która oferuje liczne narzędzia i algorytmy do różnych zadań uczenia maszynowego (ML). Jest zbudowana na bazie NumPy, SciPy i Matplotlib. Charakteryzuje się łatwością w użyciu, szeroką i dobrze opisaną dokumentacją, jest skuteczna i otwarta zarówno dla amatorów, jak i doświadczonych ekspertów. Ze względu na swoje możliwości adaptacyjne, łatwość obsługi i rozbudowaną funkcjonalność, Scikit-learn jest bardzo często stosowany w społeczności

uczenia maszynowego. Biblioteka stanowi ważne narzędzie dla różnych przedsięwzięć związanych ze sztuczną inteligencją, począwszy od prostych kwestii porządkowych, a skończywszy na złożonej strukturze i ocenie modeli [20].

Imblearn – to biblioteka Pythona utworzona w celu radzenia sobie z problemami, pojawiającymi się podczas pracy z zestawami danych, które są niezrównoważone. Niezrównoważony zbiór danych to taki, w którym jedna klasa ma znacznie wyższą lub niższą liczbę instancji niż inne, co skutkuje tendencyjną wydajnością modelu. Biblioteka pozwala stosować różne strategie dbania o niezrównoważone informacje i łagodzenia efektu nierówności klas. Płynnie integruje się z biblioteką scikit-learn i rozszerza jej możliwości [21].

Bibliografia

[1] J.C. Venter, M.D. Adams, E.W. Myers, P.W. Li, R.J. Mural, G.G. Sutton, H.O. Smith, M. Yandell, C.A. Evans, R.A. Holt, „The sequence of the human genome", Science 291 (2001) 1304–1351. https://www.science.org/doi/abs/10.1126/science.1058040

[2] E.S. Lander, et al., „Initial sequencing and analysis of the human genome", Nature 409 (2001) 860–921. https://www.nature.com/articles/35057062

[3] V. Trevino, F. Falciani, H.A. Barrera-Salda, „DNA microarrays: a powerful genomic tool for biomedical and clinical research", Mol. Med. (Cambridge, MA) 13 (2007) 527–541. https://molmed.biomedcentral.com/articles/10.2119/2006-00107.Trevino

[4] Beatriz A. Garroa, Katya Rodríguez, Roberto A. Vázquez, „Classification of DNA microarrays using artificial neural networks and ABC algorithm", Applied Soft Computing 38 (2016) 548-560. https://www.sciencedirect.com/science/article/abs/pii/S1568494615006171?via%3Dihub

[5] Heinz Fehrenbach, Christina Wagner, Michael Wegmann, „Airway remodeling in asthma: what really matters", Cell Tissue Res. 2017; 367(3): 551–569. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5320023/

[6] Lumei Liu, Brooke Stephens, Maxwell Bergman, Anne May, Tendy Chiang, „Role of Collagen in Airway Mechanics", Bioengineering (Basel). 2021 Jan; 8(1): 13. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7830870/

[7] Sumit Gupta, Salman Siddiqui, Pranab Haldar, James J Entwisle, Dean Mawby, Andrew J Wardlaw, Peter Bradding, Ian D Pavord, Ruth H Green, Christopher E, „Brightling, Quantitative analysis of high-resolution computed tomography scans in severe asthma subphenotypes", Thorax 2010;65:775e781. https://www.scienceopen.com/document_file/e4da1707-afea-4675-b0fc-4b2dafcf1e82/PubMedCentral/e4da1707-afea-4675-b0fc-4b2dafcf1e82.pdf

[8] Sara Brown, „Machine learning, explained”, 21/05/2021, MIT Management Sloan School
https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

[9] Mayank Banoula, „Machine Learning Steps: A Complete Guide”, 16/02/2023, Simplilearn
https://www.simplilearn.com/tutorials/machine-learning-tutorial/machine-learning-steps

[10] Hema Shekar Basavegowda, Guesh Dagnew, „Deep learning approach for microarray cancer data classification”, 2019; 2468-2322.
https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/trit.2019.0028

[11] Kıvanç GÜÇKIRAN, Ismail CANTÜRK, Lale ÖZYILMAZ, „DNA Microarray Gene Expression Data Classification Using SVM, MLP, and RF with Feature Selection Methods Relief and LASSO”, Journal of Natural and Applied Sciences, Volume 23, Issue 1, 126-132, 2019
https://dergipark.org.tr/tr/download/article-file/702328

[12] Sarang Narkhede, „Understanding AUC - ROC Curve”, Towards Data Science Jun 2018
https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

[13] Bradley Efron, „Estimating the Error Rate of a Prediction Rule: Improvement on Cross-Validation”, Journal of the American Statistical Association , Vol. 78, No. 382 (Jun., 1983)
https://sites.stat.washington.edu/courses/stat527/s14/readings/Efron_jasa1983.pdf

