QFEbonn

Quantitative Finance & Economics

Spotify Music Genre Analysis

Quantitative Finance & Economics Bonn

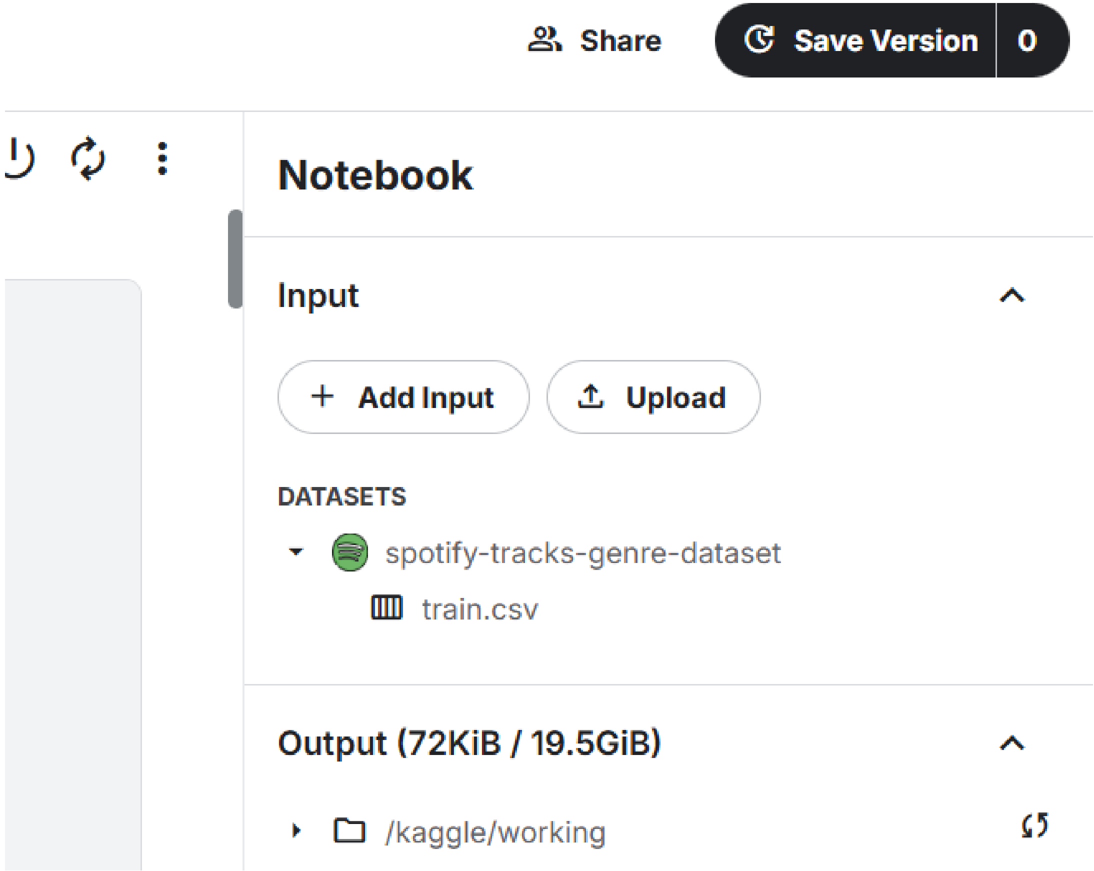
Introduction

- Dataset: Spotify music from Kaggle and Hugging Face
- Goal: Clean messy text data to prepare for analysis
- Why cleaning matters:
  dirty data -> **flawed** insights

QFEbonn
Quantitative Finance & Economics

Setting Up

- We will be use Kaggle to directly access data and run code live
- You will need to:
  - Sign up at Kaggle.com
  - Create a new notebook
  - Add this dataset in your input: kaggle.com/datasets/thedevastator/spotify-tracks-genre-dataset



Share

Save Version | 0

Notebook

Input

+ Add Input

Upload

DATASETS

spotify-tracks-genre-dataset

train.csv

Output (72KiB / 19.5GiB)

/kaggle/working

Data Inspection

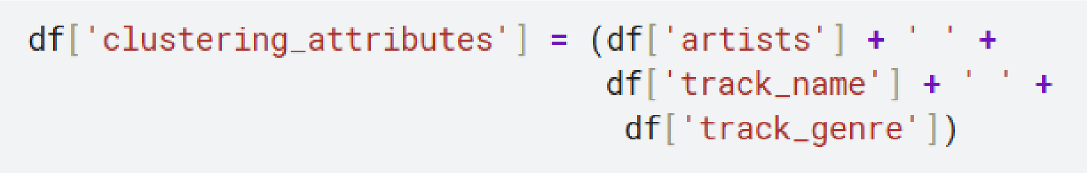
- Use head(), shape, describe(), info(), nunique() to have a quick look on the data
- Check if there's duplicates or if any of the column has missing values
- Try to find out which data entries has missing values
- Use fillna() to replace missing values (e.g. '' or 'Unknown')
- Find out which columns are not numerical ('object')
- Find out which artist has the most tracks

