**Sztuczna Inteligencja i Inżynieria Wiedzy**

Lab 4

Autor: Jakub Szwedowicz 243416

Prowadzący: dr inż. Piotr Syga

Zajęcia: czw 7:30

Spis treści

[1. eksploracja danych – przedstaw podstawowe dane statystyczne i uwagi dotyczące cech i etykiet zbioru danych. (10 punktów) 3](#_Toc138816873)

[2. Przygotowanie danych - podziel dane na zestaw uczący i walidacyjny (alternatywnie użyj walidacji krzyżowej), zbadaj wpływ różnego typu przetworzenia danych na wyniki klasyfikacji (proponowane: normalizacja, standaryzacja, dyskretyzacja, selekcja cech, PCA) - czyli wykonaj porównanie wyników bez przetworzenia danych z rezultatami po ich przetworzeniu, wykorzystując co najmniej 2 metody różnego typu (osobno). (30 punktów) Bonus – usuń 5% wartości cech i przygotuj dane stosując metody radzenia sobie z brakującymi danymi. (5 punktów 6](#_Toc138816874)

[3. klasyfikacja – przetestuj klasyfikatory i zbadaj wpływ na wyniki: naiwny klasyfikator Bayesa oraz drzewo decyzjne używając przynajmniej 3 różnych zestawów hiperparametrów. (40 punktów) Bonus – Przetestuj (ze zrozumieniem!) bardziej zaawansowane algorytmy, takie jak Las losowy czy Klasyfikator wektorów nośnych (SVM, z ang. Support Vector Machines). (5 punktów) 7](#_Toc138816875)

[4. ocena klasyfikacji – do porównania wyników różnego typu przygotowania danych oraz wykorzystanego klasyfikatora użyj poznanych metryk oceny klasyfikacji i zinterpretuj wyniki. (20 punktów) 10](#_Toc138816876)

# eksploracja danych – przedstaw podstawowe dane statystyczne i uwagi dotyczące cech i etykiet zbioru danych. (10 punktów)

Z pliku glass.names można wyczytać kilka istotnych informacji.

Przede wszystkim poznajemy dane statystyczne, z których wynika, że w przetwarzanych danych nie występują braki.

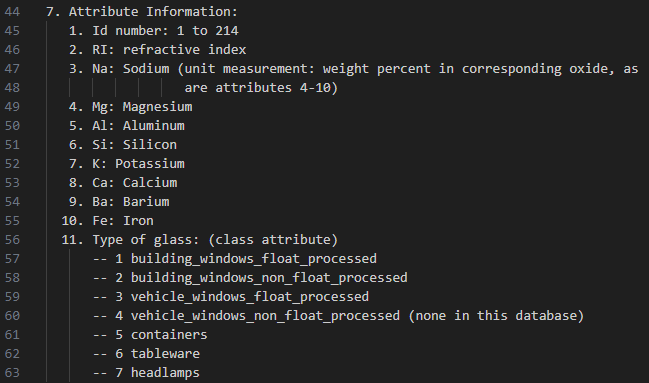
Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Opis wygenerowany automatycznie

Powyższe potwierdza również prosta analiza przy użyciu pandas. Odczytać można, że dla każdej kolumny występuje 214 wartości „non-null”, gdzie wszystkich próbek jest właśnie 214 i dodatkowo zliczając „isna” wychodzi wynik 0  
Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Z pliku dowiadujemy się również jakie pierwiastki występują oraz w jakich typach produktów szklanych występują:



Poznajemy również cząsteczki, które będą tak naprawdę zliczane:  
Obraz zawierający tekst, zrzut ekranu

