Klasyfikator gatunków muzycznych

Sprawozdanie z projektu indywidualnego

Kacper Aleksander

Spis treści

1. Omówienie projektu 2
2. Zbiór danych 2
3. Ekstraktowanie cech 2
4. Macierz korelacji cech 7
5. Wybór modelu 7
6. Sposób użycia 10
7. Napotkane problemy 10
8. Niedociągnięcia 10
9. Plany dalszego rozwoju 11
10. Źródła 11

1. Omówienie projektu

Założeniem mojego projektu jest stworzenie modelu sztucznej inteligencji pozwalającego skutecznie przewidywać gatunek muzyczny podanego utworu. W tym celu wykorzystałem klasyfikator K-najbliższych sąsiadów. Projekt napisałem w języku Python – wykorzystałem m.in. biblioteki NumPy, Pandas, Matplotlib, Librosa, Scikit-learn. Całość połączyłem z botem w komunikatorze Telegram, by zapewnić odbiorcy wygodny interfejs.

2. Zbiór danych

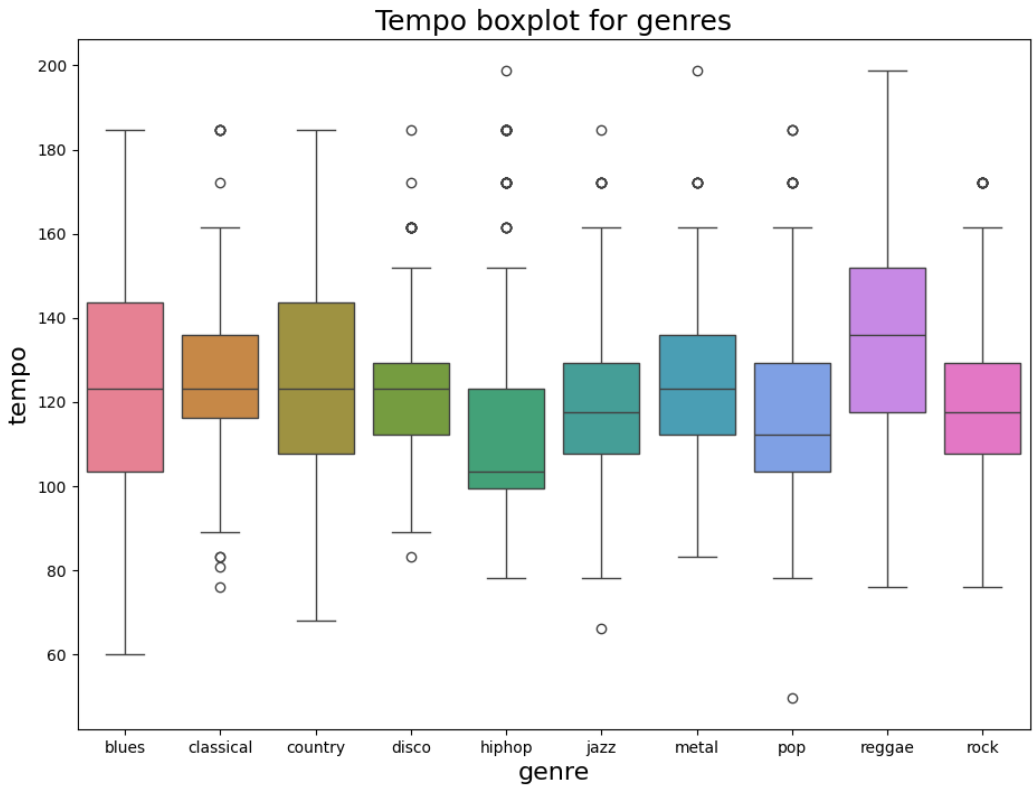
Zbiór danych składa się z 10 gatunków, każdy po 100 piosenek. Każda piosenka trwa 30 sekund. Jedna była wadliwa, więc ją usunąłem, co daje 999 utworów. Program zapewnia edytowalność zbioru danych, to znaczy, że można dodać oraz usunąć gatunki i utwory bez zmian w kodzie.

Link do zbioru umieściłem na końcu sprawozdania.

