Projekt 1 - raport

Sebastian Deręgowski
Dawid Janus

16 kwietnia 2021

Spis treści

1 Wstęp, cel, dane

Obiektem naszego zainteresowania były dane ze strony *https://api.apispreadsheets.com/api/dataset/gender-voice/*. Zbiór ten 3168 obserwacji dotyczących parametrów próbki głosu (łącznie 21 zmiennych objaśniających). Naszym zadaniem było stworzenie modelu, który przewidzi jaką płeć miała osoba wydająca dany dźwięk. Na początek dowiedzieliśmy się co oznaczają poszczególne zmienne, aby lepiej zrozumieć zadany problem.

meanfreq - średnia częstotliwość (w kHz)

sd - odchylenie standadrowe częstotliwości

median - mediana częstotliwości (w kHz)

Q25 - pierwszy kwantyl częstotliwości (w kHz)

Q75 - trzeci kwantyl częstotliwości (w kHz)

IQR - rozstęp międzykwartylowy (w kHz)

skew - współczynnik asymetrii (skośność)

kurt - kurtoza

sp.ent - entropia widmowa

sfm - widmowa płaskość

mode - moda

centroid - centroid

meanfun - średnia podstawowa częstotliwość zmierzona w całym sygnale akustycznym

minfun - minimalna podstawowa częstotliwość zmierzona w całym sygnale akustycznym

maxfun - maksymalna podstawowa częstotliwość zmierzona w całym sygnale akustycznym

label - zmienna celu określająca płeć

meandom - średnia dominującej częstotliwości zmierzonej w całym sygnale akustycznym

mindom - minimum dominującej częstotliwości zmierzonej w całym sygnale akustycznym

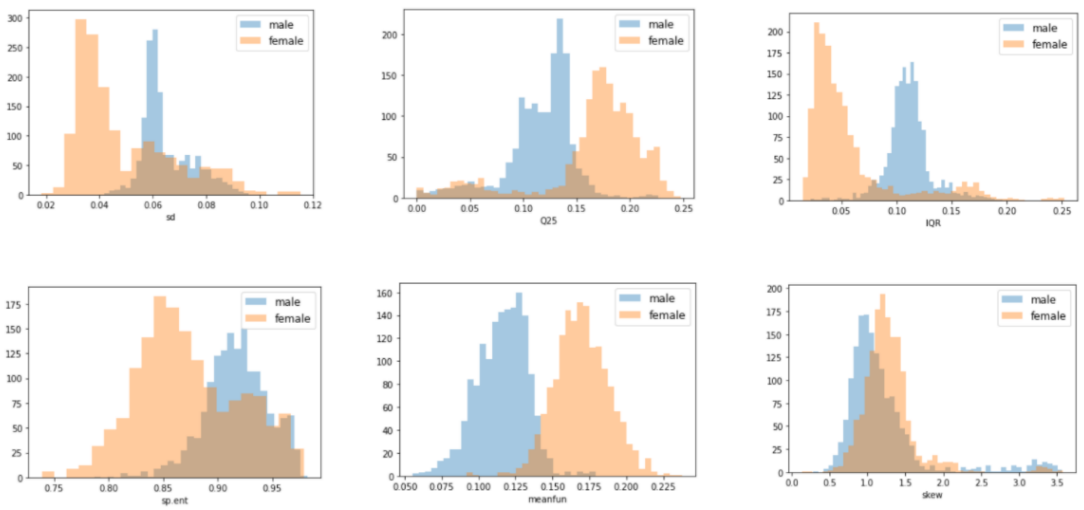
maxdom - maksimum dominującej częstotliwości zmierzonej w całym sygnale akustycznym

dfrange - zasięg dominującej częstotliwości zmierzonej w całym sygnale akustycznym

modindx - indeks modulacji.