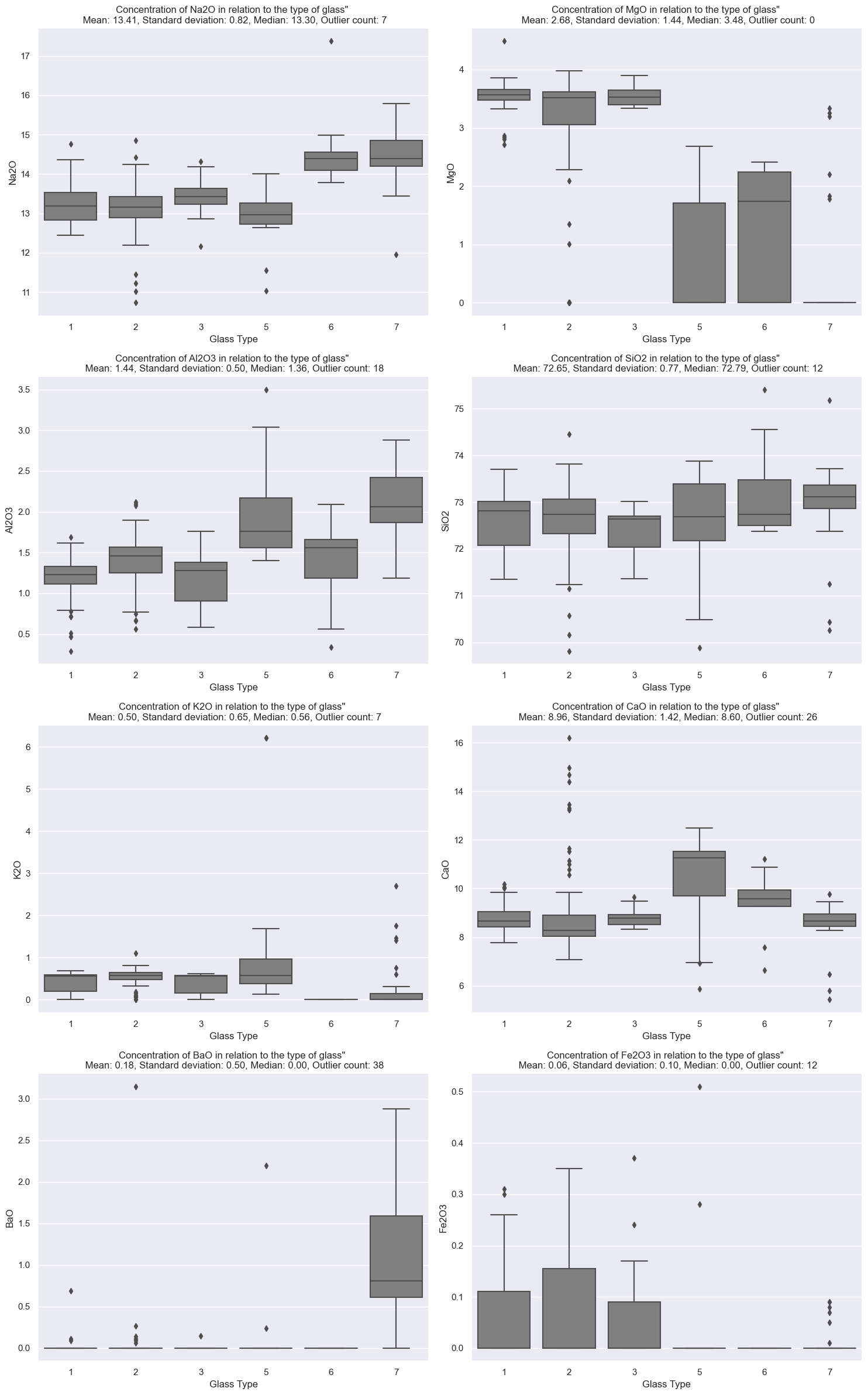
Opis wygenerowany automatycznie

Wczytane z pliku dane można przedstawić za pomocą tabeli:

Obraz zawierający tekst, zrzut ekranu, diagram, Wykres

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka, czarne i białe

Opis wygenerowany automatycznie

Ponadto przygotowane zostały wykresy pudełkowe przedstawiające cechy statystyczne poszczególnych cząsteczek. Można z nich wyczytać średnią, odchylenie standardowe, medianę oraz wartości odstające zdefiniowane jako te leżące poza przedziałem międzykwartalnym (IQR).