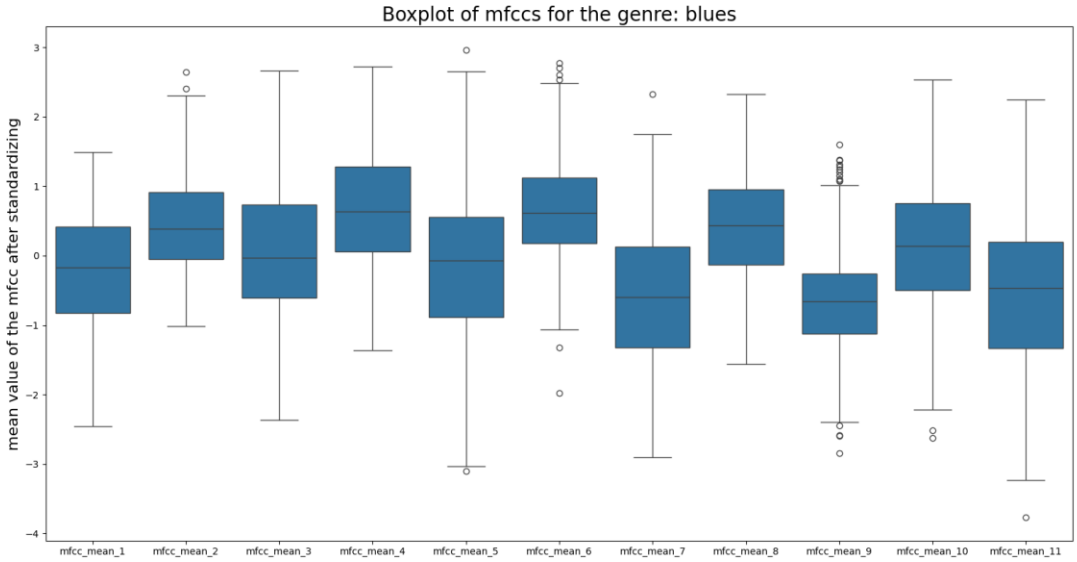
3. Ekstraktowanie cech

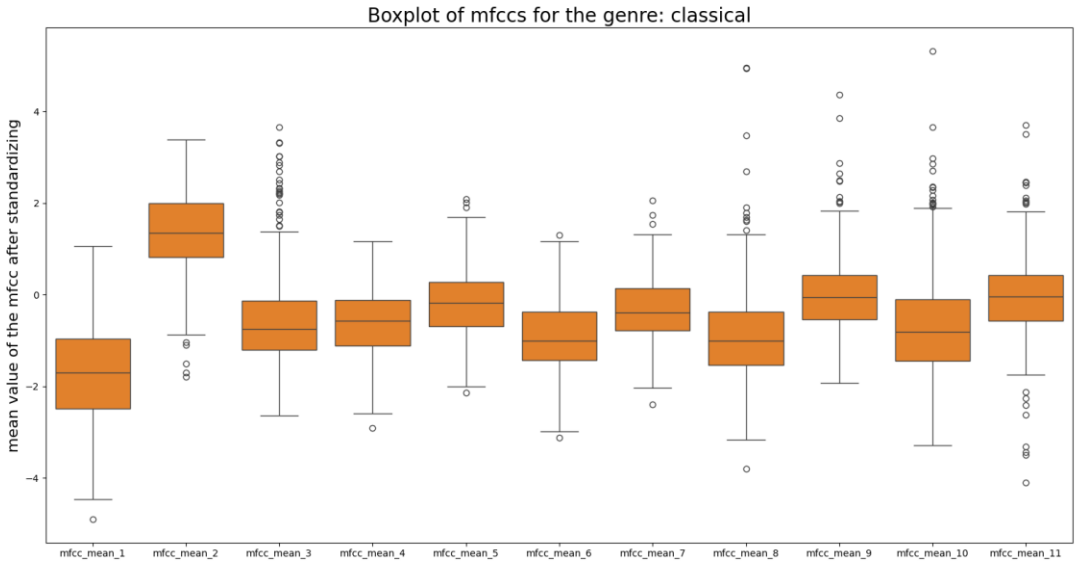
Z każdego utworu wyekstraktowałem i zapisałem średnią na przestrzeni czasu dla każdego z 11 współczynników cepstralnych (zwanych dalej MFCC) oraz tempo – razem 12 wartości. Za pomocą wykresów zweryfikowałem, czy te cechy nadają się do odróżniania gatunków od siebie.

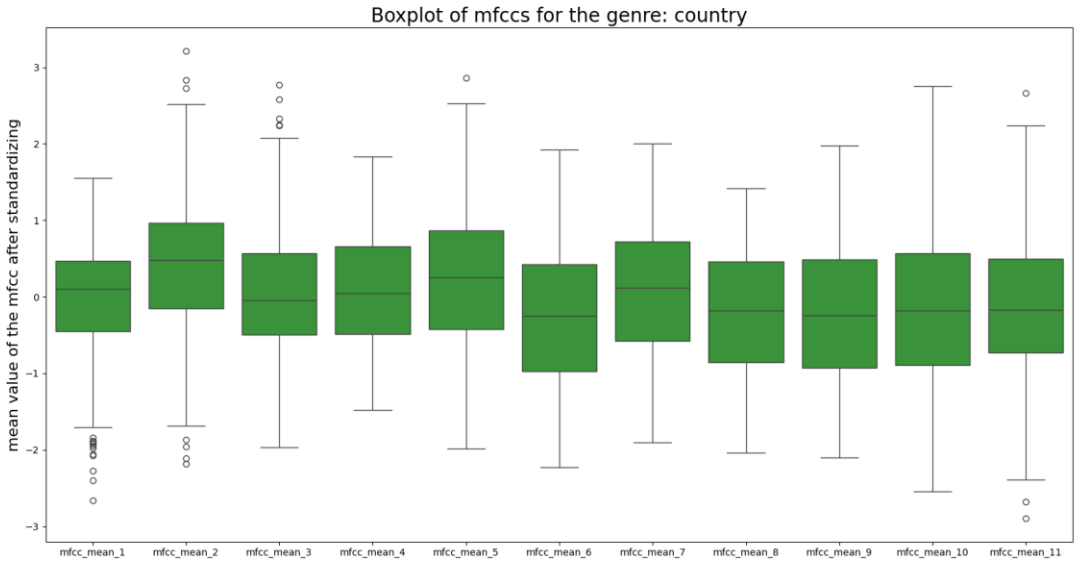
Poniżej przedstawiam wykres pudełkowy tempa dla każdego gatunku:

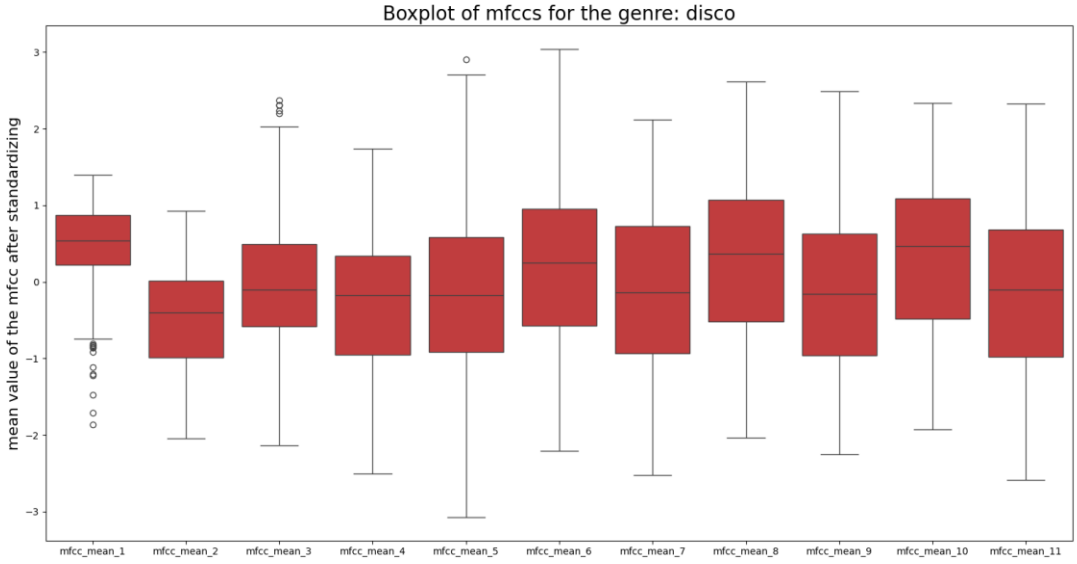


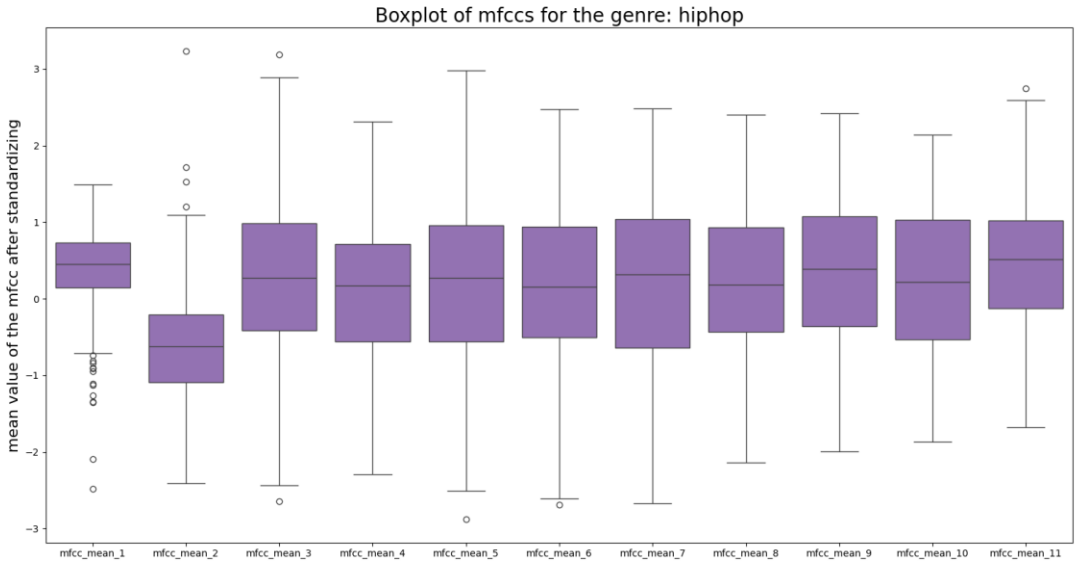
Przedstawiam również wykresy pudełkowe MFCC dla poszczególnych gatunków:

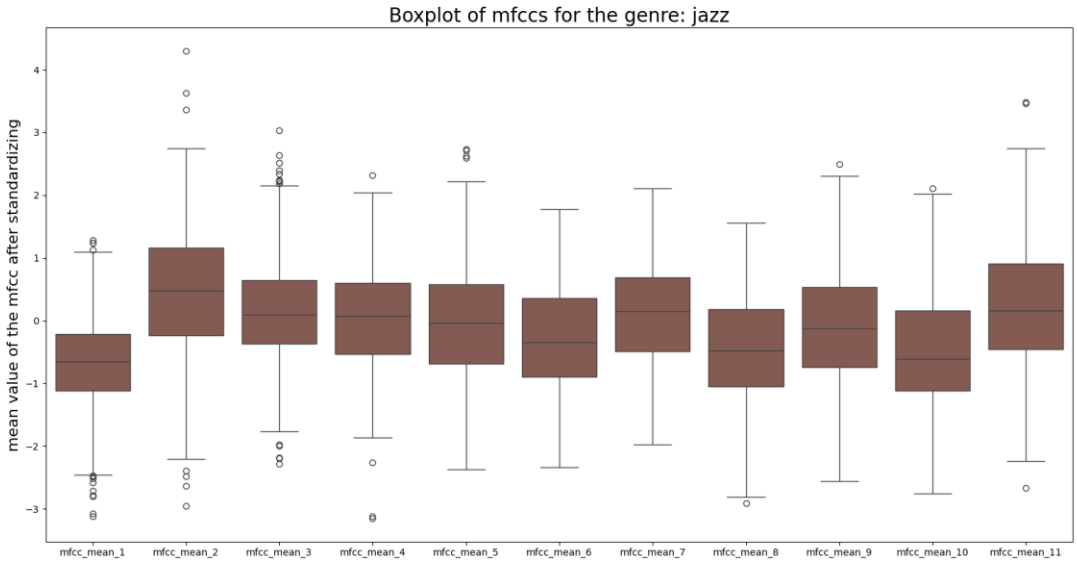


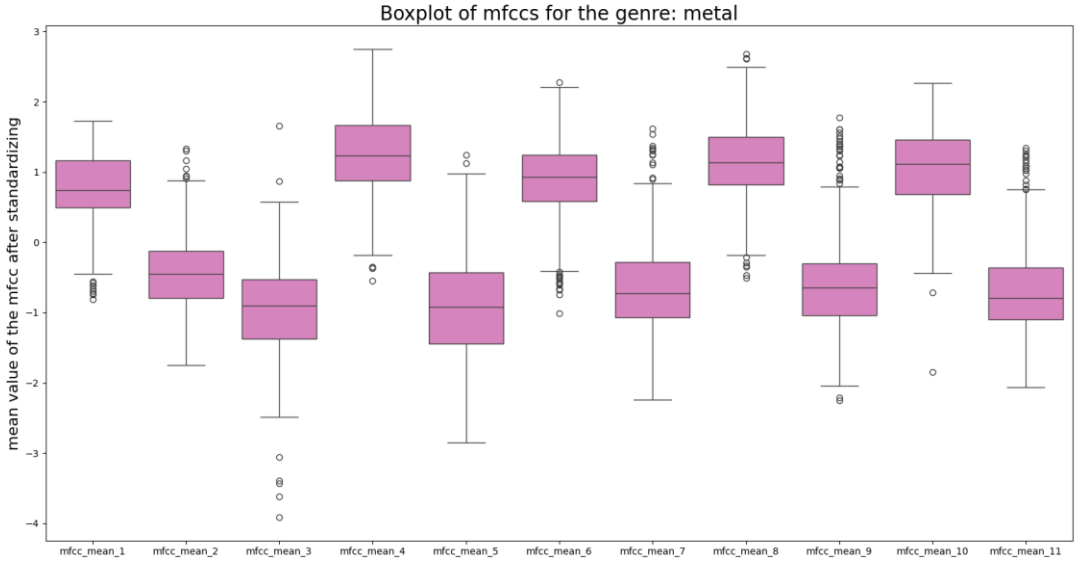


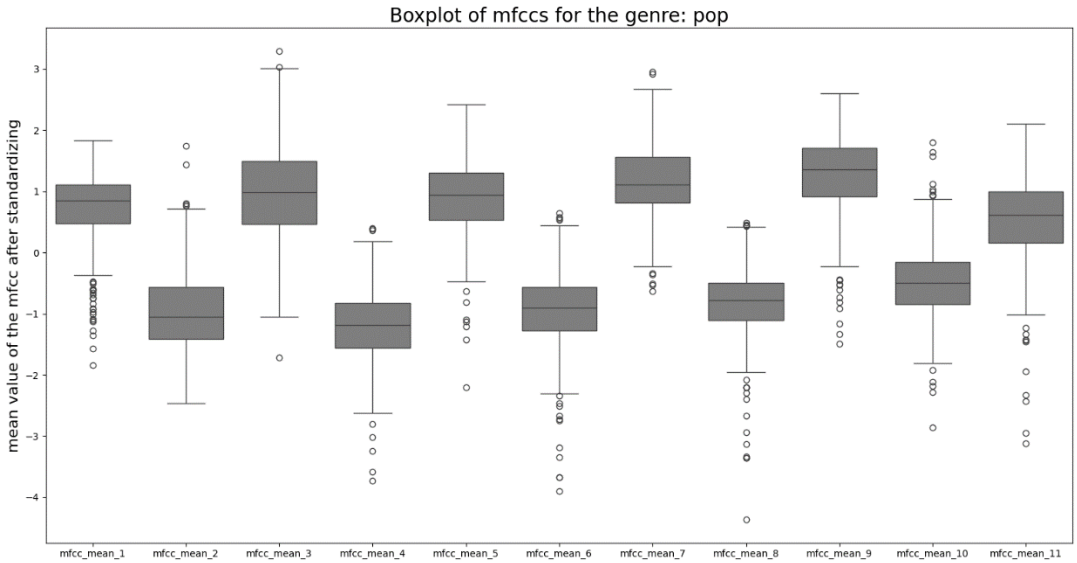

Boxplot of mfccs for the genre: country
mean value of the mfcc after standardizing
mfcc_mean_1
mfcc_mean_2
mfcc_mean_3
mfcc_mean_4
mfcc_mean_5
mfcc_mean_6
mfcc_mean_7
mfcc_mean_8
