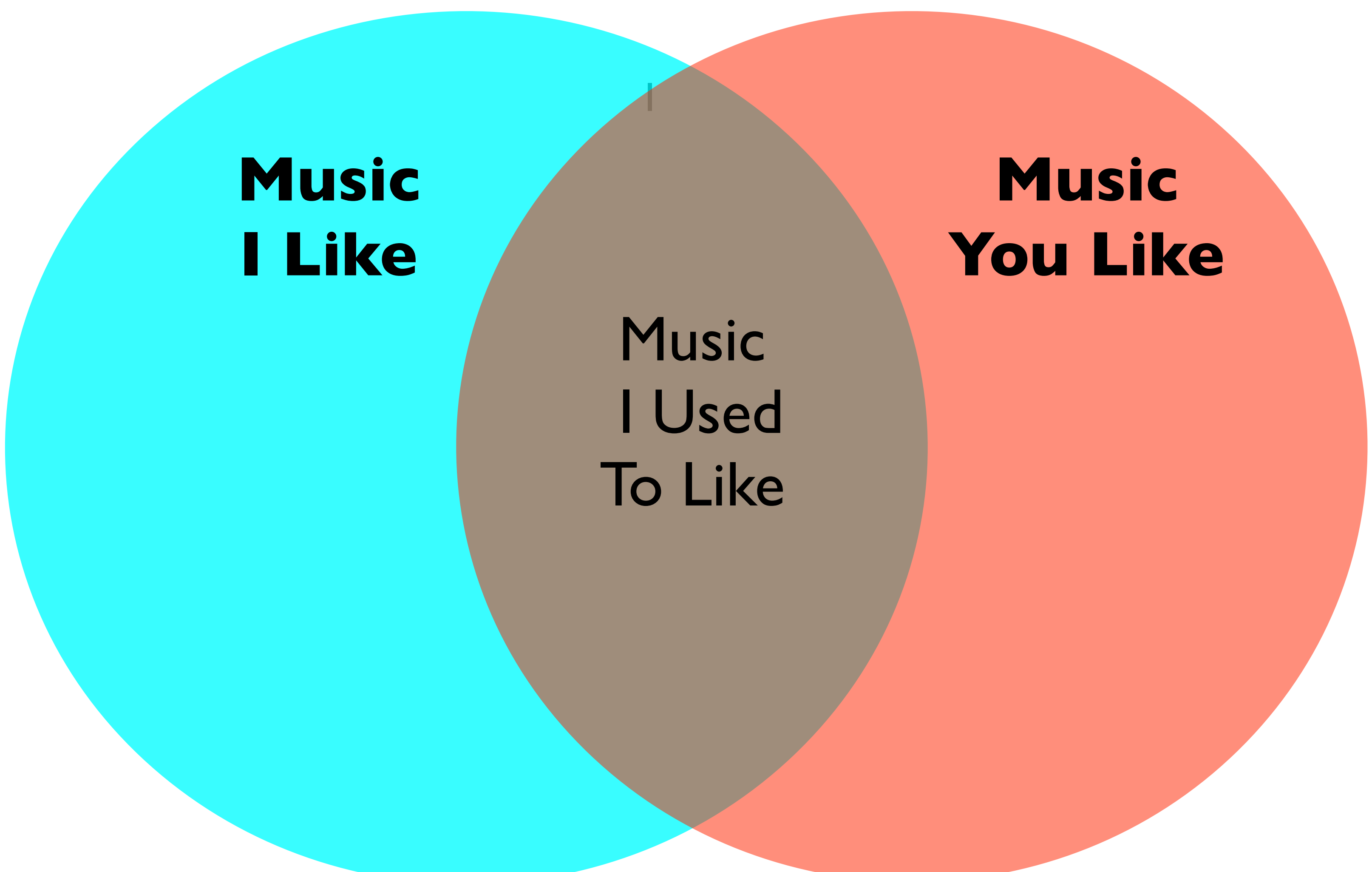
Day 5: Music Recommendation

Douglas Eck

Research Scientist, Google, Mountain View (deck@google.com)



Get this t-shirt at dieselsweeties.com

Overview

- Overview of music recommendation.
- Content-based method: autotagging.
- Side issues of interest

Three Approaches to Recommendation

- Collaborative filtering (Amazon)
  "Many people who bought A also bought B.
  You bought A, you'll probably like B."
  Cannot recommend items no one has bought.
  Suffers from popularity bias

- Social recommendation (Last.FM)
  Community members tag music. Tag clouds used as basis for similarity measure.
  Cannot recommend items no one has tagged.
  Popularity bias (all roads lead to Radiohead)

- Expert recommendation (Pandora)
  Trained experts annotate music based on ~=400 parameters
  Not scalable (thousands of new songs online daily)

Music Recommendation Point - Counterpoint:

What's the best way to help users find music they like?

Point: Use content analysis for music recommendation.

- Paul Lamere (EchoNest)
- Audio helps us know more about music in the long tail.
- Evidence: Examples, observations.

Help! My iPod thinks I'm emo.

SXSW Interactive
March 17, 2009
#sxswemo

Paul Lamere

Anthony Volodkin

Photo (CC) by Jason Rogers

Music recommendation is broken

A recommendation that no human would make

If you like Britney Spears ...

You might like the Report on Pre-War Intelligence

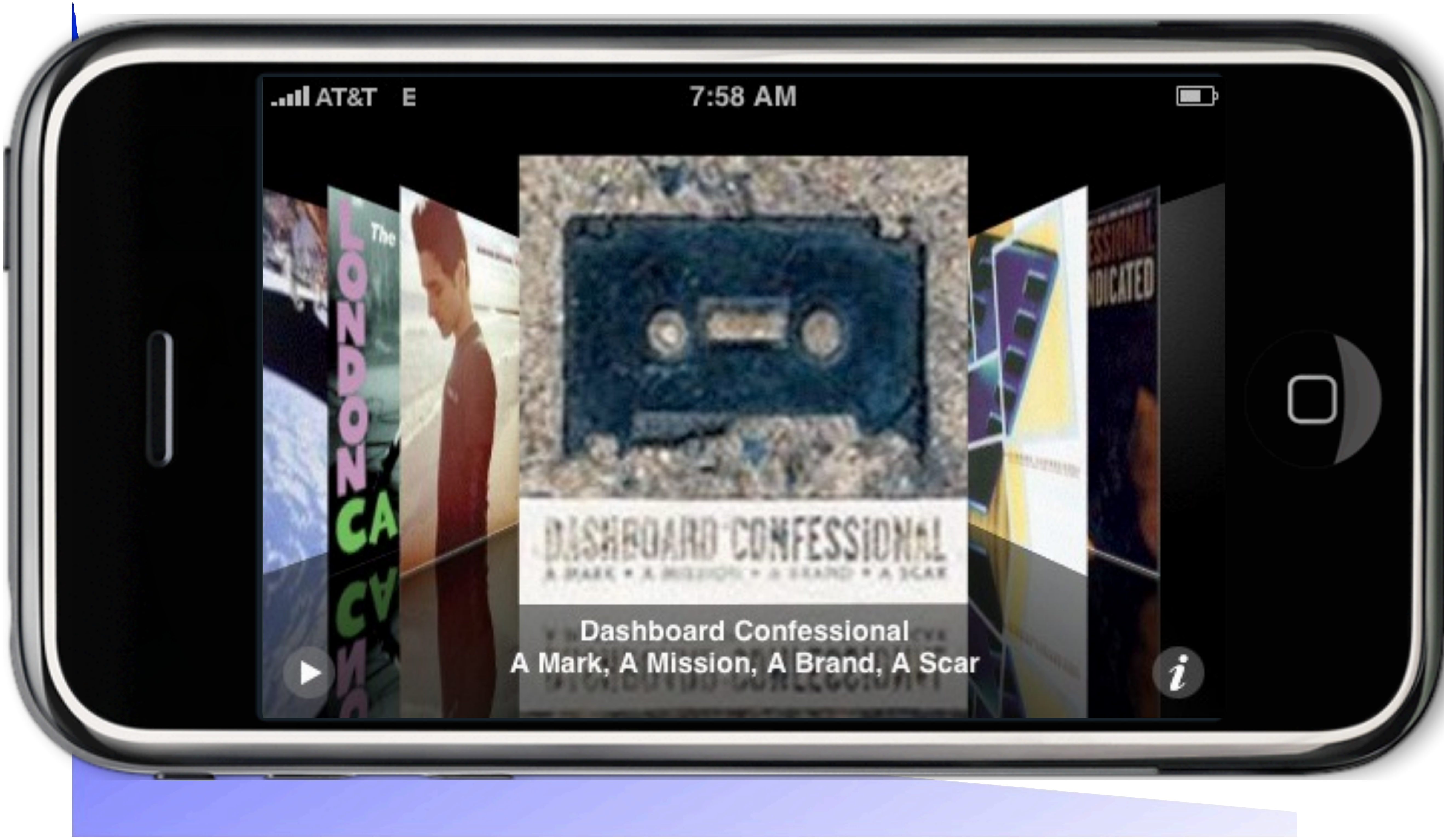
Why do we care?

Compulsory Long Tail slide

http://www.says-it.com/cassette/

Why do we care?

