研究班番号【 130 】

機械学習を用いた楽器の分類

情報班:安保　晃大、田尾　悠翔、平松　香穂理

要約

本研究では、Python・機械学習を用いて、簡易に複数の楽器音が同時に記録されている音源に対して楽器の音の分類を行うことを目的に、方法の検討・プログラムの作成を行った。特徴量の抽出法としてMFCC(メル周波数ケプストラム係数)、モデルとしてSVM(サポートベクターマシン)を利用した。

その結果、楽器一種類のみの音が収録された音源では約99%、複数種類の楽器の音が同時に収録された音源では各楽器平均86%という結果での分類を行えた。

しかし複数種類の楽器の音の分類に関して、過不足なく全楽器の音を分類できたケースに関しては約30%の結果に終わり、これは今後の課題であると考えられる。

1. はじめに

音楽を聴いているときに、その音楽に使用されている楽器が何かと疑問に感じることがある。しかしそれらの音楽に使用されている楽器を初心者の人間が聴き分けることは難しい。そこで、音楽に使用されている複数の楽器の判定を行い、任意の音楽を読み込ませるとその音楽の再生と同時に鳴っている楽器を表示するプログラムを作成しようと考えた。

具体的には、機械学習の「分類」という手法を用いたプログラムを作成した。そして、研究の第一段階として楽器一種類のみの音が収録された音源について、第二段階として複数種類の楽器の音が同時に収録された音源について分類を試みた。

2. 開発方法

2.1 開発の概要

研究を第一段階と第二段階に分割し、第一段階では楽器一種類のみの音が収録された音源について、第二段階では複数種類の楽器の音が同時に収録された音源について分類を行った。第一段階と第二段階では基本となるプログラムは同一であるが、音源データの読み込み部分やモデルデータ作成の部分等に変更を加えた。

2.2 利用した環境

プログラミング言語は、機械学習に適した言語であるPython 3系を利用した。また、Pythonの実行環境は環境構築の容易さ、開発する端末を問わないことや機械学習と親和性が高いということを考慮し、GoogleがWeb上で提供しているサービスであるGoogle Colaboratoryを利用した。

また、主なライブラリとして、音声の読み込み・特徴量抽出にlibrosa、モデルにscikit-learnを使用した。これらの選定に関しては、参考文献をもとに検討・比較を行った(Python×AI・機械学習入門編)。

2.3 利用した音声ファイル

本研究では、分類する楽器の種類として、楽器数・音声収集の容易さなどを考慮し、一般的な吹奏楽で利用される楽器9種類(図2.3-1参照)についての分類を行った。

音声ファイルは、インターネット上にある一種類の楽器が鳴っている音声を集め、無音部分の消去を行い、44.1Kbps, 1秒, wav形式の音声ファイルに加工を行った。なお、この加工はPythonを用いた。

楽器名
ピアノ
フルート
クラリネット
アルトサックス
トランペット
ホルン
トロンボーン
ユーフォニアム

図 2.3-1

2.4 利用した手法

本研究では、Python上で利用できる既存のライブラリ等を活用し、教師あり機械学習の分類という手法を利用した(第一、第二段階共通)。

機械学習の分類は、「学習」と「予測」の段階に分かれている。「学習」ではあるデータとラベルがセットになったものを入力し、入力されたデータの特徴とラベルを結び付けて

記録する。そして、「予測」ではラベルが未知のデータから特徴を取り出し、この特徴と「学習」で記録したものを基に適切なラベルに分類する。今回の研究では、データが楽器音、ラベルが楽器名に該当する。

また、機械学習の分類の中にも、ラベルへの分類方法や特徴の取り出し(抽出)方法にはいくつか種類がある。この研究では、分類手法としてサポートベクターマシン(以下SVM)、特徴量の抽出法としてメル周波数ケプストラム係数(以下MFCC)を利用した。

2.4.1 SVM

SVMとは、データの特徴量とカテゴリーを基に、異なるカテゴリーのデータ間の距離が最小となる境界線を引いたモデルのことである。

簡略化のため図2.4.1-1を用いて考えると、特徴量を基に複数のデータ(この図では丸)をX-Y座標に置いたとき、黒丸と白丸という同一カテゴリーでそれぞれ塊ができており、その間には線を引くことができる。この線が境界線のことであり、SVMではこの線をデータ間の距離が最小になるように引く。このモデルに新しいデータを入れると、そのデータが境界線より上にあるか、下にあるかによってどちらのカテゴリーに属するか分類することができるようになる。

