

Uniwersytet Warszawski
Wydział Fizyki

Joanna Nowakowska
Nr albumu: 370486

**Testowanie skuteczności algorytmu
klasyfikującego obrazy medyczne
wykorzystującego parametryzację
za pomocą konwolucyjnej sieci
neuronowej**

Praca licencjacka
na kierunku Zastosowania Fizyki w Biologii i Medycynie
specjalność Fizyka Medyczna

Praca wykonana pod kierunkiem
dra Józefa Gintera
Zakład Fizyki Biomedycznej
Instytut Fizyki Doświadczalnej

Warszawa, <miesiąc-i-rok-złożenia-pracy>

Oświadczenie kierującego pracą

Oświadczam, że niniejsza praca została przygotowana pod moim kierunkiem i stwierdzam, że spełnia ona warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data

Podpis kierującego pracą

Oświadczenie autora (autorów) pracy

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

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

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

Data

Podpis autora (autorów) pracy

Streszczenie

<Krótkie (maks. 800 znaków) streszczenie pracy, na przykład:

Lorem ipsum – tekst składający się z łacińskich i quasi-łacińskich wyrazów, mający korzenie w klasycznej łacinie, wzorowany na fragmencie traktatu Cyserona „O granicach dobra i zła” (De finibus bonorum et malorum) napisanego w 45 r. p.n.e. Tekst jest stosowany do demonstracji krojów pisma (czcionek, fontów), kompozycji kolumny itp. Po raz pierwszy został użyty przez nieznanego drukarza w XVI w.

Tekst w obcym języku pozwala skoncentrować uwagę na wizualnych aspektach tekstu, a nie jego znaczeniu.

Cytat z https://pl.wikipedia.org/wiki/Lorem_ipsum >

Słowa kluczowe

czerniak, SVM, klasyfikator, sieć neuronowa

Dziedzina pracy (kody wg programu Socrates-Erasmus)

13.2 Fizyka

Tytuł pracy w języku angielskim

<Parametrization of a convolutional neural network for testing the effectiveness of a classifying algorithm on medical images>

Spis treści

Cel pracy	3
1. Wstęp	4
1.1. Sztuczne sieci neuronowe	5
1.2. Algorytm SVM (ang. <i>Support Vector Machine</i>)	6
1.3. Klasyfikacja danych i ocena jakości klasyfikatora	8
1.4. Czerniak - różnicowanie zmian barwnikowych w praktyce klinicznej	10
2. Dane eksperymentalne	12
3. Metodologia	14
3.1. Oprogramowanie	14
3.2. Metodyka pracy	14
4. Wyniki	16
4.1. Klasyfikator losowy	16
4.1.1. Klasyfikator losowy dla zbioru zrównoważonego	17
4.1.2. Klasyfikator losowy dla zbioru niezrównoważonego	18
4.2. Klasyfikator SVM	18
4.2.1. Zbiory TR i TE zrównoważone	18
4.2.2. Zbiory TR i TE niezrównoważone	20
4.2.3. Zbiór TR zrównoważony i TE niezrównoważony	22
4.2.4. Zbiór TR niezrównoważony i TE zrównoważony	24
4.3. Perceptron	25
4.3.1. Zbiór TR niezrównoważony i TE zrównoważony	27
5. Dyskusja	30
6. Podsumowanie	31

Cel pracy

Rozdział 1

Wstęp

Nowoczesne technologie oraz coraz szybsze komputery są obecnie kluczowymi czynnikami napędzającymi rozwój niemal wszystkich gałęzi przemysłu. Jedną z najważniejszych branż, która dzięki temu może się rozwijać jest medycyna. Wprowadzane na bieżąco medyczne innowacje pozwalają ratować życie coraz większej liczbie ludzi na całym świecie.

Jednym z najbardziej dynamicznie rozwijających się obecnie obszarów informatyki jest sztuczna inteligencja, w tym jedna z jej gałęzi - tzw. sieci neuronowe (ang. *neural networks*). Służą one głównie do klasyfikacji i rozpoznawania obiektów oraz analizy danych. Posiadają wiele zastosowań, m.in. w technice – do analizy obrazów i przetwarzania sygnałów, czy w ekonomii do tworzenia prognoz i optymalizacji decyzji gospodarczych.

Dzięki ich zdolnościom do ciągłego uczenia się, sieci wykorzystuje się do przewidywania wyników na podstawie zgromadzonych danych wejściowych. Z tego powodu konwolucyjne sieci neuronowe (ang. *convolutional neural networks*), jeden z typów sieci neuronowych (o których będzie mowa w kolejnych rozdziałach), wykazują duży potencjał w obszarze diagnostyki medycznej. By wspomóc lekarzy, systemy te mogą przeprowadzać dogłębną analizę szerokiego zakresu danych na temat danego pacjenta i na tej podstawie dokonywać decyzji na temat jego stanu zdrowia. Takie zastosowanie sieci neuronowych odciążałoby lekarzy i, przy odpowiednio dobranych sposobach klasyfikacji, zwiększyło liczbę prawidłowo postawionych diagnoz.

Jednym z obszarów diagnostyki medycznej, w którym wciąż w dużej mierze diagnozę opiera się na subiektywnej ocenie lekarza jest dermatologia. Przy coraz większej liczbie ludności mającej problemy ze zmianami skórными, szybka i odpowiednia diagnoza staje się kluczowym elementem procesu leczenia. Szczególnie niebezpieczne są czerniaki – złośliwe nowotwory skóry, których częstość występowania bardzo wzrosła w przeciągu ostatnich 30 lat[10]. Są obecnie najczęstszą przyczyną zgonów spowodowanych nowotworami złośliwymi skóry. Z tego powodu w ostatnich latach wzrosło zainteresowanie badaniami mającymi na celu opracowanie algorytmów wspomagających rozpoznawanie czerniaków. Między innymi w 2017 roku na Międzynarodowym Sympozjum Obrazowania Biomedycznego (ISBI – ang. *International Symposium on Biomedical Imaging*) zorganizowano konkurs przy współpracy z *International Skin Imaging Collaboration* (ISIC), którego celem była prezentacja algorytmów klasyfikujących wyżej opisane zmiany skórne. Jako danych wejściowych użyto zestawu zdjęć zmian barwnikowych łagodnych oraz złośliwych udostępnionych przez stronę ISIC.

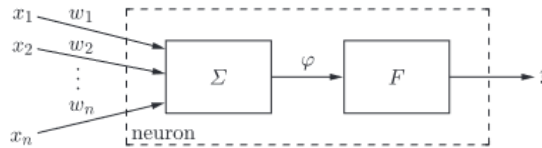
W pracy użyto tego samego zbioru zdjęć w celu przetestowania różnych algorytmów klasyfikujących obrazy medyczne.

1.1. Sztuczne sieci neuronowe

Sztuczne sieci neuronowe (ang. *Artificial Neural Networks*, ANNs) są to pewne struktury matematyczne wzorujące się na działaniu neuronów biologicznych. Naśladując zdolności ludzkiego umysłu, mają one na celu naukę podejmowania odpowiednich decyzji, klasyfikacji wzorów czy formułowania prognoz na podstawie wcześniej zgromadzonych danych i dokonania analizy retrospektywnej. Przy konstruowaniu sieci należy na początku zadać pewne ograniczenia(?) związane m.in z architekturą sieci (typ, funkcja aktywacji czy liczba neuronów w warstwach), sposobem uczenia sieci czy metody analizy i weryfikacji otrzymanych wyników.

Sieć neuronowa składa się z pojedynczych neuronów pogrupowanych w warstwy. Sieć można podzielić na trzy główne części: warstwę wejściową, warstwy ukryte oraz warstwę wyjściową. Warstwa wejściowa wprowadza dane do sieci. Ostatnia warstwa, warstwa wyjściowa, służy do wyznaczania wartości wyjściowych sieci. Pomiedzy pierwszą i ostatnią warstwą znajdują się warstwy ukryte. Neurony znajdujące się w warstwach ukrytych mają za zadanie przetwarzać informacje wejściowe poprzez kolejne etapy uczenia na informacje wyjściowe. Neurony, które znajdują się w sąsiadujących warstwach są ze sobą połączone. Poprzez te połączenia następuje przesył informacji przez sieć. Budowę oraz działanie pojedynczego neuronu przedstawiono na Rysunku 1.1.

Rysunek 1.1: Model działania pojedynczego neuronu[2]



Pojedynczy neuron, jak na powyższym rysunku, jest przetwornikiem sygnału - na wejściu dostaje wektor x danych wejściowych (sygnałów) pochodzących z warstwy wejściowej sieci lub z neuronów warstwy poprzedniej. Każdemu połączeniu przypisywana jest waga w_i dobierana w procesie uczenia. W neuronie sumowane są dane wejściowe i odpowiadające im wagi. Suma iloczynów daje sumaryczne wejście s neuronu:

$$s = \sum_{i=0}^n x_i * w_i = w^T x, \quad (1.1)$$

gdzie x_i - kolejne sygnały wejściowe, w_i - odpowiadające im wagi, n - liczba cech (sygnałów) na wejściu. Wartość s jest argumentem twz. funkcji aktywacji a według której obliczana jest wartość wyjściowa neuronu, określonej wzorem

$$a = f(s). \quad (1.2)$$

Stosuje się różne funkcje aktywacji, m.in.:

- funkcję progową:

$$f(s, p) = \begin{cases} 0 & \text{dla } s < p \\ 1 & \text{dla } s \geq p \end{cases}, \quad (1.3)$$

gdzie p - wartość progowa.