Compulsory Long Tail slide

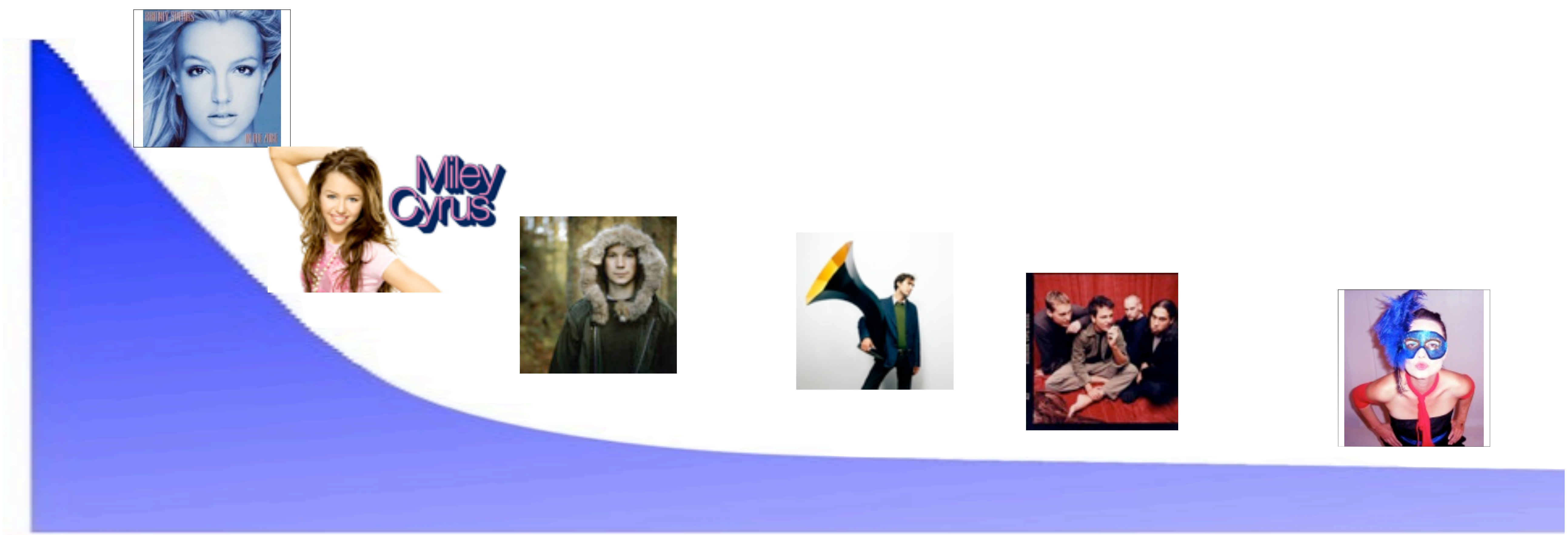


Why do we care?

Compulsory Long Tail slide

Why do we care?

Compulsory Long Tail slide



State of music discovery

We can’t seem to find the long tail

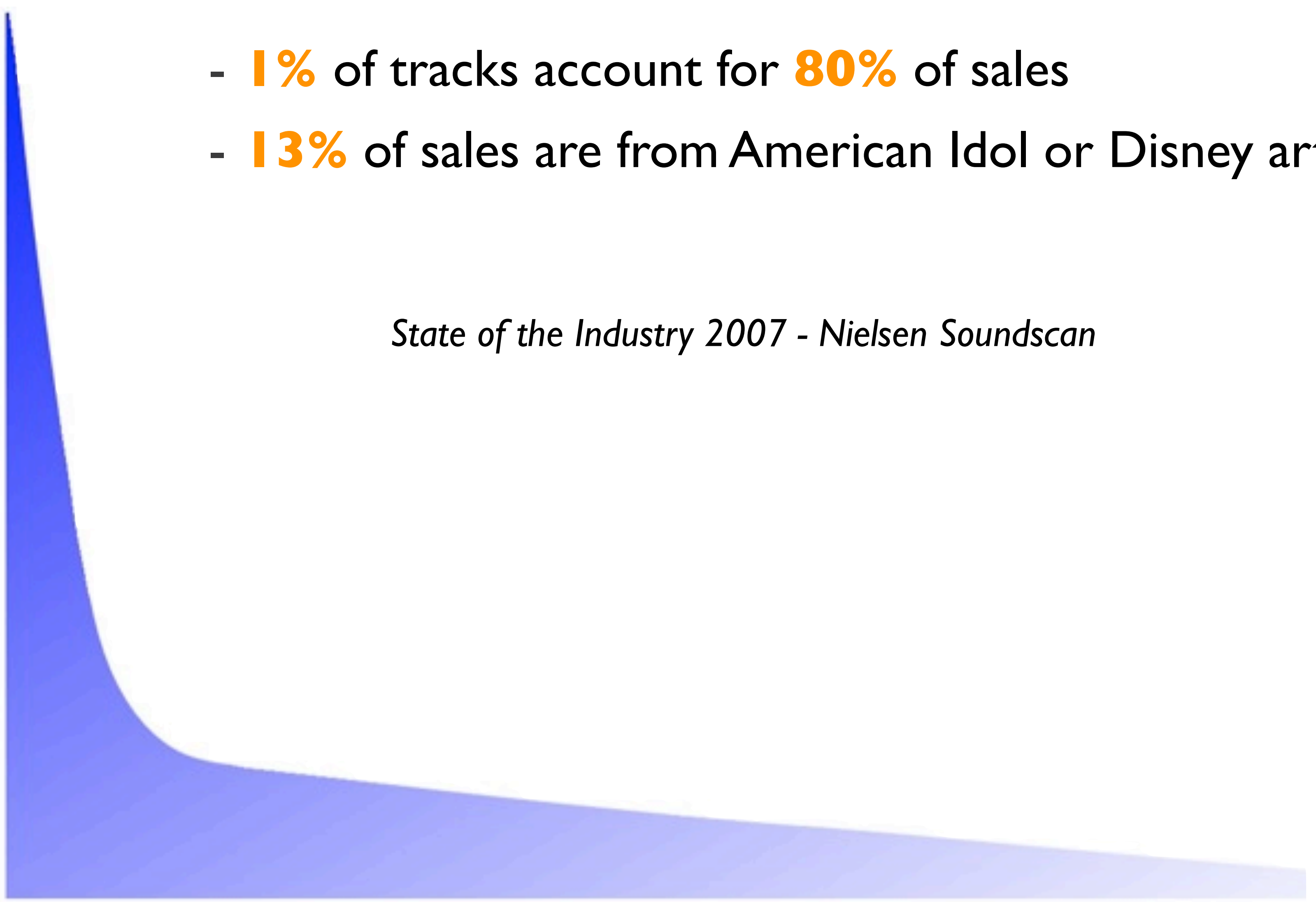
Sales data for 2007

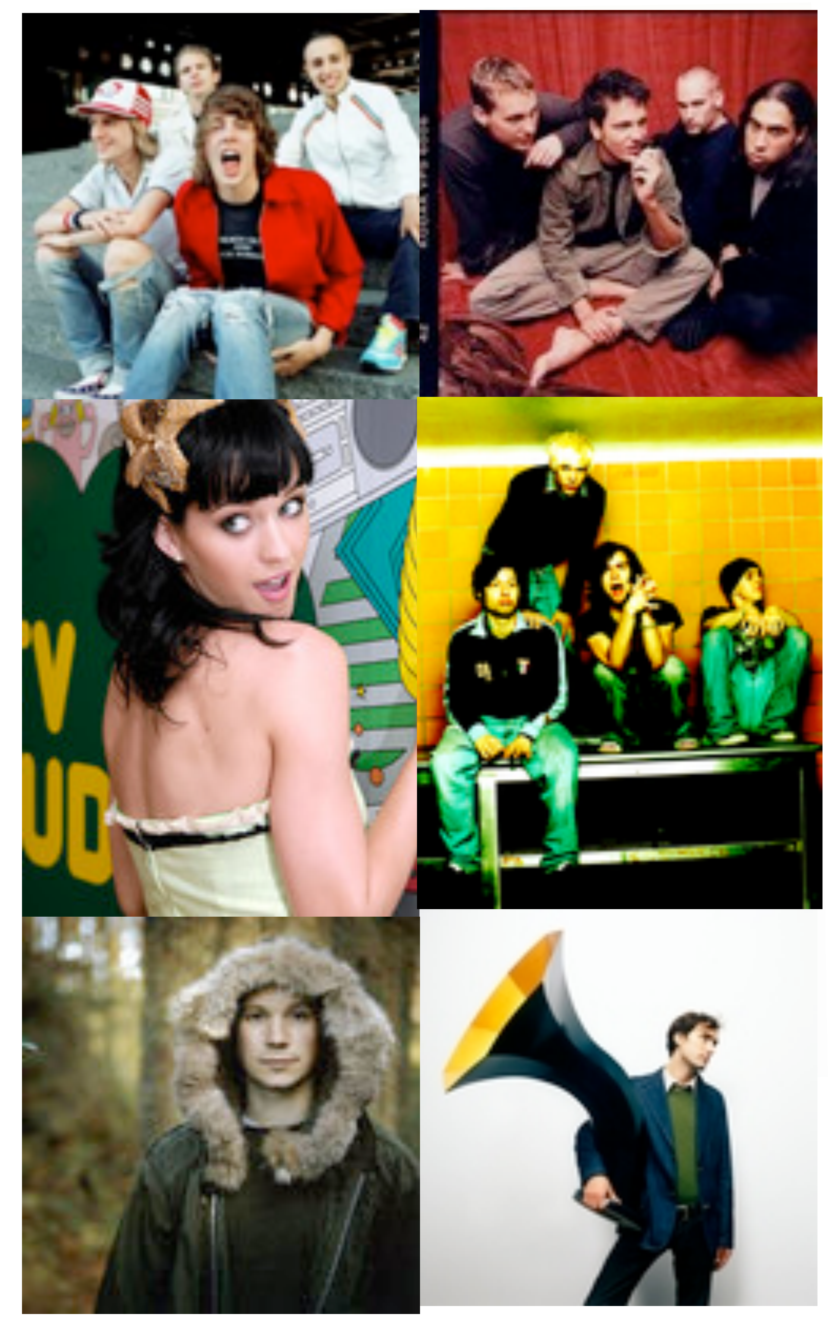
- **4 million** unique tracks sold

But ...