Data Visualization

- Use plt and sns.histplot() to plot the Distribution of Track Popularity
- Use groupby() to find the top 10 genres with the highest mean of Popularity
- Use sns.boxplot() to plot the Distribution of Popularity by Genre (Top 10 Genre)

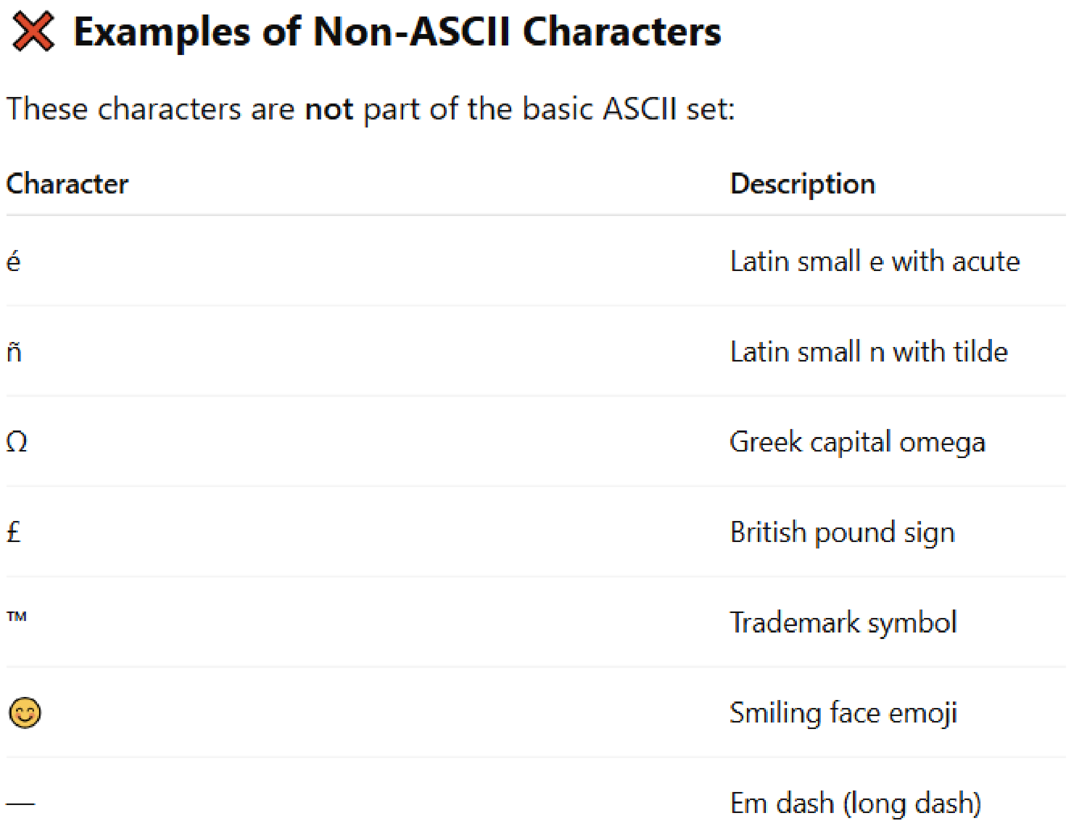
Data Cleaning

- Let’s create a new column ‘clus_att’ short for clustering attributes
- We focus now on cleaning this new column:
  - Remove Punctuation
  - Remove Non ASCII Characters
  - Remove Stop Words
  - Remove Duplicates
  - Tokenize Words
  - Lemmatize Verbs



```
df['clustering_attributes'] = (df['artists'] + ' ' +
                               df['track_name'] + ' ' +
                                df['track_genre'])
```