本研究では、SVMの精度の高さ、利用の簡易さを考慮し使用することとした。

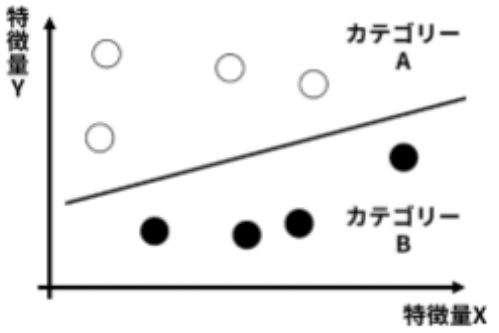

図 SEQ 図 * ARABIC 1.4.1-1

2.4.2 MFCC

MFCCとは、音声データを人間の聴覚特性を考慮することができるデータに変換する手法である。

具体的には、音の波形をフーリエ変換によって周波数についての解析に変換し、その際にメル尺度とよばれる尺度を使用する。この尺度によって人間は周波数の低い音には敏感で高い音には鈍感というように人間の聴覚特性を考慮して音を解析することができるようになる。こうして得られたスペクトルにメルフィルタバンクを掛けメルスペクトルにし、さらにメルスペクトルを離散コサイン変換し、得られたものがMFCCである。

本研究では、人間の聴覚特性が楽器の音色を分類する上で適しているため、この手法を使用することにした。

2.5 プログラムの流れ

2.5.1 第一段階プログラム

①楽器1種類のみの音声を読み込み。
②音声を特徴量に変換(抽出)。
③学習させモデルデータを作成。
④予測(正確率の調査)。

2.5.2 第二段階プログラム(図2.5.2-1)

①:楽器1種類のみの音声を読み込み。
②:9種類の楽器の音声をランダムに合成。
③:音声を特徴量に変換(抽出)。
④:学習させモデルデータを作成。
⑤:予測(正確率の調査)。

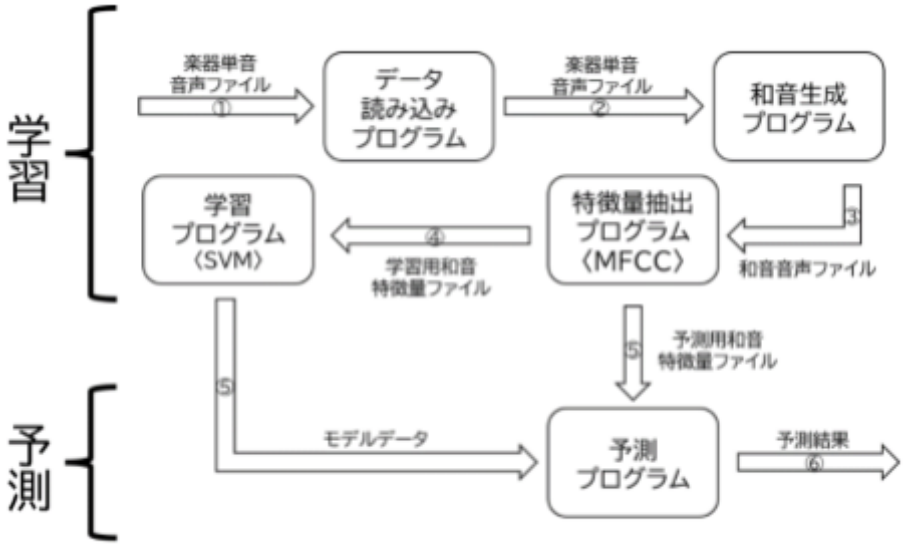

図 2.5.2-1

なお、前述の通り第一、第二段階のプログラムは基本的に同様であるが、第二段階プログラムでは、単一楽器の音声を合成することによって複数の楽器が記録された音声を作成するプログラム(図2.5.2-1「和音生成プログラム」)と、モデルデータ作成プログラム(図2.5.2-1「学習プログラム」内、2.6参照)を新たに作成し、複数楽器の分類への対応を行った。