2 Ekspolarcyjna analiza danych

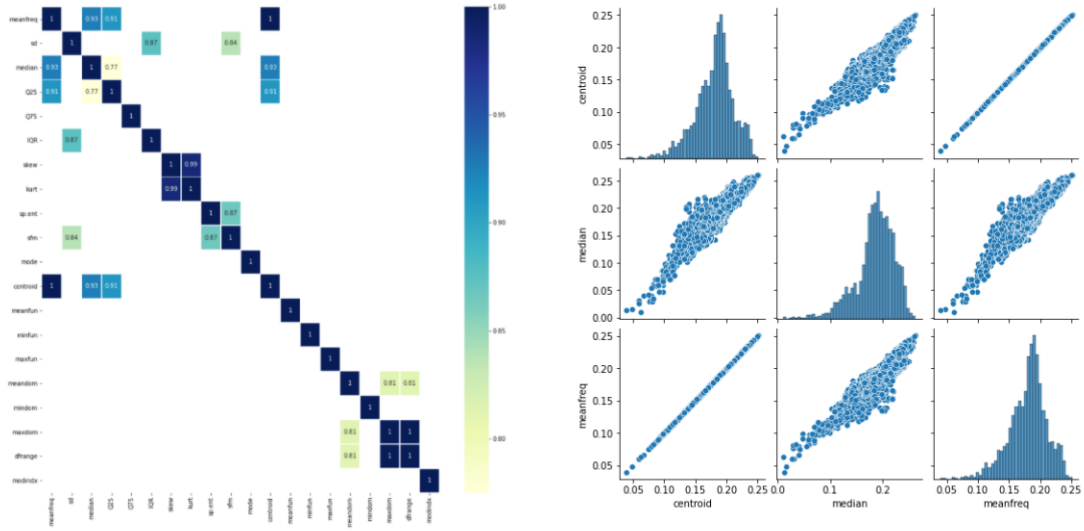
2.1 Rozkłady najważniejszych zmiennych z podziałem na płeć



2.1.1. Rozkłady wybranych zmiennych z uwzględnieniem etykiety płci

Powyżej przedstawiliśmy rozkłady niektórych zmiennych z podziałem na płeć. Na przedstawionych wykresach jest dość widoczna różnica pomiędzy mężczyzną a kobietą, więc przypuszczaliśmy, że powyższe zmienne bedą miały największy wpływ na predykcje modelu.

2.2 Korelacja



2.2.1 Macierz korelacji zmiennych/wykresy zależności pomiędzy zmiennymi *centroid, median, meanfreq*

Następnym wykonanym przez nas krokiem było sprawdzenie korelacji pomiędzy zmiennymi. Na powyższej macierzy wyświetliliśmy tylko te wartości korelacji, których wartość bezwzględna jest większa od 0.75. Najbardziej skorelowanymi zmiennymi są: *centroid* i *meanfreq*, *skew* i *kurt*, *maxdom* i *dfrange*. Obok macierzy korelacji stowrzyliśmy wykres przedstawiający zależności pomiędzy *centroid*, *median* i *meanfreq*.

3 Inżynieria cech

Ze względu na wspomnianią w poprzednim rozdziale korelacje pomiędzy niektórymi zmiennymi postanowiliśmy usunąć z naszej ramki danych zmienne: *kurt*, *centroid* oraz *dfrange*. Zmieniliśmy zmienną label zamieniając *male* na 1 oraz *female* na 0. Natomiast na zmiennej *skew* przeprowadziliśmy transformacje logarytmiczną ze względu na *długi ogon* widoczny na jej rozkładzie.

4 Modele

Pod rozwagę wzięliśmy 3 modele - LogisticRegression(), `KNearestNeighbors()` oraz `XGBClassifier()`. Każdy z trzech modeli poddaliśmy różnym sposobom oceny skuteczności dokonywanych przez nie predykcji. Wyniki poszczególnych weryfikacji są opisane poniżej.

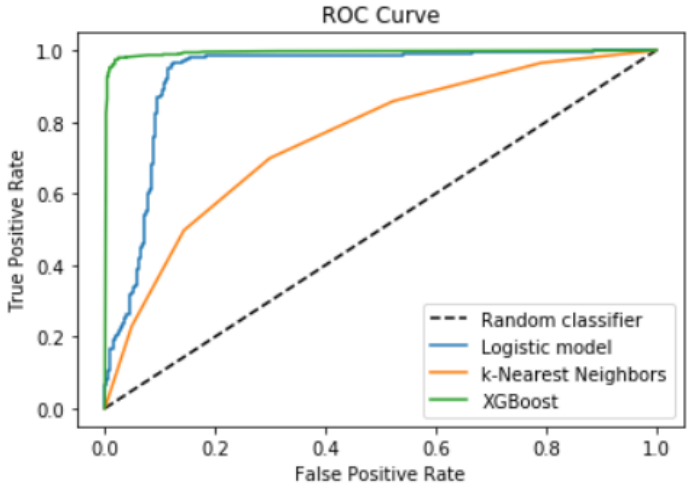
4.1 Macierze pomyłek

LogisticRegression()			KNearestNeighbors()			XGBClassifier()		
	Actual positives	Actual negatives		Actual positives	Actual negatives		Actual positives	Actual negatives
Positive predictions	435	23	Positive predictions	347	166	Positive predictions	486	20
Negative predictions	60	528	Negative predictions	148	385	Negative predictions	9	531