[14] Bradley Efron, Robert J. Tibshirani, „An Introduction to the Bootstrap”, CHAPMAN & HALL/CRC, 1993

[15] TIOBE Index
https://www.tiobe.com/tiobe-index/

[16] Dokumentacja języka Python
https://www.python.org/doc/

[17] Dokumentacja biblioteki NumPy

https://numpy.org/doc/stable/

[18] Dokumentacja biblioteki Pandas

https://pandas.pydata.org/docs/

[19] Dokumentacja biblioteki matplotlib

https://matplotlib.org/stable/

[20] Dokumentacja biblioteki scikit-learn

https://scikit-learn.org/stable/

[21] Dokumentacja biblioteki imbalanced-learn

https://imbalanced-learn.org/stable/

[22] Alex Smola, S.V.N. Vishwanathan, „Introduction to Machine Learning", Cambridge University Press 2008

https://alex.smola.org/drafts/thebook.pdf

[23] Sebastián Maldonado, Julio López, Carla Vairetti, „An alternative SMOTE oversampling strategy for high-dimensional datasets", Applied Soft Computing Volume 76, March 2019, Pages 380-389

https://www.sciencedirect.com/science/article/abs/pii/S1568494618307130

[24] Sarang Narkhede, „Understanding Confusion Matrix", Towards Data Science May 2018

https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

[25] Stanisława Bazan-Socha, Sylwia Buregwa-Czuma, Bogdan Jakiela, Lech Zaręba, Izabela Zawlik, Aleksander Myszka, Jerzy Soja, Krzysztof Okoń, Jacek Zarychta, Paweł Kozlik, Sylwia Dziedzina, Agnieszka Padjas, Krzysztof Wójcik, Michał Kępski, Jan G. Bazan, „Reticular Basement Membrane Thickness Is Associated with Growth- and Fibrosis-Promoting Airway Transcriptome Profile-Study in Asthma Patients", Int J Mol Sci. 2021 Feb; 22(3): 998

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7863966/

[26] Nosayba Al-Azzam, Ibrahem Shatnawi Ph.D, PE, PMP, PTOE, „Comparing supervised and semi-supervised Machine Learning Models on Diagnosing Breast Cancer ", Annals of Medicine and Surger Volume 62, February 2021, Pages 53-64

https://www.sciencedirect.com/science/article/pii/S2049080120305604

[27] Marcel Neunhoeffer, Sebastian Sternberg „How Cross-Validation Can Go Wrong and What to Do About It", Political Analysis , Volume 27 , Issue 1 , January 2019 , pp. 101 – 106

https://doi.org/10.1017/pan.2018.39

[28] Tam D Tran-The, „How cross-validation can go wrong and how to fix it", Towards Data Science Jul 7, 2022

https://towardsdatascience.com/how-cross-validation-can-go-wrong-and-how-to-fix-it-feature-selection-use-case-with-sample-code-abf928be9080

[29] Soledad Galli, „Feature selection with Lasso in Python", Train In Data, 16 Aug 2022

https://www.blog.trainindata.com/lasso-feature-selection-with-python/

Spis ilustracji

Spis tabel

Załącznik nr 2 do Zarządzenia nr 228/2021 Rektora Uniwersytetu Rzeszowskiego z dnia 1 grudnia 2021 roku w sprawie ustalenia procedury antyplagiatowej w Uniwersytecie Rzeszowskim

OŚWIADCZENIE STUDENTA O SAMODZIELNOŚCI PRACY

DANIEL MIKOŁAJ CZYŻ
Imię (imiona) i nazwisko studenta

Kolegium Nauk PRZYRODNICZYCH

INFORMATYKA
Nazwa kierunku

106530
Numer albumu:

1. Oświadczam, że moja praca dyplomowa pt.: PRZEWIDYWANIE REMODELINGU OSUSZELI W OPARCIU O DANE MIKROMACIERZOWE

   1) została przygotowana przeze mnie samodzielnie*,
   2) nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2021 r., poz. 1062) oraz dóbr osobistych chronionych prawem cywilnym,
   3) nie zawiera danych i informacji, które uzyskałem/am w sposób niedozwolony,
   4) nie była podstawą nadania dyplomu uczelni wyższej ani mnie, ani innej osobie.

2. Jednocześnie *wyrażam zgodę/~~nie wyrażam zgody~~* ** na udostępnienie mojej pracy dyplomowej do celów naukowo-badawczych z poszanowaniem przepisów ustawy o prawie autorskim i prawach pokrewnych.

RZESZÓW, 22.06.2023 r. Daniel Czyż
(miejscowość, data) *(czytelny podpis studenta)*

* *uwzględniając merytoryczny wkład promotora pracy*
** - *niepotrzebne skreślić*