- **1%** of tracks account for **80%** of sales
- **13%** of sales are from American Idol or Disney artists

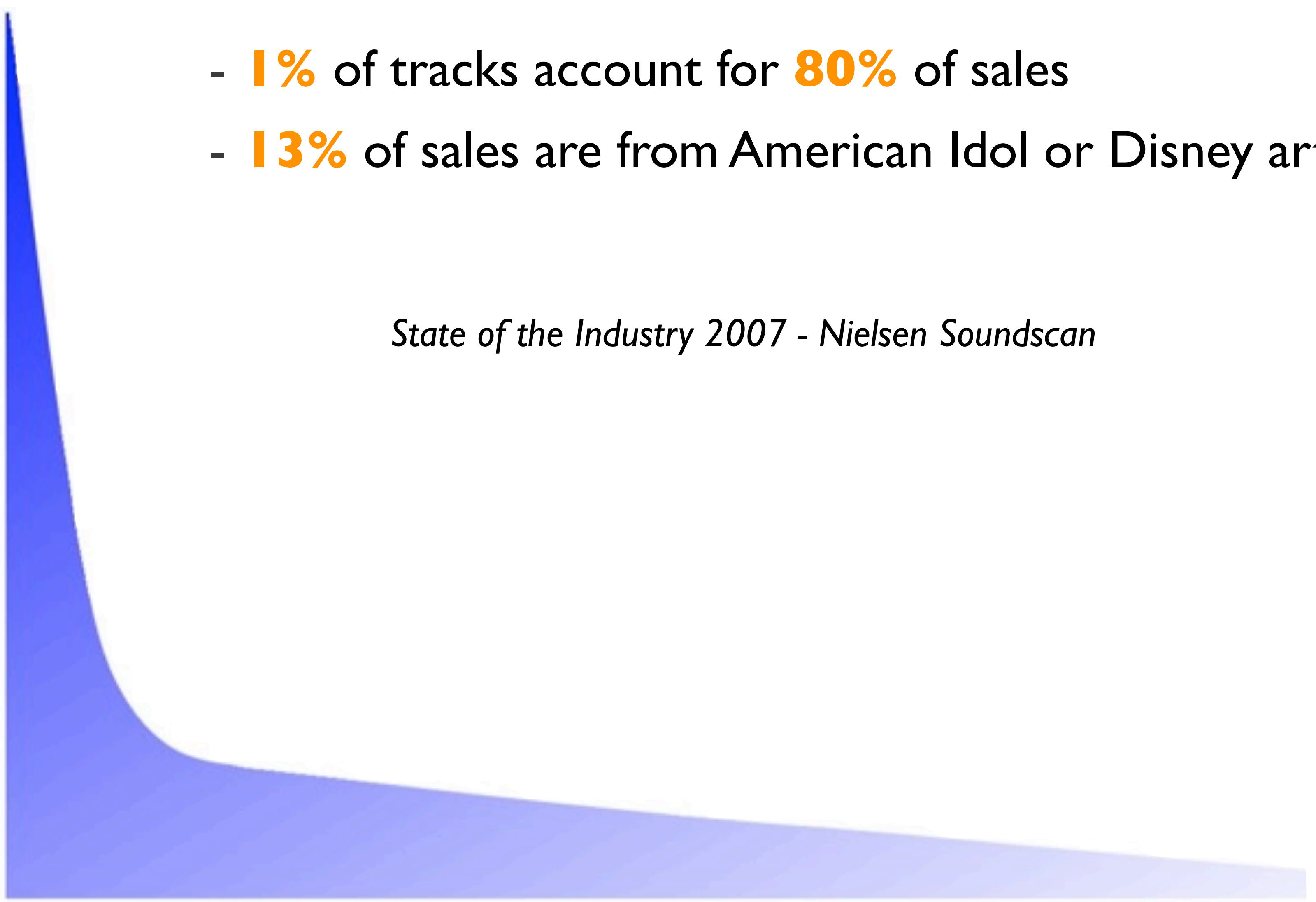
State of the Industry 2007 - Nielsen Soundscan





State of music discovery

We can’t seem to find the long tail



Sales data for 2007

- **4 million** unique tracks sold

But ...

- **1%** of tracks account for **80%** of sales
- **13%** of sales are from American Idol or Disney artists

State of the Industry 2007 - Nielsen Soundscan

State of music discovery

We can’t seem to find the long tail

Sales data for 2007

- **4 million** unique tracks sold

But ...

- **1%** of tracks account for **80%** of sales
- **13%** of sales are from American Idol or Disney artists

State of the Industry 2007 - Nielsen Soundscan

Make everything available

State of music discovery

We can't seem to find the long tail

Sales data for 2007

- **4 million** unique tracks sold

But ...

- **1%** of tracks account for **80%** of sales
- **13%** of sales are from American Idol or Disney artists

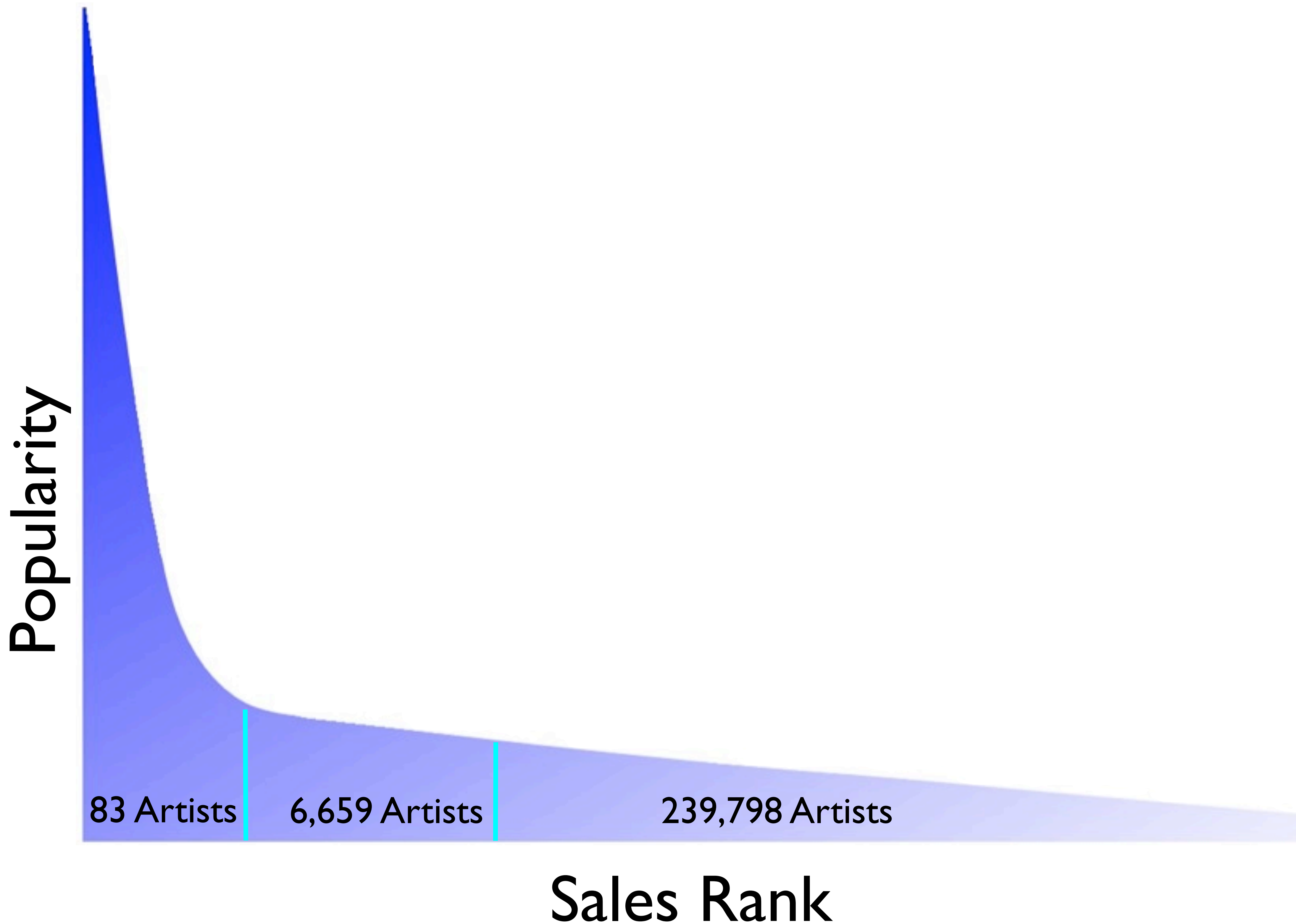
State of the Industry 2007 - Nielsen Soundscan