mfcc_mean_9
mfcc_mean_10
mfcc_mean_11

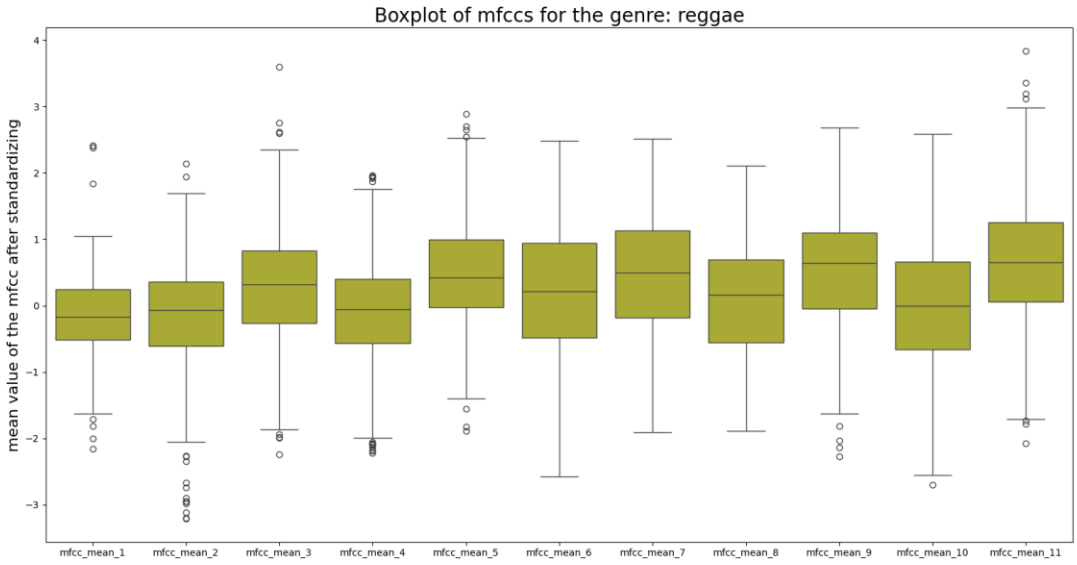

Boxplot of mfccs for the genre: disco
mean value of the mfcc after standardizing
mfcc_mean_1
mfcc_mean_2
mfcc_mean_3
mfcc_mean_4
mfcc_mean_5
mfcc_mean_6
mfcc_mean_7
mfcc_mean_8
mfcc_mean_9
mfcc_mean_10
mfcc_mean_11