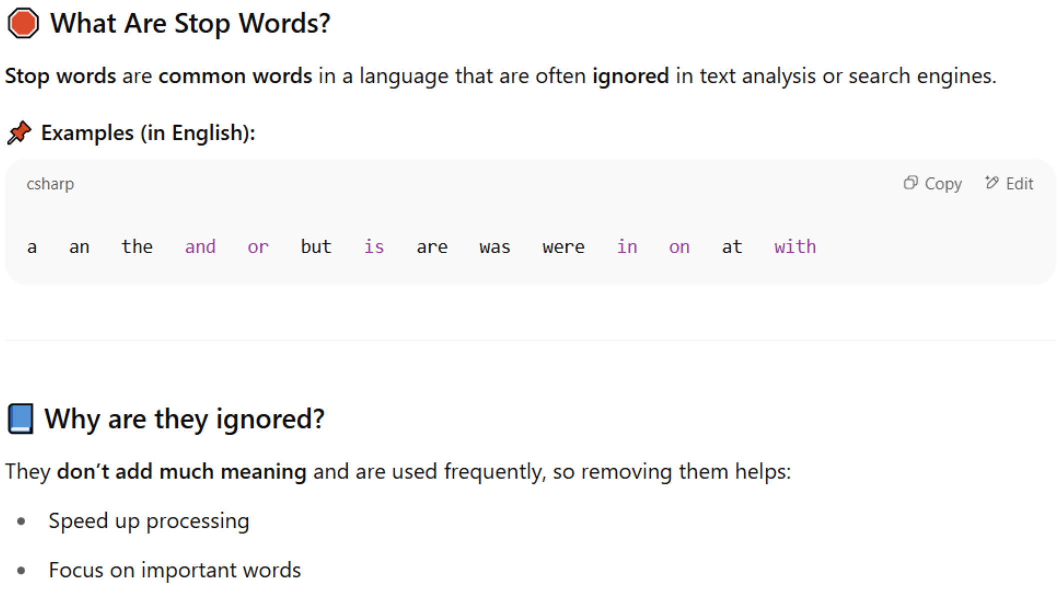
Non-ASCII Characters and Stop Words



❌ Examples of Non-ASCII Characters

These characters are **not** part of the basic ASCII set:

Character	Description
é	Latin small e with acute
ñ	Latin small n with tilde
Ω	Greek capital omega
£	British pound sign
™	Trademark symbol
😊	Smiling face emoji
—	Em dash (long dash)



🛑 What Are Stop Words?

Stop words are **common words** in a language that are often **ignored** in text analysis or search engines.

📌 **Examples (in English):**

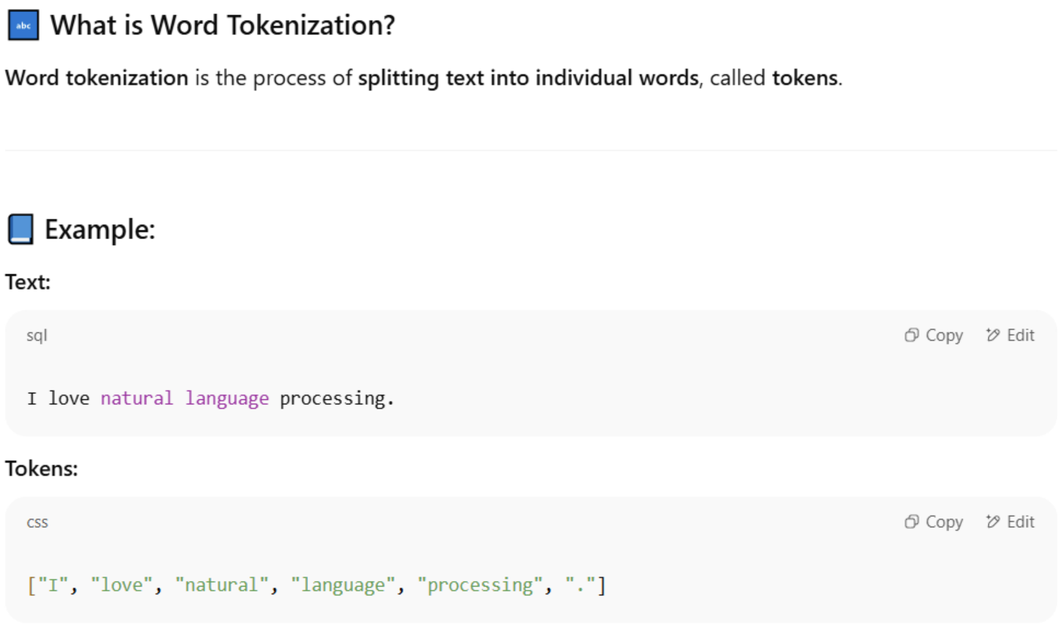
```csharp
a   an   the   and   or   but   is   are   was   were   in   on   at   with
```

📘 Why are they ignored?

They **don't add much meaning** and are used frequently, so removing them helps:

- Speed up processing
- Focus on important words

Tokenize and Lemmatize Words



🔤 What is Word Tokenization?

Word tokenization is the process of **splitting text into individual words**, called **tokens**.

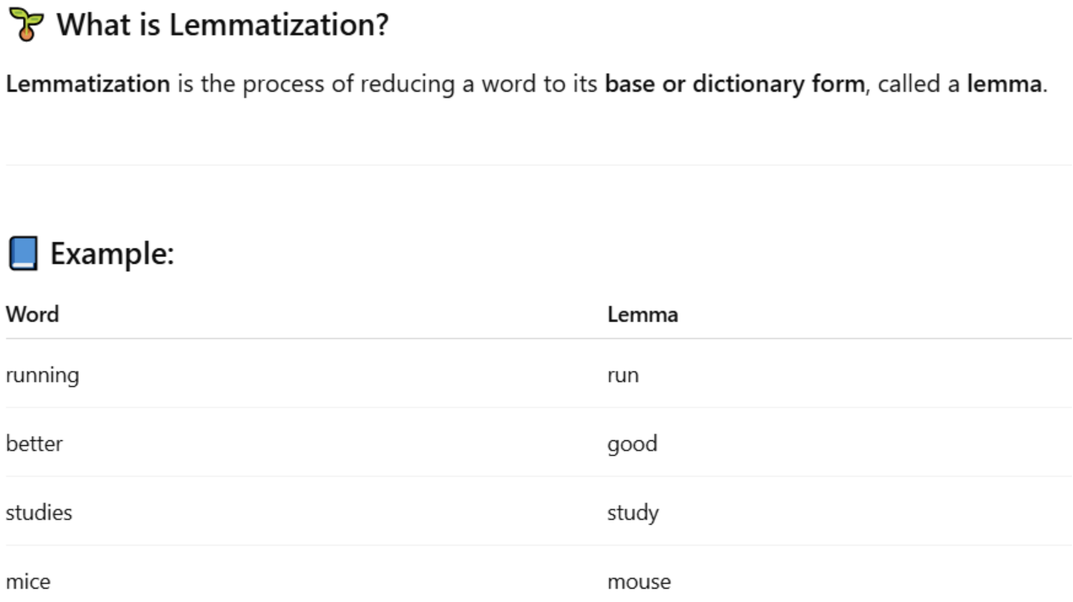
📘 Example:

Text:

```sql
I love natural language processing.
```

Tokens:

```css
["I", "love", "natural", "language", "processing", "."]
```



🌱 What is Lemmatization?

Lemmatization is the process of reducing a word to its **base or dictionary form**, called a **lemma**.

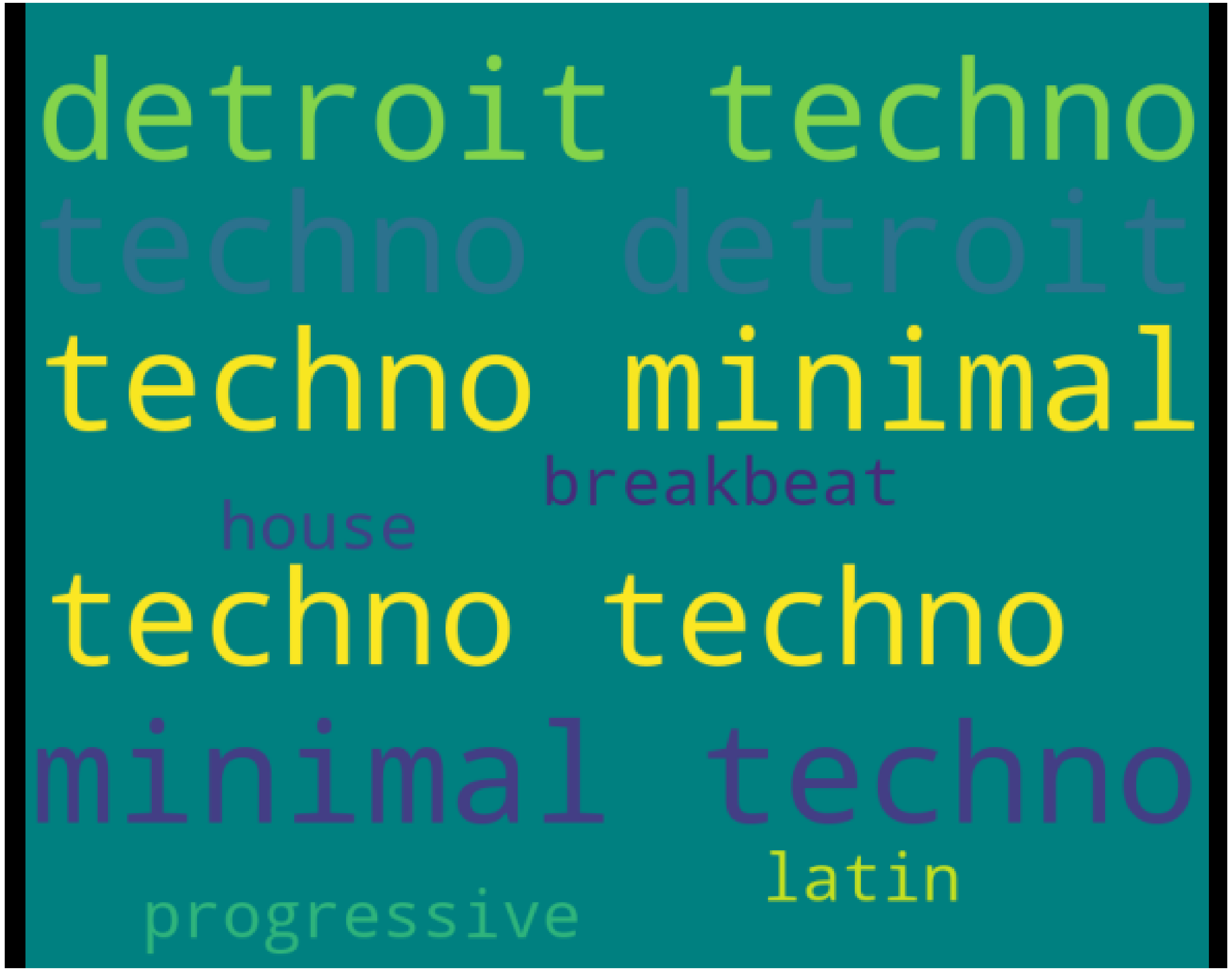
📘 Example:

Word	Lemma
running	run
better	good
studies	study
mice	mouse

Word Cloud

Common music genres of the 1st cluster

Computed by **K-means clustering** (K=6)

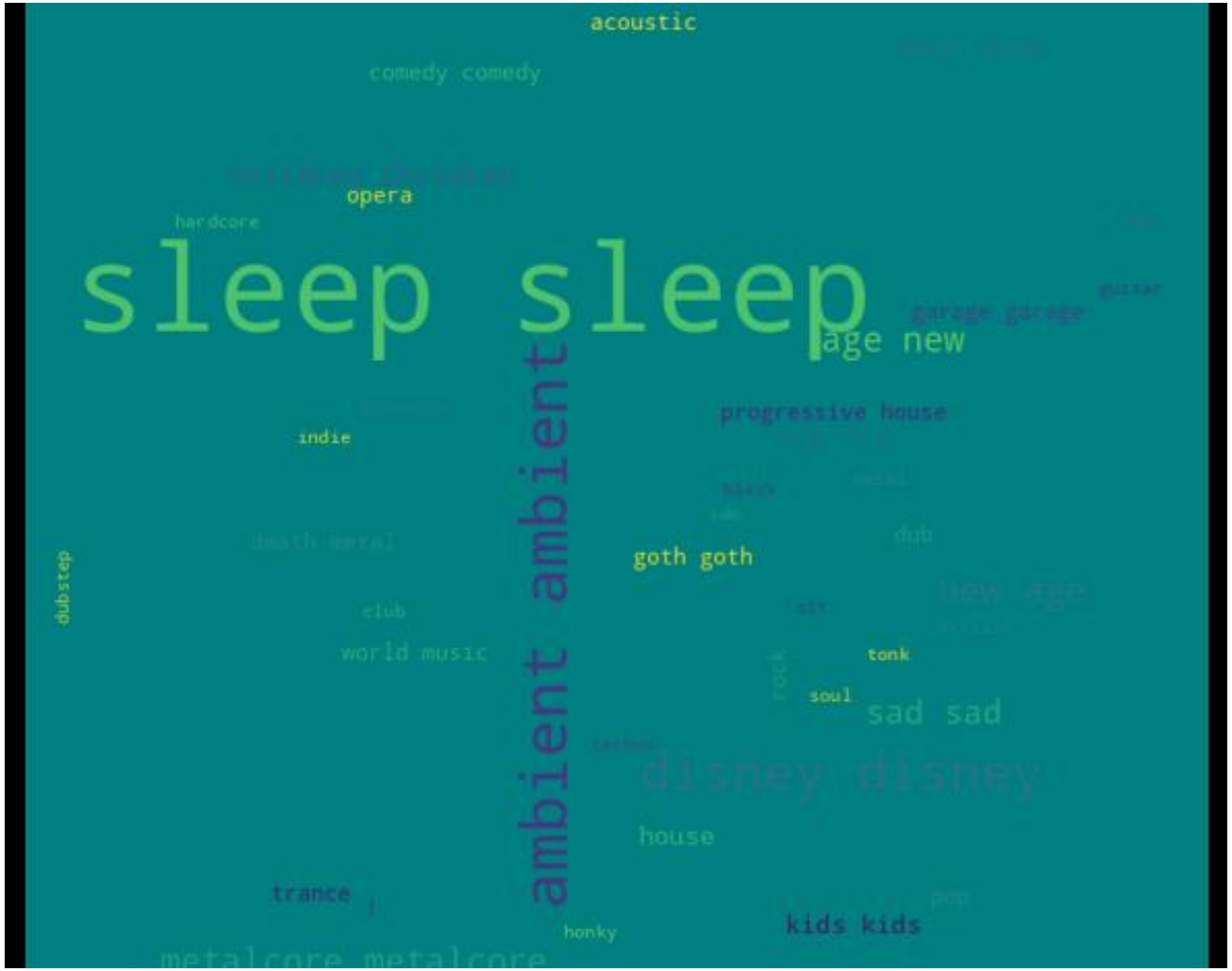


