Welcome to the humdrum$_R$ Workshop 2023!

Hosts:

Nat Condit-Schultz, natcs@gatech.edu

Claire Arthur, claire.arthur@gatech.edu

Day 0, Morning Session
R and Rstudio

Why R?

- R is awesome.
- R is language for data analysts, by data analysts.
  - (But free, unlike Matlab, SAS)
- Basic language, and many packages, designed for easy/quick data manipulation and statistics.
  - (Including many *primary* source statistical algorithms.)
- A wonderful “echosystem”: Rstudio, Rmarkdown/Shiny, the tidyverse, etc.

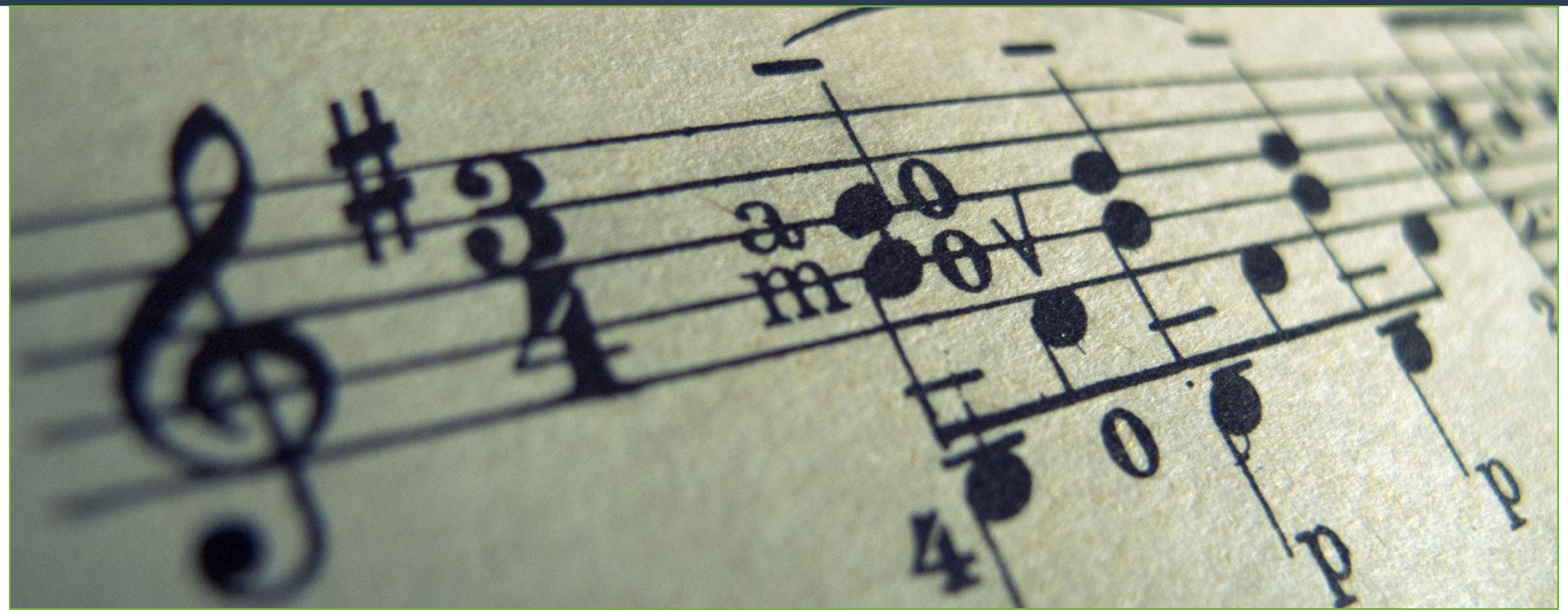
Why not Python?

- The Python community has developed wonderful resources for data science.
- However, as a mega-popular *general purpose* language, Python also has lots and lots and lots of…less good packages.
  - It's a much more mess echo system. Package/version management can be hell.
- R has core features that need to be imported in Python.
  - Makes data analysis "on the fly" quicker and more concise.
  - R also has features that *cannot* be replicated in Python (multiple-dispatch, metaprogramming, better lambdas, etc.).