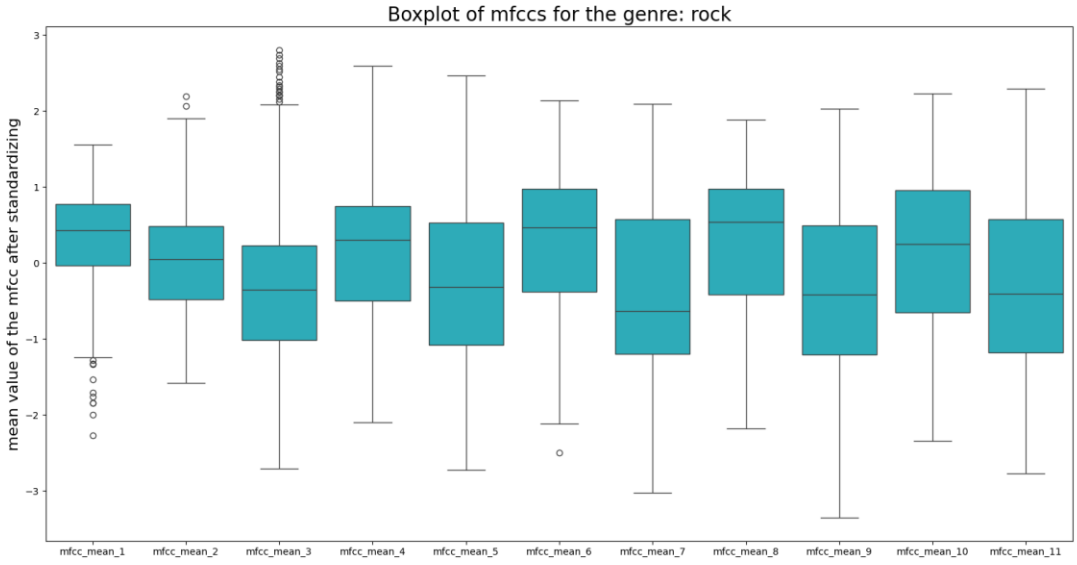

Boxplot of mfccs for the genre: hiphop
mean value of the mfcc after standardizing
mfcc_mean_1
mfcc_mean_2
mfcc_mean_3
mfcc_mean_4
mfcc_mean_5
mfcc_mean_6
mfcc_mean_7
mfcc_mean_8
mfcc_mean_9
mfcc_mean_10
mfcc_mean_11


Boxplot of mfccs for the genre: jazz
mean value of the mfcc after standardizing
mfcc_mean_1
mfcc_mean_2
mfcc_mean_3
mfcc_mean_4
mfcc_mean_5
mfcc_mean_6
mfcc_mean_7
mfcc_mean_8
mfcc_mean_9
mfcc_mean_10
mfcc_mean_11


Boxplot of mfccs for the genre: metal
mean value of the mfcc after standardizing
mfcc_mean_1
mfcc_mean_2
mfcc_mean_3
mfcc_mean_4
mfcc_mean_5
mfcc_mean_6
mfcc_mean_7
mfcc_mean_8
mfcc_mean_9
mfcc_mean_10
mfcc_mean_11