Make everything available

Help me find it

Help! I'm stuck in the head

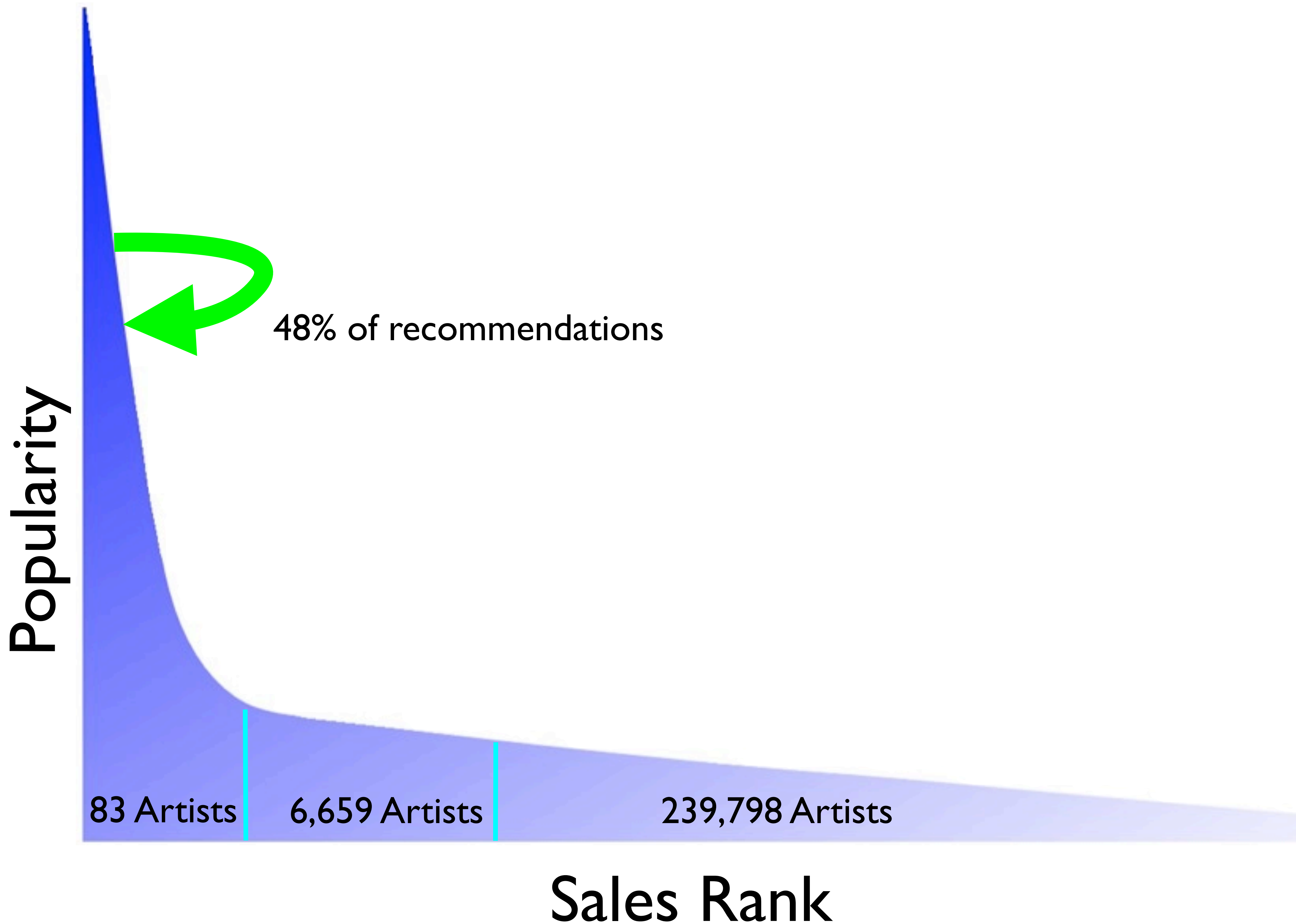
The limited reach of music recommendation



Study by Dr. Oscar Celma - MTG UPF

Help! I'm stuck in the head

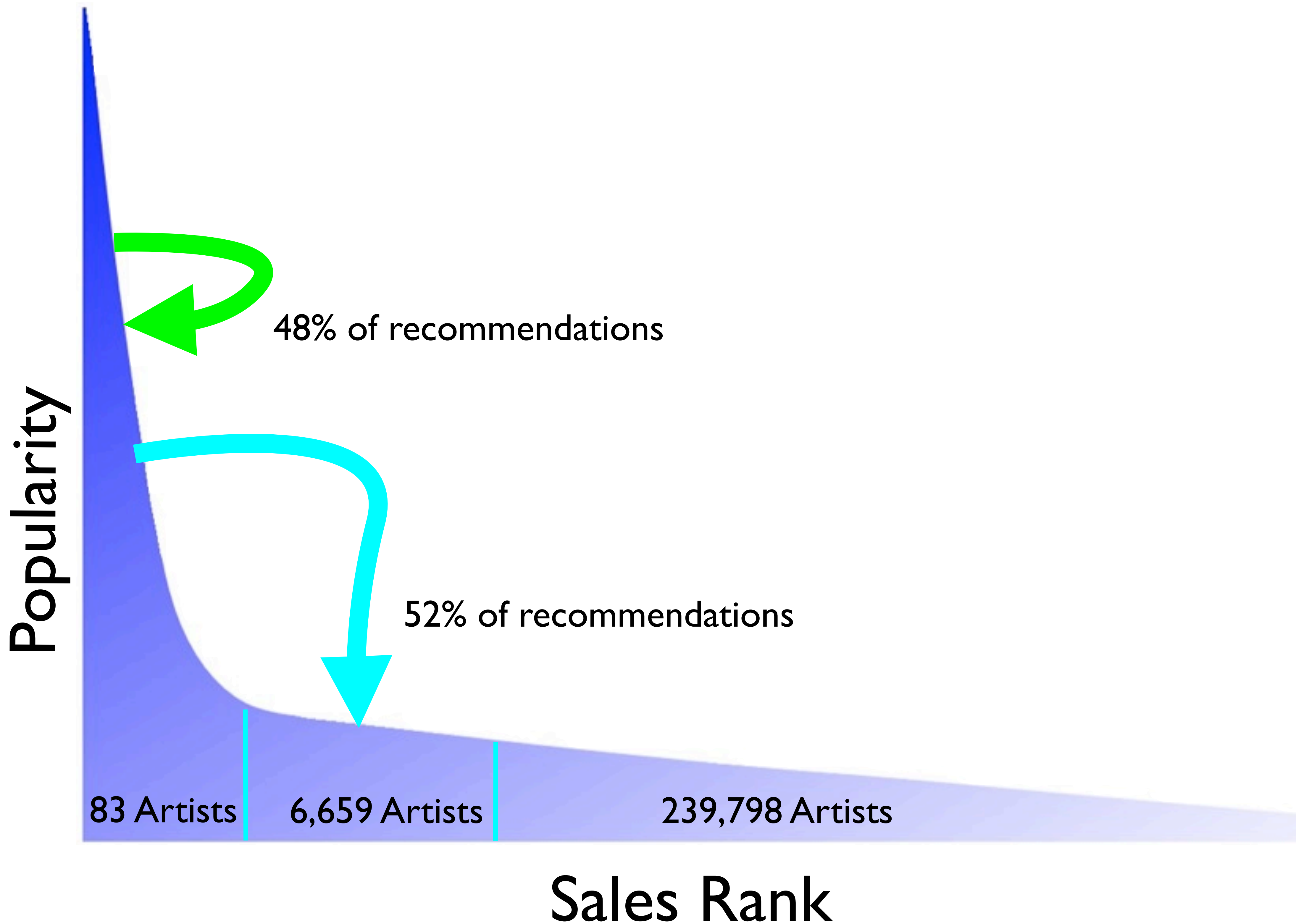
The limited reach of music recommendation