- funkcję logistyczną:

$$f(s) = \frac{1}{1 + \exp^{-s}}, \quad (1.4)$$

- tangens hiperboliczny:

$$f(s) = \frac{\exp^s - \exp^{-s}}{\exp^s + \exp^{-s}}. \quad (1.5)$$

Sam proces uczenia maszynowego polega na ciągłym poprawianiu i dobieraniu odpowiednich wag w_i w taki sposób, aby sygnał przechodzący przez sieć dawał na wyjściu klasyfikację zbioru treningowego ze zminimalizowanym błędem. Błąd jest minimalizowany przy pomocy tzw. funkcji kosztu, która oblicza różnicę między wartościami predykowanymi a prawdziwymi. Mając zbiór uczący w postaci

$$(X^{(j)}, z^{(j)})_{j=1, \dots, N} \quad (1.6)$$

gdzie X - zbiór uczący, z - odpowiadająca mu prawidłowa klasyfikacja, funkcja kosztu $J(w)$ może np. obliczać błąd średniokwadratowy przyjmując postać:

$$J(w) = \frac{1}{2} \sum_{j=1}^N (y^{(j)} - z^{(j)})^2 \quad (1.7)$$

gdzie y i z to odpowiednio predykowane i prawdziwe wartości dla zbioru uczącego. W celu dobrania odpowiednich wag stosuje się propagację wsteczną, która działa w oparciu o algorytm zwany minimalizacją gradientową:

$$w_j := w_j - \alpha \frac{\partial}{\partial w_j} J(w) \quad (1.8)$$

gdzie α to parametr szybkości uczenia. Proces dobierania wag można podzielić na 2 etapy. Najpierw przez sieć przepuszczany jest sygnał j przez każdą kolejną warstwę aż do warstwy wyjściowej, a następnie operacja powtarzana jest do tyłu. Podczas tego procesu zwracana jest informacja o błędzie wag - za pomocą funkcji kosztu - a po jej minimalizacji następuje aktualizacja wektora wag.

1.2. Algorytm SVM (ang. *Support Vector Machine*)

Maszyny wektorów nośnych (ang. *Support Vector Machine* - SVM) to jeden z najczęściej stosowanych algorytmów w nadzorowanym uczeniu maszynowym jako klasyfikator. W modelu SVM każdy element badanego zbioru danych traktowany jest jako punkt w n -wymiarowej przestrzeni, gdzie n to liczba parametrów opisująca dany element. Zadaniem klasyfikatora jest znalezienie hiperpłaszczyzny, która 'najlepiej' rozseparuje dane należące do różnych klas. Proces uczenia się algorytmu to proces dobierania wag, w którym maksymalizowany jest margines separacji oddzielający skrajne punkty obu klas (wektory nośne) leżące najbliżej wyznaczonej hiperpłaszczyzny.

Klasyfikator SVM przyjmuje na wejściu parę zmiennych (x, y) , gdzie x to n -wymiarowy wektor cech (parametrów), a y to liczba wskazująca prawdziwą przynależność zmiennej do wybranej klasy (dla wygody późniejszych obliczeń, -1 lub +1). Przy założeniu o liniowej separalności klas, równanie hiperpłaszczyzny możemy zapisać w postaci

$$h(x) = w^T x + b = 0, \quad (1.9)$$

gdzie w - wektor wag (w_1, w_2, \dots, w_n) , x - wektor danych wejściowych, b - polaryzacja. W takim wypadku równania przynależności dla obydwu klas wyglądają następująco:

$$\text{jeśli } w^T x + b \geq 0, \quad \text{to } y_i = +1, \quad (1.10)$$

$$\text{jeśli } w^T x + b \leq 0, \quad \text{to } y_i = -1, \quad (1.11)$$

co można uogólnić do postaci:

$$y_i(w^T x + b) \geq 1. \quad (1.12)$$

Spełnienie równania 1.12 przez pary punktów (x_i, y_i) definiuje wektory nośne decydujące o położeniu hiperpłaszczyzny i wielkości marginesu separacji. Odległość $r(x_{SV})$ wektorów nośnych x_{SV} od hiperpłaszczyzny określona jest wzorem:

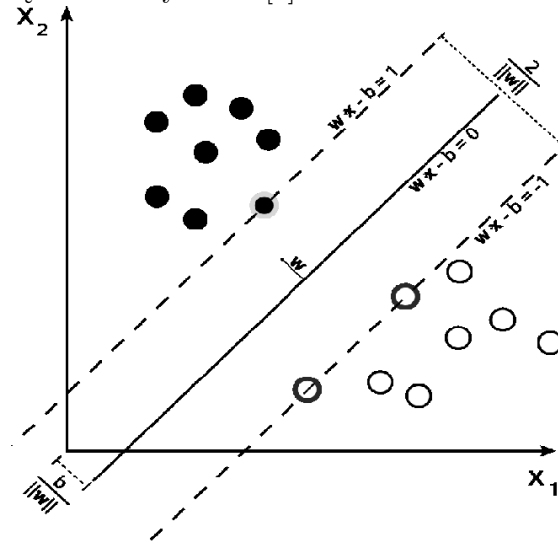
$$r(x_{SV}) = \frac{y_i(x_{SV})}{\|w\|} = \frac{\pm 1}{\|w\|}, \quad (1.13)$$

gdzie $\|w\|$ to normalizacja wektora w . Szerokość d marginesu separacji możemy w takim wypadku określić jako podwojoną bezwzględną odległość wektorów nośnych od hiperpłaszczyzny [5]:

$$d = 2 * \|r(x_{SV})\| = \frac{2}{\|w\|}. \quad (1.14)$$

Ideę działania algorytmu przedstawiono na Rysunku 1.2.

Rysunek 1.2: Idea działania algorytmu SVM - szukanie hiperpłaszczyzny jak najlepiej rozseparowującej dane należące do różnych klas[1]



Pragnąc uzyskać zmaksymalizowany margines separacji d trzeba zminimalizować $\|w\|$, co jest równoważne z minimalizacją wyrażenia $\frac{1}{2}\|w\|^2$. Podana funkcja jest funkcją wypukłą, a więc mającą jedno minimum globalne. Istnieje wiele gotowych bibliotek numerycznych przeprowadzających bardzo szybko wspomnianą minimalizację. Jednak do rozwiązania podanego wyrażenia często stosuje się tzw. mnożniki Lagrange'a, ułatwiające rozwiązanie problemu przy bardzo wielowymiarowych wektorach cech. Więcej o tej metodzie w [Przypiss]. Trzeba jednak pamiętać, że bardzo często pracuje się na początkowo nieseparowalnym liniowo zbiorze danych. W takim wypadku można zastosować pewne przekształcenia (tzw. mapowanie) pozwalające na przejście do większowymiarowej przestrzeni, w której dane stają się liniowo separowalne. Wprowadza się tzw. funkcję mapującą $\phi(x)$, która przenosi punkty z oryginalnej przestrzeni wejściowej do 'powiększonej' przestrzeni cech: (w algorytmie uczącym zamienia się x na $\phi(x)$).

$$\begin{aligned} \phi : R^n &\rightarrow R^{n'} \\ x &\rightarrow \phi(x) \\ (x_1, x_2, \dots, x_n) &\rightarrow (z_1, z_2, \dots, z_{n'}) \end{aligned}$$

Dla każdego mapowania możemy zdefiniować tzw. jądro (ang. *kernel*) K :

$$K(x, z) = \phi(x)^T \phi(z) \quad (1.15)$$

Wtedy za odwzorowanie danych w nowej przestrzeni odpowiada funkcja kernelowa, mogąca przyjmować różne postaci. Do najważniejszych funkcji kernelowych należą:

- Kernel RBF (ang. *Radial Basis Function*):

$$K(x, z) = e^{-\frac{\|x-z\|^2}{2\sigma^2}} \quad (1.16)$$

- Kernel wielomianowy:

$$K(x, z) = (\langle x, z \rangle + 1)^p \quad (1.17)$$

- Kernel sigmoidalny:

$$K(x, z) = \tanh(\langle x, z \rangle + 1) \quad (1.18)$$

gdzie p to parametr jądrowy. W pracy zastosowano kernel RBF, którego używa się najczęściej.

Algorytm SVM domyślnie zwraca wartości binarne. Jednak do niektórych analiz statystycznych, również w niniejszej pracy, potrzebne jest otrzymanie prawdopodobieństw przynależności danego elementu do każdej z klas (zamiast wartości zero-jedynkowych). W tym celu stosuje się pewne algorytmy, w tym tzw. skalowanie Platt'a (ang. *Platt scaling*), zwracające prawdopodobieństwo przynależności zmiennej do klasy 1, określone wzorem:

$$P(y = 1|f(x)) = \frac{1}{1 + \exp(Af(x) + B)}, \quad (1.19)$$

gdzie A i B - parametry wyznaczone na drodze uczenia przy użyciu funkcji maksymalnej wiarygodności na zbiorze uczącym $(f(x_i, y_i))$ [8].