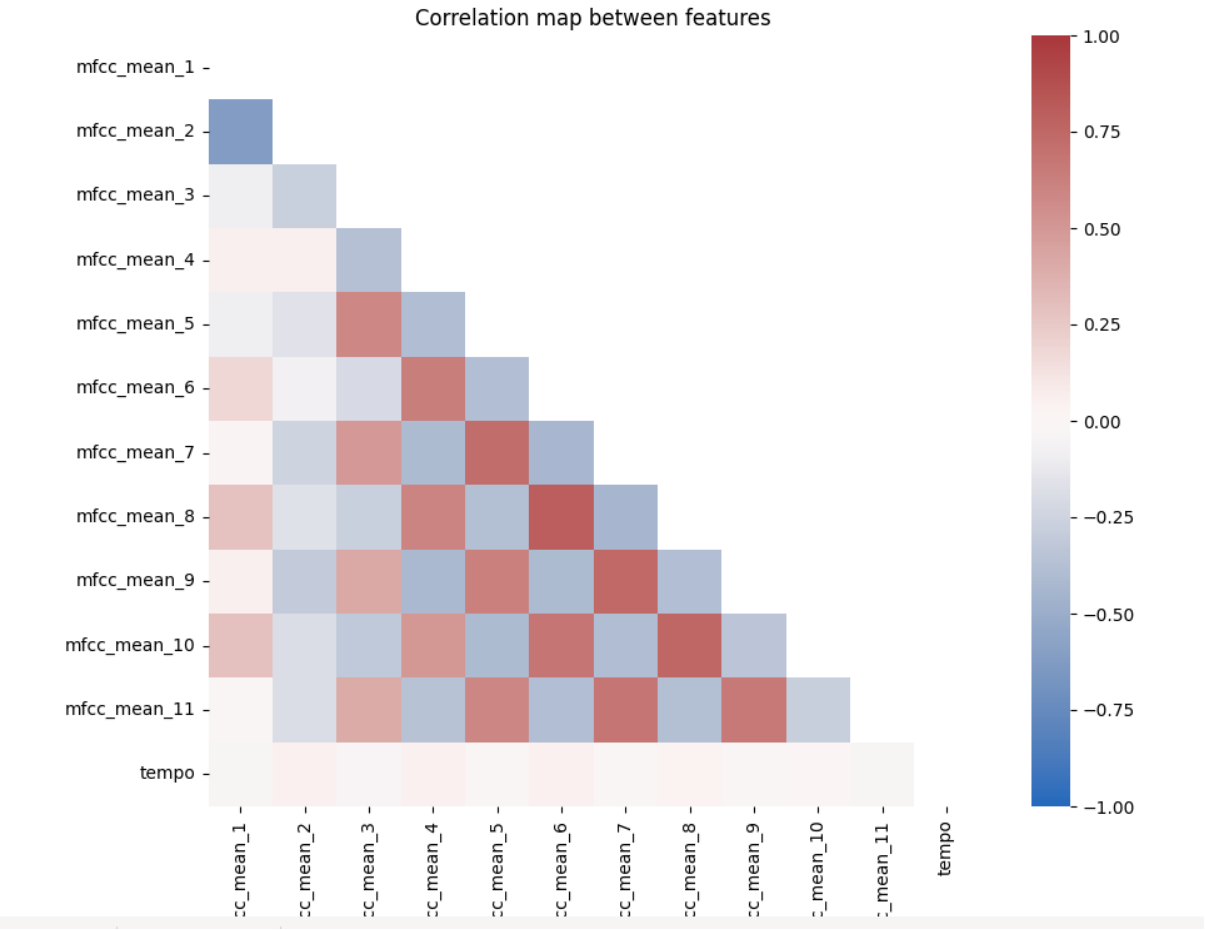

Boxplot of mfccs for the genre: pop
mean value of the mfcc after standardizing
mfcc_mean_1
mfcc_mean_2
mfcc_mean_3
mfcc_mean_4
mfcc_mean_5
mfcc_mean_6
mfcc_mean_7
mfcc_mean_8
mfcc_mean_9
mfcc_mean_10
mfcc_mean_11





Na podstawie powyższych wykresów widać różnice w wartościach MFCC oraz tempa pomiędzy gatunkami, zatem te cechy nadają się do odróżniania gatunków od siebie.

4. Macierz korelacji cech



Zwiększanie ilości MFCC, od pewnej ich ilości, nie zwiększało znacząco dokładności klasyfikacji. Jak można zauważyć, poszczególne cechy są ze sobą skorelowane w dosyć przewidywalny sposób – na zmianę raz dodatnio, raz ujemnie. Kolejne, silnie skorelowane cechy nie są zatem użyteczne dla algorytmu klasyfikującego, ponieważ jest to powtórzenie informacji, które niosą poprzednie cechy.

5. Wybór modelu

Wypróbowałem 3 algorytmy uczenia maszynowego: maszynę wektorów nośnych, las losowy oraz k-najbliższych sąsiadów. Przedstawiam rezultaty dla każdego z nich.

Maszyna wektorów nośnych (SVM):

```
Fitting 5 folds for each of 32 candidates, totalling 160 fits
              precision    recall  f1-score   support

       blues       0.78      0.88      0.83        92
   classical       0.91      0.93      0.92       108
     country       0.64      0.69      0.66        96
       disco       0.73      0.82      0.77        94
      hiphop       0.75      0.74      0.75       107
        jazz       0.76      0.74      0.75       100
       metal       0.91      0.86      0.89        99
         pop       0.89      0.84      0.86       105
      reggae       0.76      0.77      0.76        92
        rock       0.76      0.64      0.70       106

    accuracy                           0.79       999
   macro avg       0.79      0.79      0.79       999
weighted avg       0.79      0.79      0.79       999
```

Las losowy (RF):

```
              precision    recall  f1-score   support

       blues       0.96      0.81      0.88       110
   classical       0.83      0.87      0.85        95
     country       0.75      0.74      0.74        96
       disco       0.70      0.79      0.74        92
      hiphop       0.79      0.77      0.78        95
        jazz       0.77      0.81      0.79        98
       metal       0.80      0.94      0.86        93
         pop       0.84      0.89      0.87       110
      reggae       0.78      0.80      0.79       101
        rock       0.84      0.64      0.73       109

    accuracy                           0.80       999
   macro avg       0.81      0.81      0.80       999
weighted avg       0.81      0.80      0.80       999
```