Study by Dr. Oscar Celma - MTG UPF

Help! I'm stuck in the head

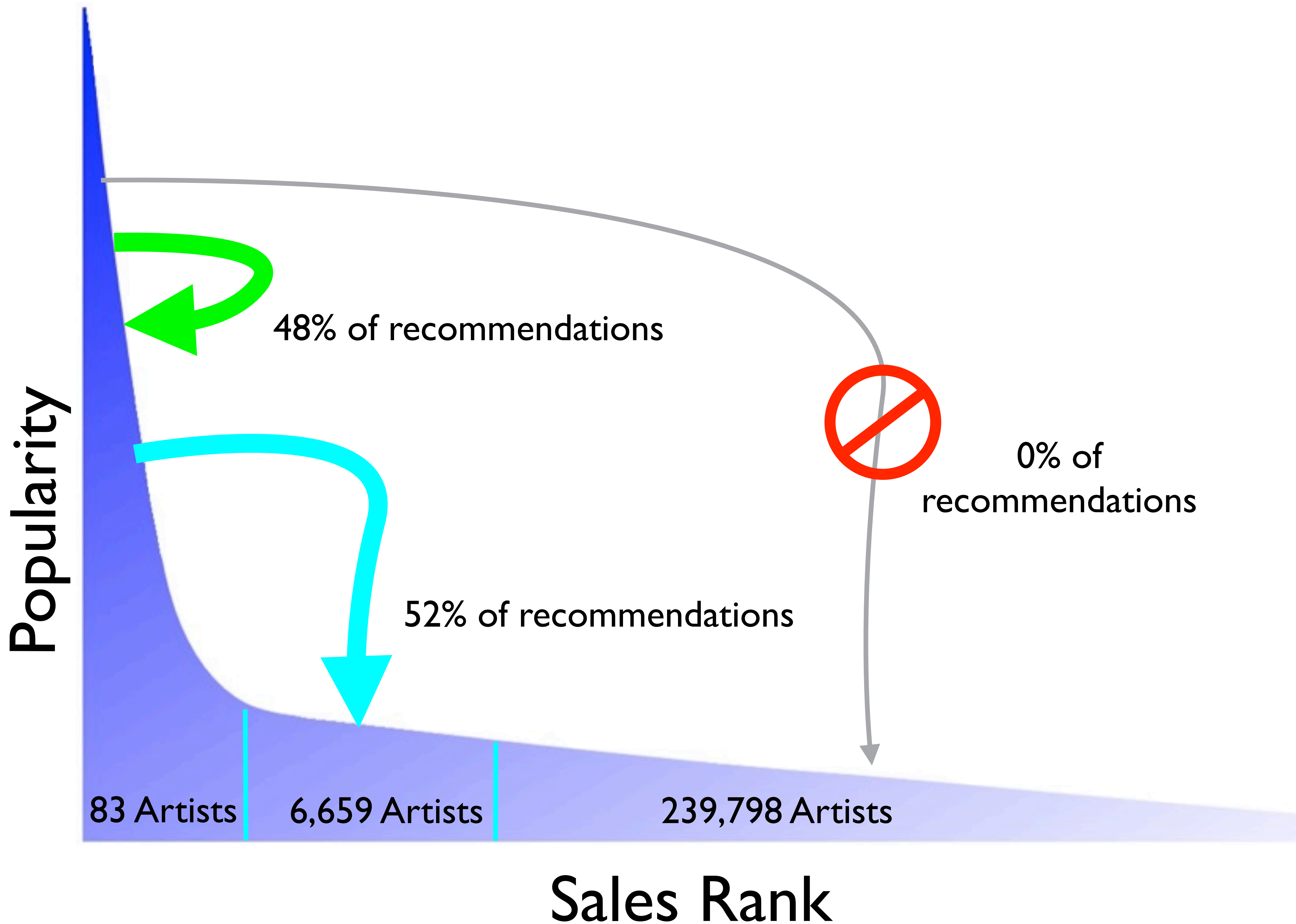
The limited reach of music recommendation



Study by Dr. Oscar Celma - MTG UPF

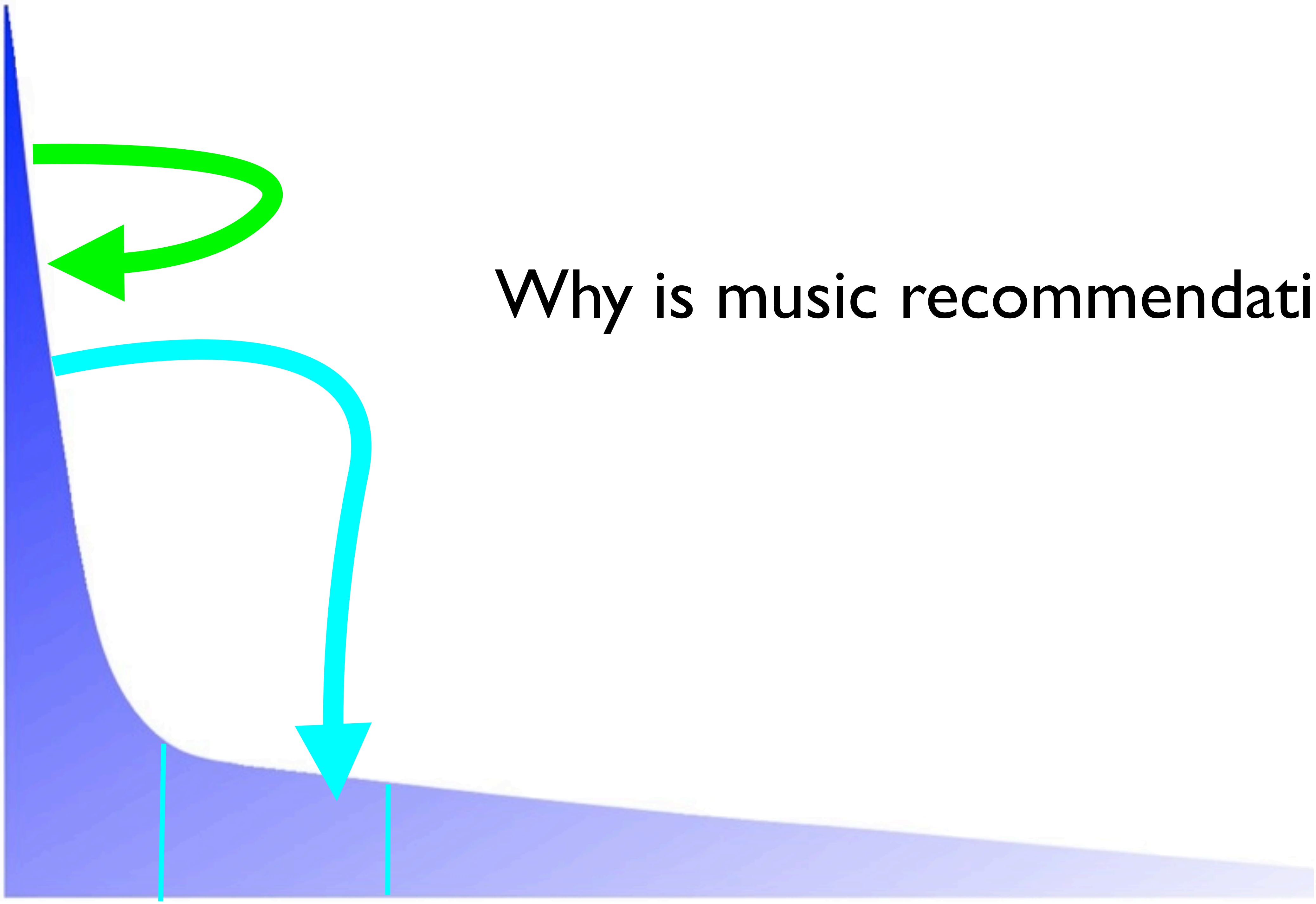
Help! I'm stuck in the head

The limited reach of music recommendation



Study by Dr. Oscar Celma - MTG UPF

Help! My iPod thinks I’m emo



Why is music recommendation broken?

The Wisdom of Crowds

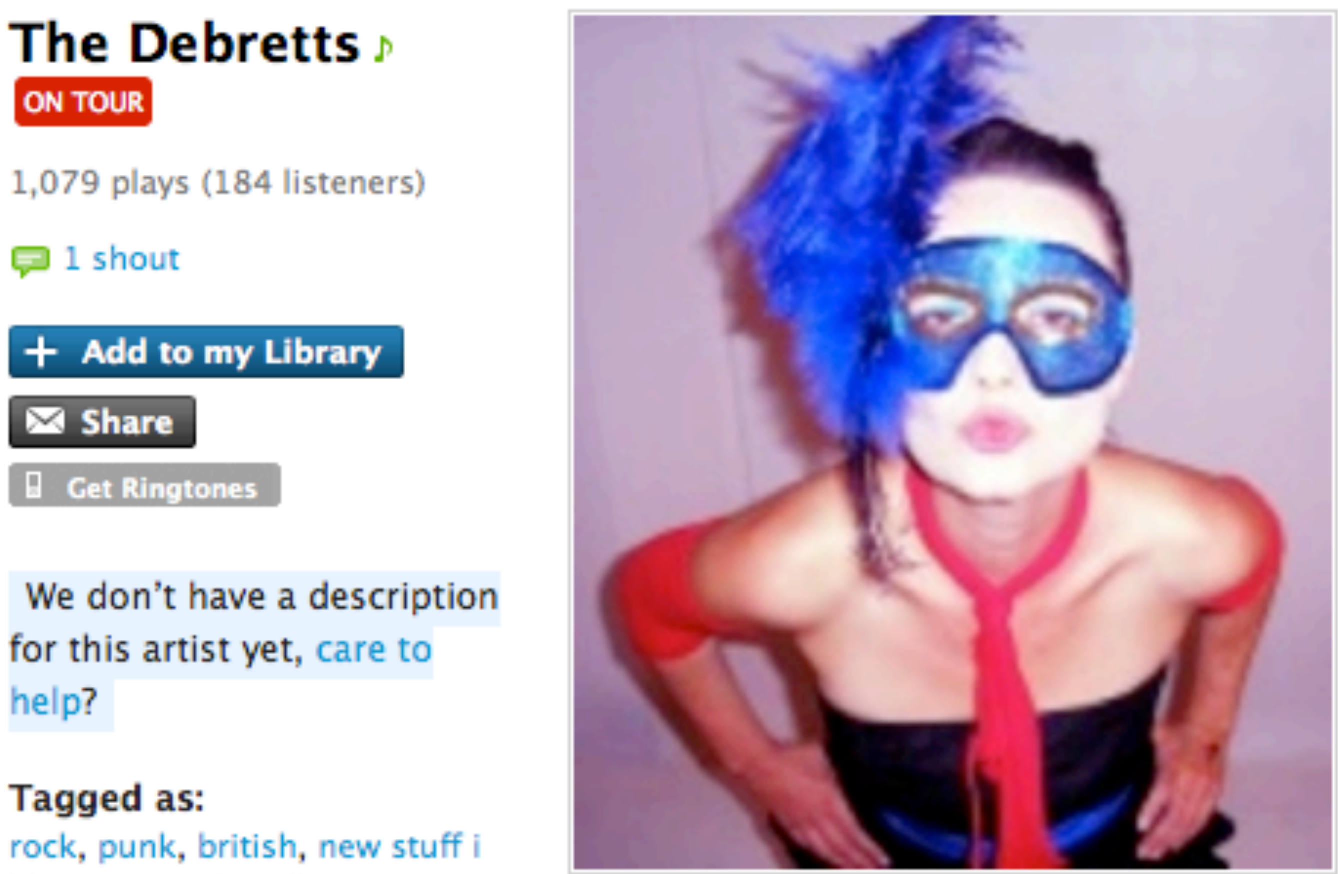
How does collaborative filtering work?