Intro to R

- Open the file `Day0.1_IntroToR.Rmd` in Rstudio.

Day 0, Afternoon Session
Humdrum Representation & Syntax



Why Humdrum?

- Humdrum (David Huron, 1995) is a complete software system comprising of both a musical representation scheme (and file format) *and* a toolkit.
- Today we will only discuss Humdrum as a representation:
  - Human readable (Plain ASCII text) & Computer readable
  - Tailor representations to user interests (established and user-defined representations)
  - Availability of large volume of high-quality encoded materials.

Humdrum

- While Humdrum format can be used to store almost *any* form of data, we will focus later on extra-musical representations.

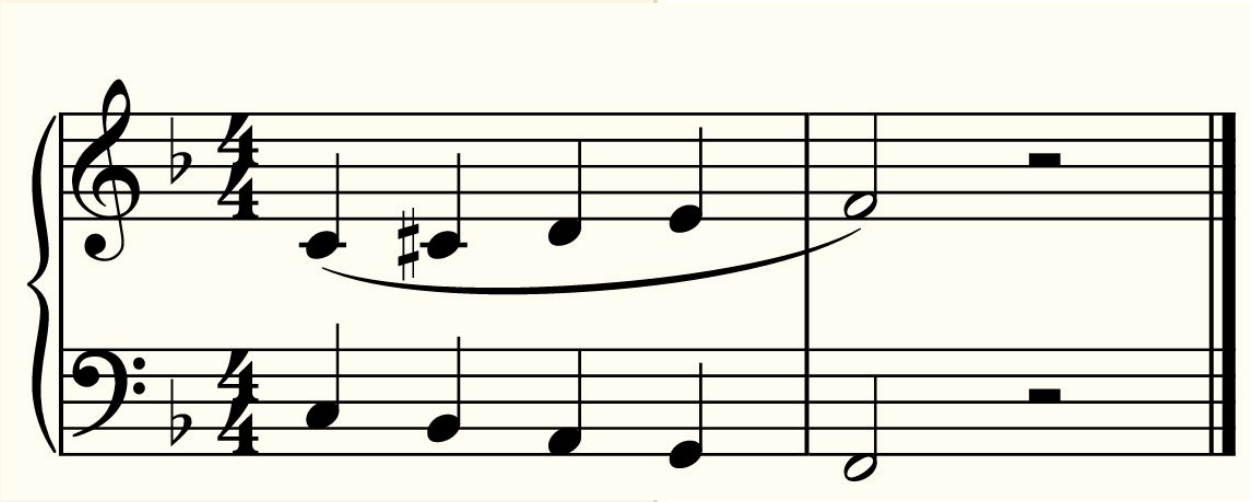
- When using Humdrum to encode traditional Western notation, you will typically use its native *kern* format

Humdrum Data Structure

- Basic structure uses columns (i.e., “spines”) and cells (like a spreadsheet) to organize data into parts with typically one spine per part
- Set of symbols distinguish *Humdrum data* from different varieties of *metadata*