2.6 モデルデータ作成の手法

今回の研究で独自性を持たせた部分として、モデルデータ作成の手法が挙げられる。一般的な手法では、モデル作成の際に予測したい音声の組み合わせの数だけデータが必要 になり、複雑な多クラス分類を行う必要がある(今回の場合では510クラス分類)。これは、膨大な時間がかかり非効率である。また、今回利用したSVMは2クラス分類が最適であり、それ以上の多クラス分類には適していない。そこで本研究では、モデルを楽器の数だけ作成、各モデルに対して担当の楽器を設定し、「担当の楽器が鳴っている、もしくは鳴っていない」という2クラス分類を複数組み合わせることによって効率の向上を目指した。

3. 結果

研究第一段階・第二段階の両方において、用意した楽器音が収録された音源を学習用・検証用の二つに分割を行い、学習用データをもとに学習・分類をさせて、検証用データを使用し予測、正答率を求めることによってこの手法・プログラムの精度を検証した。

3.1 第一段階プログラムの結果

第一段階プログラムでは、テストデータを45個用意し、結果の偏りを無いようにするため100回実行を行った。その結果、約99.6%の精度で予測することができた。(図3.1-1)

第一段階プログラム - 実行結果

予測が正解したデータ数	全テストデータ数	予測の正解率
44.81	45	約99.58%

図 3.1-1

3.2 第二段階プログラムの結果

第二段階プログラムでは、和音生成・モデル作成・テストを10回繰り返し、合計2437件のテストデータに対して予測を行った。その結果、楽器ごとの結果に注目すると、約86.5%の精度で予測することができた。(図3.2-1)

しかし、複数楽器の音が記録された音声全体で見たときに、すべての楽器が正しく・過不足なく予測されているものに関しては、約30%にとどまった。

第二段階プログラム - 実行結果
各楽器ごと

楽器名	ピアノ	フルート	クラリネット	サックス	トランペット
正解率(平均)	78.42	87.97	95.65	88.30	97.33
楽器名	ホルン	トロンボーン	ユーフォニアム	チューバ	平均
正解率(平均)	80.1	79.45	89.38	81.66	86.47

図 3.2-1

4. 考察

第一段階プログラムによる、楽器一種類の音が記録された音声の分類については99%超という非常に高い精度で予測することができたと考えられる。

また、第二段階プログラムによる複数種類の楽器の音が同時に収録された音声の分類についても、楽器ごとに考えれば86%という高い精度で予測することができた。しかし、音声全体で考慮した場合には正解率が約30％と著しく低下している。これは、図3.2-1のピアノやトロンボーンなどのように正解率が低い楽器が影響しているのだと考えられる。また、特定の楽器の組み合わせによって誤認識が発生している場合も考慮する必要があると考えられる。

5. 結論

第一段階プログラムは、正解率より完成に近い状態であり、今回の研究で行った手法が有用であると考えられる。

しかし第二段階プログラムに関しては、正解率の観点から一定の成果は上げているものの改善の余地も多数残されていると考えられる。考察にて言及した音声全体の正解率低下の原因に関しては、音声全体で予測が不正解のデータによく使用されている楽器の組み合わせを調べることや、楽器の種類によって正解率にばらつきが発生している原因を検討することなどによってより考察を進めることができると考えられる。

また、プログラムのパラメータの調整や、特徴量・モデルに使用している手法を変更することによってもさらに予測精度を高めることも可能であると考えられる。

そして、使用楽器を増やすなどのことを行い、現在の限られた楽器数にとどまらず本来の目的であった様々な楽器が入った私たちが普段耳にする曲にも使用できるようにしたい。

6. 参考文献ならびに参考Webページ

paiza株式会社. Python体験編 - paizaラーニング. https://paiza.jp/works/python/trial

paiza株式会社. Python3入門編 - paizaラーニング. https://paiza.jp/works/python3/primer

paiza株式会社. Python×AI・機械学習入門編 - paizaラーニング. https://paiza.jp/works/ai_ml/primer

山森 一人, 青島 大河, 相川 勝(2016).「くし型フィルタと Multi-class SVMによる混合音からの演奏楽器推定」. 宮崎大學工學部紀要, 45, 211-215.

7. 謝辞

研究を進めるにあたり、大阪工業大学　小林裕之教授にご指導いただきました。この場をお借りして、お礼を申し上げます。