# Przygotowanie danych - podziel dane na zestaw uczący i walidacyjny (alternatywnie użyj walidacji krzyżowej), zbadaj wpływ różnego typu przetworzenia danych na wyniki klasyfikacji (proponowane: normalizacja, standaryzacja, dyskretyzacja, selekcja cech, PCA) - czyli wykonaj porównanie wyników bez przetworzenia danych z rezultatami po ich przetworzeniu, wykorzystując co najmniej 2 metody różnego typu (osobno). (30 punktów) Bonus – usuń 5% wartości cech i przygotuj dane stosując metody radzenia sobie z brakującymi danymi. (5 punktów

Usunięto kolumny Id, type oraz Ri a następnie stworzono macierz korelacji przedstawiającą korelację występowania pomiędzy poszczególnymi cząsteczkami. Na podstawie rysunku można stwierdzić, że np. obecnośc BaO koreluje z występowaniem Al2O3.

Obraz zawierający tekst, zrzut ekranu, kwadrat, numer

Opis wygenerowany automatycznie

Do podzielenia danych na zestawy uczące i walidacyjne napisana została klasa ModelsData, która będzie gromadzić:

* Modele (knn, SVC, itd.),
* Dane testowe wspólne dla modeli wartości funkcji: y\_test, y \_train
* Specyficzne dla każdego modelu zbiory argumentów funkcji X\_train\_scaled oraz X\_test\_scaled.
* Zestaw najlepszych scalerów dla każdego modelu, które dały najlepszy rezultat.

Przetestowano kombinacje: braku scalera, StandardScaler, ScandardScaler z PCA(i=1…8), PCA(i=1…8), MinMaxScaler(). Na tej podstawie uzyskano najlepsze kombinacje skalerów dla każdego modelu:  
Obraz zawierający tekst, Czcionka, zrzut ekranu

Opis wygenerowany automatycznie

Gdzie w przypadku nie użycia scalerów, podstawowe modele (bez strojenia hiperapametrów) uzyskują niższe dokładności klasyfikacji danych:

Obraz zawierający tekst, Czcionka, zrzut ekranu

Opis wygenerowany automatycznie

# klasyfikacja – przetestuj klasyfikatory i zbadaj wpływ na wyniki: naiwny klasyfikator Bayesa oraz drzewo decyzjne używając przynajmniej 3 różnych zestawów hiperparametrów. (40 punktów) Bonus – Przetestuj (ze zrozumieniem!) bardziej zaawansowane algorytmy, takie jak Las losowy czy Klasyfikator wektorów nośnych (SVM, z ang. Support Vector Machines). (5 punktów)

Dla GNB użyto zestawu różnych wartości ”var\_smoothing”. Hiperparametr ten sztucznie dodaje zdefiniowaną wartość do wariancji rozkładu. Zmienna ta zwiększa zatem wagę próbek bliżej średniej rozkładu. W zależności od charakteru badanego zjawiska – może to być porząden gdy badane zmienne mają charakter rozkładu normalnego lub nie.

Dla drzewa decyzyjnego (które w sklearn jest binarne, gdyż jest przystosowane do danych numerycznych) modyfikowane są 4 parametry:

* criterion – kryterium którego algorytm używa do określenia w jaki sposób należy podzielić przetwarzane dane w każdym poziomie drzewa aby uzyskać najlepszą dokładność.
* max\_depth – maksymalna wysokość drzewa. Istotnie wpływa na wymagania pamięciowe i obliczeniowe dla działania algorytmu gdyż liczba węzłów drzewa rośnie wykładniczo w stosunku do wysokości drzewa. Istotne również jest, że zbytnia głebokość drzewa może doprowadzić do zbytniego dopasowania modelu do danych (overfitting), co w rezultacie może bardzo pogorszyć klasyfikację danych spoza zestawu trenującego i testowego. Zbyt niska głębokość sprawi, że niewielkie różnice w próbkach będą powodować diametralnie różne wyniki klasyfikacji.
* min\_samples\_split, - zmienna ograniczająca od dołu liczbę wymaganych próbek w węźle aby dokonać podziału i rozrostu drzewa. Zbyt duża wartość może doprowadzić to uniewrażliwienia drzewa na małe zmiany wartości próbek, natomiast zbyt mała wręcz odwrotnie.
* min\_samples\_leaf – podobnie jak powyżej ale dotyczy liści.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, Czcionka, czarne

Opis wygenerowany automatycznie

Wyniki dla różnych modeli (SVC, Regresji logistycznej, Losowego lasu klasyfikatorów, GNB oraz drzewa decyzyjnego) przedstawiają się następująco:

============= non Scaled =============

Finding best models using non scaled data

Training model SVC

Best score: 0.6963025210084033

Best parameters: {'C': 256, 'class\_weight': None, 'gamma': 'auto', 'kernel': 'rbf'}

Training model LogisticRegression

Best score: 0.5789915966386554

Best parameters: {'C': 0.3, 'dual': False, 'penalty': 'l2', 'solver': 'newton-cg', 'tol': 0.5}

Training model GaussianNB

Best score: 0.40369747899159664

Best parameters: {'var\_smoothing': 1e-07}

Training model DecisionTreeClassifier

Best score: 0.6610084033613445

Best parameters: {'criterion': 'gini', 'max\_depth': 5, 'min\_samples\_leaf': 2, 'min\_samples\_split': 2}

Training model RandomForestClassifier

Best score: 0.7245977011494253

Best parameters: {'criterion': 'gini', 'max\_depth': 90, 'max\_features': 'sqrt', 'min\_samples\_leaf': 2, 'min\_samples\_split': 3, 'n\_jobs': -1}

============= Scaled =============

Finding best models using scaled data

Training model SVC

Best score: 0.6611764705882354

Best parameters: {'C': 16, 'class\_weight': None, 'gamma': 'scale', 'kernel': 'rbf'}

Training model LogisticRegression

Best score: 0.5966386554621849

Best parameters: {'C': 0.5, 'dual': False, 'penalty': 'l2', 'solver': 'sag', 'tol': 0.05}

Training model GaussianNB

Best score: 0.40369747899159664

Best parameters: {'var\_smoothing': 5e-07}

Training model DecisionTreeClassifier

Best score: 0.6489075630252101

Best parameters: {'criterion': 'entropy', 'max\_depth': 5, 'min\_samples\_leaf': 2, 'min\_samples\_split': 2}

Training model RandomForestClassifier

Best score: 0.7377011494252873

Best parameters: {'criterion': 'gini', 'max\_depth': 90, 'max\_features': 'sqrt', 'min\_samples\_leaf': 1, 'min\_samples\_split': 4, 'n\_jobs': -1}

Obraz zawierający tekst, Czcionka, numer, linia

Opis wygenerowany automatycznieSame modele zostały wyznaczone przy użyciu GridSearchCV, który przeszukuje N-wymiarową przestrzeń hiperparametrów oraz dokonuje cross-validacji na zbiorze danych treningowych w celu określenia skuteczności modelu. Najpowszechniejszą formą cross-validacji jest K-fold Cross-Validation, która polega na iteracyjnym dzieleniu zbioru treningowego na k partycji, a następnie w każdej iteracji jedna partycja zostaje użyta jako zbiór testowy, a pozostałe k-1 jako zbiór treningowy.

K-Fold Cross Validation (Image by Gufosowa from WikiMedia)

# ocena klasyfikacji – do porównania wyników różnego typu przygotowania danych oraz wykorzystanego klasyfikatora użyj poznanych metryk oceny klasyfikacji i zinterpretuj wyniki. (20 punktów)

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający zrzut ekranu, kwadrat, Wielobarwność, wzór

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka, menu

Opis wygenerowany automatycznieDla danych zeskalowanych:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający zrzut ekranu, kwadrat, Wielobarwność, linia

Opis wygenerowany automatycznie Obraz zawierający zrzut ekranu, kwadrat, Wielobarwność, tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, Czcionka, menu

Opis wygenerowany automatycznieObraz zawierający zrzut ekranu, kwadrat, tekst, Wielobarwność

Opis wygenerowany automatycznieObraz zawierający zrzut ekranu, kwadrat, Wielobarwność, wzór

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

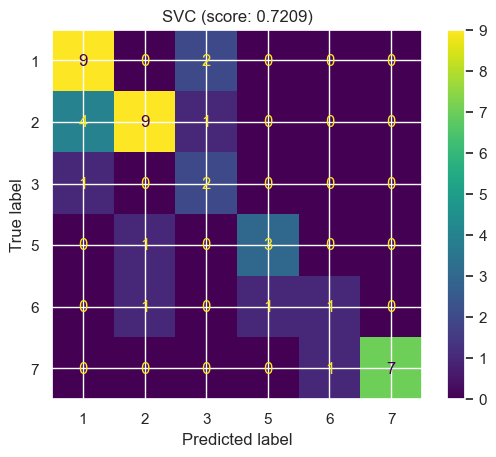
Obraz zawierający zrzut ekranu, kwadrat, Wielobarwność, tekst