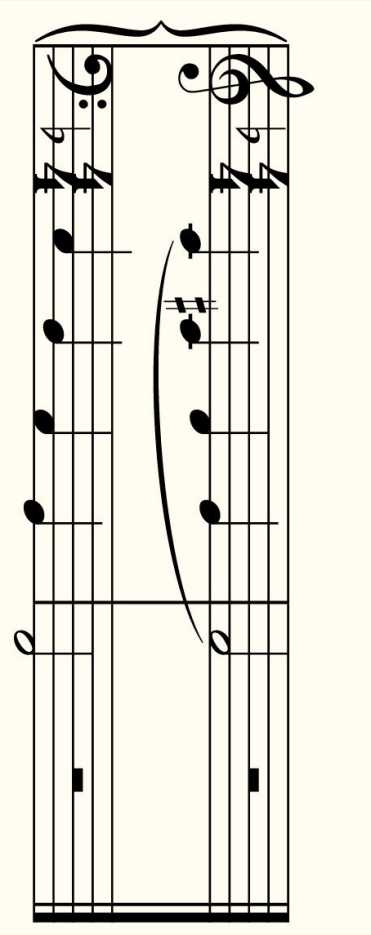
VerovioHumdrumViewer

1	**kern	**kern
2	*M4/4	*M4/4
3	*clefF4	*clefG2
4	*k[b-]	*k[b-]
5	=1-	=1-
6	4C	(4c
7	4BB-	4c#
8	4AA	4d
9	4GG	4e
10	=2	=2
11	2FF	2f)
12	2r	2r
13	==	==
14	*-	*-



VerovioHumdrumViewer

1	**kern	**kern
2	*M4/4	*M4/4
3	*clefF4	*clefG2
4	*k[b-]	*k[b-]
5	=1-	=1-
6	4C	(4c
7	4BB-	4c#
8	4AA	4d
9	4GG	4e
10	=2	=2
11	2FF	2f)
12	2r	2r
13	==	==
14	*-	*-