4.1.1. Macierze pomyłek dla modeli

Powyżej znajdują się wyniki predykcji naszych trzech modeli na zbiorze testowym. Składał się on z 495 próbek męskich i 551 próbek żeńskich, a zatem był dość zrównoważony. Jak widzimy, zdecydowanie najlepiej poradził sobie `XGBClassifier()`, który w aż 97% skutecznie sklasyfikował głosy jako męskie lub żeńskie. Nie dużo gorzej poradził sobie model regresji logistycznej, która osiągnęła skuteczność na poziomie 92%. Najsłabiej poradził sobie model `KNearestNeighbors()` - w jego przypadku zaledwie 70%.

4.2 Krzywa ROC i pole AUC



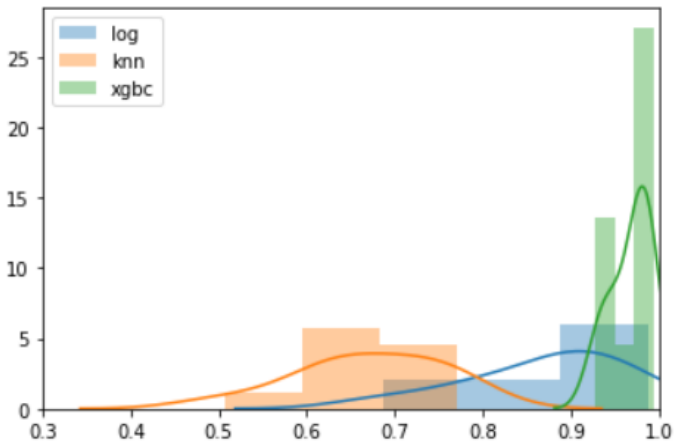
4.2.1. Porównanie krzywych ROC dla modeli

Następnie stworzyliśmy wykresy krzywej ROC dla naszych trzech modelów w celu lepszej wizualizacji różnic pomiędzy modelami. Jak widzimy, w przypadku modelu `XGBoostClassifier()` krzywa jest niemal pionowa dla *false positive rate* i pozioma dla *true positive rate*. Pole AUC dla tego modelu wynosi 0.99.

Model regresji logistycznej ma podobny wygląd krzywej jeśli chodzi o *true positive rate*, ustępuje jednak poprzednikowi w *false positive rate*. W przypadku tego modelu AUC jest równe 0.93. Najsłabszy spośród naszych trzech modeli, `KNearestNeighbors()`, osiągnął AUC równe 0.76.

4.3 Kroswalidacja

Dotychczasowe testowanie modeli odbywało się na zbiorze testowym, który stanowił 1/3 całego zbioru. Mimo że był on dobrze zbalansowany, postanowiliśmy sprawdzić, jakie wyniki osiągną nasze modele dla różnych próbek spośród naszych danych. W tym celu zastosowaliśmy kroswalidację, dzieląc zbiór na 10 podzbiorów. Poniżej znajdują się wizualizacja skuteczności naszych modeli:



4.3.1. Rozkład skuteczności modeli

Bez żadnych zaskoczeń, ponownie najlepiej wypadł model `XGBoostClassifier()`. Miał on nie tylko najlepszą średnią (0.97), ale również najmniejsze odchylenie standardowe (0.02). Model regresji logistycznej osiągnął ponownie drugi najlepszy wynik (średnio 0.87 przy odchyleniu standardowym 0.08). Najsłabszy model uzyskał średnią wyników na poziomie zaledwie 0.67, odchylenie standardowe wyniosło 0.07. Nikogo zatem nie powinno dziwić, że zdecydowaliśmy się na wybór modelu `XGBoostClassifier()`.

5 Strojenie parametrów

Zdajemy sobie sprawę, że strojenie parametrów mogłoby, a nawet powinno odbywać się przed wyborem modelu. Zważywszy jednak na znaczącą przewagę modelu XGBoostClassifier, a także na długi czas strojenia parametrów, zdecydowaliśmy się dostroić parametry jedynie dla tego jednego modelu.