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający zrzut ekranu, kwadrat, Wielobarwność, linia

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie Dla danych niezeskalowanych:

Obraz zawierający zrzut ekranu, kwadrat, Wielobarwność, wzór

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

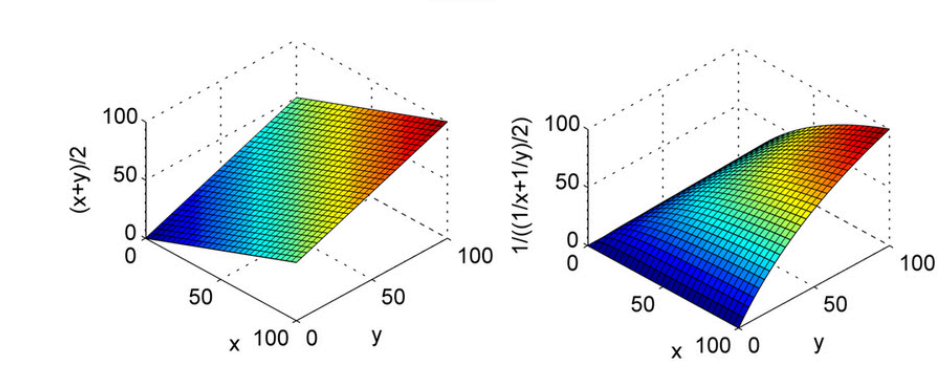
Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający zrzut ekranu, kwadrat, Wielobarwność, wzór

Opis wygenerowany automatycznie

Na podstawie powyższych rysunków można odczytać:

* precision – TP / (TP + FP) – wskazuje ile prawdziwie pozytywnych klasyfikacji dokonał model w stosunku do wszystkich pozytywnie udzielonych odpowiedzi. Metryka przydatna w sytuacji, gdy koszt udzielenia fałszywie pozytywnej klasyfikacji jest bardzo wysoki (np. przy diagnozie raka)
* recall – TP / (TP + FN) – mierzy zdolność modelu do określania pozytywnych przewidywań. Innymi słowy: im więcej razy model wskaże odpowiedź fałszywie negatywną (czyli taką, która powinna być prawdziwie pozytywna) tym mniejszy będzie wynikowy ułamek. Przydatna miara w przypadku gdy koszt pominięcia odpowiedzi pozytywnej jest wysoki i chcemy maksymalizować liczbę prawdziwie poprawnych odpowiedzi (np. wykrywanie pułapek - min)
* f1-score – średnia harmoniczna z precision oraz recall. Pozwala ona łatwiej zauważyć gdy któraś z wartości (precision lub recall) zbliża się do zera (wtedy cała średnia również dąży do 0)



Wykresy 2D dla średniej arytmetycznej (z lewej) i średniej harmonicznej (z prawej).

* Suport: Jest to liczba wystąpień danej klasy w zestawie danych. Ta metryka jest przydatna, wtedy gdy chcemy zrozumieć, jak dystrybuowane są dane i jak model może działać na różnych klasach. Dla przykładu, model może działać bardzo dobrze na klasie, która ma dużo próbek (wysokie wsparcie), ale nie tak dobrze na klasie, która ma tylko kilka próbek (niskie wsparcie).

Reasumując można zauważyć, że GNB gorzej sobie radzi z klasyfikacją szkła od drzewa decyzyjnego. Prawdopodobnie wynika to z faktu, że GNB zakłada wzajemną niezależność zmiennych (cząstek), co w przypadku inżynierii materiałowej jest nieprawdą. Ponadto GNB właśnie z racji na uproszczenia jakie stosuje algorytm w opisie modelu to jest on szybszy co może mieć swoje zastosowanie w systemach czasu rzeczywistego.

Na podstawie różnic w dokładności GNB oraz drzewa decyzyjnego dla danych przetworzonych i nieprzetworzonych przez scalery można zauważyć, że GNB jest bardziej wrażliwy na brak standaryzacji danych.