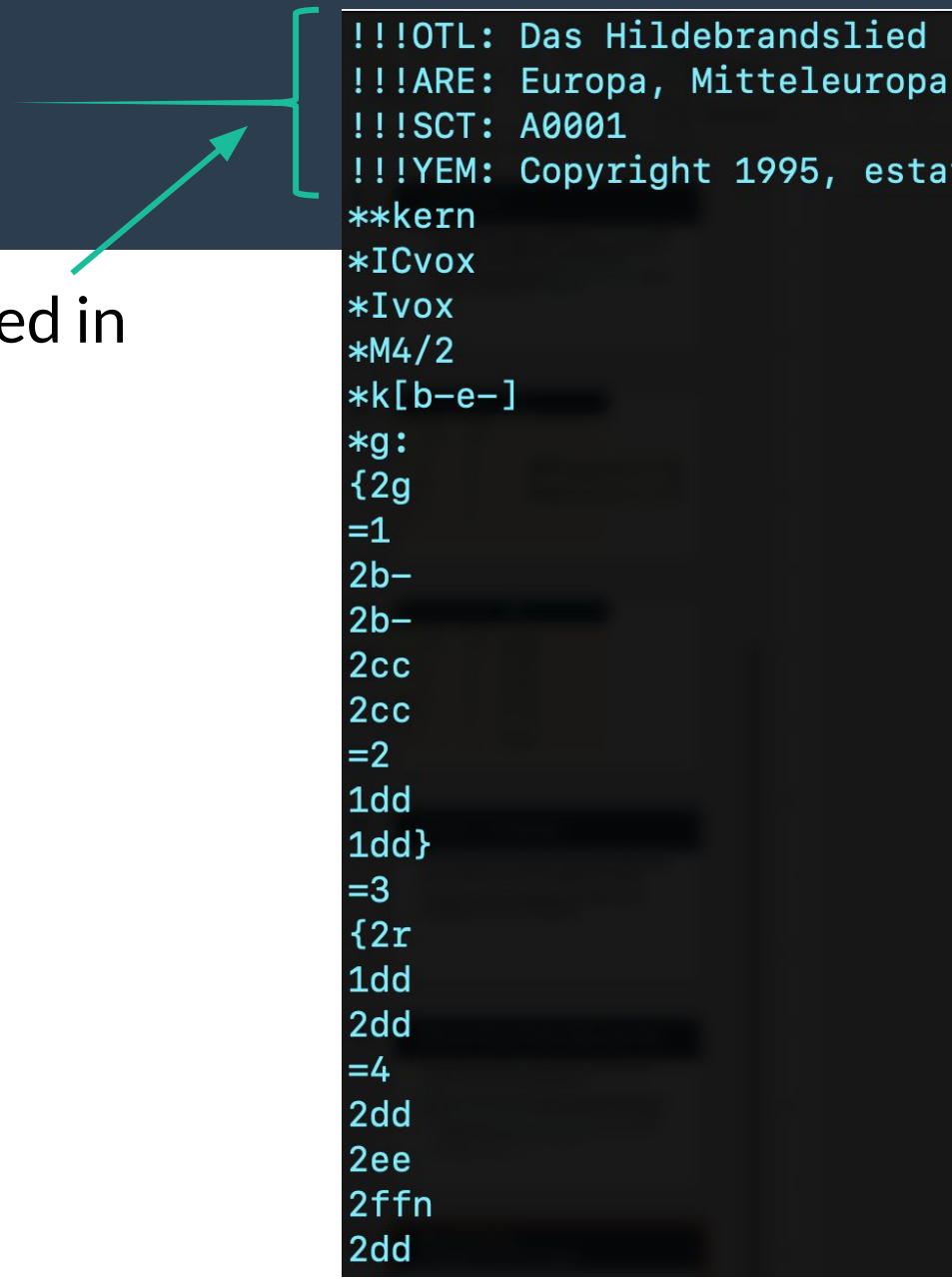


Minimum Criteria: What makes a humdrum file?

- A humdrum-format file is a simple plain text file with data written in columnar format (called spines).
- In order for a humdrum-format file to be properly encoded so that it is *computer readable* it needs a minimum of two things:
  1. Each file needs an exclusive representation (for a kern file that would be: **kern) at the top of each spine
  2. A terminator token (*-)

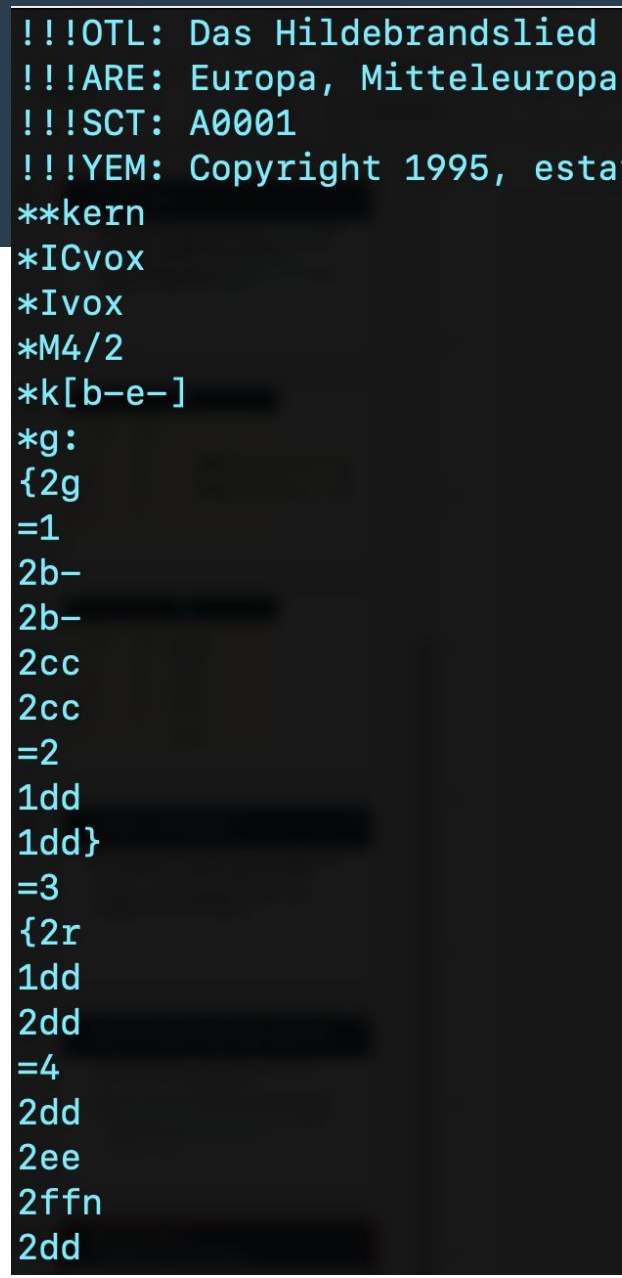
Humdrum Information: types

- Metadata (Work, title, composer, year, etc.); placed in *comment records*



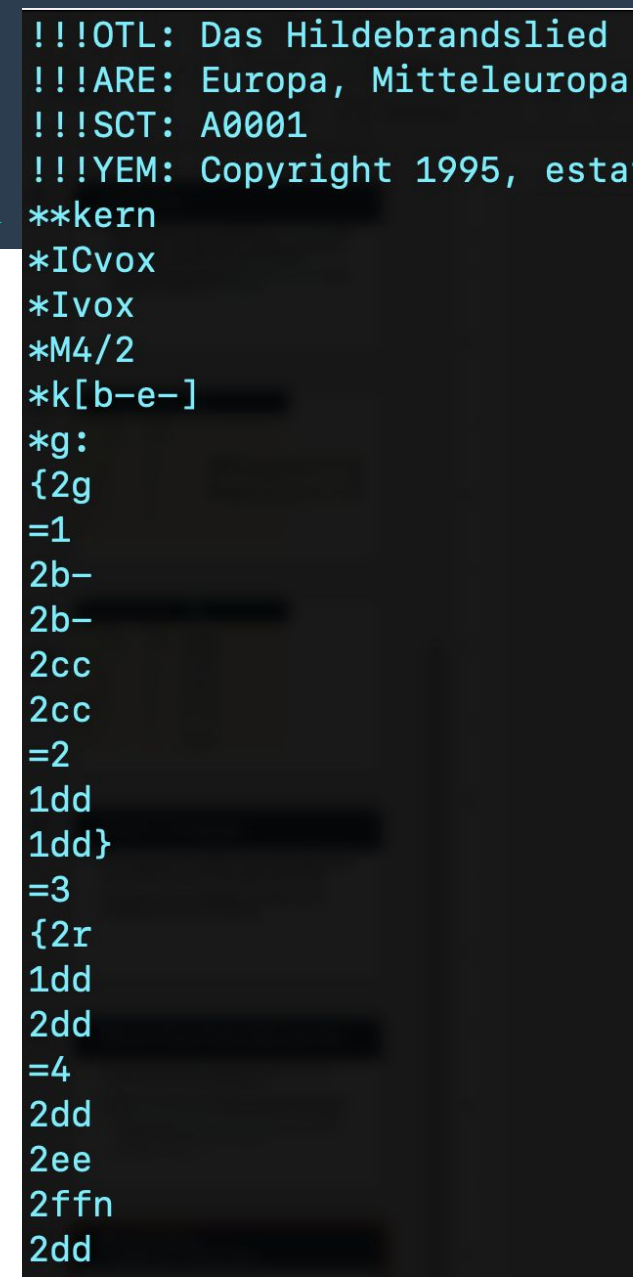
Humdrum Information: types

- Metadata (Work, title, composer, year, etc.); placed in *comment records*
- The *exclusive interpretation* (**) tells us the representation scheme for the data in that spine (e.g., kern, mint, fret, harm, etc.)



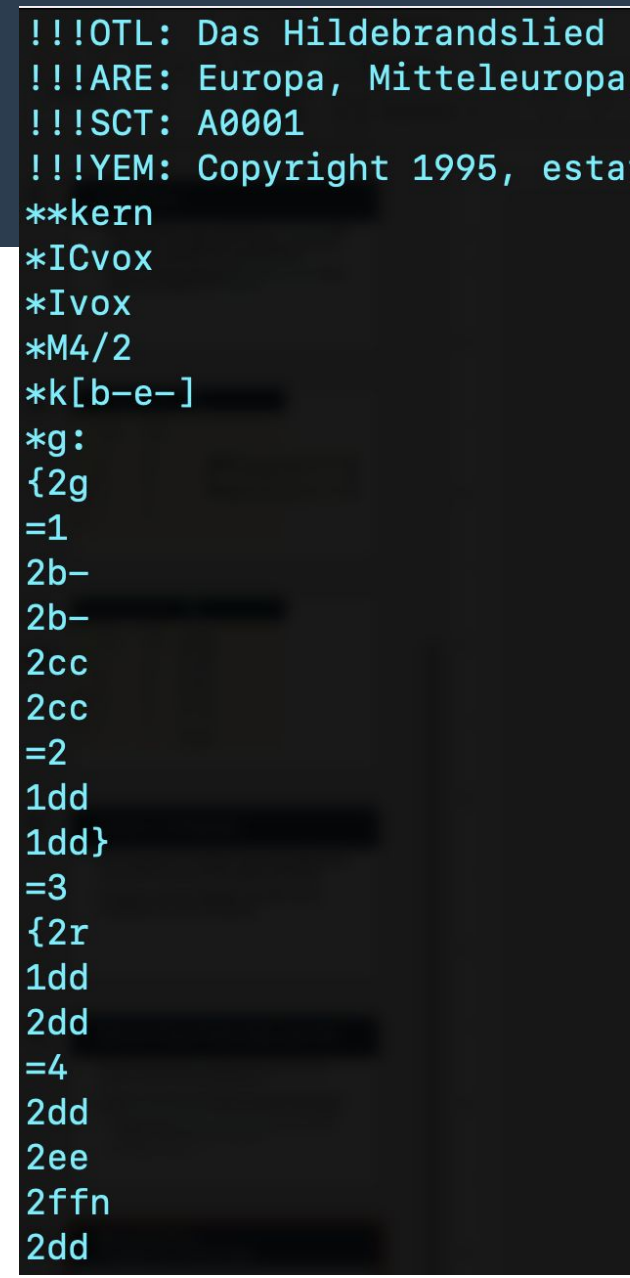
Humdrum Information: types

- Metadata (Work, title, composer, year, etc.); placed in *comment records*
- The *exclusive interpretation* (**) tells us the representation scheme for the data in that spine (e.g., kern, mint, fret, bhatke, etc.)
- Type of data Contextual data (key signature, time signature, clef, instrument, tempo, etc.); placed in *tandem interpretation records*



Humdrum Information: types

- Metadata (Work, title, composer, year, etc.); placed in *comment records*
- The *exclusive interpretation* (**) tells us the representation scheme for the data in that spine (e.g., kern, mint, fret, bhatke, etc.)
- Type of data Contextual data (key signature, time signature, clef, instrument, tempo, etc.); placed in *tandem interpretation records*
- Musical data (notes, rests, accidentals, barlines, ties, slurs, ornamentation, phrase groupings, etc.); placed in *data records*



Kern Data: Cheatsheet

- Musical data (for a cheatsheet, click here)
  - Rhythm
    - Follows denominator of traditional note values using American note names (e.g., quarter note (crotchet) = 4)
  - Pitch
    - Letter & octave, accidentals (always comes *after rhythm*)
    - Octaves switch at every C (ASA style) c4 = “c”
    - Sharp #, Flat -
  - Grouping (Phrasing)
    - Ties [ ], Phrases { }
  - Articulation
    - Slurs ( ), Stacato ‘ , Accents ^v
  - Bars & barlines
    - Single =, Double ==, Repeat back =:| , Repeat from =|:

```
!!!OTL: Das Hildebrandslied
!!!ARE: Europa, Mitteleuropa
!!!SCT: A0001
!!!YEM: Copyright 1995, esta
**kern
*ICvox
*Ivox
*M4/2
*k[b-e-]
*g:
{2g
=1
2b-
2b-
2cc
2cc
=2
1dd
1dd}
=3
{2r
1dd
2dd
=4
2dd
2ee
2ffn
2dd
```

Function vs. Orthography

- Kern (and other representations) designed to facilitate analytical applications rather than visual representations (primarily)
- However, many orthographic details *can* be included but are not necessary.

Part 1: Kern data

- We will be using the online tool, Verovio Humdrum Viewer (VHV) to assist with our encoding and learning of kern
  - (VHV full music encoding tutorial here)
  - A note about VHV & Orthography…

Exercise 1: Kern Encoding Practice

- Open your “Day0_KernEncodingVHV.pdf” file
- Using VHV, see how far you can get with the encoding practice (we’ll do the first one together)

AFTERNOON BREAK?

PLEASE BE BACK BY 3:30! :)

Multiple Spines, multiple stops

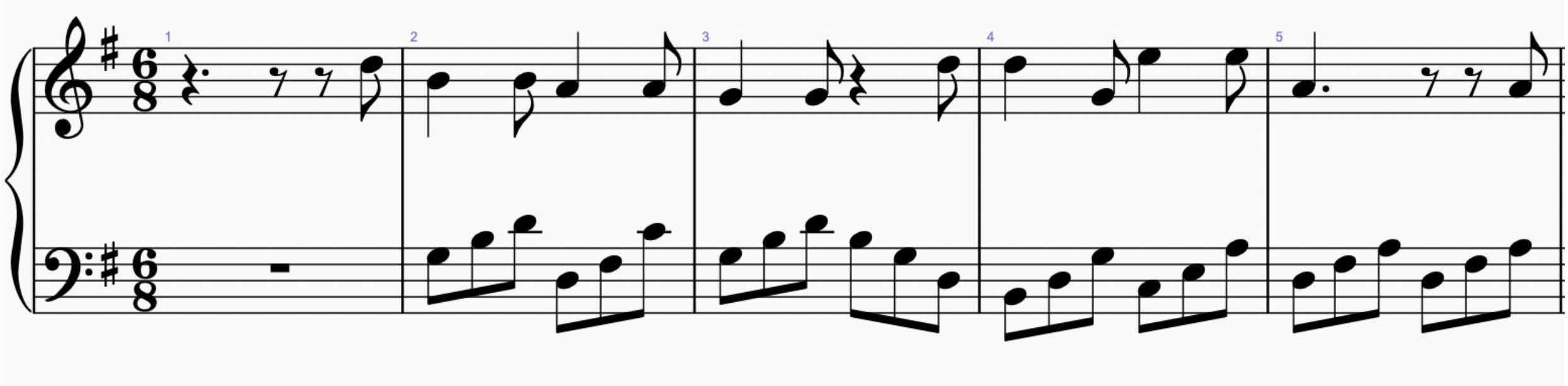
- Humdrum data is encoded row-wise, meaning that items on the same row are occurring (musically) simultaneously.
- For notes & rests, this can take different forms:
  1. Multiple voices, (multi-stops)
  2. Multiple parts, different staves

Humdrum data examples

- Example files in VHV

Exercise 2: Kern Encoding Practice

- Using VHV, try to encode the following:



Multiple Spines, multiple stops

- This situation is not limited to concurrent kern spines, it can be anything happening simultaneously that we need represented, e.g.,

Multiple Spines, multiple stops

- This situation is not limited to concurrent kern spines, it can be anything happening simultaneously that we need represented, e.g.,
  - Notes and associated lyrics

Multiple Spines, multiple stops

- This situation is not limited to concurrent kern spines, it can be anything happening simultaneously that we need represented, e.g.,
  - Notes and associated lyrics
  - Notes and chord labels

Multiple Spines, multiple stops

- This situation is not limited to concurrent kern spines, it can be anything happening simultaneously that we need represented, e.g.,
  - Notes and associated lyrics
  - Notes and chord labels
  - Notes and timestamps…

Non-kern representations

- Pre-existing exclusive interpretations & associated syntax defined in humdrum, e.g.,

**harm	Roman numerals
**silbe	Lyrics
**recip	Rhythm (isolated)
**mint	Melodic intervals
**hint	Harmonic intervals
**semits	Semitone distance from middle C
**deg	Scale degree numbers
**time	Timestamps (expressed in s or ms)

Non-kern representations

- Can include custom defined interpretations of any kind, e.g.,

**function	Function labels (T, P, D) -- as used in the TAVERN corpus
**rhyme	Rhyme scheme information -- as used in the MCFLOW corpus
**cadence	Cadence labels (e.g., CC, IAC) – as used in the SWTC

Humdrum data examples

- Example files in VHV

Getting humdrum/kern data

- Existing repos
- Conversion tools
- humdrumR native capabilities from spreadsheet data