Przy użyciu GridSearchCV() stroiliśmy następujące parametry:

1. learning_rate - rozmiar kroku podczas kolejnej iteracji
2. max_depth - maksymalna głębokość drzewa
3. min_child_weight - minimalna suma instancji wag do utworzenia dziecka, jeśli liść jej nie ma, to nie dokonuje się dalszy podział
4. n_estimators - liczba drzew

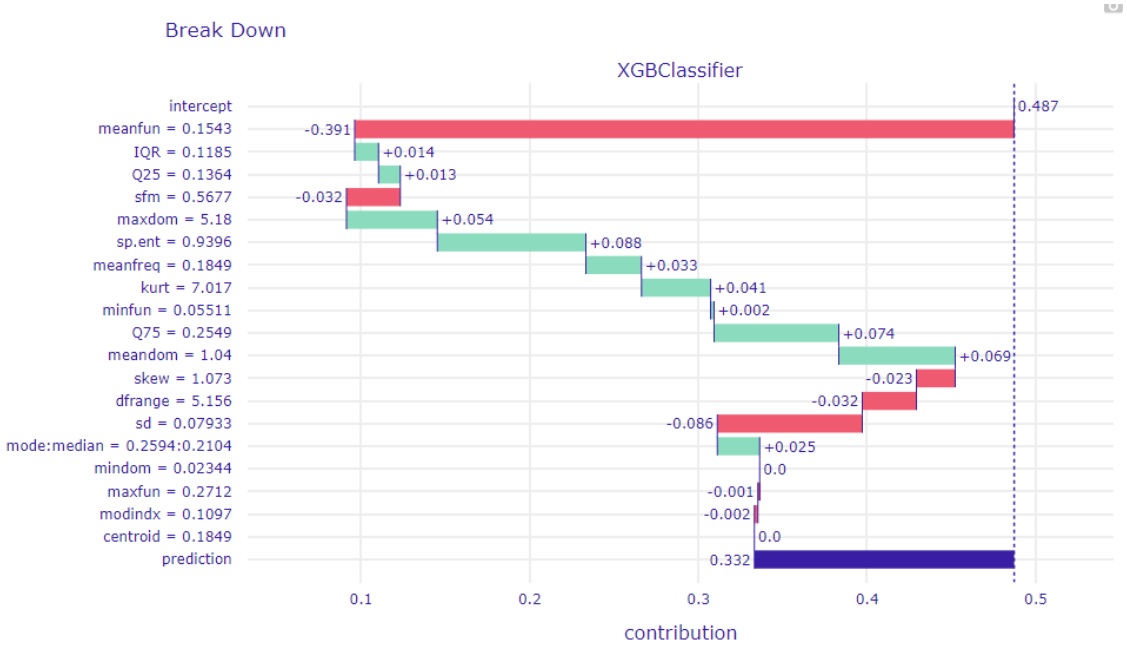
5. gamma - minimalna redukcja straty potrzebna do wykonania podziału drzewa

Strojenie było dość czasochłonne. Za najkorzystniejszą kombinację parametrów uznano:

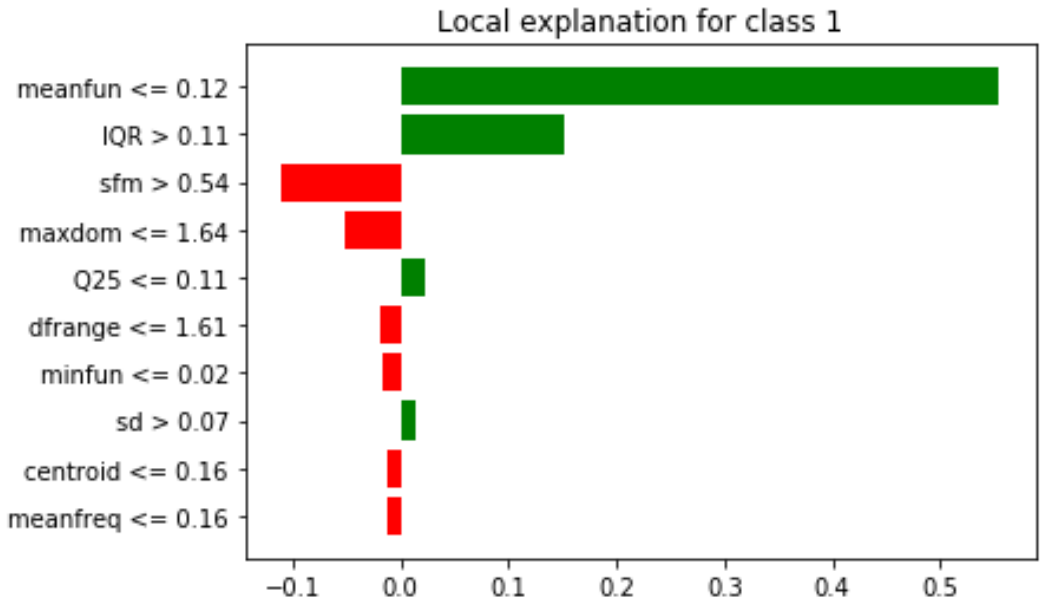
1. learning_rate = 0.01
2. max_depth = 5
3. min_child_weight = 1
4. n_estimators = 500
5. gamma = 0

