

# Audio-Coding for Python

By Christoph Fuchs

MultimediaArt-Audio | University of Applied Sciences Salzburg

## Introduction

We are currently living in a digital age, where machine learning applications are taking over more and more tasks of our daily lives. For example, the music suggestions that Spotify gives to its users, are generated by an algorithm that is powered by a neural network – a machine learning application that trains itself to understand the taste of its users, in order to give the perfect suggestions to them.

With a basic understanding of the programming language Python, it is possible to write codes that analyze music data automatically and categorize it automatically. Because of those current trends and possibilities, my MMP3 is a research project, which shows some of the tasks that can be fulfilled with Python in the field of Audio Signal Processing as well as Music Information Retrieval.

## Materials and Mehods

To begin with, it was necessary to get basic knowledge in Python programming in general. I did this with the help of “Python 3 Crashkurs” by Eric Mathes.

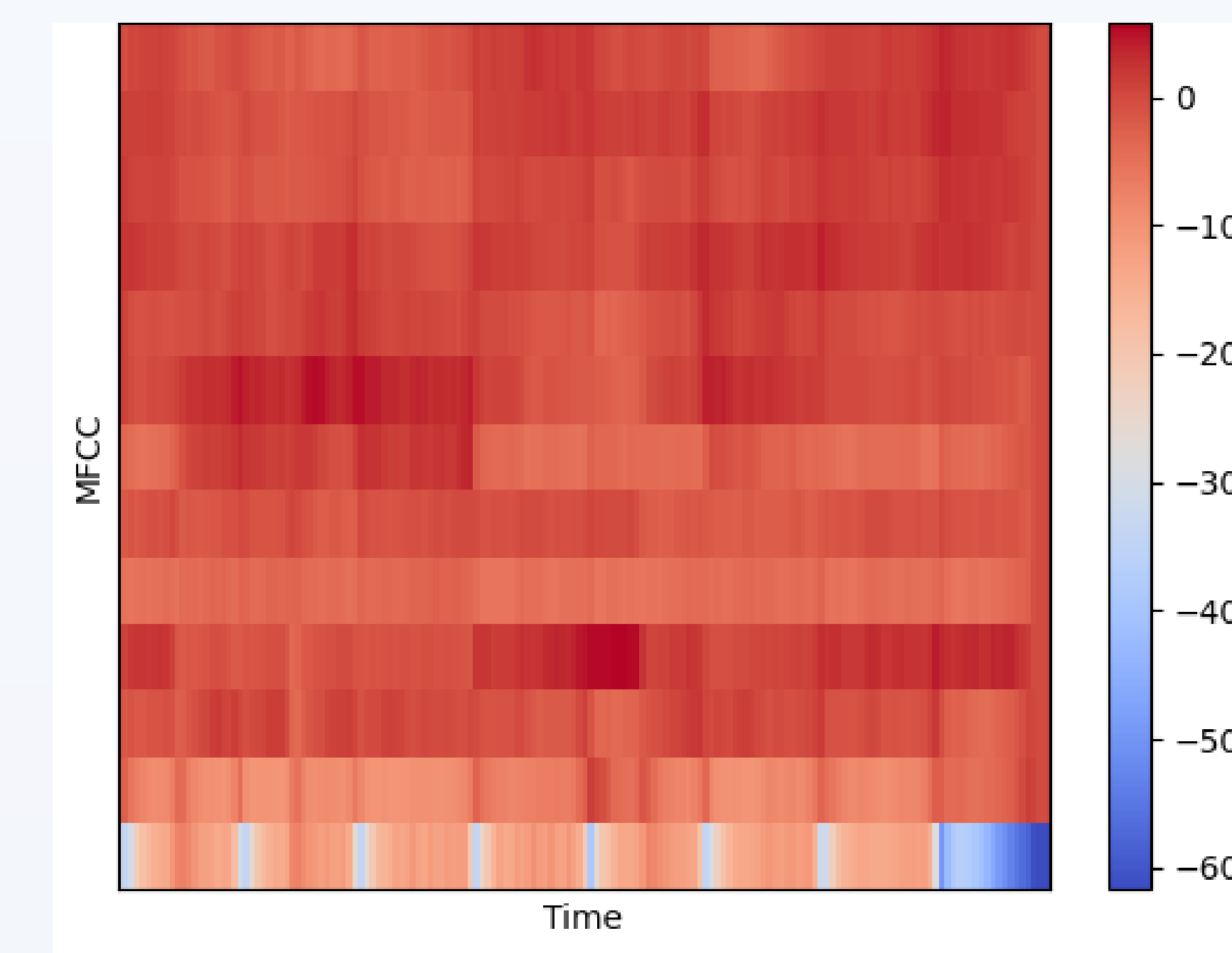
After knowing how to write code in general, I started my research on how to extract features from audio data, which would help me to classify it according to its genre and attributes.

To accomplish this, I watched a Youtube series called “The Sound AI” by Valerio Velardo, who teaches the basics of sound characteristics and Music Information Retrieval (MIR).

Lastly, it was necessary to create a neural network for machine learning, which would train itself in order to classify audio data correctly. This was also explained by Valerio Velardo.

## Results

Here are some of the features that I extracted, visualized as a Mel-spectrogram:



Those kind of visualizations can be used to feed a neural network in order to build a genre classification system with the help of Python.

```
1/2 [=====>.....] - ETA: 0
2/2 [=====] - 0s 53m
Epoch 30/30
1/2 [=====>.....] - ETA: 0
2/2 [=====] - 0s 52m
1/1 [=====] - ETA: 0
1/1 [=====] - 0s 21m
Accuracy on test set is: 0.2000000298023224
Expected index: 9, Predicted index: [4]
[Finished in 18.1s]
```

In this case, if only 10 audio samples were used to train the network, it has already an accuracy of 20%. For commercially used programs it is common to use the Million Song Data Set by “The Echo Nest”.

## References

Eric Matthes (2020): Python 3 Crashkurs. dpunkt.verlag GmbH

Valerio Velardo: The Sound of AI (Youtube Channel)

Brian McFee et al (2015): librosa: Audio and Music Signal Analysis in Python

Corey Schafer (Youtube Channel)

Keith Galli (Youtube Channel)

Allen Downey (Github)

## Special thanks

Martin Löcker

David Fasching

Marius Schebella

Patrik Lechner

Selina Milla

Leonhard Jauch

## Contact

christophfuchs@gmx.at