1.3. Klasyfikacja danych i ocena jakości klasyfikatora

Proces klasyfikacji danych podczas uczenia maszynowego składa się zazwyczaj z 3 etapów. Pierwszym jest konstrukcja modelu najlepiej odpowiadającego zbiorowi danych wejściowych (danym treningowym), a następnie zastosowanie uzyskanego dopasowania do klasyfikacji nowych danych (danych testowych). Na koniec dokonuje się weryfikacji "jakości" wybranego modelu, czyli bada się jego parametry takie jak trafność, szybkość czy stopień generalizacji. Aby ocenić poprawność otrzymanego klasyfikatora niezbędna jest znajomość prawdziwej przynależności danych do badanych klas i porównanie jej z przyporządkowaniem do klas zaproponowanych w procesie klasyfikacji. Oczywiście wydaje się być to, że im większą ilość danych dysponujemy oraz gdy zbiory treningowe i testowe pochodzą z różnych "źródeł" (różnych serii badań), tym otrzymamy lepszy klasyfikator. Niestety w praktyce często pracuje się na pojedynczym zbiorze danych, na dodatek o małej liczebności. W takim wypadku stosuje się różne techniki podziału zbioru na elementy treningowe i testowe. Jedną z nich jest zastosowana w poniższej pracy tzw. metoda K -krotnej walidacji (ang. *K-fold cross-validation*). Metoda K -krotnej walidacji polega na podziale dysponowanego zbioru danych na K podzbiorów. Następnie kolejno jeden podzbiór przyjmuje się jako zbiór testowy, a wszystkie pozostałe

jako zbiór uczący, na którym trenuje się klasyfikator. Proces klasyfikacji zachodzi więc niezależnie K razy. Na koniec uśrednia się otrzymane wyniki z K przejść w celu otrzymania końcowego rezultatu. Zaletą powyższej metody jest wykorzystanie w trakcie uczenia całego kompletu informacji. Jednak z drugiej strony użycie tych samych danych na etapie trenowania i testowania może skutkować nieobiektywną oceną skuteczności klasyfikatora.

Przy binarnej ocenie jakości klasyfikacji (podział na dwie klasy) często stosuje się tzw. macierz błędów (ang. *confusion matrix*). Każdemu elementowi zbioru przypisuje się dwie etykiety: jedną zgodną ze stanem faktycznym oraz jedną uzyskaną w procesie klasyfikacji. Wszystkie możliwe kombinacje połączeń obu etykiet przedstawia ww. macierz błędów (Tabela 1.1). Zazwyczaj zamiast etykiet zero-jedynkowych stosuje się podział na klasę pozytywną oraz negatywną.

Tabela 1.1: macierz błędów

	klasa prawdziwa pozytywna	klasa prawdziwa negatywna
klasa predykowana pozytywna	prawdziwie pozytywna TP	fałszywie pozytywna FP
klasa predykowana negatywna	fałszywie negatywna FN	prawdziwie negatywna TN

Gdy przydział do danej klasy w procesie klasyfikacji zgodził się ze stanem faktycznym, przypisujemy dany element albo do zbioru „prawdziwie pozytywnych” rozpoznań (ang. *true positive*, TP) albo do „prawdziwie negatywnych” (ang. *true negative*, TN). W przeciwnym razie, gdy przypisane etykiety nie są zgodne, przypisujemy danym wartość „fałszywie negatywną” (ang. *false negative*, FN) lub „fałszywie pozytywną” (ang. *false positive*, FP). Zliczenia wszystkich przypadków są podstawą do wyznaczenia różnych wartości diagnostycznych testu pozwalających na ocenę klasyfikatora. Podstawowym parametrem wykorzystywanym przede wszystkim w diagnostyce medycznej (ale nie tylko) jest tzw. czułość (ang. *sensitivity*). Można się również spotkać z innymi angielskimi nazwami takimi jak *true positive ratio* (TPR) czy *recall*. Mówiąc o przypadkach medycznych, określa ona jaką część chorych prawidłowo zakwalifikowano do chorych, czyli inaczej, prawdopodobieństwo prawidłowej klasyfikacji osoby chorej. Z tego powodu klasyfikatory mające pomóc rozpoznaniom medycznym powinny dążyć do jak największej wartości tego parametru. Na podstawie macierzy błędów, czułość określona jest wzorem:

$$TPR = \frac{TP}{TP + FN} \quad (1.20)$$

„Dopełnieniem” czułości jest specyficzność (ang. *specificity* (SPC), *true negative ratio*). Określa ona jaką część osób zdrowych zakwalifikowano poprawnie, czyli prawdopodobieństwo, że test pokaże wynik negatywny dla osoby zdrowej. Obliczana jest według wzoru:

$$SPC = \frac{TN}{FP + TN} \quad (1.21)$$

Ważną miarą jest również określenie ilości „fałszywych alarmów” czyli tzw. FPR (ang. *false positive ratio*). Wskazuje ono prawdopodobieństwo zakwalifikowania osoby zdrowej jako chorej. Określana jest wzorem:

$$FPR = \frac{FP}{FP + TN} \quad (1.22)$$

Innymi parametrami badanymi w pracy były również precyzja pozytywna PPV (ang. *positive predictive value*) mówiąca o procencie liczby osób chorych wśród wszystkich kwalifikacji pozytywnych, dokładność ACC (ang. *accuracy*) określająca całkowitą dokładność klasyfikacji oraz parametr F1, czyli średnia harmoniczna z precyzji i czułości. wymienione wielkości obliczane są kolejno według wzorów:

$$PPV = \frac{TP}{TP + FP} \quad (1.23)$$

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \quad (1.24)$$

$$F1 = 2 \frac{PPV * TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN} \quad (1.25)$$

Wyżej wymienione parametry stosuje się w przypadku klasyfikacji binarnych. Większe możliwości mogą dawać klasyfikatory zwracające wartości z ciągłego rozkładu prawdopodobieństwa. Istnieje wtedy możliwość „ręcznego” ustalenia progu z przedziału $[0,1]$, powyżej którego odpowiedź klasyfikatora traktuje się jako pozytywną, a poniżej jako negatywną. W takim wypadku ocenę jakości modelu przeprowadza się za pomocą tzw. krzywej ROC. Kolejne punkty na tej krzywej otrzymywane są dla ustalonej wartości progu. Na osi X znajdują się wartości 1-SPC, czyli parametr false positive ratio FPR, natomiast na osi Y wartości czułości TPR. Idealna krzywa zawierałaby punkt (0,1) oznaczający, że wszystkie dane zakwalifikowano prawidłowo. Parametr AUC (ang. *Area Under Curve*), definiowany jako pole powierzchni pod krzywą, wskazuje moc diagnostyczną testu.

W celu zbadania jakości klasyfikatora można również porównać go z innymi, prostymi klasyfikatorami, np. klasyfikatorem losowym – gdzie ostateczne wyniki są przypisywane losowo na podstawie przyjętego rozkładu prawdopodobieństwa lub klasyfikatorem zerowym/jedynkowym, kiedy predykcja zawsze wynosi odpowiednio 0 lub 1.

1.4. Czerniak - różnicowanie zmian barwnikowych w praktyce klinicznej

Czerniak jest obecnie jednym z najgroźniejszych zmian nowotworowych skóry. Wywodzi się z komórek barwnikowych skóry – melanocytów. Zdecydowana większość przypadków dotyczy tkanki skórnej, jednak w związku z występowaniem komórek barwnikowych w innych tkankach może się on pojawiać np. wewnątrz gałki ocznej, jamie ustnej czy jelitach. Charakteryzuje się wysokim stopniem złośliwości i częstymi przerzutami nawet we wczesnych stadiach choroby. Przyczyny rozwoju wciąż nie są do końca wyjaśnione. Wiadomo, że znaczenie mają zarówno czynniki genetyczne (około 10% zachorowań), jak i środowiskowe. Za najważniejszy czynnik uważa się nadmierną ekspozycję na promieniowanie UV, która ma mutageny wpływ na DNA. Czerniak jest nowotworem wyleczalnym, jednak zasadnicze znaczenie ma jak najwcześniejsza diagnoza, ponieważ dzięki niej możliwe jest wycięcie zmiany przed rozwinieniem i przerzutami do innych organów. Przeżywalność powyżej 5 lat od rozpoznania spada z powyżej 90% na początkowych etapach nowotworzenia do 14% w ostatnich stadiach[4]. Wstępne rozpoznanie i różnicowanie zmian barwnikowych skóry przeprowadza dermatolog. Zazwyczaj używa się tzw. dermatoskopu, który pozwala na obejrzenie zmian z kilkunastokrotnym powiększeniem. Lekarz stawia wstępną diagnozę na podstawie oceny różnych parametrów zmiany, takich jak: asymetria zmiany, nieregularne kropki/globule, duży rozmiar (średnica > 6 mm), niejednolity kolor (obecność głównie czerwonego i niebieskiego), nierówne, postrzępione brzegi czy ewolucja zmiany w krótkim czasie.

Po przeprowadzeniu badania dermatoskopowego i oceny według przyjętej skali (zazwyczaj reguła ABCD lub jedna z innych stosowanych metod[9]), lekarz dokonuje subiektywnej oceny i w przypadku pozytywnej diagnozy zleca wycięcie zmiany. Ostateczne rozpoznanie następuje dopiero po badaniu histopatologicznym wyciętej zmiany.

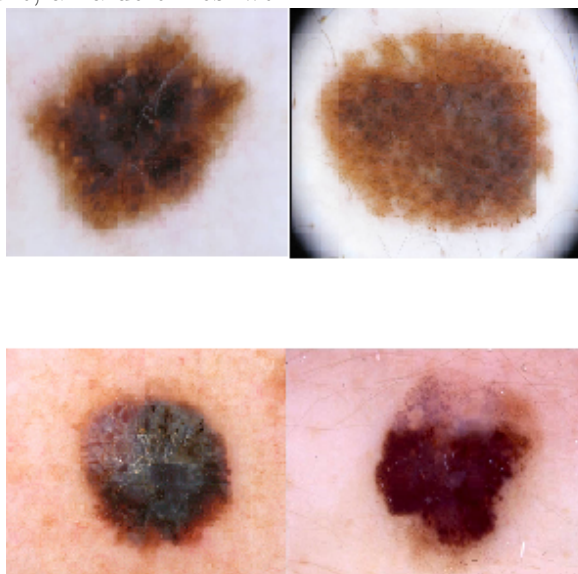
Podawana trafność stawianych przez lekarzy prawidłowych diagnoz różni się w zależności od źródła. Badania przeprowadzone przez *Skin and Cancer Foundation* w Australii osiągnęły trafność na poziomie 65%[6], co pokazuje potrzeby poprawy w zakresie diagnostyki czerniaka oraz szukania różnych algorytmów wspomagających.

Rozdział 2

Dane eksperymentalne

Danymi eksperymentalnymi w niniejszej pracy był zbiór 1279 zdjęć zmian barwnikowych skóry, podzielony na zdjęcia zmian łagodnych (1031 zdjęć) oraz złośliwych (248 zdjęć). Obrazy zostały pobrane ze strony *International Skin Imaging Collaboration*[przypis] - organizacji posiadającej jedną z największych na świecie bazy zdjęć dermatologicznych, które sprawiają duże trudności diagnostyczne. Przykładowe zdjęcia pobrane z bazy pokazano na Rysunku 2.1.

Rysunek 2.1: Przykładowe zdjęcie zmian skórnych pobranych z bazy. Zdjęcia na górze przedstawiają zmiany łagodne, a na dole złośliwe.



Stosunek liczby zdjęć złośliwych i niezłośliwych dobrze odzwierciedla rzeczywistość, gdzie w większości przypadków mamy do czynienia ze zmianami łagodnymi, a tylko nieznaczna część badanych zmian skórnych to pozytywne przypadki choroby. Zgodnie z pracą [3], większość eksperymentów jest przeprowadzana właśnie na nie zrównoważonych danych, które dokładniej odzwierciedlają stan faktyczny, kiedy zazwyczaj więcej badanych osób jest zdrowych niż chorych. Jednakże, w odniesieniu do pracy z konwolucyjnymi sieciami neuronowymi, postanowiono sprawdzić czy zrównanie liczebności zbiorów, zarówno w przypadku danych treningowych i testowych, zwiększy efektywność nauki sieci oraz poprawi wyniki końcowe.

W pracy użyto parametrów uzyskanych w wyniku przepuszczenia obrazów przez wytreno-

waną wcześniej sieć konwolucyjną udostępnioną przez *Berkeley Vision and Learning Center* (BVLC). Model sieci wykorzystany do uzyskania potrzebnych danych nazywa się VGG-CNN-S i został stworzony w ramach *Image Large Scale Visual Recognition Challenge* (ILSVRC) 2012 - konkursu w ramach którego bada się najlepsze algorytmy do klasyfikacji obrazów. Sieć została wcześniej wytrenowana na bazie danych z ImageNet[7]. Pełną budowę sieci można zobaczyć w [przypiss].

Rozdział 3

Metodologia

3.1. Oprogramowanie

W celu analizy danych oraz przedstawienia wyników końcowych napisano skrypty w języku Python 3.4. Skorzystano w wielu wbudowanych bibliotek: NumPy, Matplotlib, Pandas i przede wszystkim Scikit-Learn. Scikit-Learn to biblioteka używana do uczenia maszynowego zawierająca różne algorytmy, m.in. do klasyfikacji, regresji czy analizy skupień.

3.2. Metodyka pracy

Do analizy w ramach niniejszej pracy licencjackiej posłużono się wektorem 4096. parametrów uzyskanych w procesie parametryzacji obrazów. Przycięte do kwadratu zdjęcia przepuszczono przez wyżej opisaną sieć VGG-CNN-S. Z sieci wyodrębniono warstwę szóstą w pełni połączoną (FC6) o długości 4096. W ten sposób uzyskano wektor 4096 parametrów opisujących dany obraz. Powyższa operacja została przeprowadzona w ramach pracy licencjackiej Urszuli Romaniuk[prypis].

Danymi wejściowymi przeprowadzonej analizy była macierz X o wymiarach $m \times n$, gdzie m to liczba wierszy odpowiadająca ilości zdjęć użytych w pracy (1279), a n to liczba kolumn równa długości wektora parametrów opisującego pojedynczy obraz (4096). Przed procesem uczenia przeskalowano wektor cech za pomocą funkcji sigmoidalnej

$$x'_j = \frac{1}{1 + \exp^{-x_j}} \quad (\text{dla } j = 1, \dots, n), \quad (3.1)$$

gdzie x'_j - przeskalowana wartość j -tej cechy, x_j - oryginalna wartość j -tej cechy, a n to długość wektora cech (4096). Dodatkowo jednowymiarowa macierz Y o długości równej liczbie zdjęć (1279) zawierała kolejno zera i jedynki, wskazując na przynależność każdego zdjęcia do jednej z dwóch klas ("0" - klasa negatywna, zmiany niezłośliwe; "1" - klasa pozytywna, zmiany złośliwe). Zbiór danych podzielono na obrazy testowe i treningowe kolejno na cztery sposoby: najpierw trenowano i testowano klasyfikatory na zbiorach zrównoważonych, następnie na zbiorach niezrównoważonych (w stosunku 4:1 zmian łagodnych do złośliwych), a na koniec trenowano na zbiorze zrównoważonym a testowano na niezrównoważonym i odwrotnie. Tworzono instancję danego klasyfikatora, dopasowywano model do danych treningowych, a następnie sprawdzano go na danych testowych. Za pomocą funkcji *predict_proba* otrzymywano prawdopodobieństwo przynależności zmiennej do danej klasy. Zmieniało próg prawdopodobieństwa z zakresu (0,1) poniżej którego obraz klasyfikowano jako "0", a powyżej jako "1".

Za każdym razem obliczano parametry wskazujące jakości danego klasyfikatora (rozdział-przypis). Następnie wybierano jeden z parametrów, sprawdzano dla jakiego progu dał on najlepsze wyniki i jakie wartości wskazują w tej sytuacji pozostałe parametry. Za każdym razem wyniki porównywano z wynikami wyliczonymi dla klasyfikatora losowego pokazanymi w Rozdziale 4.

Rozdział 4

Wyniki

Zbiór danych zawierał 1279 zdjęć zmian barwnikowych skóry, w tym 1031 zdjęć zmian łagodnych o przypisanej wartości binarnej 0 i 248 zdjęć zmian złośliwych o przypisanej wartości 1. Za pomocą walidacji krzyżowej tworzono dwa podzbiory — treningowy TR i testowy TE . W zależności od analizowanego przypadku zbiory były albo zrównoważone (zawierały taką samą ilość zmian łagodnych co złośliwych) albo niezrównoważone (stosunek ilości zmian złośliwych do niezłośliwych wynosił 1:4). Sprawdzano wyniki dla czterech różnych przypadków:

- gdy obydwa zbiory TR i TE były zrównoważone,
- gdy obydwa zbiory TR i TE były niezrównoważone,
- gdy zbiór TR był zrównoważony, a TE niezrównoważony,
- gdy zbiór TR był niezrównoważony, a TE zrównoważony.

Za każdym razem wyniki były porównywane z klasyfikatorem losowym.

Rozdział został podzielony na trzy podrozdziały. Pierwszy opisuje przyjęty model klasyfikatora losowego. Dwa pozostałe mówią o różnych typach zastosowanych klasyfikatorów. Dla pierwszego klasyfikatora (SVM) analizowano wyniki dla czterech wymienionych wyżej przypadków, a dla perceptronu zanalizowano przypadek ostatni (zbiór TR niezrównoważony, a TE zrównoważony), który w wypadku SVM dał najlepsze wyniki.

4.1. Klasyfikator losowy

Na początku postanowiono wyznaczyć oczekiwane wartości dla klasyfikatora losowego. W tym celu przeprowadzono analizę pokazaną w poniższym podrozdziale. Przyjmując, że w zbiorze testowym mamy a jedynek i $(1 - a)$ zer, gdzie a to liczba z przedziału $[0,1]$ określająca część danych należących do podanego zbioru (tu: jedynek), natomiast prawdopodobieństwo klasyfikacji danej jako jedynka (klasa pozytywna) wynosi b , a jako zero $(1 - b)$ (klasa negatywna), możemy stworzyć "teoretyczną" macierz błędów, pokazaną w Tabeli 4.1:

Tabela 4.1: Teoretyczna macierz błędów dla klasyfikatora losowego

	klasa prawdziwa pozytywna	klasa prawdziwa negatywna
klasa predykowana pozytywna	$a*b$	$(1-a)*b$
klasa predykowana negatywna	$a*(1-b)$	$(1-a)*(1-b)$

Na podstawie powyższej Tabeli oraz definicji badanych parametrów (Rozdział 1.3) możemy określić parametry jakości klasyfikatora jako:

$$TPR = \frac{a * b}{a} = b, \quad (4.1)$$

$$F1 = 2 \frac{a * b}{a + b}, \quad (4.2)$$

$$ACC = a * b + (1 - a)(1 - b), \quad (4.3)$$

$$PPV = \frac{a * b}{b} = a. \quad (4.4)$$

Analizując powyższe równania można zauważyć, że dla klasyfikatora losowego wartości TPR nie da się 'zmaksymalizować'. Możemy zbadać pozostałe parametry przy zadaniu konkretnego prawdopodobieństwa (np. $b = 0,95$) i dla niego wyznaczyć pozostałe parametry. Analogicznie jest w przypadku PPV - jej losowa wartość zależy jedynie od początkowej liczebności zbiorów. Inaczej jest w przypadku parametrów $F1$ i ACC . Chcąc zmaksymalizować $F1$ możemy obliczyć pochodną:

$$F1'(b) = 2 \frac{a^2}{(a + b)^2}. \quad (4.5)$$

Pochodna ta jest zawsze większa od 0, z tego wynika że funkcja podana równaniem 4.2 jest rosnąca, a więc największą wartość przyjmuje dla maksymalnego b , czyli $b = 1$ oraz danego a .

Chcąc zmaksymalizować ACC obliczamy jej pochodną:

$$ACC'(b) = 2a - 1, \quad (4.6)$$

Widać, że maksymalna wartość będzie zależna od zadanej wartości a .

Dalszą analizę podzielono na dwie części: badanie zbioru testowego zrównoważonego oraz niezrównoważonego.

4.1.1. Klasyfikator losowy dla zbioru zrównoważonego

W przypadku zbioru zrównoważonego wartość a wynosi 0,5. Chcąc sprawdzić pozostałe wartości w przypadku $TPR = 0,95$ otrzymujemy:

$$b = 0,95 \text{ (bo } TPR = b),$$

$$F1 = 2 \frac{0,5 * 0,95}{0,5 + 0,95} = 0,66, \quad (4.7)$$

$$ACC = 2 * 0,5 * 0,95 + 1 - 0,95 - 0,5 = 0,5$$

$$PPV = a = 0,5$$

Przy maksymalizacji parametru $F1$ otrzymujemy:

$$b = 1 \text{ (co wyprowadzono powyżej)}$$

$$TPR = b = 1,$$

$$F1 = 2 \frac{0,5 * 1}{(1 + 0,5)} = \frac{2}{3} = 0,66, \quad (4.8)$$

$$ACC = 2 * 0,5 * 1 + 1 - 1 - 0,5 = 0,5,$$

$$PPV = 0,5.$$

Natomiast dla maksymalizacji ACC dla $a = 0,5$ wartość pochodnej ACC równa się 0 niezależnie od wartości b , dlatego w tym wypadku nie da się obliczyć oczekiwanej wartości b dla klasyfikatora losowego.

4.1.2. Klasyfikator losowy dla zbioru nie zrównoważonego

W przypadku zbioru nie zrównoważonego o stosunku liczebności 4:1 zer do jedynek wartość a wynosi 0,2. Chcąc sprawdzić pozostałe wartości parametrów w przypadku $TPR = 0,95$ otrzymujemy:

$$\begin{aligned} b &= 0,95 \text{ (bo } TPR = b), \\ F1 &= 2 \frac{0,2 * 0,95}{0,2 + 0,95} = 0,33, \\ ACC &= 2 * 0,2 * 0,95 + 1 - 0,95 - 0,2 = 0,23 \\ PPV &= a = 0,2 \end{aligned} \tag{4.9}$$

Przy maksymalizacji parametru $F1$ otrzymujemy:

$$\begin{aligned} b &= 1 \text{ (co wyprowadzono powyżej)} \\ TPR &= b = 1, \\ F1 &= 2 \frac{0,2 * 1}{(1 + 0,2)} = \frac{1}{3} = 0,33, \\ ACC &= 2 * 0,2 * 1 + 1 - 1 - 0,2 = 0,2, \\ PPV &= 0,2. \end{aligned} \tag{4.10}$$

Dla maksymalizacji ACC pochodna $ACC'(b) = 2a - 1 < 0$ - równanie określające trafność jest funkcją malejącą, a więc największą wartość parametr przyjmie dla $b = 0$. W takim wypadku:

$$\begin{aligned} TPR &= 0, \\ F1 &= 0, \\ ACC &= 2 * 0,2 * 0 + 1 - b - 0,2 = 0,8, \\ PPV &= a = 0,2 \end{aligned}$$

4.2. Klasyfikator SVM

4.2.1. Zbiory TR i TE zrównoważone

Ze względu na dużą dysproporcję liczby zdjęć zmian łagodnych i złośliwych, w celu uzyskania zbiorów zrównoważonych potrzebne było obcięcie ilości zdjęć o wartości binarnej 0. Stosując walidację krzyżową liczbę jedynek (248) podzielono na 5 jak najbardziej równolicznych podzbiorów i do każdego dodano dokładnie taką samą liczbę zer. Za każdym razem jeden z tak stworzonych podzbiorów pełnił funkcję zbioru testowego TE , a pozostałe cztery rolę zbioru TR . Na początek zmieniając wysokość j progu w zakresie $[0,1]$ wybierano taką wartość j_{best} dla którego parametr TPR był jak najbliższy wartości 0,95. Następnie obliczano pozostałe parametry dla opisanego przypadku. Sprawdzano również wszystkie parametry ze względu na wartość C (odpowiadającej generalizacji modelu) w zakresie od 1 do 464 (wartość zwiększano logarytmicznie) i wybierano wartość C , dla której model dawał najlepsze wyniki. Uzyskany model porównywano z klasyfikatorem losowym. Wyniki przedstawiono w Tabeli 4.2.

Tabela 4.2: Wyniki dla zbiorów TR i TE zrównoważonych dla parametru $C = 46,4$ i ustalonym $TPR = 0,95$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_{best}	0,21	-
TPR	$0,952 \pm 0,020$	0,95
F1	$0,704 \pm 0,009$	0,66
ACC	$0,601 \pm 0,013$	0,50
PPV	$0,559 \pm 0,008$	0,50

Następnie sprawdzono dla którego progu i dla której wartości C wartość parametru $F1$ była największa i jakie wartości przyjmowały w takim wypadku pozostałe parametry. Wyniki przedstawiono w Tabeli 4.3.

Tabela 4.3: Wyniki dla zbiorów TR i TE zrównoważonych dla parametru $C = 100$ i maksymalizacji $F1$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_{best}	0,37	-
TPR	$0,855 \pm 0,050$	1,00
F1	$0,738 \pm 0,037$	0,66
ACC	$0,696 \pm 0,045$	0,50
PPV	$0,649 \pm 0,035$	0,50

Analogicznie postąpiono maksymalizując pozostałe dwa parametry - trafność ACC oraz precyzję PPV . Otrzymane wyniki przedstawiono kolejno w Tabelach 4.4 i 4.5.

Tabela 4.4: Wyniki dla zbiorów TR i TE zrównoważonych dla parametru $C = 21,5$ i maksymalizacji trafności ACC

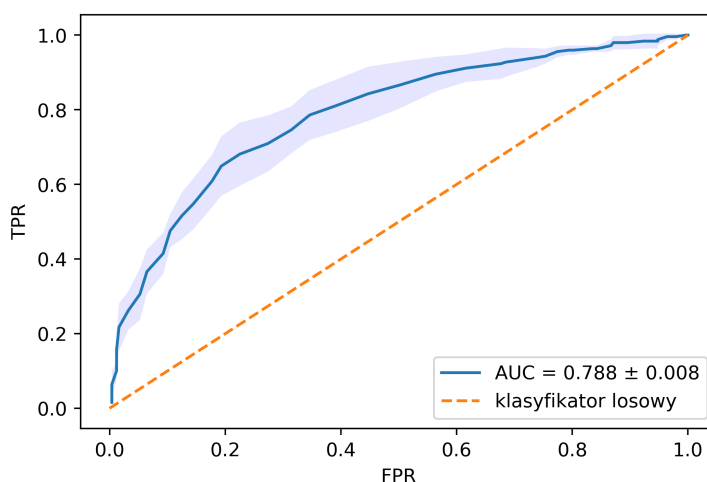
	Klasyfikator SVM	Klasyfikator losowy
Próg j_{best}	0,55	-
TPR	$0,653 \pm 0,078$?
F1	$0,704 \pm 0,080$?
ACC	$0,726 \pm 0,076$	0,50
PPV	$0,767 \pm 0,090$	0,50

Tabela 4.5: Wyniki dla zbiorów TR i TE zrównoważonych dla parametru $C = 46,4$ i maksymalizacji precyzji PPV

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,82	-
TPR	$0,217 \pm 0,075$?
F1	$0,347 \pm 0,106$?
ACC	$0,603 \pm 0,036$	0,50
PPV	$0,957 \pm 0,057$	0,50

Na Rysunku 4.1 przedstawiono krzywe ROC dla zastosowanego klasyfikatora i klasyfikatora losowego dla najlepszej wartości parametru C otrzymanej dla czułości $TPR = 0,95$.