. Skuteczność predykcji wyniosła aż $0,98$.

6 Dekompozycje wybranych obserwacji



6.1. Dekompozycja BreakDown dla losowej obserwacji ze zbioru



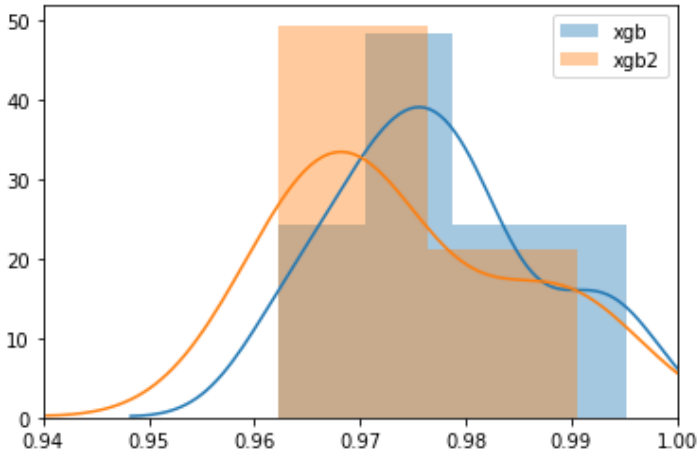
6.2. Dekompozycja Lime dla losowej obserwacji ze zbioru

Z wykresów wynika, że ogromny wpływ na predykcje modelu ma te kilka zmiennych, których rozkłady jednoznacznie dzielą zbiór na głosy żeńskie i męskie, natomiast reszta marginalnie wpływa na przewidywania.

7 Eksperyment

W związku z tym, że większość zmiennych ma marginalny wpływ na predykcję, aż się prosi, aby spróbować znacząco uprościć nasz model. W tym celu usunęliśmy wszystkie zmienne objaśniające poza sześcioma, których rozkłady zamieściliśmy w rozdziale 2.1.

Następnie dokonaliśmy porównania obu modeli przy pomocy kroswalidacji:



7.1. Porównanie modelu pełnego (xgb) i uproszczonego (xgb2)

Jak widzimy, minimalnie lepszy okazał się być model pełen, ale różnice są naprawdę nieznaczne. Model pełen osiągnął skuteczność na poziomie 0.977 przy odchyleniu standardowym 0.009, a uproszczony 0.974 przy takim samym odchyleniu.

Ostatecznie zatem zdecydowaliśmy się na model pełen, jednak w przypadku gdy liczba zmiennych obserwujących znacząco wydłużyłaby czas dokonywania predykcji bądź fitowania, ograniczenie ich do zaledwie tych sześciu kolumn będzie dobrym rozwiązaniem.

Analiza przypadków pomyłek wykazała, że co do większości z nich oba modele były mocno przekonany co do swojej predykcji, tzn. ostateczny wynik nie balansował na granicy głosu męskiego i żeńskiego, tylko był zbliżony w stronę ekstremum. Może to sugerować błędne oznaczenie próbek, a jeżeli nie to pokazuje jak niesamowicie trudne do prawidłowej predykcji były to obserwacje.

8 Podsumowanie

Zbiór danych okazał się być szalenie interesujący, niemalże za każdym razem odkrywał przed nami nowe ukryte właściwości. Bardzo istotna w zrozuieniu danych okazała się ich eksploracyjna analiza, dokonana nie raz, ale kilkukrotnie, za każdym razem z innego punktu widzenia. Ostatecznie udało nam się stworzyć model, który klasyfikuje głosy z dokładnością 0.98, co uważamy za bardzo dobry wynik.