QFEbonn
Quantitative Finance & Economics

Word Cloud

Common music genres of the 2nd cluster

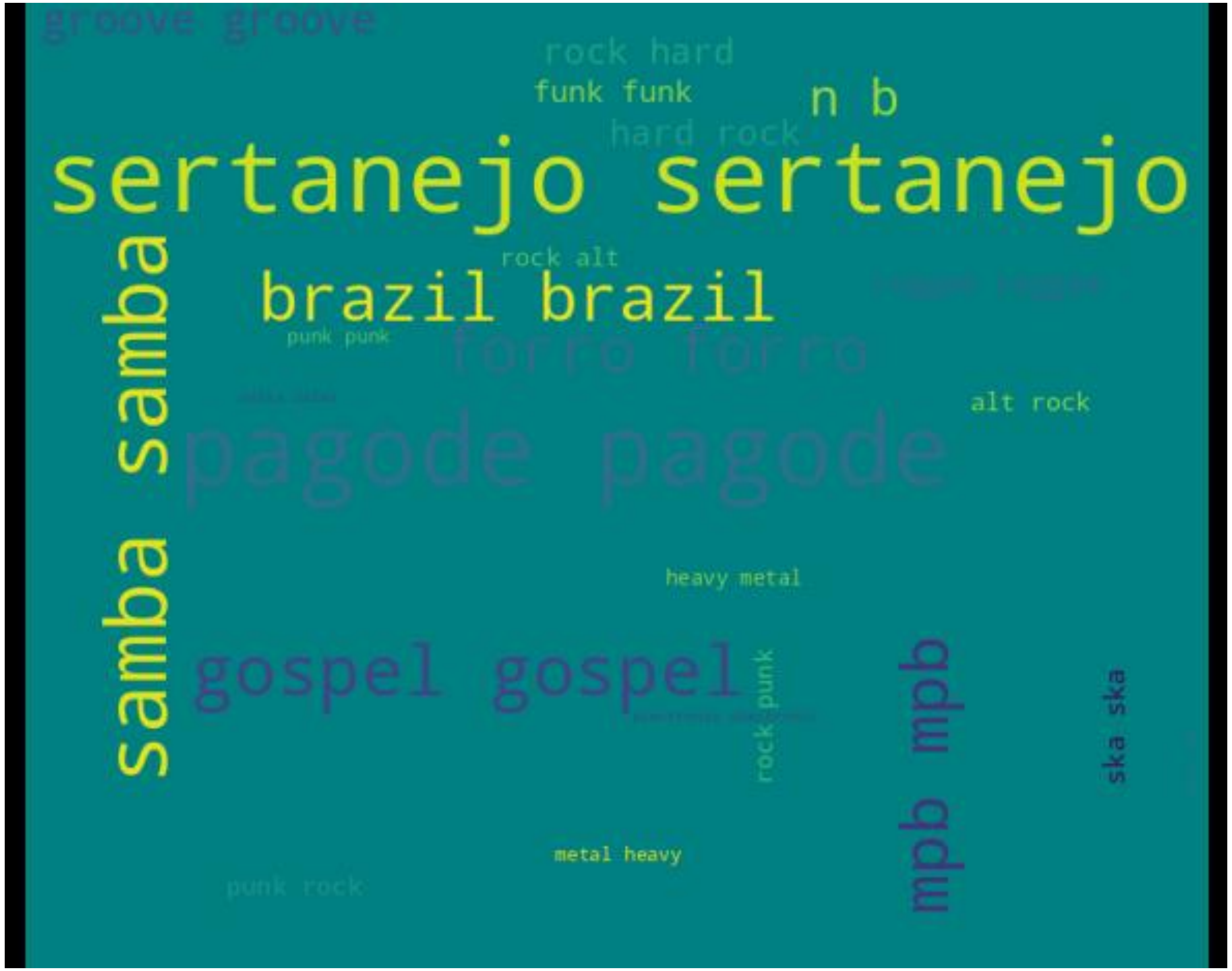
Computed by **K-means clustering** (K=6)