Rysunek 4.1: Krzywe ROC dla zastosowanego klasyfikatora i klasyfikatora losowego dla zbiorów TR i TE zrównoważonych



Pole AUC pod krzywą jest ponad 50% większe od pola AUC_{luck} dla klasyfikatora losowego, co świadczy o dużej mocy diagnostycznej otrzymanego klasyfikatora.

Analizując powyższe Tabele można zauważyć, że maksymalizacja parametru $F1$ daje "najlepsze" wyniki - największe wartości pozostałych parametrów. Zachowana jest duża czułość TPR na poziomie 85%. Na dodatek próg 0,37 zapewnia wysoką trafność wynoszącą ok. 70%. W przypadku ustawienia bardzo wysokiej czułości 95% próg j spada do wartości 0,21 co powoduje obniżenie trafności. Natomiast maksymalizacja trafności nie podnosi znacząco wyników w porównaniu do maksymalizacji $F1$, a jedynie pogarsza czułość o prawie 25%. Zdecydowanie najgorsze wyniki powoduje maksymalizacja precyzji.

4.2.2. Zbiory TR i TE nie zrównoważone

W tej części analizy zarówno zbiór TE jak i TR były zbiorami nie zrównoważonymi o stosunku zer i jedynek 4:1. Wszystkie kroki obliczeń powtórzono tak jak w podrozdziale 4.1.1. Wyniki dla $TPR = 0,95$, maksymalizacji $F1$, ACC i PPV przedstawiono kolejno w Tabelach 4.6, 4.7, 4.8 i 4.9.

Tabela 4.6: Wyniki dla zbiorów TR i TE nie zrównoważonych dla parametru $C = 10$ i ustalonym $TPR = 0,95$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,07	-
TPR	$0,944 \pm 0,030$	0,95
F1	$0,412 \pm 0,012$	0,33
ACC	$0,460 \pm 0,022$	0,23
PPV	$0,263 \pm 0,009$	0,20

Tabela 4.7: Wyniki dla zbiorów TR i TE nie zrównoważonych dla parametru $C = 100$ i maksymalizacji $F1$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,15	-
TPR	$0,814 \pm 0,074$	1
F1	$0,709 \pm 0,042$	0,33
ACC	$0,867 \pm 0,018$	0,20
PPV	$0,630 \pm 0,033$	0,20

Tabela 4.8: Wyniki dla zbiorów TR i TE nie zrównoważonych dla parametru $C = 100$ i maksymalizacji ACC

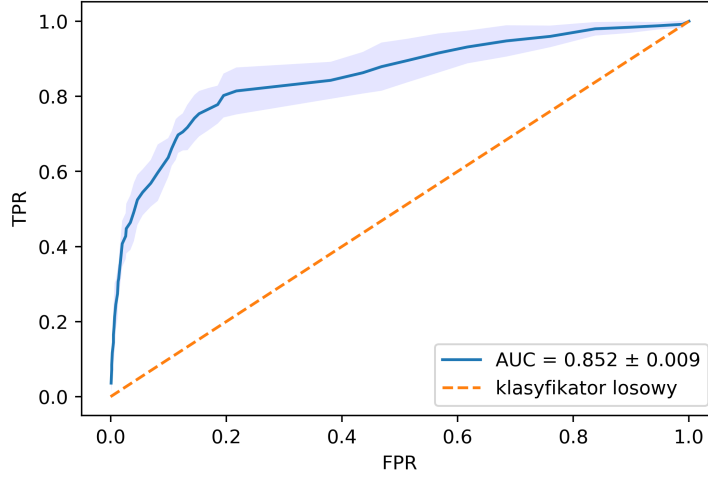
	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,22	-
TPR	$0,701 \pm 0,053$	0
F1	$0,706 \pm 0,054$	0
ACC	$0,883 \pm 0,023$	0,80
PPV	$0,713 \pm 0,063$	0,20

Tabela 4.9: Wyniki dla zbiorów TR i TE nie zrównoważonych dla parametru $C = 4,64$ i maksymalizacji PPV

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,82	-
TPR	$0,177 \pm 0,071$?
F1	$0,292 \pm 0,011$?
ACC	$0,834 \pm 0,013$?
PPV	$0,965 \pm 0,043$	0,20

Na Rysunku 4.2 przedstawiono krzywe ROC dla zastosowanego klasyfikatora i klasyfikatora losowego dla najlepszej wartości parametru C otrzymanej dla czułości $TPR = 0,95$.

Rysunek 4.2: Krzywe ROC dla zastosowanego klasyfikatora i klasyfikatora losowego dla zbiorów TR i TE nie zrównoważonych



AUC otrzymanego klasyfikatora jest ponad 60% większe od AUC_{luck} klasyfikatora losowego, co ponownie świadczy o poprawności i jakości użytego klasyfikatora. Dla zadanej wartości TPR reszta parametrów znacząco spadła w porównaniu do wartości z podrozdziału 4.1.1. Równie kiepskie wyniki otrzymano dla maksymalizacji precyzji. Natomiast większe wartości wszystkich parametrów zauważono dla maksymalizacji $F1$ oraz ACC . Dla trafności uzyskano maksymalną wartość 88% przy wysokich, powyżej 70%, pozostałych parametrach. Przy maksymalizacji $F1$ otrzymano również bardzo wysoką trafność oraz czułość, przy jednoczesnym lekkim spadku precyzji.

4.2.3. Zbiór TR zrównoważony i TE nie zrównoważony

W tej części przetestowano algorytmy gdy klasyfikator uczony jest na zbiorze TR zrównoważonym i testowany na zbiorze TE nie zrównoważonym (stosunek zer i jedynek 4:1). Wyniki przedstawiono w Tabelach 4.10, 4.11, 4.12 i 4.13.

Tabela 4.10: Wyniki dla zbioru TR zrównoważonego i TE nie zrównoważonego dla parametru $C = 464$ i ustalonym $TPR = 0,95$

	Klasyfikator SVM	Klasyfikator losowy
Próg j.best	0,20	-
TPR	0,952 ± 0,035	0,95
F1	0,426 ± 0,012	0,33
ACC	0,486 ± 0,029	0,23
PPV	0,274 ± 0,010	0,20

Tabela 4.11: Wyniki dla zbioru TR zrównoważonego i TE niezrównoważonego dla parametru $C = 100$ i maksymalizacji $F1$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,55	-
TPR	$0,653 \pm 0,091$	1
F1	$0,569 \pm 0,060$	0,33
ACC	$0,802 \pm 0,034$	0,20
PPV	$0,509 \pm 0,058$	0,20

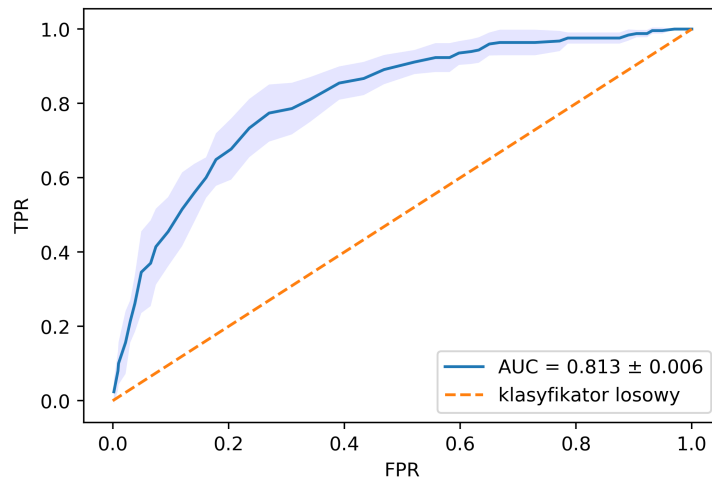
Tabela 4.12: Wyniki dla zbioru TR zrównoważonego i TE niezrównoważonego dla parametru $C = 46,4$ i maksymalizacji ACC

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,67	-
TPR	$0,484 \pm 0,056$	0
F1	$0,539 \pm 0,058$	0,0
ACC	$0,835 \pm 0,023$	0,80
PPV	$0,613 \pm 0,076$	0,20

Tabela 4.13: Wyniki dla zbioru TR zrównoważonego i TE niezrównoważonego dla parametru $C = 46,4$ i maksymalizacji PPV

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,88	-
TPR	$0,089 \pm 0,053$?
F1	$0,158 \pm 0,033$?
ACC	$0,814 \pm 0,004$?
PPV	$0,900 \pm 0,122$	0,20

Rysunek 4.3: Krzywe ROC dla zastosowanego klasyfikatora i klasyfikatora losowego dla zbioru TR zrównoważonego i zbioru TE niezrównoważonego



Podobnie jak w poprzednich podrozdziałach, pole AUC jest znacząco większe od AUC_{luck} .

Wyniki dla ustalonego progu TPR są bardzo zbliżone do tych w rozdziale 4.2.2. Maksymalizacja parametru $F1$ dała dużo gorszy wynik, na poziomie 60%, pogarszając równocześnie pozostałe parametry. Maksymalizacja ACC spowodowała spadek czułości poniżej 50%, obniżając także trafność i precyzję. Wyszukanie największej precyzji spowodowało podwyższenie progu do 0.88, co poskutkowało obniżeniem czułości prawie do zera, a przez co także drastycznym obniżeniem współczynnika $F1$.

4.2.4. Zbiór TR niezrównoważony i TE zrównoważony

W ostatniej części uczono klasyfikator na zbiorze niezrównoważonym a testowano na zrównoważonym. Wyniki w Tabelach 4.14, 4.15, 4.16 4.17, ukazują otrzymane wyniki.

Tabela 4.14: Wyniki dla zbioru TR niezrównoważonego i TE zrównoważonego dla parametru $C = 215,4$ i ustalonym $TPR = 0,95$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,07	-
TPR	$0,948 \pm 0,047$	0,95
F1	$0,715 \pm 0,031$	0,66
ACC	$0,623 \pm 0,039$	0,50
PPV	$0,575 \pm 0,025$	0,50

Tabela 4.15: Wyniki dla zbioru TR niezrównoważonego i TE zrównoważonego dla parametru $C = 464$ i maksymalizacji $F1$

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,11	-
TPR	$0,895 \pm 0,053$	1
F1	$0,841 \pm 0,042$	0,66
ACC	$0,831 \pm 0,044$	0,50
PPV	$0,794 \pm 0,046$	0,50

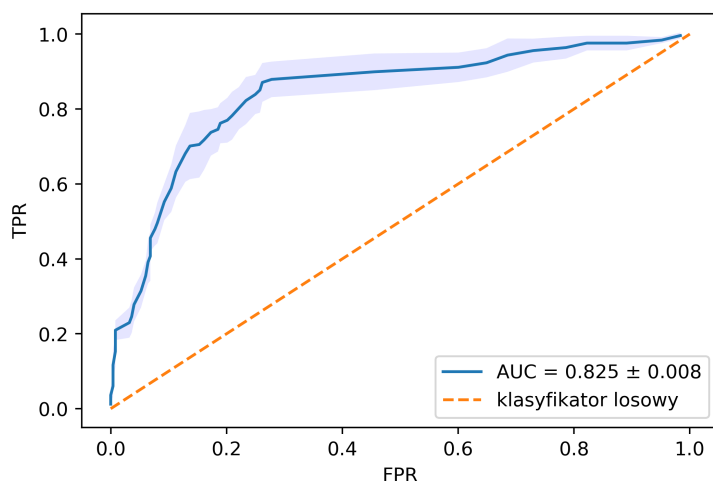
Tabela 4.16: Wyniki dla zbioru TR niezrównoważonego i TE zrównoważonego dla parametru $C = 464$ i maksymalizacji ACC

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,11	-
TPR	$0,895 \pm 0,053$?
F1	$0,841 \pm 0,042$?
ACC	$0,831 \pm 0,044$	0,50
PPV	$0,794 \pm 0,046$	0,50

Tabela 4.17: Wyniki dla zbioru TR niezrównoważonego i TE zrównoważonego dla parametru $C = 464$ i maksymalizacji PPV

	Klasyfikator SVM	Klasyfikator losowy
Próg j_best	0,76	-
TPR	$0,121 \pm 0,052$?
F1	$0,210 \pm 0,082$?
ACC	$0,558 \pm 0,023$	0,50
PPV	$0,982 \pm 0,036$	0,50

Rysunek 4.4: Krzywe ROC dla zastosowanego klasyfikatora i klasyfikatora losowego dla zbioru TR niezrównoważonego i TE zrównoważonego



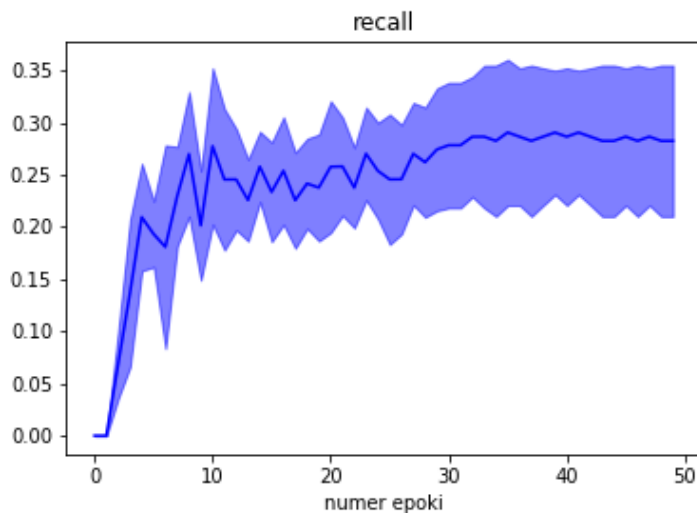
Pole AUC jest większe od AUC_{luck} o prawie 70%, co może wskazywać na wysoką skuteczność użytego klasyfikatora. Analizując otrzymane w tym podrozdziale Tabele, daje on najlepsze wyniki. Dla zadanego TPR wyniki nie są znacząco wyższe niż w odpowiadających im wynikach w pozostałych podrozdziałach, głównie ze względu na bardzo niski próg - 0.07. Jednak maksymalizacja $F1$ oraz ACC daje bardzo dobre wyniki na poziomie 80% dla każdego z parametrów. Po raz kolejny maksymalizacja PPV dała najgorsze wyniki.

4.3. Perceptron

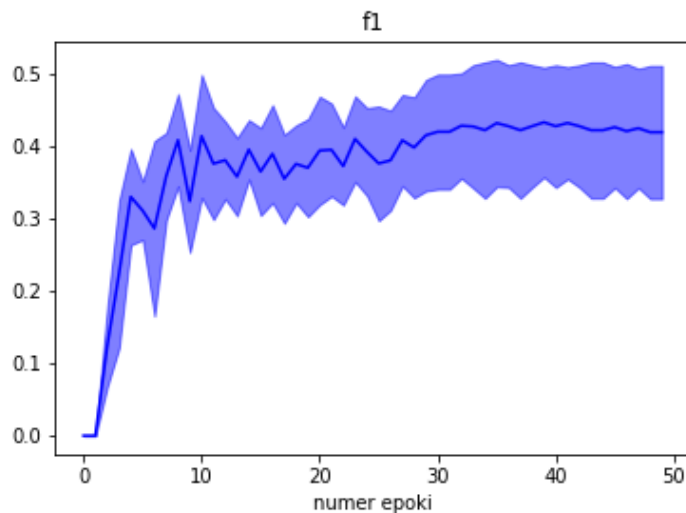
Na koniec postanowiono sprawdzić, czy równie dobre wyniki jak w podrozdziale 4.2.4. można uzyskać stosując inny rodzaj klasyfikatora. Podobną analizę jak dla algorytmu SVM przeprowadzono dla pojedynczego perceptronu, którego użyto jako klasyfikator. Na początku uczono sieć na zadanym zbiorze treningowym TR niezrównoważonym, a następnie testowano na zbiorze zrównoważonym (tak jak w podrozdziale 4.2.4). Na samym początku za pomocą wykresów badano przy jakiej liczbie epok (liczbie cykli uczenia sieci) sieć zostanie nauczona (od kiedy badane parametry nie będą ulegać poprawie podczas zwiększania liczby epok). Następnie dla wybranej epoki postępowano analogicznie jak przy klasyfikacji algorytmem SVM: zmieniano kolejno próg w zakresie $[0,1]$ i badano kiedy nastąpi maksymalizacja wybranego parametru. W celu wybrania odpowiedniego numeru epoki dla którego sieć została nauczona, wartości wybranych parametrów pokazano na wykresach w zależności od numeru epoki.

Poniżej na Rysunkach 4.5, 4.6, 4.7, 4.8 przedstawiono otrzymane zależności.

Rysunek 4.5: Wykres wartości parametru czułości TPR ($recall$) w zależności od numeru epoki dla uczenia sieci na zbiorze nie zrównoważonym

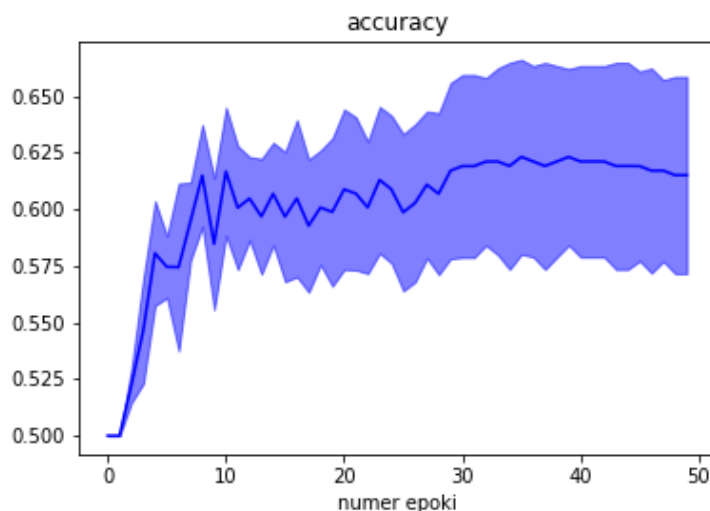


Rysunek 4.6: Wykres wartości parametru $F1$ w zależności od numeru epoki dla uczenia sieci na zbiorze nie zrównoważonym

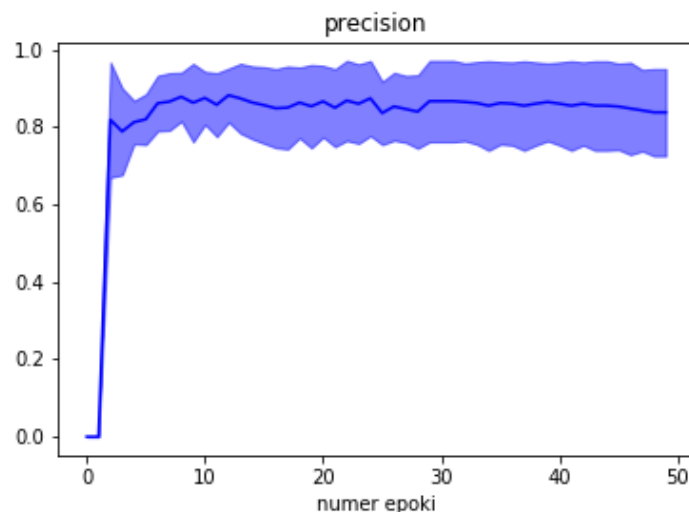


Analizując powyższe wykresy można zauważyć, że od ok. trzydziestu epok wartości parametrów pozostają na stałym poziomie. Z tego powodu wybrano czterdzieści epok jako liczbę cykli uczenia, kiedy sieć została nauczona na odpowiednim zbiorze i dla tej liczby przeprowadzono dalszą analizę.

Rysunek 4.7: Wykres wartości parametru trafności ACC w zależności od numeru epoki dla uczenia sieci na zbiorze nie zrównoważonym



Rysunek 4.8: Wykres wartości parametru $F1$ w zależności od numeru epoki dla uczenia sieci na zbiorze nie zrównoważonym



4.3.1. Zbiór TR nie zrównoważony i TE zrównoważony

Przy uczeniu perceptronu liczba parametrów na wejściu powinna być mniejsza niż liczebność zbioru treningowego. Z tego powodu na samym początku zredukowano liczbę parametrów wejściowych do 256 - za pomocą !!!!! wybrano parametry najlepiej różnicujące dwie klasy. Następnie dla wybranego wyżej numeru epoki przeprowadzono analizę analogiczną jak dla klasyfikatora SVM. Badano wartości parametrów dla zadanego $TPR = 95\%$, a następnie zmaksymalizowano współczynniki $F1$, ACC oraz PPV . Otrzymane wyniki przedstawiono w Tabelach 4.18, 4.19, 4.20, 4.21.

Na koniec, na Rysunku 4.9, pokazano krzywą ROC uzyskaną dla powyższych danych, wraz z obliczonym AUC oraz porównaniem z klasyfikatorem losowym.

Tabela 4.18: Wyniki dla perceptronu (zbiór TR niezrównoważony i TE zrównoważony) dla zadanego $TPR = 0,95$

	Perceptron	Klasyfikator losowy
Próg j_best	0,015	-
TPR	$0,951 \pm 0,055$	0,95
F1	$0,709 \pm 0,015$	0,66
ACC	$0,607 \pm 0,047$	0,50
PPV	$0,568 \pm 0,041$	0,50

Tabela 4.19: Wyniki dla perceptronu (zbiór TR niezrównoważony i TE zrównoważony) dla maksymalizacji $F1$

	Perceptron	Klasyfikator losowy
Próg j_best	0,03	-
TPR	$0,911 \pm 0,067$	1
F1	$0,725 \pm 0,009$	0,66
ACC	$0,655 \pm 0,018$	0,50
PPV	$0,605 \pm 0,025$	0,50

Tabela 4.20: Wyniki dla perceptronu (zbiór TR niezrównoważony i TE zrównoważony) dla maksymalizacji ACC

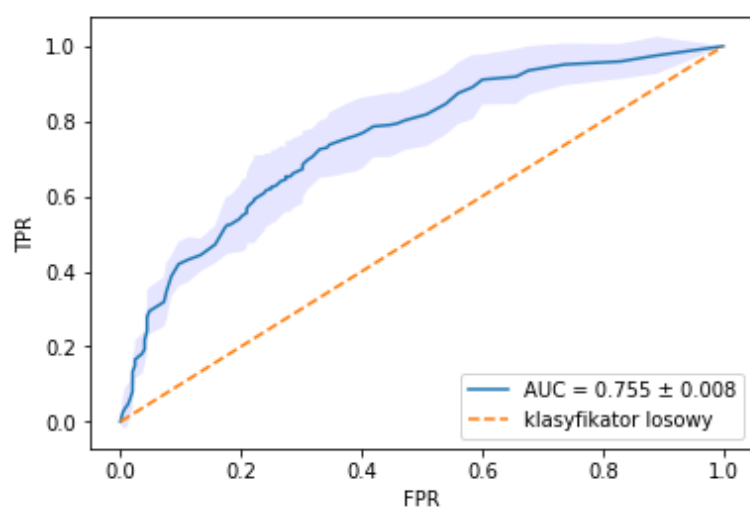
	Perceptron	Klasyfikator losowy
Próg j_best	0,10	-
TPR	$0,726 \pm 0,076$?
F1	$0,704 \pm 0,034$?
ACC	$0,697 \pm 0,022$	0,50
PPV	$0,689 \pm 0,026$	0,50

Tabela 4.21: Wyniki dla perceptronu (zbiór TR niezrównoważony i TE zrównoważony) dla maksymalizacji PPV

	Perceptron	Klasyfikator losowy
Próg j_best	0,90	-
TPR	$0,061 \pm 0,047$?
F1	$0,108 \pm 0,075$?
ACC	$0,522 \pm 0,012$	0,50
PPV	$0,900 \pm 0,126$	0,50

Uzyskane w tym podrozdziale wyniki są porównywalne z osiągniętymi w podrozdziale 4.2.1, jednak są one wyraźnie gorsze od tych uzyskanych w rozdziale 4.2.4 - czyli dla odpowiadających im typów zbiorów testowych i treningowych dla klasyfikatora SVM. Największe wartości wszystkich parametrów daje ponownie maksymalizacja współczynnika $F1$, a najmniejsze współczynnika PPV . Otrzymana krzywa ROC, a dokładniej pole AUC jest mniejsze niż jakikolwiek przypadek w rozdziale 4.2, co może znaczyć o gorszej jakości użytego klasyfikatora.

Rysunek 4.9: Krzywa ROC otrzymana dla perceptronu wraz z wartością AUC oraz klasyfikatorem losowym



Rozdział 5

Dyskusja

Rozdział 6

Podsumowanie

Bibliografia

- [1] Support vector machines. <http://python.astrotech.io/machine-learning/support-vector-machines.html>.
- [2] Krystyna Ambroch. Sztuczne sieci neuronowe. <http://www.smp.uph.edu.pl/msn/32/ambroch.pdf>.
- [3] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.
- [4] Andre Esteva. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542:115–118, 2017.
- [5] Adrian Horzyk. Maszynaaa wektorów nośnych. <http://home.agh.edu.pl/horzyk/lectures/miw/MIW-SVM.pdf>. [dostęp: 2018-06-4 13:42].
- [6] de Launey JW MacKenzie-Wood AR, Milton GW. Melanoma: accuracy of clinical diagnosis. *The Australasian Journal of Dermatology*, 1(39):31–33, 1998.
- [7] Hao Su Jonathan Krause Sanjeev Satheesh Sean Ma Zhiheng Huang Andrej Karpathy Aditya Khosla Michael Bernstein Alexander C. Berg Li Fei-Fei Olga Russakovsky, Jia Deng. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
- [8] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 1999.
- [9] Ahn CSi Rao BK. Dermatoscopy for melanoma and pigmented lesions. *Dermatologic clinics*, 3(30):413–34, 2012.
- [10] Elena Sotiriou Elizabeth Lazaridou Demetrios Ioannides Zoe Apalla, Aimilios Lallas. Epidemiological trends in skin cancer. *Dermatol. Pract. Concept.*, 7(2):1–6, 2017.