					
	35%	4%	62%	8%	60%

Overlap Data based on listening behavior of 12,000 Last.fm Listeners

The stupidity of solitude
The **Cold Start** problem

If you like Blondie, you might like the DeBretts ...

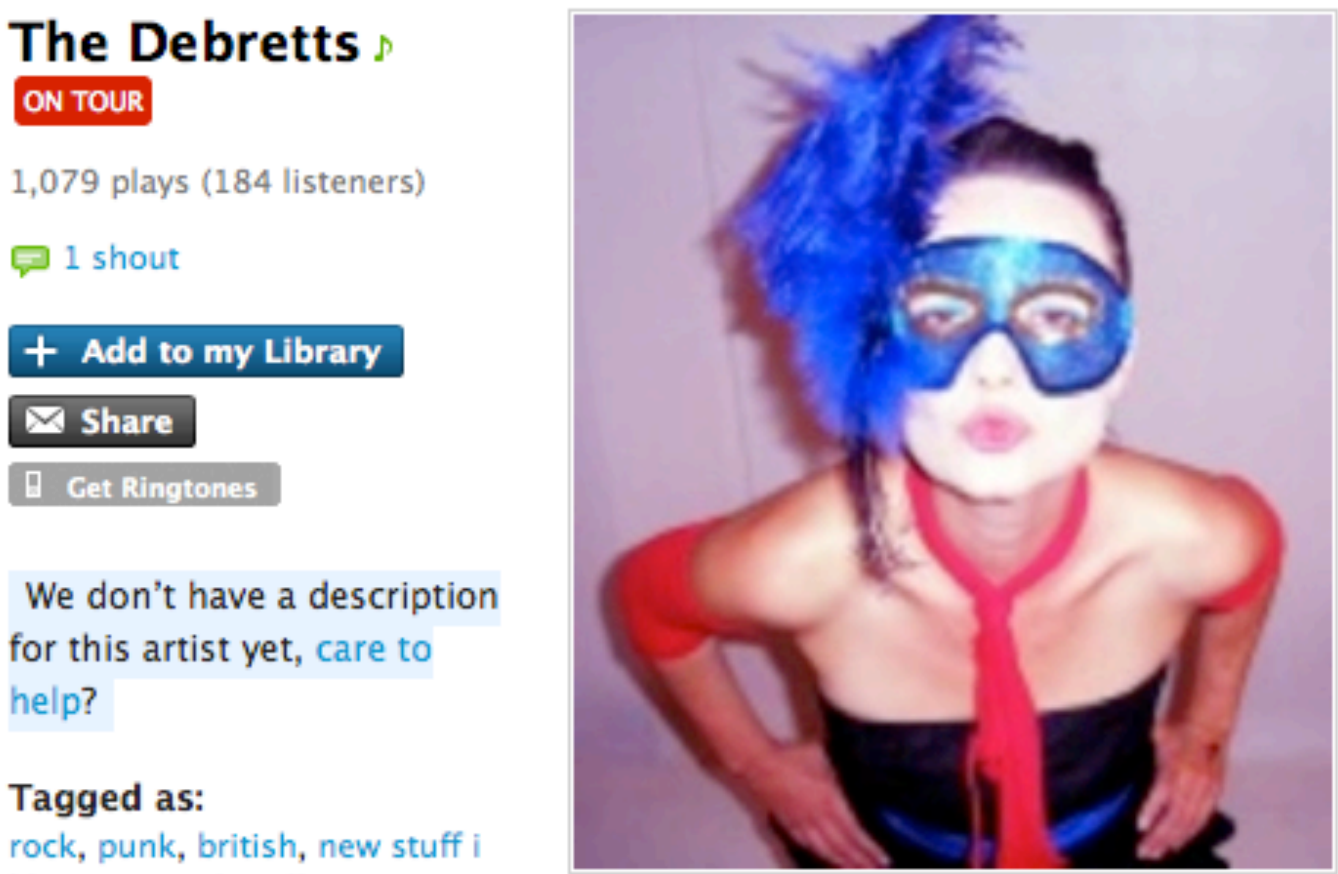


But the recommender will never tell you that.

The stupidity of solitude

The **Cold Start** problem

If you like Blondie, you might like the DeBretts ...

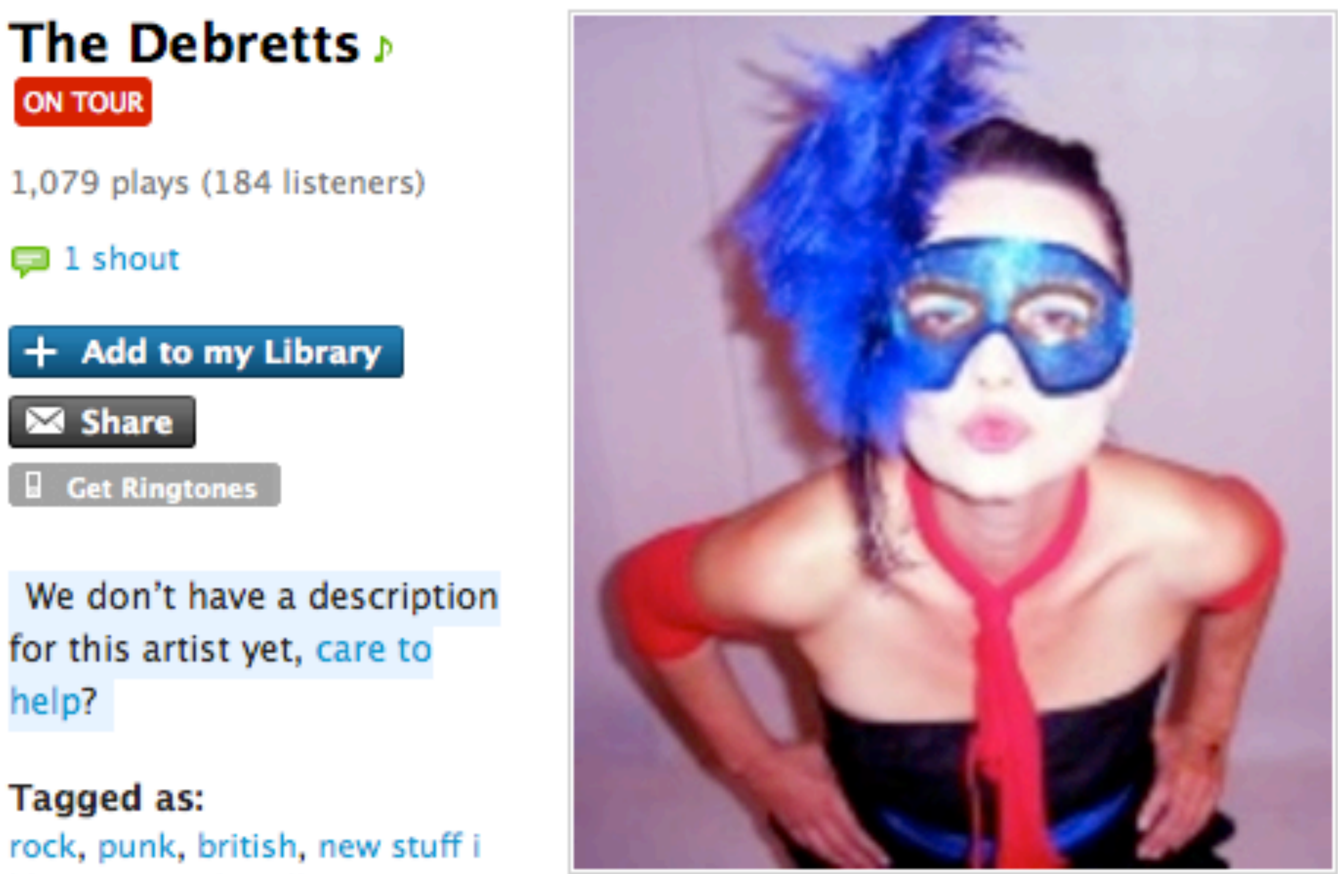


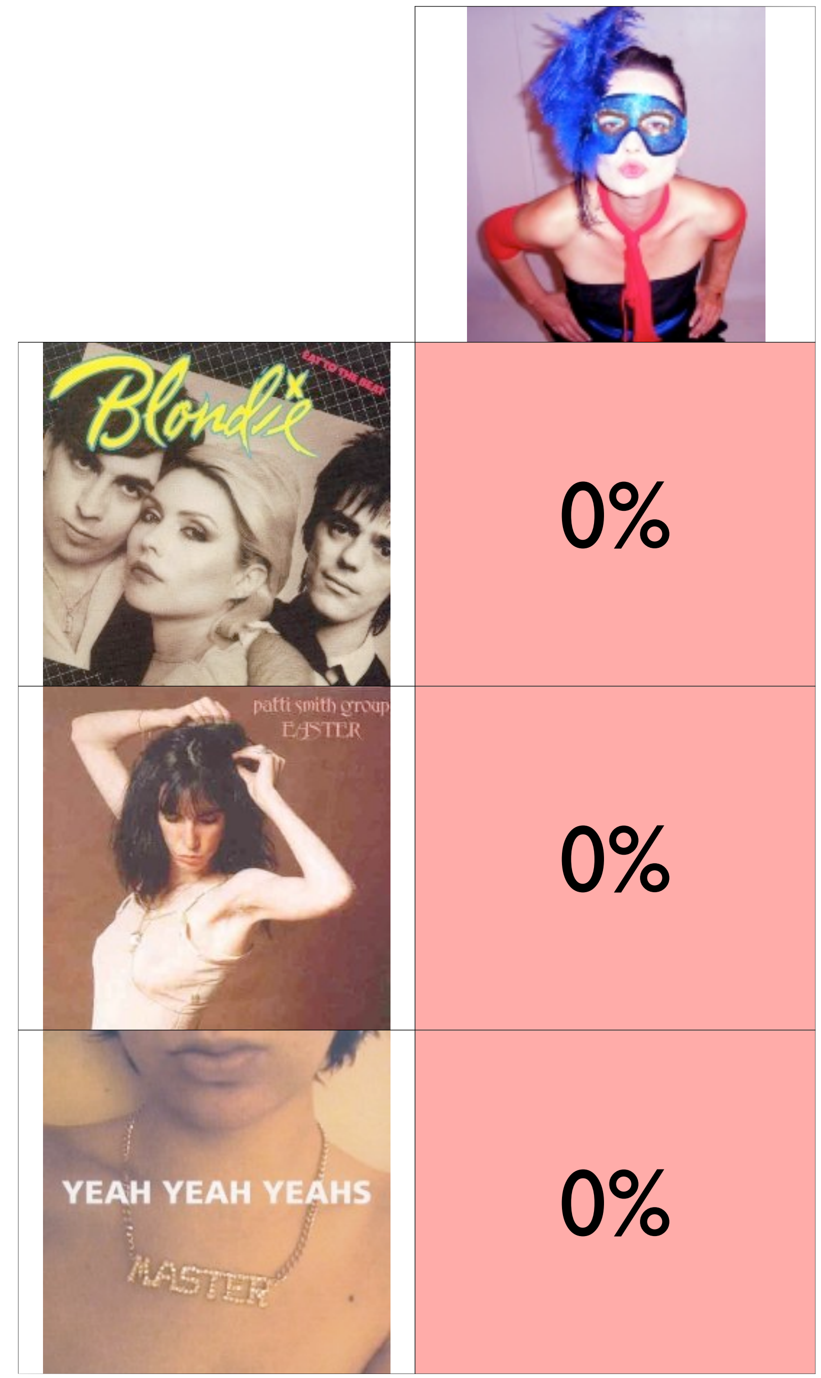
But the recommender will never tell you that.

The stupidity of solitude

The **Cold Start** problem

If you like Blondie, you might like the DeBretts ...





But the recommender will never tell you that.

The Harry Potter Problem

If you like X you might like Harry Potter

Powell's Recommendations

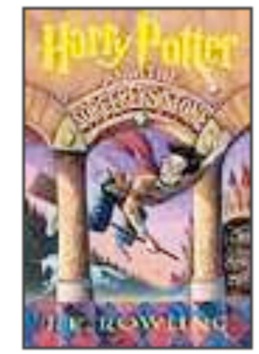
If you enjoyed Java RMI by *William Grosso*, you might also enjoy the following titles:

$29.95
New Trade Paper
ADD TO CART
add to wishlist

Pragmatic Unit Testing in Java with JUnit (Pragmatic Programmers)
Andrew Hunt

$41.00
Used Hardcover
ADD TO CART
add to wishlist

Design Patterns: Elements of Reusable Object-Oriented Software (Addison-Wesley Professional Computing)
Erich Gamma