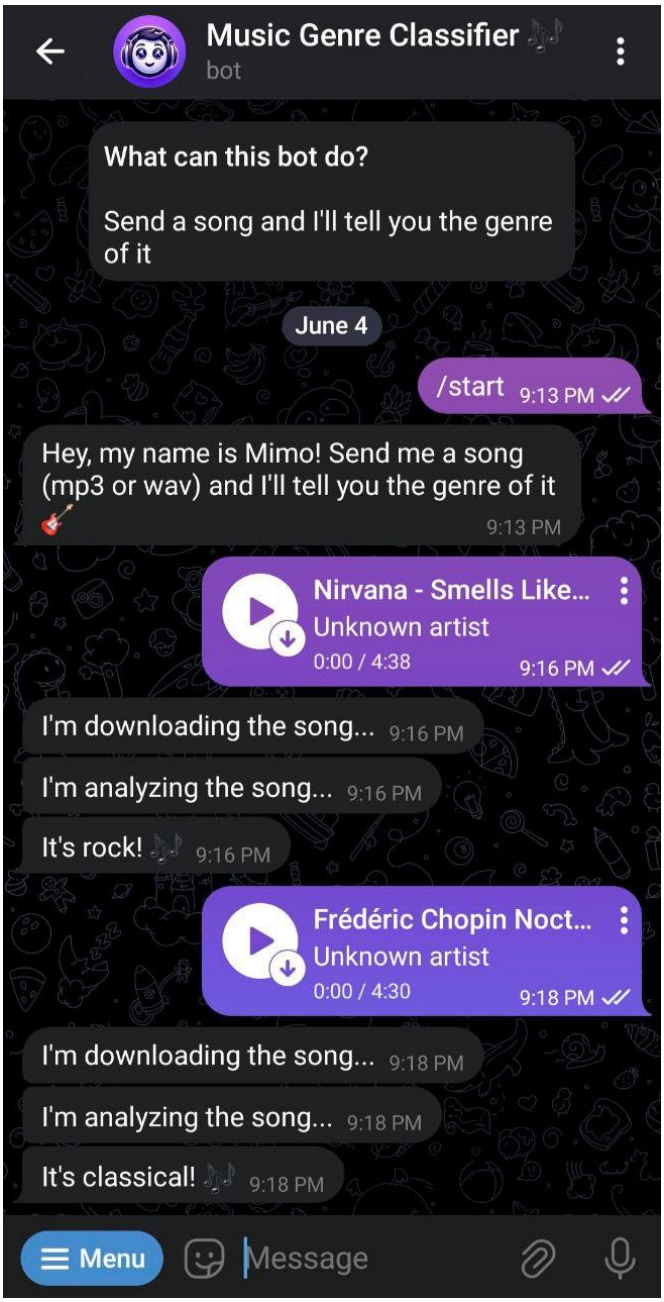
K-najbliższych sąsiadów (KNN):

```
Fitting 5 folds for each of 42 candidates, totalling 210 fits
              precision    recall  f1-score   support

       blues       0.90      0.88      0.89        52
   classical       0.93      0.89      0.91        64
     country       0.89      0.79      0.84        42
       disco       0.85      0.97      0.91        36
      hiphop       0.92      0.89      0.91        55
        jazz       0.77      0.90      0.83        40
       metal       0.90      0.97      0.94        37
         pop       0.94      0.92      0.93        66
      reggae       0.92      0.94      0.93        48
        rock       0.93      0.87      0.90        60

    accuracy                           0.90       500
   macro avg       0.90      0.90      0.90       500
weighted avg       0.90      0.90      0.90       500
```

Jak można zauważyć, klasyfikator KNN oferuje dla mojego zbioru danych najwyższą skuteczność spomiędzy powyższych algorytmów, więc to z niego zdecydowałem się skorzystać.

6. Sposób użycia

Telegramowy interfejs pozwala na intuicyjne korzystanie z programu. Należy przesłać plik w formacie mp3.



7. Napotkane problemy

- Sporadyczne niepoprawne odczytywanie MFCC z plików mp3 (wartości równe 0). Wstępnie dostrzegłem, że nie dotyczy to plików wav, jednak tych nie obsługuję poprzez Telegramowy interfejs. Można użyć konwertera z mp3 na wav i sprawdzić, czy problem zostanie rozwiązany.

8. Niedociągnięcia

- Brak wyświetlania użytkownikowi informacji o błędzie (np. timeout).

9. Plany dalszego rozwoju

- Poprawienie skuteczności klasyfikacji zamieniając klasyfikator KNN na sieć neuronową oraz poszerzając zbiór danych do trenowania.
- Przejście z plików w formacie JSON na bazę danych.
- Wprowadzenie możliwości uczenia programu z pozycji użytkownika – gdy program źle sklasyfikuje utwór, użytkownik jest proszony o sklasyfikowanie utworu. Program zapisuje utwór i jego gatunek, następnie dodaje go do zbioru danych, z którego ponownie się uczy.
- Dodanie obsługi plików wav bezpośrednio poprzez Telegramowy interfejs, bez konieczności konwertowania ich na mp3.

10. Źródła

GitHub projektu: https://github.com/kacperfin/music-genre-prediction

Zbiór danych: https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification