Overview/Review

FluCoMa: Fluid Corpus Manipulation

Slice Audio
- onset slice
- transient slice
- novelty slice
- amplitude slice
- amplitude gate

Decompose Audio
- extract transients
- harmonic/percussive separation
- model as sine waves
- non-negative matrix factorisation

Analyse Audio
- pitch
- loudness
- mel-bands
- mel-frequency cepstral coefficients
- spectral centroid
- spectral flatness
- chromagram

Transform Audio
- audio transport
- non-negative matrix factorisation filters & morphing

Analyse Data
- datasets
- labelsets
- statistical analyses
- normalization
- standardization
- robust scaler
- principal component analysis
- MDS
- KDTree
- K Nearest Neigh[illegible]s
- neural networks
- SQL-type query
- KMeans
- UMAP
- grid

… and more

Supervised Learning

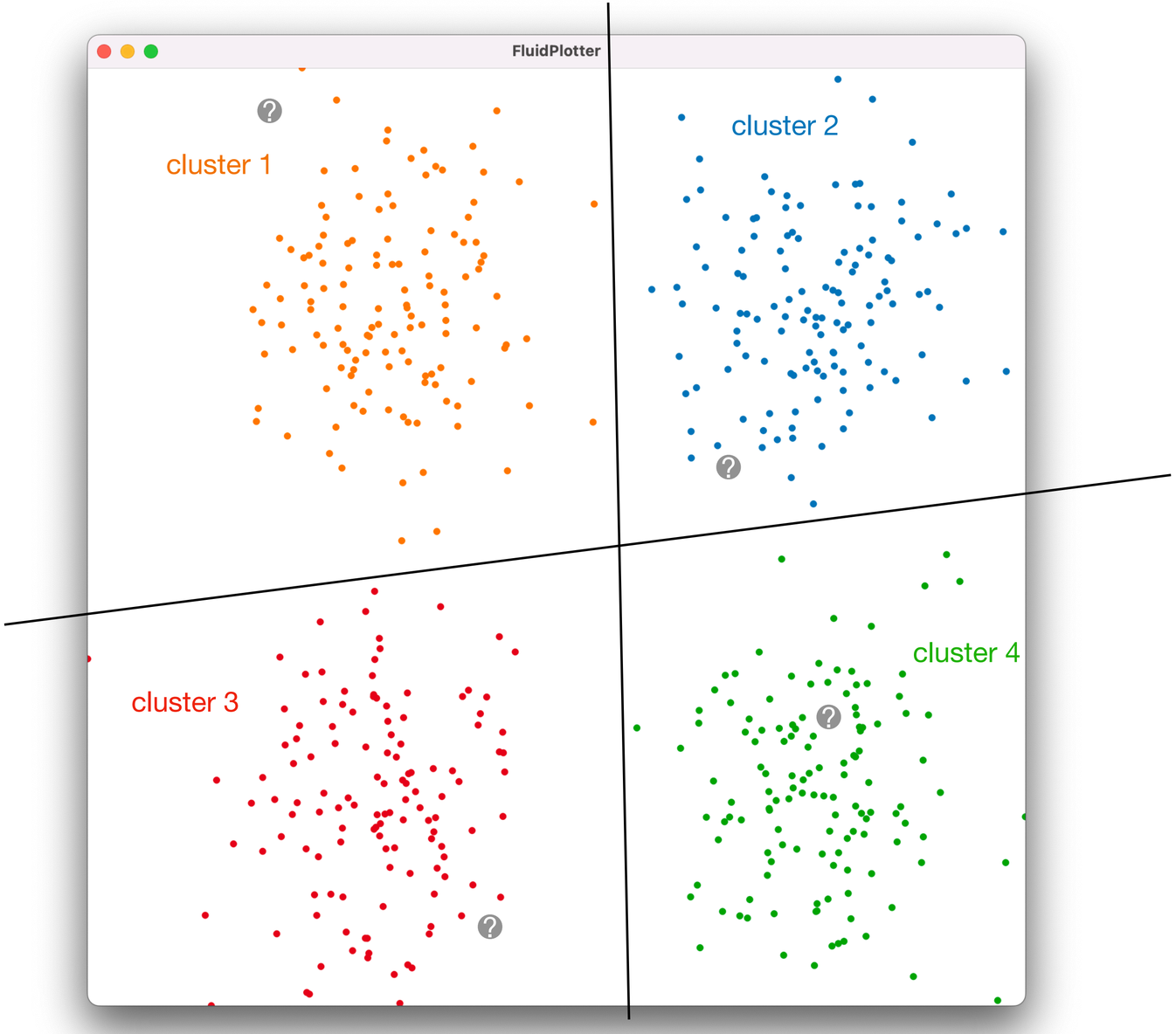
Learning patterns, associations, or relationships from data that is pre-labeled

Classification
Regression

Neural Networks
KNN

Unsupervised Learning

Learning/finding patterns and relationships in data that is not labeled



Dimensionality Reduction

Clustering

Feature Learning

scikit-learn

Machine Learning in Python

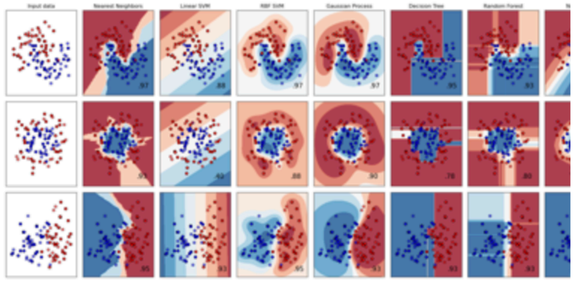
Getting Started | Release Highlights for 1.2 | GitHub

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

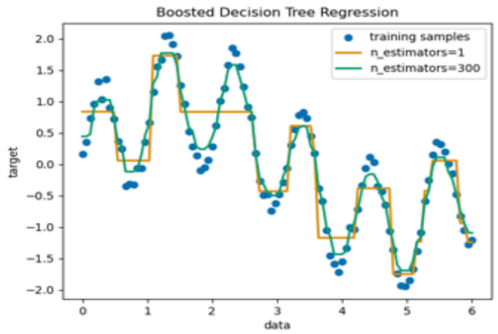


Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...

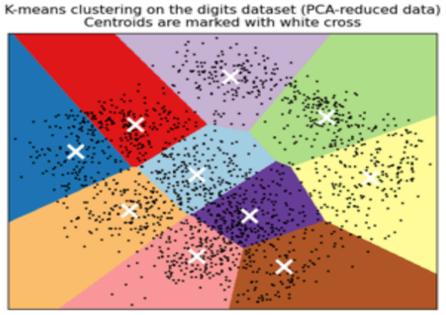


Examples

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...



Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Model selection

Comparing, validating and choosing parameters and models.

Preprocessing

Feature extraction and normalization.

Applications: Transforming input data such as text

Supervised Learning

Classification

- fluid.mlpclassifier~
- fluid.knnclassifier~

Regression

- fluid.mlpregressor~
- fluid.knnregressor~

Preprocessing

- fluid.normalize~
- fluid.standardize~
- fluid.robustscale~

Unsupervised Learning

Dimensionality Reduction

- fluid.pca~
- fluid.umap~
- fluid.mds~
- autoencoder (fluid.mlpregressor~)

Clustering

- fluid.kmeans~
- fluid.skmeans~

Linear Assignment

- fluid.grid~

Also…
fluid.bufnmf~

Fluid Corpus Manipulation

More help / discussion at our forum

Check out the learn platform

Tutorials | Objects | Tasks | Inspiration | Examples

We've worked toward making a set of concise tutorials that cover a very wide range of objects and workflows found in the FluCoMa toolkit. We think that if you at least watch these three tutorials, you'll be familiar with most of the objects and be able to start building your own patches and systems out of the toolkit. Select a tutorial to explore below.

Select a tutorial...

- Building a 2D Corpus Explorer
- Classifying Sounds with a Neural Network
- Controlling a Synth with a Neural Network

Be aware, some of the patches you can open from within the package will differ slightly from the tutorial video's final result.

YouTube Video | Open Patch

Imagine you have a large corpus of sounds that you've collected from a studio session, some outside sound walks or experimenting with a synthesiser on a rainy afternoon. This tutorial teaches you how to build a '2D Corpus Explorer', a patch that will enable you to interrogate and listen to those sounds in a structured manner. The end result is similar to CataRT and AudioStellar in that small segments of your corpus sounds are analysed and mapped to a two-dimensional space that can be explored using your mouse. It will cover topics such as segmentation, analysis, data processing and visualisation.

Fluid Corpus Manipulation

More help / discussion at our forum

Check out the learn platform

Tutorials | Objects | Tasks | Inspiration | Examples

Select a category...

Slice Audio | Transform Audio

Analyse Audio | Analyse Data

Decompose Audio | Helpers and Utilites

If you have some corpus data, you'll probably want to analyse, manipulate and transform it. A variety of objects fall under this category and help you to make different manoeuvres around your data.

Containers

- fluid.dataset~
- fluid.labelset~

Statistical Analysis

- fluid.bufstats~
- fluid.stats

Scaling and Preprocessing

- fluid.normalize~
- fluid.standardize~
- fluid.robustscale~

Searching and Querying

- fluid.kdtree~
- fluid.datasetquery~

Unsupervised Machine Learning

- fluid.pca~
- fluid.kmeans~
- fluid.skmeans~
- fluid.mds~
- fluid.umap~
- fluid.grid~

Supervised Machine Learning

- fluid.knnregressor~
- fluid.knnclassifier~
- fluid.mlpregressor~
- fluid.mlpclassifier~

Fluid Corpus Manipulation

More help / discussion at our forum

Check out the learn platform

Tutorials | Objects | Tasks | Inspiration | Examples

A major part of the FluCoMa project involved commissioning composers to create works using the toolkit. These works are recorded and have also been investigated in detail by Jacob Hart (one of the team members) in "explore" articles. These long-form articles draw out how objects were used, and he has even created patches for you to plunder for your own creative endeavours. This is a great place to start if you're searching for musical inspiration with the FluCoMa Toolkit.

Select a composer(s)...

- Lauren Sarah Hayes
- Leafcutter John
- Olivier Pasquet
- Rodrigo Constanzo
- Alex Harker
- Sam Pluta
- Richard Devine
- Hans Tutschku
- Alice Eldridge & Chris Kiefer

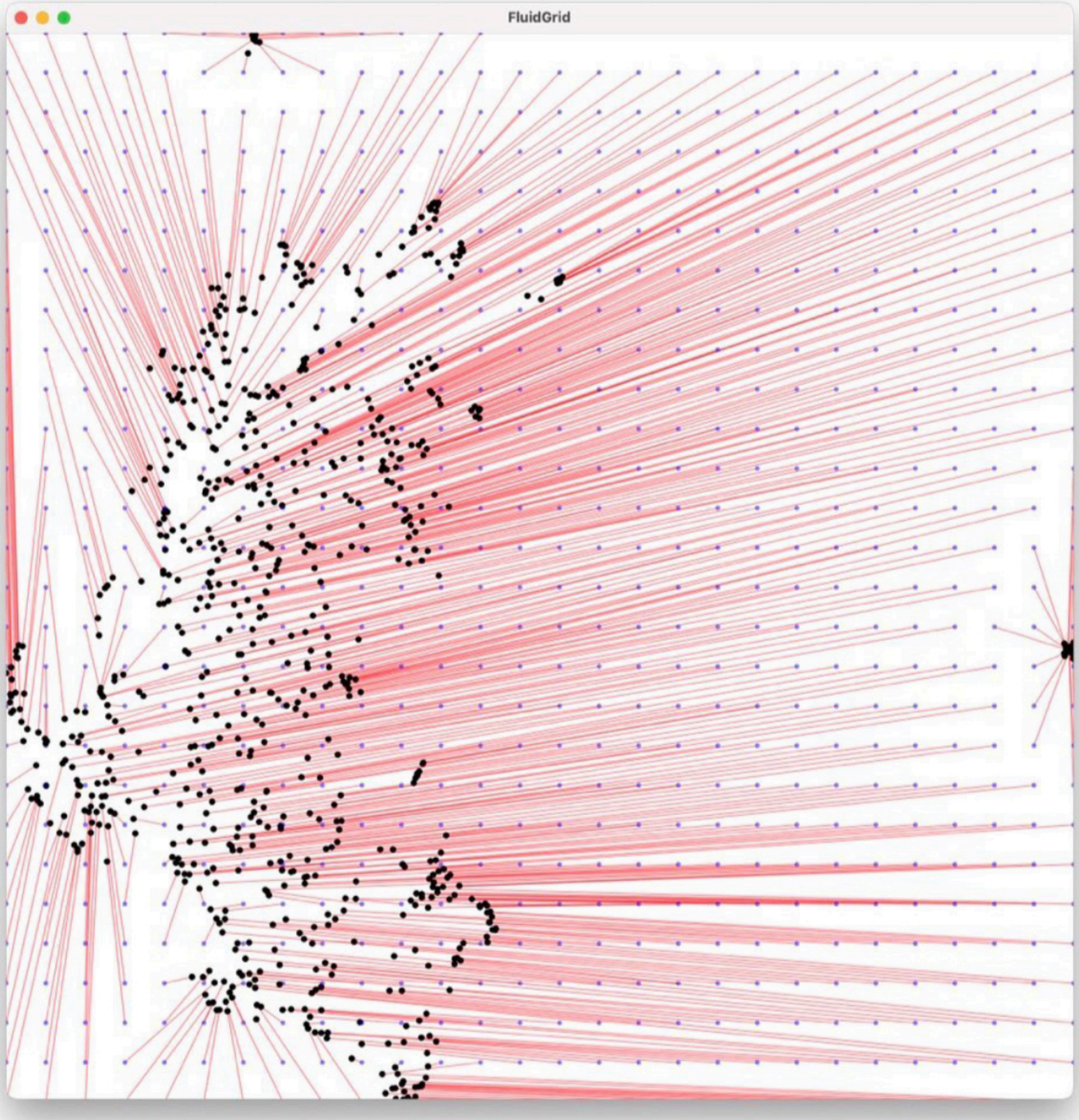
YouTube Video | Learn Article

Lauren Sarah Hayes is a Scottish improviser and sound artist. Her music is a mix of experimental pop/live electronics/techno/noise/free improvisation and has been described as 'voracious' and 'exhilarating'. She is a sculptress of sound, manipulating, remixing, and bending voice, drum machines, analogue synths and self-built software live and physically. She is excited by what can happen in the vulnerable relationships between sound, space, and audience. Her shows are highly physical, making the performance of live electronic music more engaging for audiences. Over the last decade she has developed and honed a deliberately challenging and unpredictable performance system that explores the relationships between bodies, sound, environments, and technology.

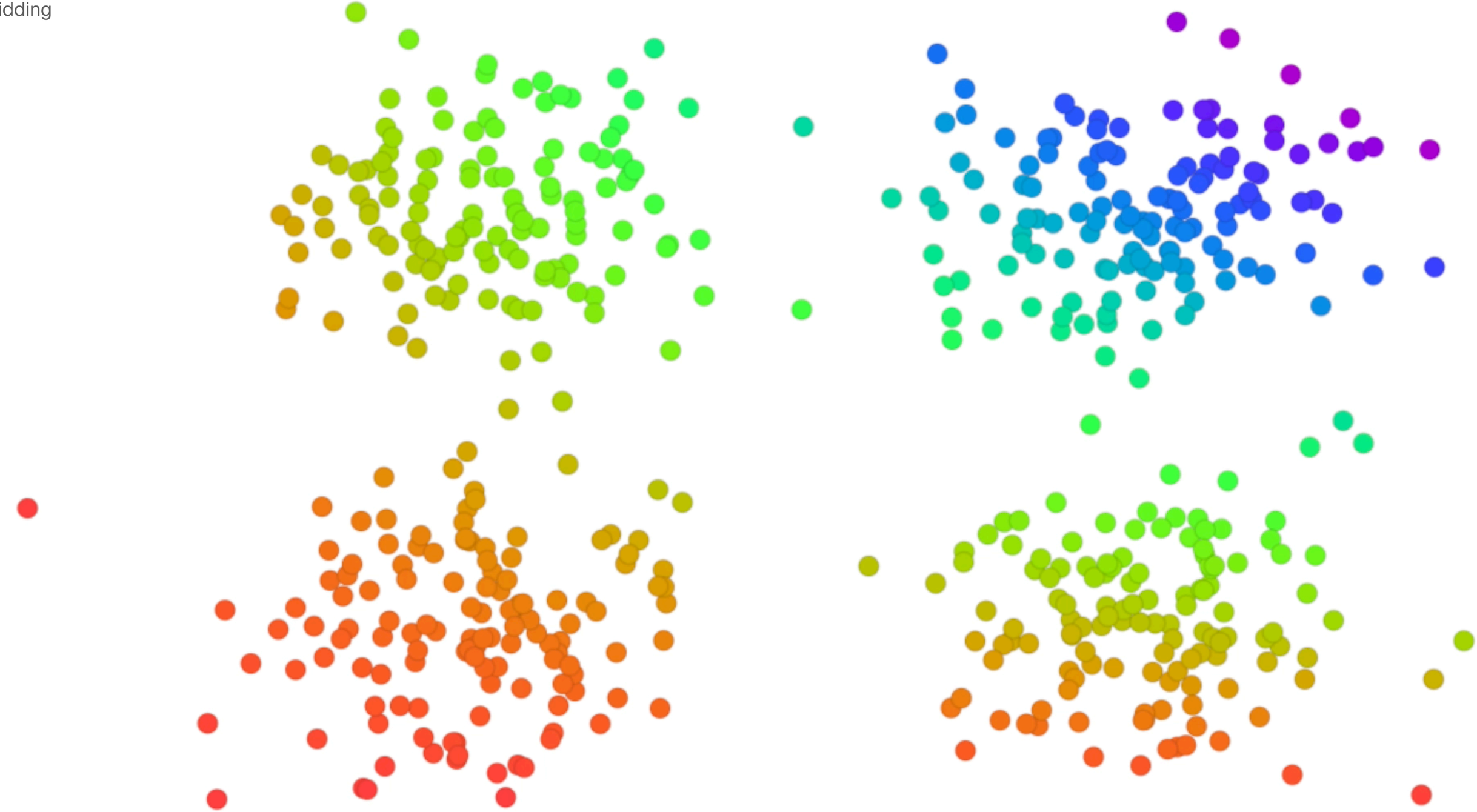
Grid (Jonker-Volgenant algorithm)

assign each point in 2D space to a point on a 2D grid

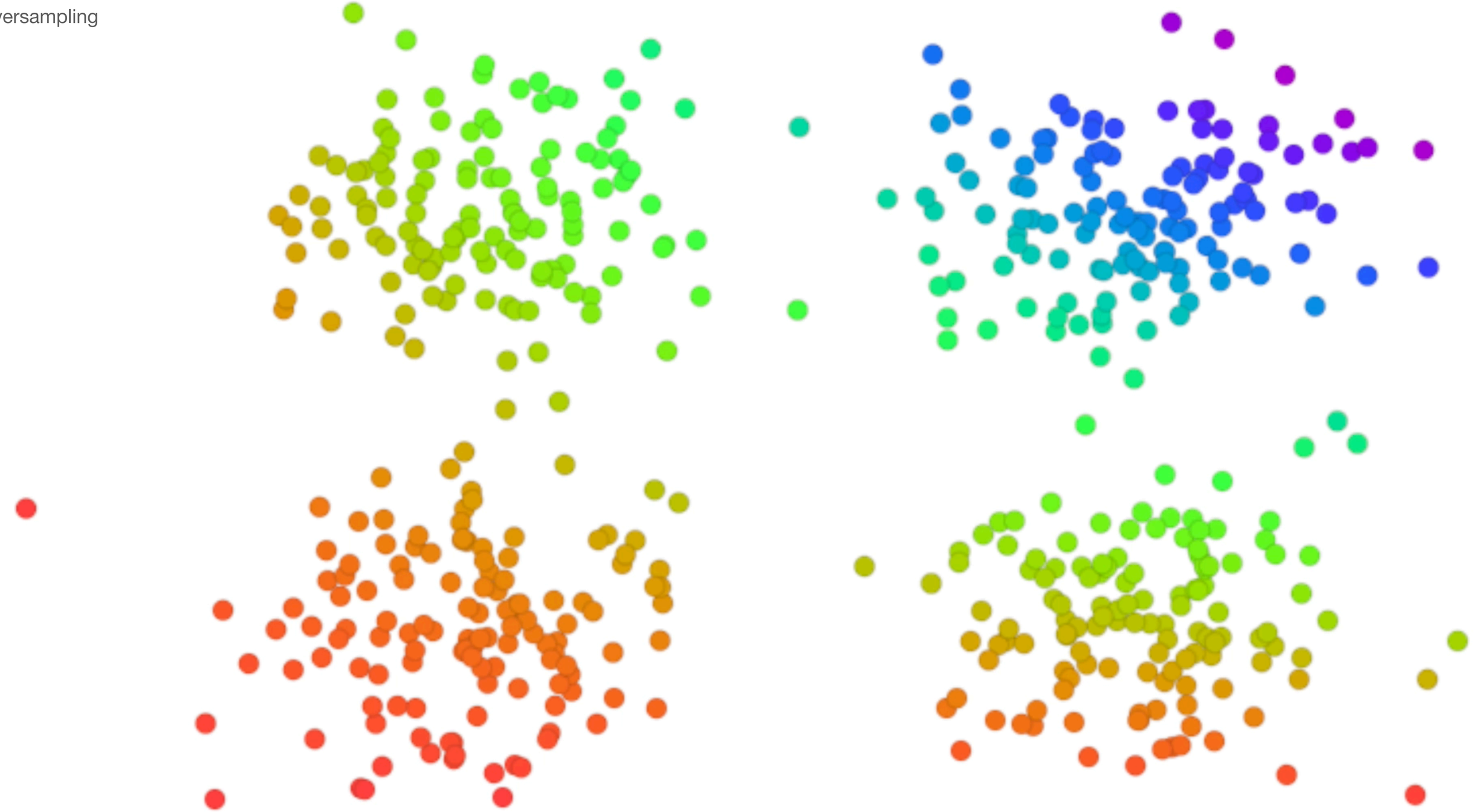
point-0: [0.75, 0.65] --> [1.0, 1.0]
point-1: [0.50, 0.80] --> [0.5, 1.0]
point-2: [0.35, 0.68] --> [0.0, 0.5]
point-3: [0.64, 0.41] --> [1.0, 0.5]
point-4: [0.36, 0.36] --> [0.0, 0.0]
point-5: [0.37, 0.68] --> [0.5, 0.5]
point-6: [0.34, 0.73] --> [0.0, 1.0]
point-7: [0.45, 0.09] --> [0.5, 0.0]
point-8: [0.55, 0.49] --> [1.0, 0.0]

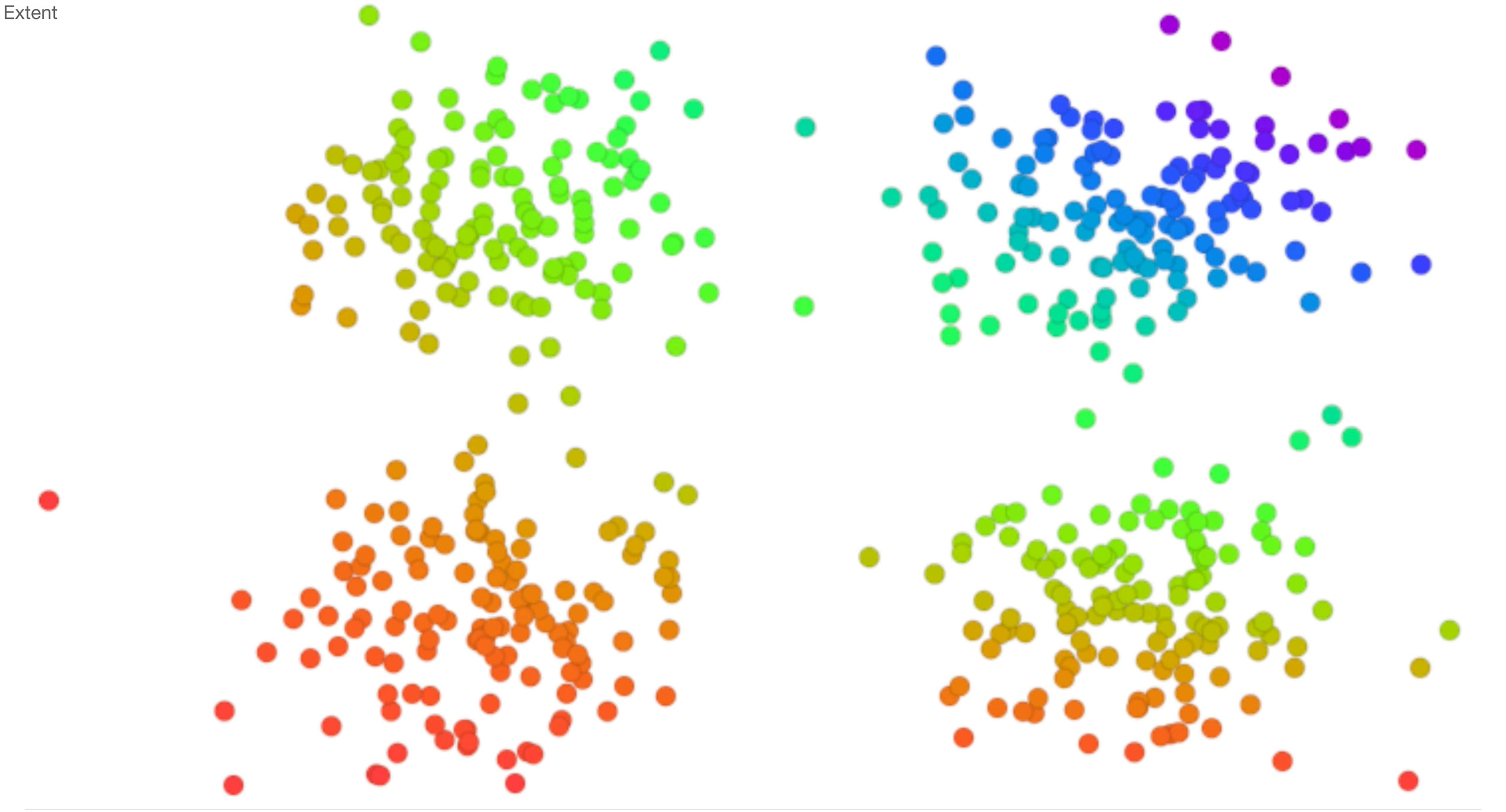

FluidGrid

Gridding



Oversampling




Extent

Pedagogy

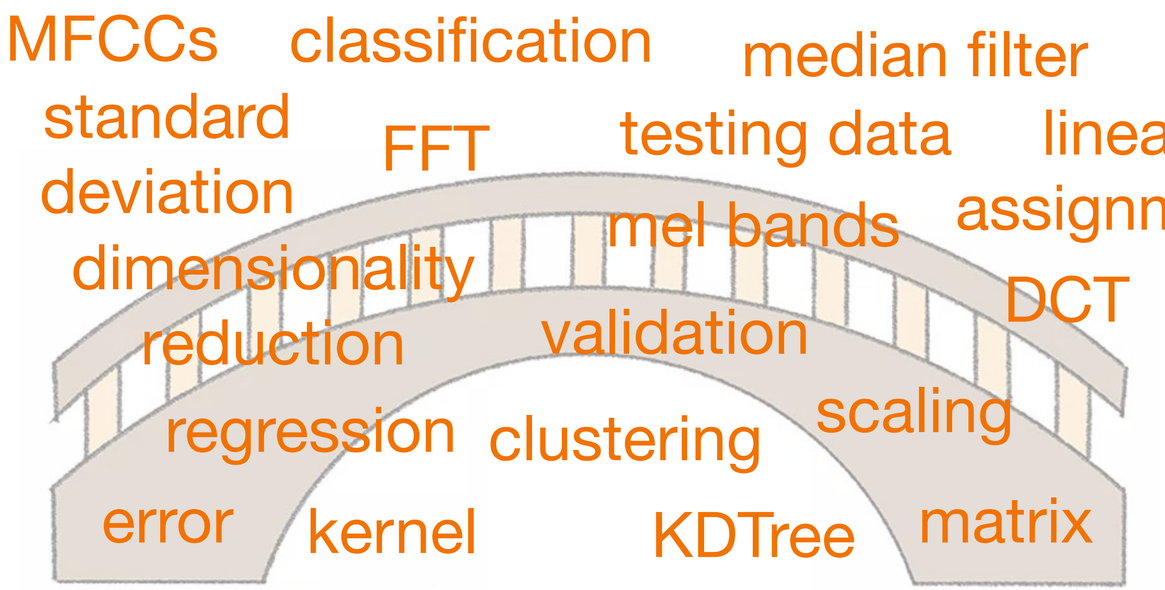
Pedagogical Goals

- “*enable techno-fluent musicians to use machine listening and machine learning in their creative practices*”
- Just “providing the tools” is not enough…
- Make the tools Accessible, Inviting, Learnable, Musical, Inspiring, Expressive
- Build Intuition (Musical Intuition <—> Statistical Intuition)
- Apply knowledge of Machine Learning & AI in broader society

Our Learners

- “Techno-fluent” Musicians
- >= 1 semester of computer music (Max, SuperCollider, or Pure Data)

Techno-fluent Musicians



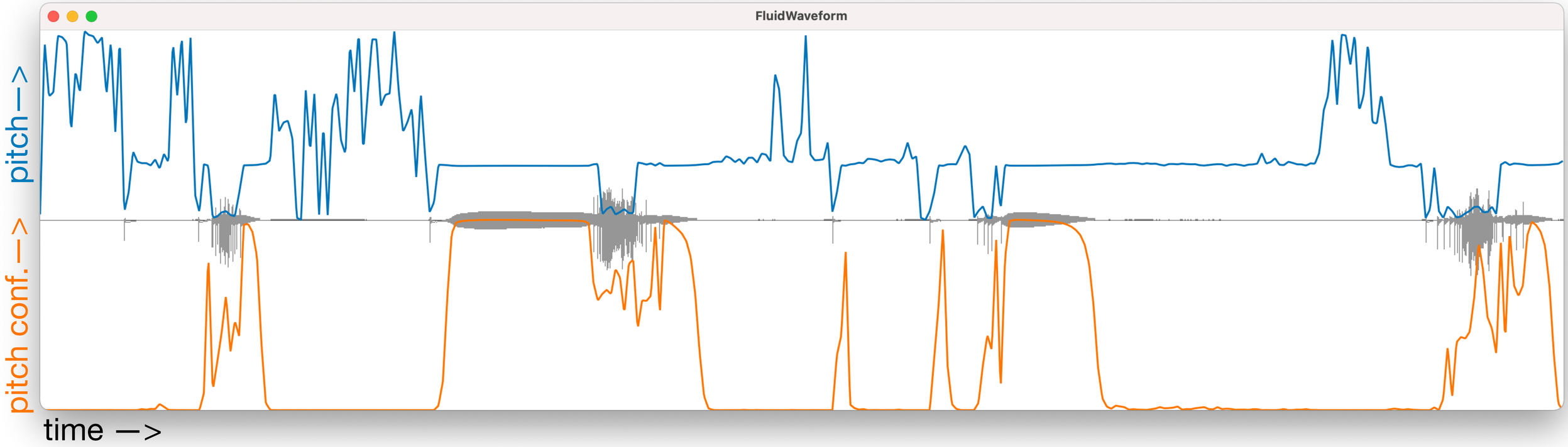
curriculum

Musically expressive uses of Machine Learning & Data Science

Challenges for Learners
& Pedagogical Approaches

Metaphor of Machine “Listening”

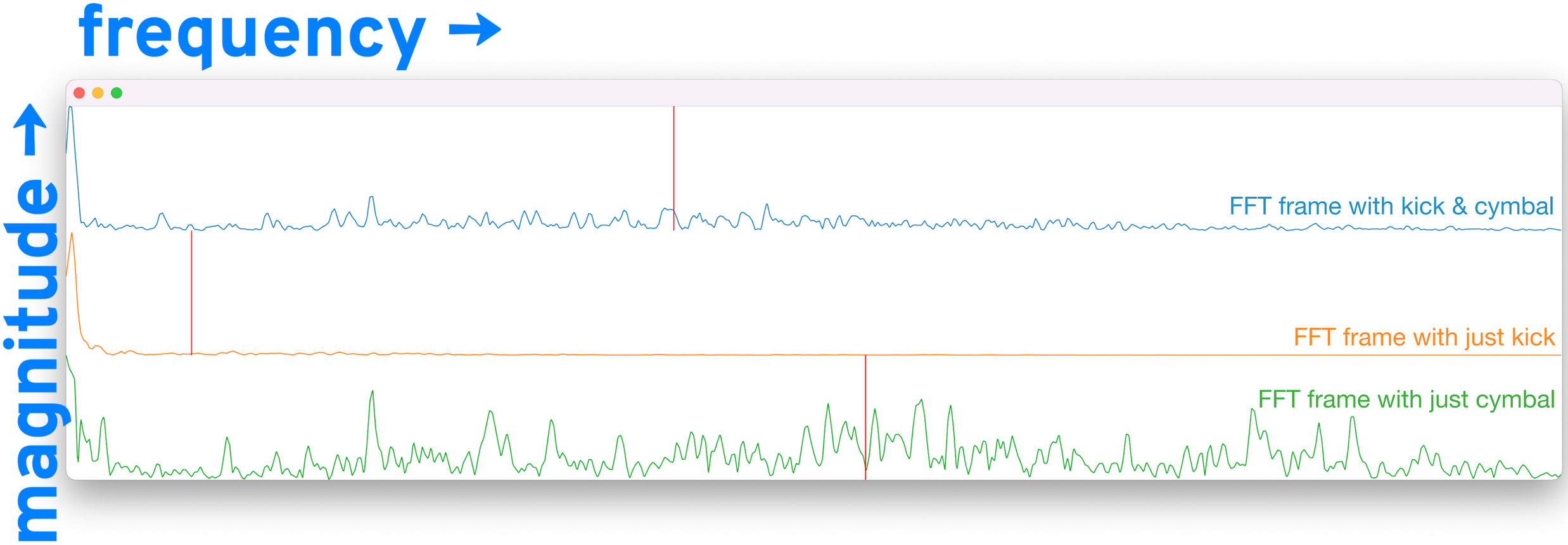
Metaphor of Machine “Listening”



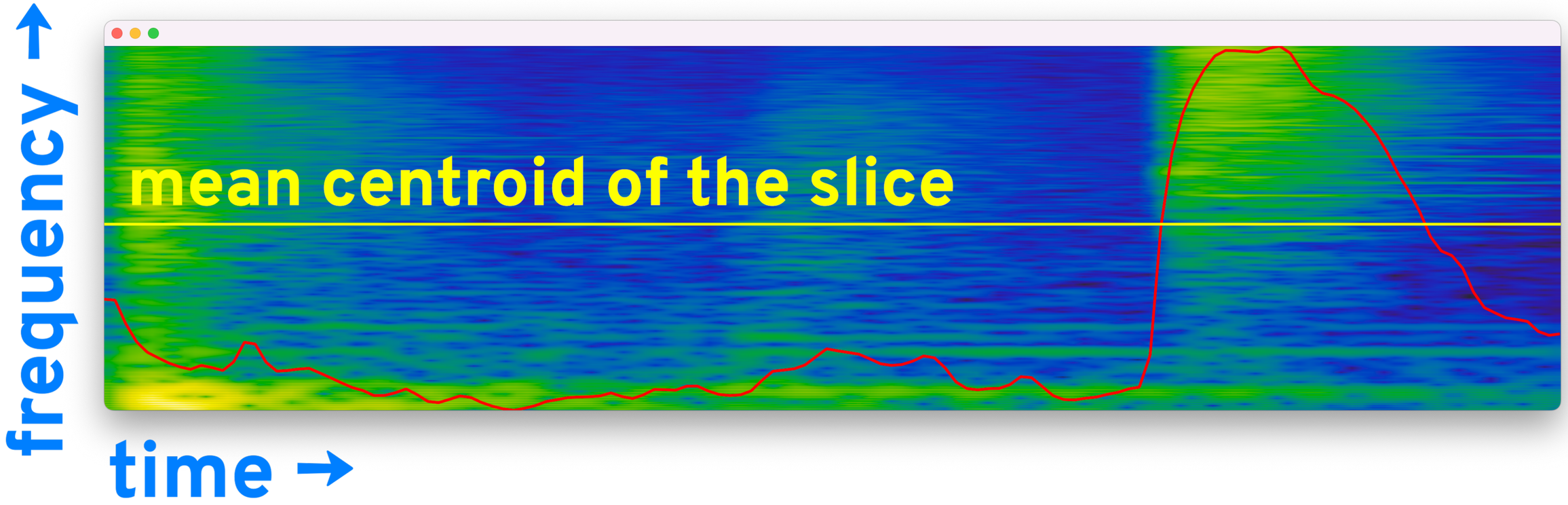
Metaphor of Machine “Listening”

- Audio descriptors & Statistical measures
- Humans do multi-modal listening that
  - rapidly switches modes
  - weights perception according to different criteria

Statistics: spectral centroid



Statistics: mean spectral centroid



*color = magnitude

Managing Data (buffer as data container)

FFT frames (time): —>

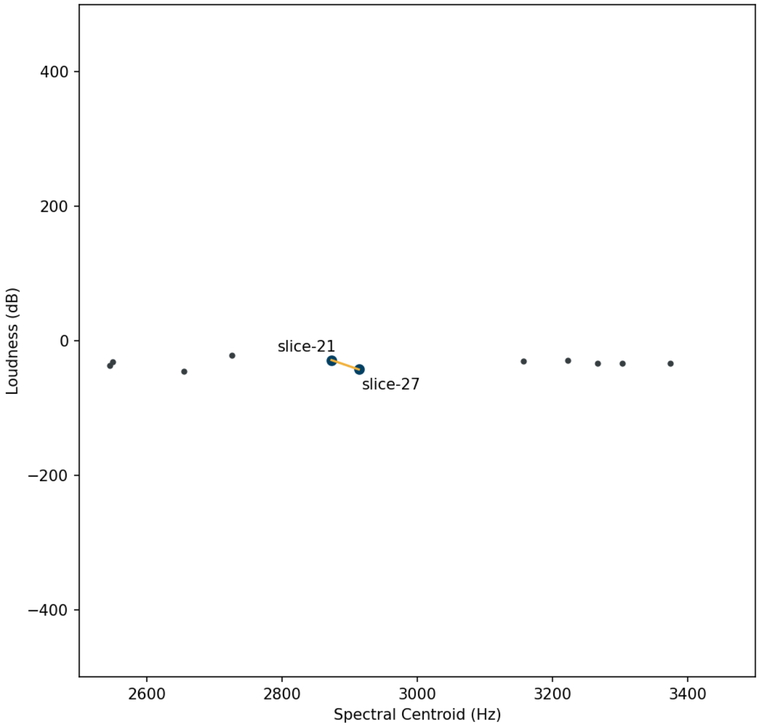
chan: \ frame:	0	1	2	3	4	5	6	7…
0	mfcc 0	mfcc 0	mfcc 0	mfcc 0	mfcc 0	mfcc 0	mfcc 0	…
1	mfcc 1	mfcc 1	mfcc 1	mfcc 1	mfcc 1	mfcc 1	mfcc 1	…
2	mfcc 2	mfcc 2	mfcc 2	mfcc 2	mfcc 2	mfcc 2	mfcc 2	…
3	mfcc 3	mfcc 3	mfcc 3	mfcc 3	mfcc 3	mfcc 3	mfcc 3	…
4	mfcc 4	mfcc 4	mfcc 4	mfcc 4	mfcc 4	mfcc 4	mfcc 4	…
5	mfcc 5	mfcc 5	mfcc 5	mfcc 5	mfcc 5	mfcc 5	mfcc 5	…
…								

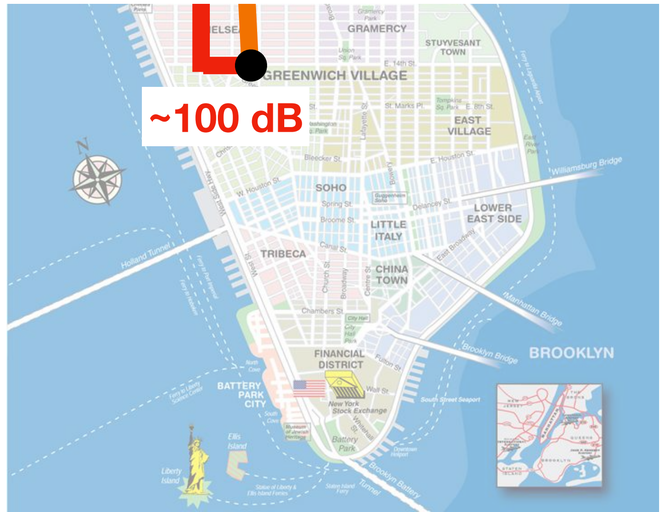
analysis feature —>

interface redesign to *hide* buffers (in Max)

Proximity as Similarity

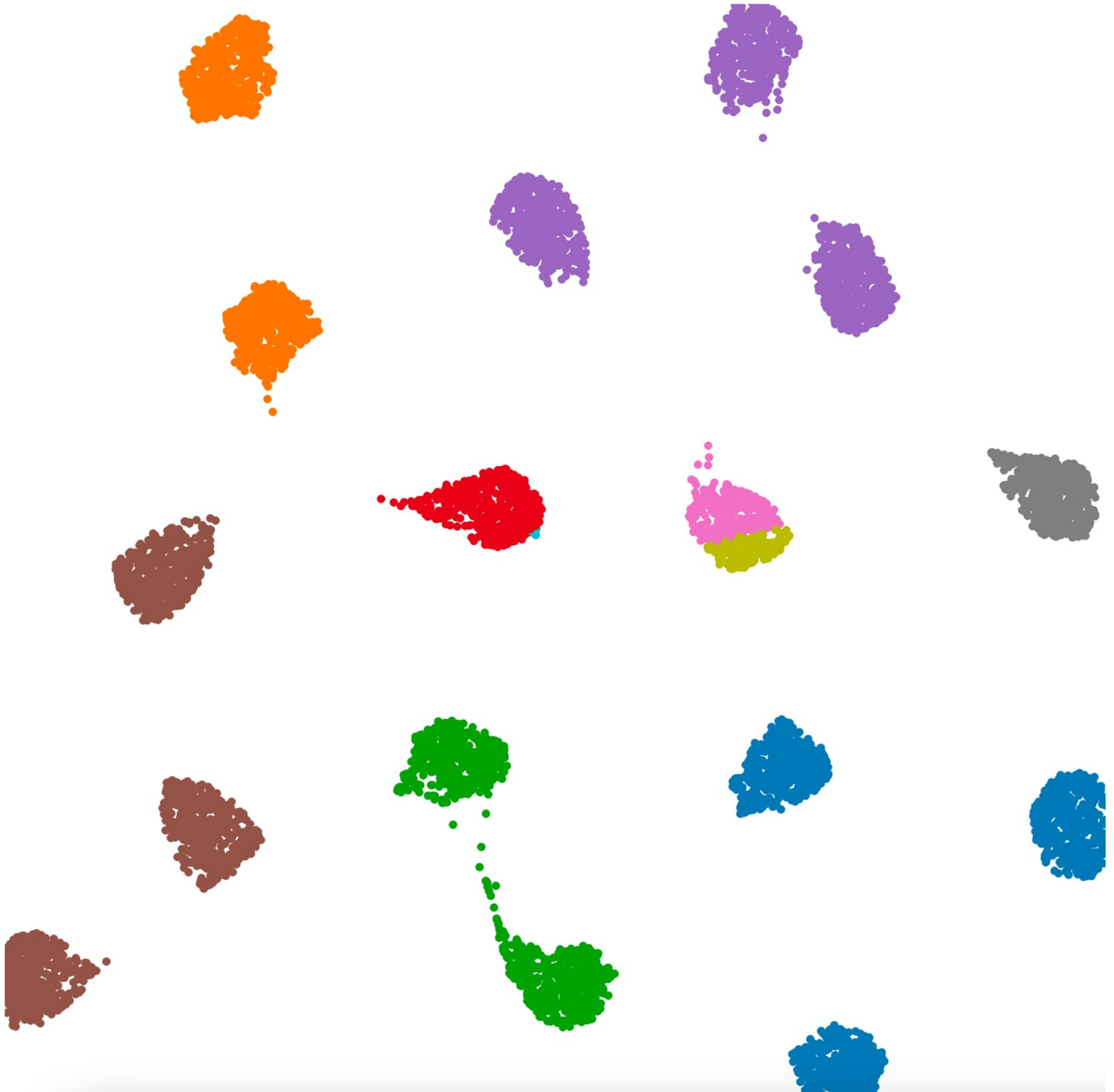
- What does it mean for two sounds to be “similar”?
- Computers compare numerically & measure distance
- Human assumptions of similarity (1 vs. 12 semitones)
- Logarithmic vs. Linear scales
- Mismatched Ranges & Scaling (1 dB ?= 1 Hz)





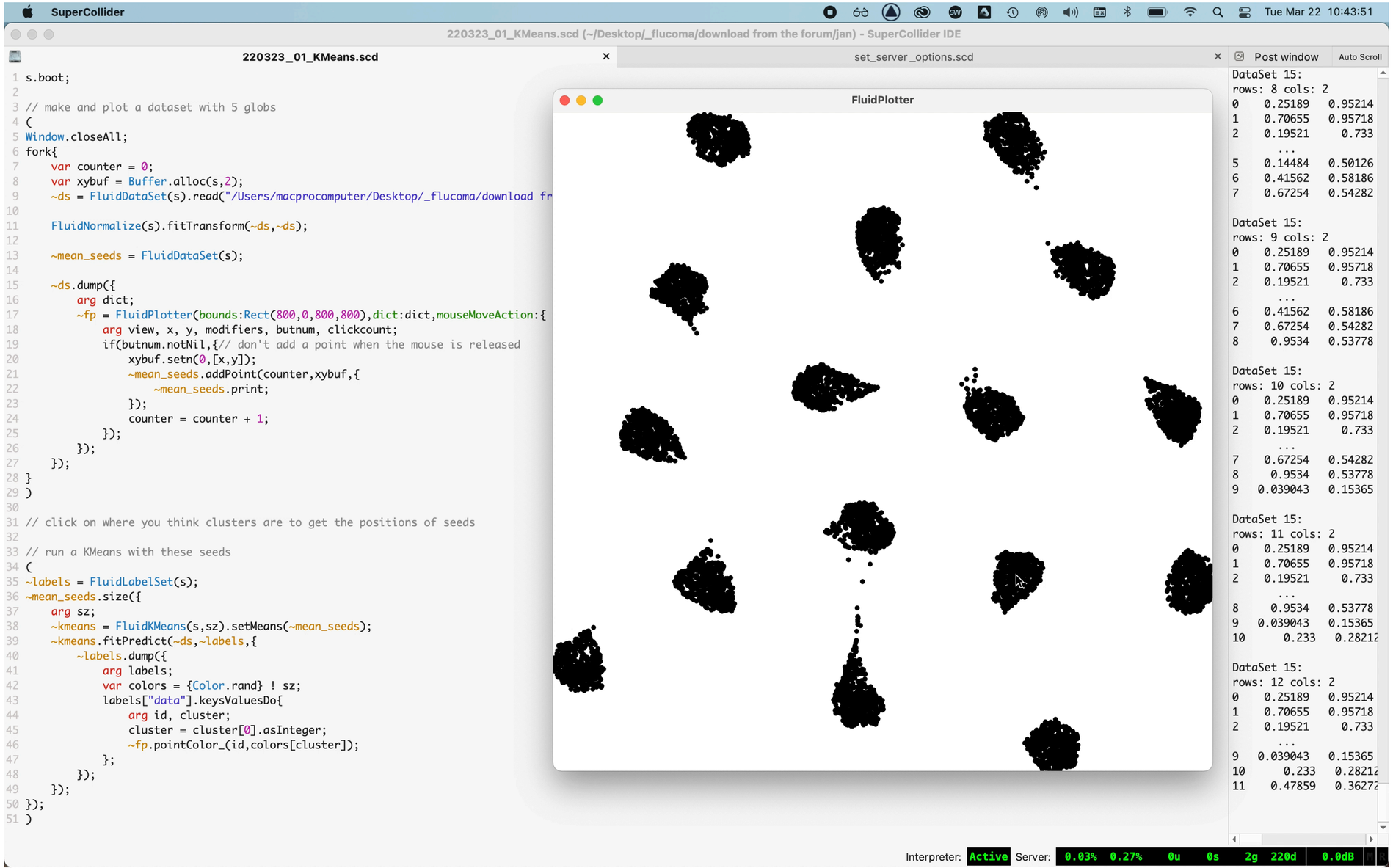
Human Perception vs Machine Perception

- Clustering
- Putting a “human-in-the-loop”



Human Perception vs Machine Perception

- Clustering
- Putting a “human-in-the-loop”



Outliers

- But, what *is* an outlier?
- Or, do I want to use these “outliers”?

A plot with outlier candidates in the upper left section.

Might they be interesting? 🔗

Outliers are not always problematic. They may be musically more relevant than the other points in the dataset. For example, if one were to analyse and plot a folder containing a whole bunch of snare drum hits, many of the snare hits might be very similar and form a cluster of data points. What might be more *musically* interesting is to look for the snare hits that have unique characteristics and stand out when compared to the majority? Another way of considering this is to ask what snare hits are *significantly different* from the majority of the data points in the dataset?

The data that created this plot is a UMAP reduction of MFCC analyses from the sound file below.

Hoping that ML Performs Magic

- Dreaming big and asking “How can this work?…”
  - Instead of “Could this work?”
- Resistance to lots of trial-and-error in building & testing pipelines

Hoping that ML Performs Ma

- Dreaming big and asking “How can this work…
  - Instead of “Could this work?”
- Resistance to lots of trial-and-error in building & testing pipelines

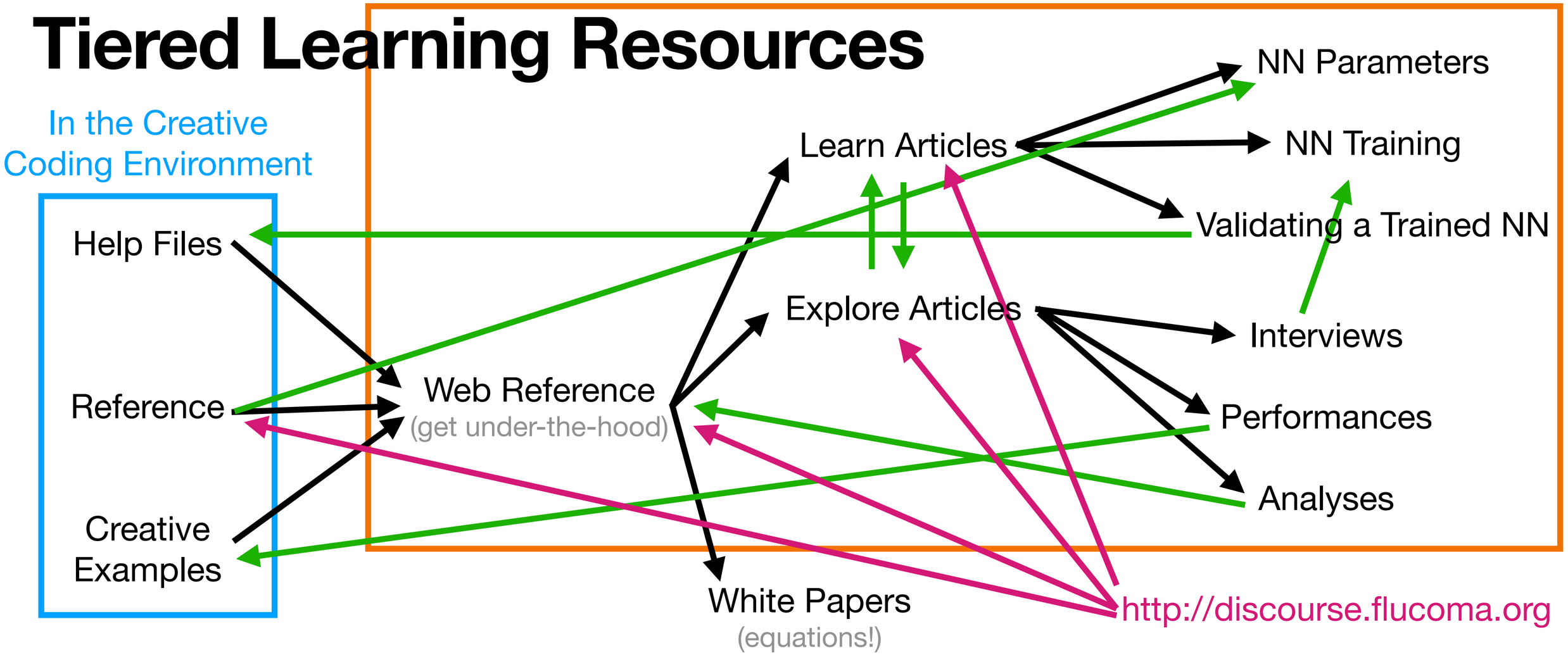
Knowing when to use Machine Learning

Assuming that ML Performs Magic

- Disclaimer: ***If it sounds good keep it.***
- And…is it doing what you think it is doing?
- ***Validation*** can:
  - Offer ways to improve our system to get even closer to our desired outcome.
  - Reveal our assumptions and/or misunderstandings about the data, providing opportunities to deepen our knowledge.
  - Reveal nuances in the system and/or data that might offer more paths of exploration and creativity.

http://learn.flucoma.org

Tiered Learning Resources



Allows learners to pursue knowledge as far as they deem appropriate to feed their creative practice in a given moment.
What they *need* to know *when* they need to know it. Granularity of Pedagogy.
Stay focused on a creative idea, not overwhelmed by the deep rabbit hole or steep learning curve.

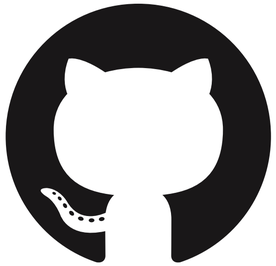
flucoma.org
learn.flucoma.org

download FluCoMa package
toolkit reference
learn from tutorials
hear & see works made with FluCoMa

discourse.flucoma.org

share works in progress
ask questions
get ideas

https://github.com/flucoma



bug reports
pull requests

https://www.youtube.com/c/
fluidcorpusmanipulation

tutorials
performances


Tell me the truth...I'm...I'm ready to hear it.
YOU DON'T NEED MACHINE LEARNING FOR THAT
40
1,406
10.3K
Tweet your reply