INTRODUCTION TO DIGITAL MUSICOLOGY

Fabian C. Moss

2025-07-05

Table of contents

Preface

This page contains material for the course **Introduction to Digital Musicology**, held at Julius-Maximilians-Universität, Würzburg (Germany) in Fall 2025.

⚠ Warning

This is work in progress.

Test citation Knuth (1984).

Part I.

INTRODUCTION

1. What is Digital Musicology?

Introduction and terminology

ℹ Goal

Understanding what "digital musicology" means.

- Introduction
- Overview of the field
- Terminology
  - e.g. digital vs computational; the latter in 2nd semester
  - digital vs empirical vs quantiative
  - how does DM relate to "traditional" subdivisions of musicology?

Exercise

Read.

2. Digital Musicology today

ℹ Goal

Acquiring and overview of current activities in Digital Musicology.

- Current research topics
- Important institutions and people (also, e.g. NFDI4Culture)
- Central journals and conferences

3. The history of Digital Musicology

ℹ Goal

Knowing the beginnings and the major stages of DM.

Part II.

DATA ABOUT MUSIC

4. RISM metadata

Goal

Learn what metadata are and how to search for music sources on RISM Online.

- What is RISM?
- What is RISM Online?

Exercise

Understand basic SPARQL and design queries via prompting.

5. Spotify and MusicBrainz metadata

ℹ Goal

Understand the kind of metadata provided by Spotify vs MusicBrainz.

6. Music and the streaming industry

ℹ Goal

Gain first insights into the music market and its workings.

💡 Exercise

Work with sales data.

7. Analyzing song survival

In this session, we will analyze songs from the Billboard 100 charts and trace their ‘course of life’ in the charts.

The data was obtained from Kaggle, a large community website for data analysis challenges.

As before, we first import the `pandas` library for data analysis and load the data using the `read_csv` fundtion that takes as its main argument the path to the data file, in our case `charts.csv`.

```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("data/charts.csv")
```

Inspecting the first 5 lines with the `.head()` method of pandas DataFrames, we obtain an understanding of the structure of the data.

```
df.head()
```

	date	rank	song	artist	last-week	peak-rank	weeks-on-bo
0	2021-11-06	1	Easy On Me	Adele	1.0	1	3
1	2021-11-06	2	Stay	The Kid LAROI & Justin Bieber	2.0	1	16
2	2021-11-06	3	Industry Baby	Lil Nas X & Jack Harlow	3.0	1	14
3	2021-11-06	4	Fancy Like	Walker Hayes	4.0	3	19
4	2021-11-06	5	Bad Habits	Ed Sheeran	5.0	2	18

Think: What do the columns represent? Provide verbal descriptions of their meaning and write it down.

After this general overview, we might want to achieve a slightly deeper understanding. For instance, it is not difficult to interpret the `date` column, but from only the first few entries, we cannot know the temporal extend of our data.

Let’s find out what the earliest and latest dates are using the `.min()` and `.max()` methods, respectively.

```
df["date"].min(), df["date"].max()
```

```
('1958-08-04', '2021-11-06')
```

This tells us that the data stored in `charts.csv` runs from August 1958 to November 2021 and thus allows us to trace the movement of songs in the Billboard charts across more than 60 years.

```
# Top artists
df.artist.value_counts()
```

```
artist
Taylor Swift                                                  1023
Elton John                                                     889
Madonna                                                        857
Drake                                                          787
Kenny Chesney                                                  769
                                                              ...
YoungBoy Never Broke Again Featuring Sherhonda Gaulden           1
Drake Featuring Chris Brown                                      1
Kehlani Featuring Jhene Aiko                                     1
DaBaby Featuring A Boogie Wit da Hoodie & London On Da Track     1
The Shins                                                        1
Name: count, Length: 10205, dtype: int64
```

```
# Longest in charts
df.sort_values(by="weeks-on-board", ascending=True).iloc[50_000:]
```

	date	rank	song	artist	last-week	peak-rank	we
213768	1980-11-22	69	Turn And Walk Away	The Babys	79.0	69	2
106440	2001-06-16	41	Fill Me In	Craig David	69.0	41	2
106443	2001-06-16	44	Bootylicious	Destiny's Child	66.0	44	2
106448	2001-06-16	49	All Or Nothing	O-Town	60.0	49	2
213674	1980-11-29	75	My Mother's Eyes	Bette Midler	85.0	75	2
...	...	...	...	...	...	...	...
39148	2014-05-10	49	Radioactive	Imagine Dragons	48.0	3	87
1215	2021-08-14	16	Blinding Lights	The Weeknd	17.0	1	87
1117	2021-08-21	18	Blinding Lights	The Weeknd	16.0	1	88
1020	2021-08-28	21	Blinding Lights	The Weeknd	18.0	1	89
919	2021-09-04	20	Blinding Lights	The Weeknd	21.0	1	90

```
df["date"] = pd.to_datetime(df["date"])
```

```
df[df.artist=="Drake"].song.value_counts()
```

```
song
Hotline Bling     36
God's Plan        36
Controlla         26
Fake Love         25
Nice For What     25
                  ..
Trust Issues       1
Too Much           1
Own It             1
Tuscan Leather     1
Come Thru          1
Name: count, Length: 108, dtype: int64
```

```
df[df.artist=="Elton John"].song.value_counts()
```

```
song
Candle In The Wind 1997/Something About The Way You Look Tonight               42
Can You Feel The Love Tonight (From "The Lion King")                           26
I Guess That's Why They Call It The Blues                                      23
The One                                                                        22
Candle In The Wind                                                             21
Little Jeannie                                                                 21
The Last Song                                                                  20
Recover Your Soul                                                              20
Believe                                                                        20
Circle Of Life (From "The Lion King")                                          20
Blessed                                                                        20
Sad Songs (say So Much)                                                        19
I Don't Wanna Go On With You Like That                                         18
Nikita                                                                         18
Bennie And The Jets                                                            18
Mama Can't Buy You Love                                                        18
Blue Eyes                                                                      18
You Can Make History (Young Again)                                             17
Sacrifice                                                                      17
Empty Garden (Hey Hey Johnny)                                                  17
```

```
Crocodile Rock                                                                 17
Goodbye Yellow Brick Road                                                      17
I'm Still Standing                                                             16
Club At The End Of The Street                                                  16
Simple Life                                                                    16
Someday Out Of The Blue                                                        15
Don't Let The Sun Go Down On Me                                                15
Island Girl                                                                    15
Daniel                                                                         15
Rocket Man                                                                     15
Healing Hands                                                                  15
The Bitch Is Back                                                              14
Who Wears These Shoes?                                                         14
Wrap Her Up                                                                    14
Sorry Seems To Be The Hardest Word                                             14
Lucy In The Sky With Diamonds                                                  14
Your Song                                                                      14
Nobody Wins                                                                    13
A Word In Spanish                                                              13
Someone Saved My Life Tonight                                                  13
You Gotta Love Someone                                                         13
In Neon                                                                        13
Chloe                                                                          13
(Sartorial Eloquence) Don't Ya Wanna Play This Game No More?                   12
Kiss The Bride                                                                 12
Saturday Night's Alright For Fighting                                          12
Grow Some Funk Of Your Own/I Feel Like A Bullet (In The Gun Of Robert Ford)    11
Made In England                                                                10
Levon                                                                          10
Victim Of Love                                                                 10
Part-Time Love                                                                 10
Honky Cat                                                                      10
Friends                                                                         9
Heartache All Over The World                                                    8
Ego                                                                             8
Tiny Dancer                                                                     7
Bite Your Lip (Get up and dance!)                                               6
Border Song                                                                     5
Name: count, dtype: int64
```

```python
def chart_performance(artist, song):
    data = df[(df["artist"] == artist) & (df["song"] == song)]
    data = data.sort_values(by="date").reset_index(drop=True)
    data["date_rel"] = pd.to_timedelta(data["date"] - data["date"][0]).dt.days
    return data
```

```
test_cases = {
    "Taylor Swift": "You Belong With Me",
    "Drake": "God's Plan",
    "Elton John": "Candle In The Wind 1997/Something About The Way You Look Tonight",
    "The Weeknd": "Blinding Lights",
    "Elvis Presley": "Please Don't Stop Loving Me"
}
```

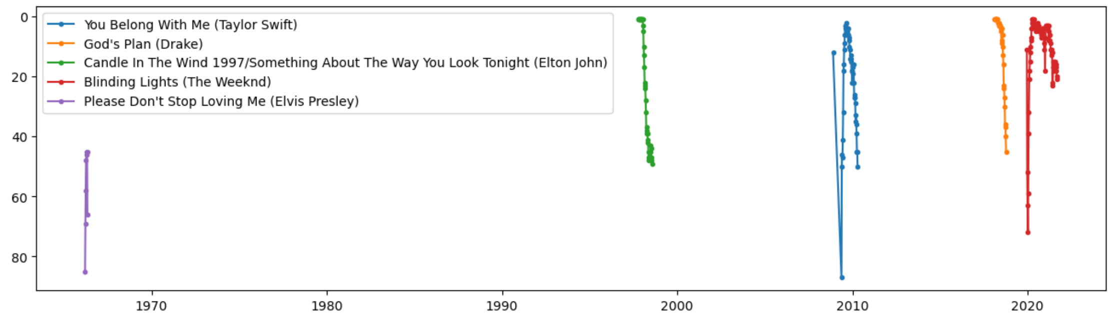
```
taylor = chart_performance("Taylor Swift", "You Belong With Me")
drake = chart_performance("Drake", "God's Plan")
elton = chart_performance("Elton John", "Candle In The Wind 1997/Something About The Way You L
weeknd = chart_performance("The Weeknd", "Blinding Lights")
elvis = chart_performance("Elvis Presley", "Please Don't Stop Loving Me")
```

```
_, ax = plt.subplots(figsize=(15,4))

for artist, song in test_cases.items():
    data = chart_performance(artist, song)
    x = data["date"].values
    y = data["rank"].values

    ax.plot(x, y, marker=".", label=f"{song} ({artist})")

plt.gca().invert_yaxis()
plt.legend()
plt.show()
```

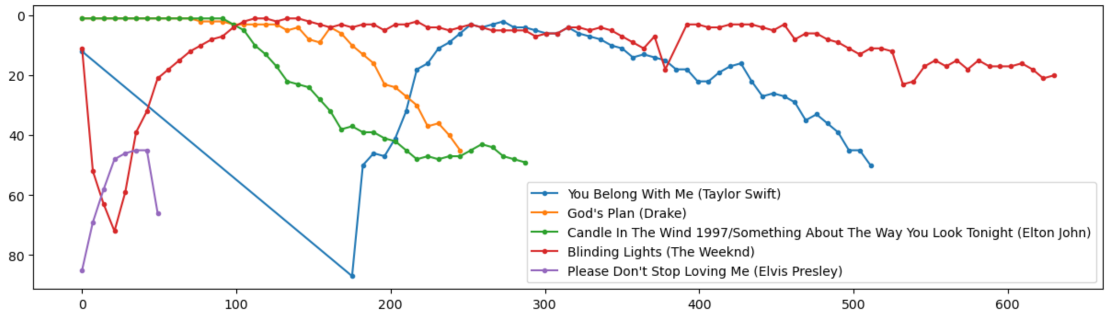


```
_, ax = plt.subplots(figsize=(15,4))

for artist, song in test_cases.items():
    data = chart_performance(artist, song)
    x = data["date_rel"].values
    y = data["rank"].values

    ax.plot(x, y, marker=".", label=f"{song} ({artist})")

plt.gca().invert_yaxis()
plt.legend()
plt.show()
```



```
# TODO: remove lines for missing weeks (gaps in curves)
# add two cases:
#  - short duration but high peak
#  - long duration but low peak
```

```
# Q: can we predict a song's survival using the features given in the data?
# --> at least introduce notion of training/test data and discuss the epistemologic
# sources for explanation
```

```
# Try other data: https://www.kaggle.com/datasets/thedevastator/billboard-hot-100-a
```

```
df_charts = pd.read_csv("Hot Stuff.csv", index_col=0)
df_charts["WeekID"] = pd.to_datetime(df_charts["WeekID"])
```

```
df_charts.head()
```

	url	WeekID	Week Position	Song
index				
0	http://www.billboard.com/charts/hot-100/1965-0...	1965-07-17	34	Don't Just Stand Th
1	http://www.billboard.com/charts/hot-100/1965-0...	1965-07-24	22	Don't Just Stand Th
2	http://www.billboard.com/charts/hot-100/1965-0...	1965-07-31	14	Don't Just Stand Th
3	http://www.billboard.com/charts/hot-100/1965-0...	1965-08-07	10	Don't Just Stand Th
4	http://www.billboard.com/charts/hot-100/1965-0...	1965-08-14	8	Don't Just Stand Th

```
df_audio = pd.read_csv("Hot 100 Audio Features.csv", index_col=0)
```

```
df_audio.head()
```

	SongID	Performer	Song
index			
0	-twistin'-White Silver SandsBill Black's Combo	Bill Black's Combo	-twistin'-White Silver Sands
1	¿Dònde Està Santa Claus? (Where Is Santa Claus...	Augie Rios	¿Dònde Està Santa Claus?
2	......And Roses And RosesAndy Williams	Andy Williams	......And Roses And Roses
3	...And Then There Were DrumsSandy Nelson	Sandy Nelson	...And Then There Were Dr
4	...Baby One More TimeBritney Spears	Britney Spears	...Baby One More Time

```
d = df_charts.merge(df_audio)
```

```
d.shape
```

```
(330208, 29)
```

```
d["WeekID"] = pd.to_datetime(d["WeekID"])
```

```
d.sample(10)
```

	url	WeekID	Week Position	Song
275610	http://www.billboard.com/charts/hot-100/1964-0...	1964-06-20	85	My Dreams
189145	http://www.billboard.com/charts/hot-100/2014-0...	2014-02-08	97	Radio
245340	http://www.billboard.com/charts/hot-100/1969-0...	1969-08-02	92	Let's Call It A Day
141579	http://www.billboard.com/charts/hot-100/2009-0...	2009-08-08	3	Knock You Down
117570	http://www.billboard.com/charts/hot-100/1960-0...	1960-02-20	93	Sleepy Lagoon
150750	http://www.billboard.com/charts/hot-100/1966-0...	1966-09-17	36	Flamingo

	url	WeekID	Week Position	Song
23582	http://www.billboard.com/charts/hot-100/1985-1...	1985-10-05	63	Soul K
21493	http://www.billboard.com/charts/hot-100/2003-0...	2003-04-12	21	Rock
71236	http://www.billboard.com/charts/hot-100/2008-1...	2008-12-13	52	My Li
208184	http://www.billboard.com/charts/hot-100/1984-1...	1984-12-15	60	Missin

```
## BOOTSTRAP!

# d = d.sample(500_000, replace=True)
```

```
d.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 330208 entries, 0 to 330207
Data columns (total 29 columns):
 #   Column                      Non-Null Count   Dtype
---  ------                      --------------   -----
 0   url                         330208 non-null  object
 1   WeekID                      330208 non-null  datetime64[ns]
 2   Week Position               330208 non-null  int64
 3   Song                        330208 non-null  object
 4   Performer                   330208 non-null  object
 5   SongID                      330208 non-null  object
 6   Instance                    330208 non-null  int64
 7   Previous Week Position      298048 non-null  float64
 8   Peak Position               330208 non-null  int64
 9   Weeks on Chart              330208 non-null  int64
 10  spotify_genre               315700 non-null  object
 11  spotify_track_id            287066 non-null  object
 12  spotify_track_preview_url   169915 non-null  object
 13  spotify_track_duration_ms   287066 non-null  float64
 14  spotify_track_explicit      287066 non-null  object
 15  spotify_track_album         287004 non-null  object
 16  danceability                286508 non-null  float64
 17  energy                      286508 non-null  float64
 18  key                         286508 non-null  float64
 19  loudness                    286508 non-null  float64
 20  mode                        286508 non-null  float64
 21  speechiness                 286508 non-null  float64
 22  acousticness                286508 non-null  float64
 23  instrumentalness            286508 non-null  float64
 24  liveness                    286508 non-null  float64
 25   valence                       286508 non-null   float64
 26   tempo                         286508 non-null   float64
 27   time_signature                286508 non-null   float64
 28   spotify_track_popularity      287066 non-null   float64
dtypes: datetime64[ns](1), float64(15), int64(4), object(9)
memory usage: 73.1+ MB
```

```
from IPython.display import Audio, HTML
```

```
Audio(url=d.loc[1000,"spotify_track_preview_url"])
```

```
<IPython.lib.display.Audio object>
```

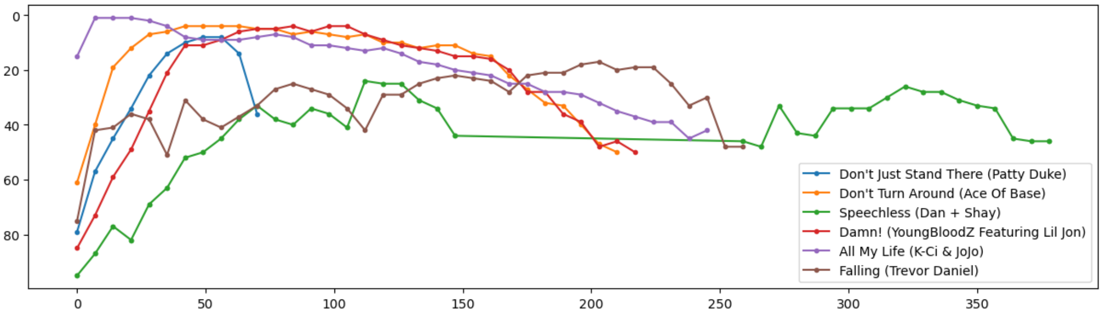
```
def curves(performer, song):
    data = d[(d.Performer == performer) & (d.Song == song)].sort_values(by="WeekID").reset_inc
    data["date_rel"] = pd.to_timedelta(data["WeekID"] - data["WeekID"][0]).dt.days
    x = data["date_rel"].values # or date_rel or WeekID
    y = data["Week Position"].values
    return x,y
```

```
test_cases2 = {
    "Patty Duke": "Don't Just Stand There",
    "Ace Of Base": "Don't Turn Around",
    "Dan + Shay": "Speechless",
    "YoungBloodZ Featuring Lil Jon": "Damn!",
    "K-Ci & JoJo": "All My Life",
    "Trevor Daniel": "Falling"
}
```

```
_, ax = plt.subplots(figsize=(15,4))

for performer, song in test_cases2.items():
    x,y = curves(performer, song)
    ax.plot(x, y, marker=".", label=f"{song} ({performer})")

plt.gca().invert_yaxis()
plt.legend()
plt.show()
```



Modeling the life of a song in the Top 100:

We assume that once a song has left the Top 100, it is impossible to re-enter (even though that does happen, of course)

1. Each song has a starting rank r_0.
2. For each following week, there is a bernoulli dropout probability θ that determines whether a song remains in the charts.
3.

```
# Observation: Genres tend to leave the Top 100 higher than they entered them
```

```
entrances = []
peaks = []
exits = []

for _, group in d.groupby("SongID"):
    weeks = group.sort_values(by="WeekID")["Week Position"].values
    entrances.append(weeks[0])
    peaks.append(weeks.min())
    exits.append(weeks[-1])
```

```
import numpy as np
```

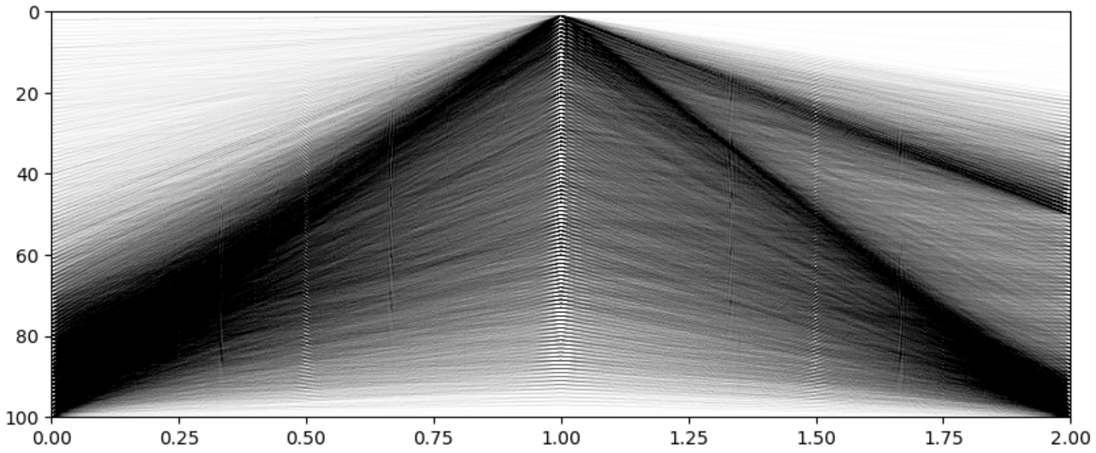
```
# from matplotlib.collections import LineCollection
```

```
_, ax = plt.subplots(figsize=(10,4))

K = len(entrances) + 1

for a, b, c in zip(entrances[:K], peaks[:K], exits[:K]):
    if a != b != c: # remove constants
        ax.plot([0, 1, 2], [a, b, c], c="k", lw=.5, alpha=.01)
```

```
plt.xlim(0,2)
plt.ylim(0,100)
plt.gca().invert_yaxis() # smaller is better
plt.savefig("img/rise-decline.png", dpi=600)
plt.show()
```



OBSERVATION: At least 3 types:

- constants
- low in, peak, low out
- low in, peak, mid out

Try to disentangle what causes the difference

Part III.

MUSIC AS DATA

8. Audio

ℹ Goal

Understand what an audio signal is and how it is represented digitally .

- Waveform to spectrogram
- Harmonics
- Timbre
- Audible range and volume
- reading melodies from a spectrogram
- digital audio: sampling

9. MIDI

ℹ Goal

Be able to name use cases for MIDI. Translate MIDI numbers to pitches.

10. MEI - header

ℹ Goal

Understand basic XML encoding and the skeleton structure of MEI.

- mei friend

11. MEI - the body

ℹ Goal

Understand the relation between CWMN and the MEI music element.

- MuseScore export
- mei friend

Part IV.

WORKING WITH MUSIC DATA

12. Digital music analysis: harmony

ℹ Goal

Understand what labeling is and why labels can be useful.

- further MuseScore practice
- segmentation and labeling
- Counting chords, finding cadences

13. Digital music analysis: melody

ℹ Goal

Understand how melodic pattern matching works in principle.

- Pattern finding in melodies (Non-Western)

Part V.

CRITICAL DIGITAL MUSICOLOGY

14. Copyright

ℹ Goal

Know a few famous copyright infringement cases and why data analysis is important here.

- Plagiarism cases and copyright

15. Representation and representativeness

ℹ Goal

Understand the difference between representativeness and representation. Obtain a critical understanding of biases relevant for data selection.

- Representation and the canon
- Representing means modeling means abstraction (what is “music” in “music encoding”?)
- biases: how to recognize them, how to deal with them, and when biases are a good thing.
- FAIR and CARE

16. Discussion

Goal

References

Knuth, Donald E. 1984. “Literate Programming.” *Comput. J.* 27 (2): 97–111. https://doi.org/10.1093/comjnl/27.2.97.