Word Cloud

Common music genres of the 3rd cluster

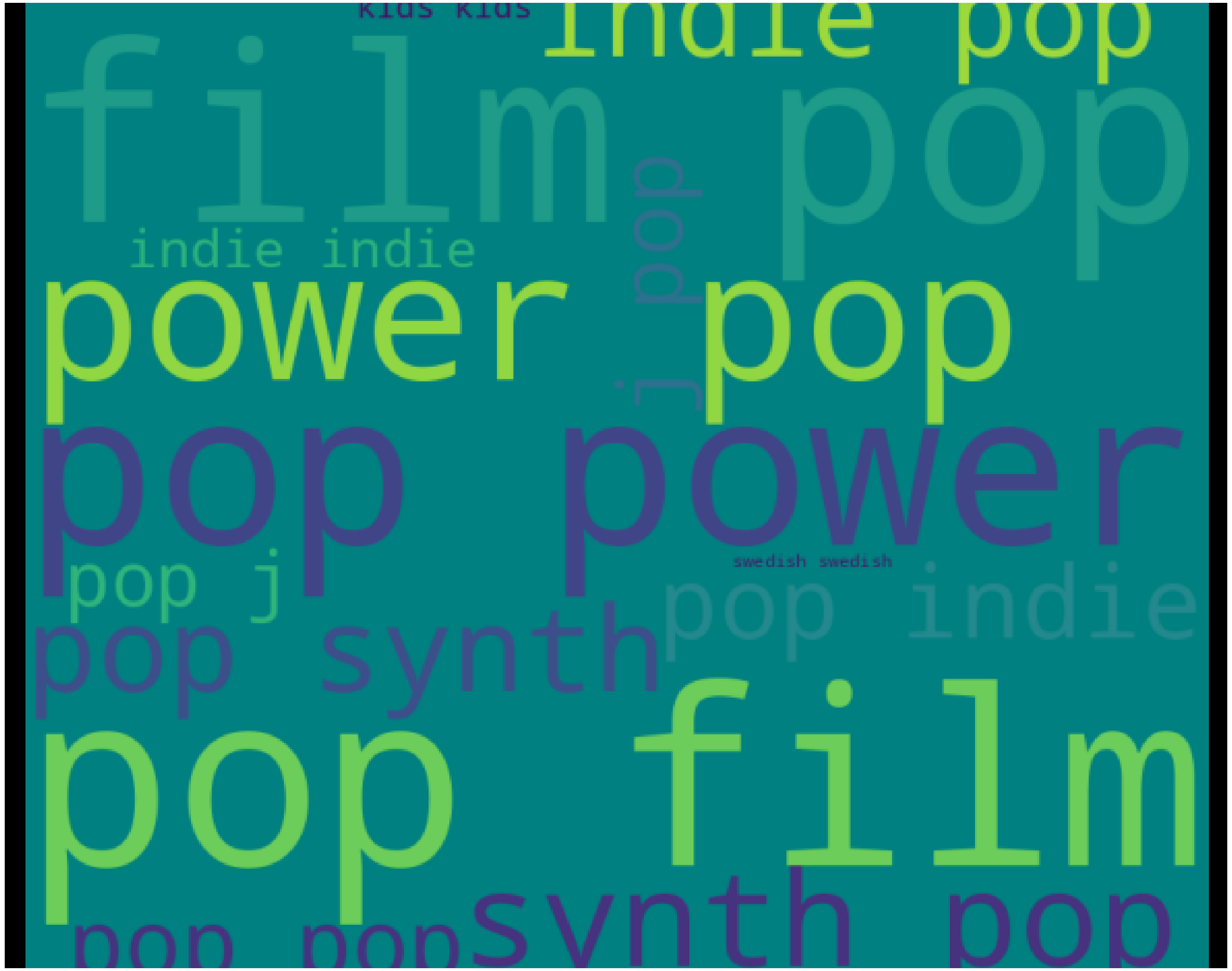
Computed by **K-means clustering** (K=6)



Word Cloud

Common music genres of the 4th cluster

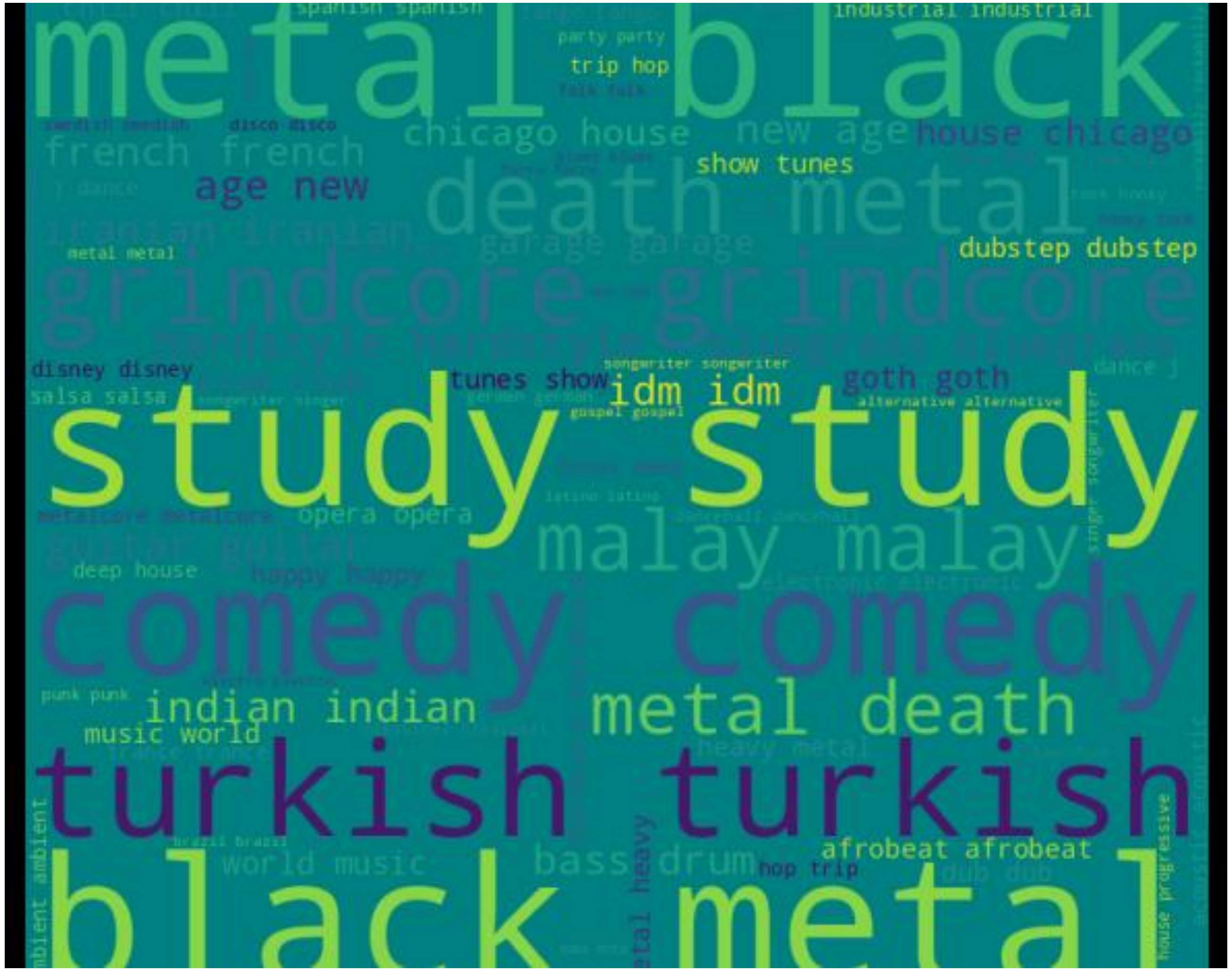
Computed by **K-means clustering** (K=6)



Word Cloud

Common music genres of the 5th cluster

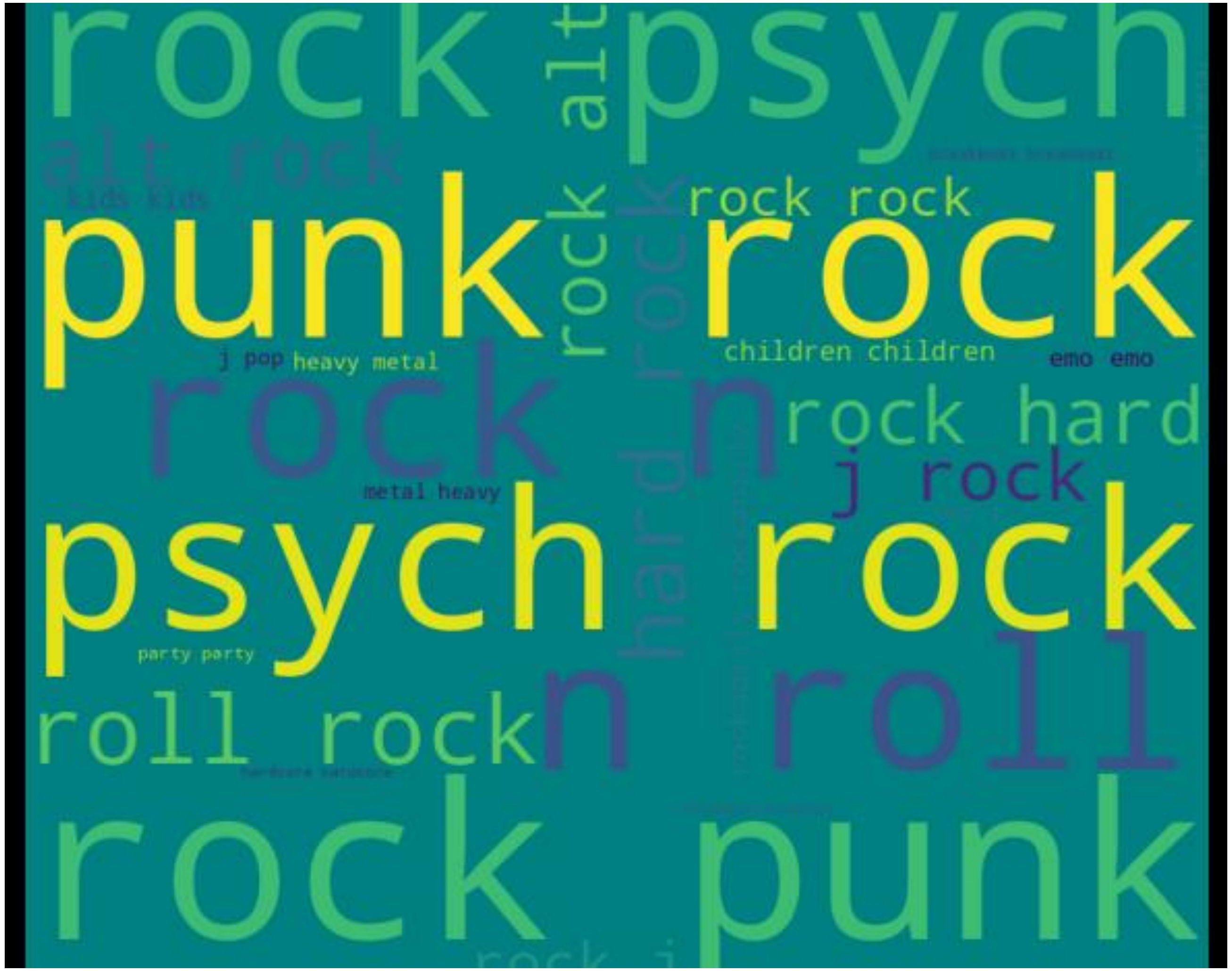
Computed by **K-means clustering** (K=6)



Word Cloud

Common music genres of the 6th cluster

Computed by **K-means clustering** (K=6)



Thank You

Quantitative Finance & Economics Bonn