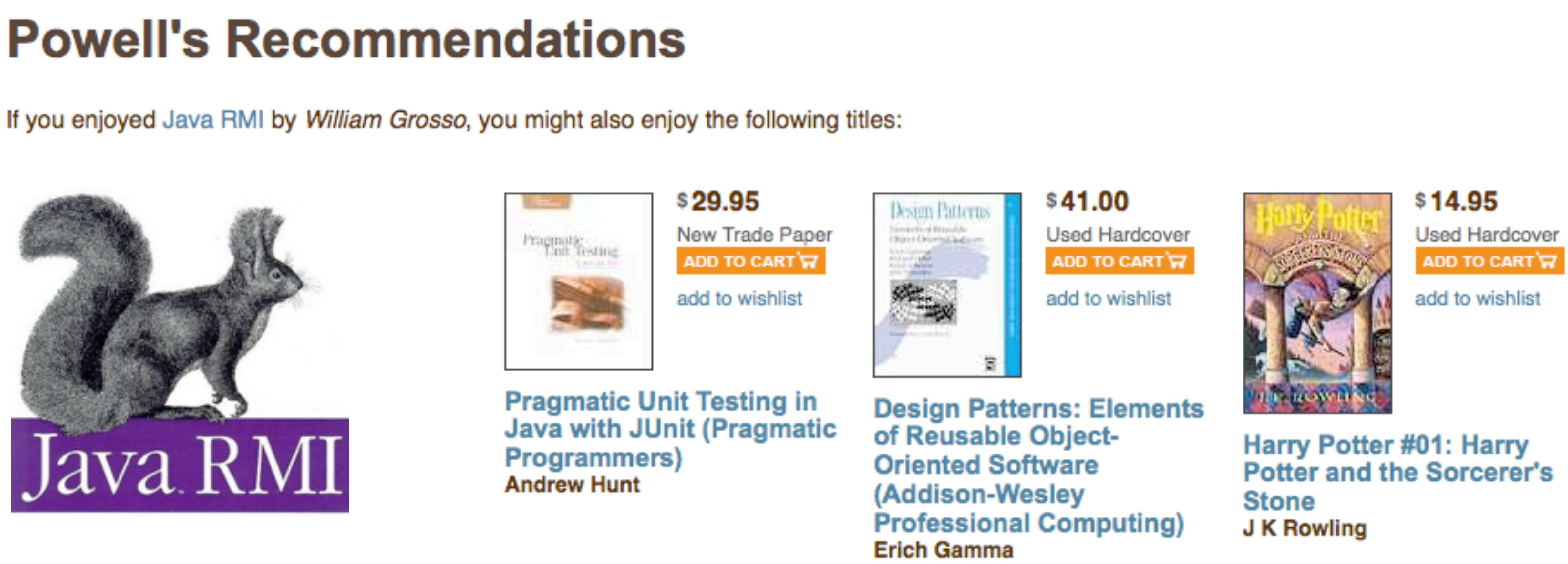


$14.95
Used Hardcover
ADD TO CART
add to wishlist

Harry Potter #01: Harry Potter and the Sorcerer's Stone
J K Rowling

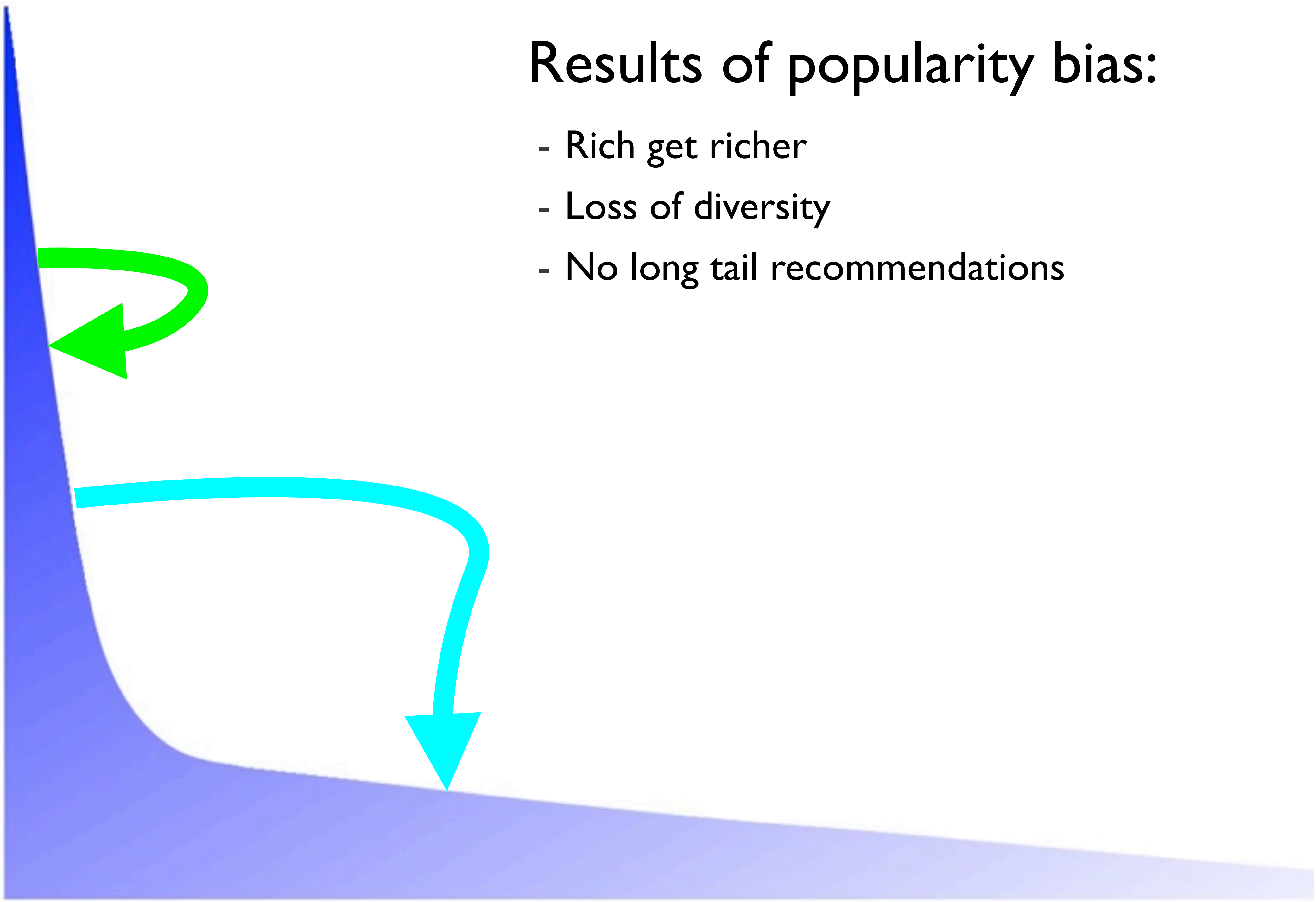
The Harry Potter Problem

If you like X you might like Harry Potter



Popularity Bias

Rich get richer - diversity is the biggest loser

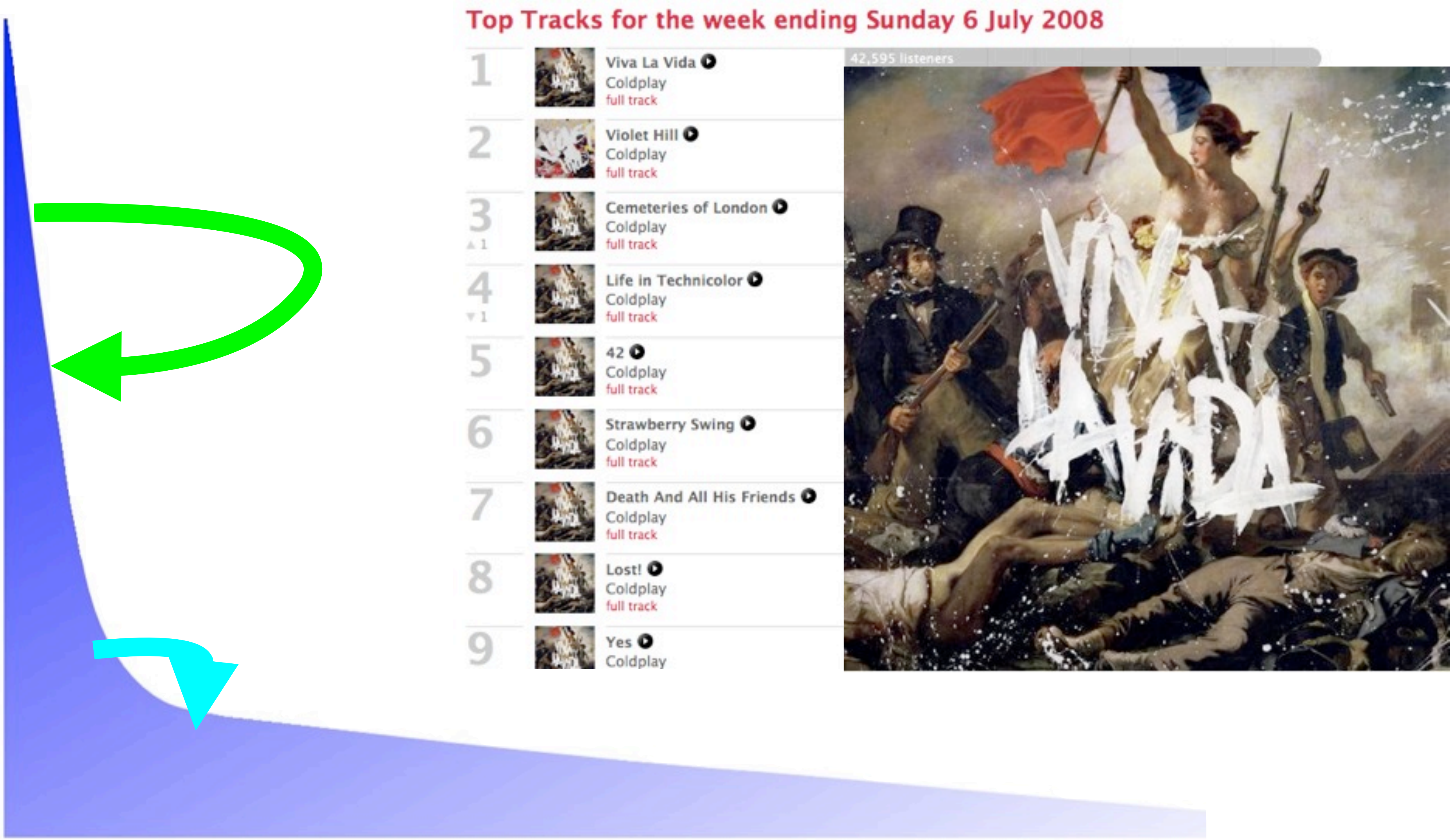


Results of popularity bias:

- Rich get richer
- Loss of diversity
- No long tail recommendations

Popularity Bias

Rich get richer - diversity is the biggest loser



The Novelty Problem

If you like The Beatles you might like ...

Sgt. Pepper's Lonely Hearts Club Band

The Beatles

(1,192 customer reviews) | More about this product

List Price: ~~$18.98~~

Price: $10.99 & eligible for free shipping with **Amazon Prime**

You Save: $7.99 (42%)

In Stock.

Ships from and sold by **Amazon.com**. Gift-wrap available.

Want it delivered Monday, January 26? Order it in the next 6 hours and 5 min at checkout. Details

60 new from $8.15   41 used from $7.24   20 collectible from $18.98

Customers Who Bought This Item Also Bought

Abbey Road ~ The Beatles
(1,106) $13.99

Help! [UK] ~ The Beatles
(240) $14.99

Please Please Me ~ The Beatles
(229) $14.99

With the Beatles ~ The Beatles
(186) $13.97

The Beatles 1 ~ The B
(1,144) $12.

The Napoleon Dynamite Problem

Some items are not easy to categorize

1 of 8 people found the following review helpful:

★☆☆☆☆ **Pure Garbage**, January 20, 2009

By [Tristan Briggs](#) - [See all my reviews](#)

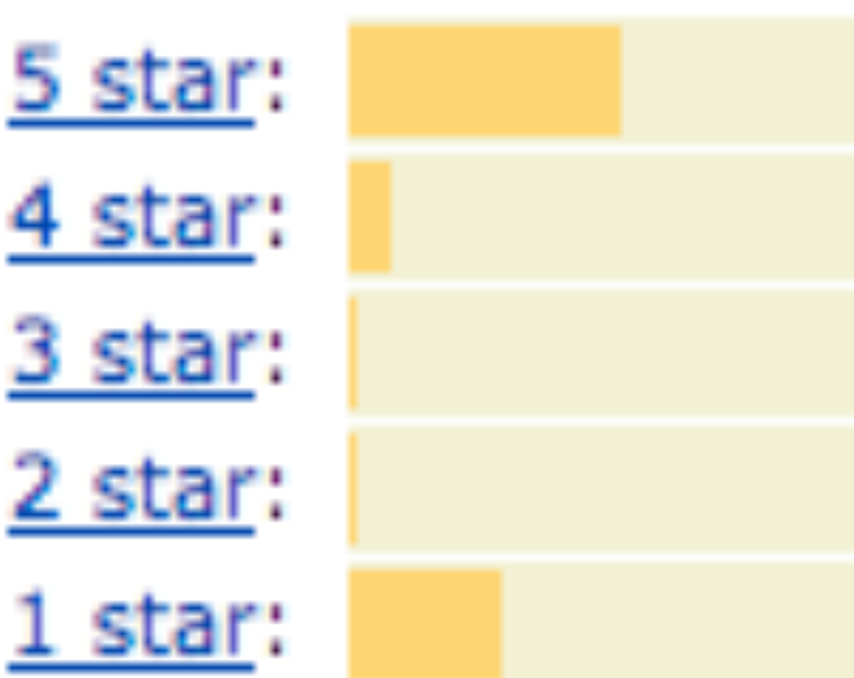
REAL NAME™

2 of 2 people found the following review helpful:

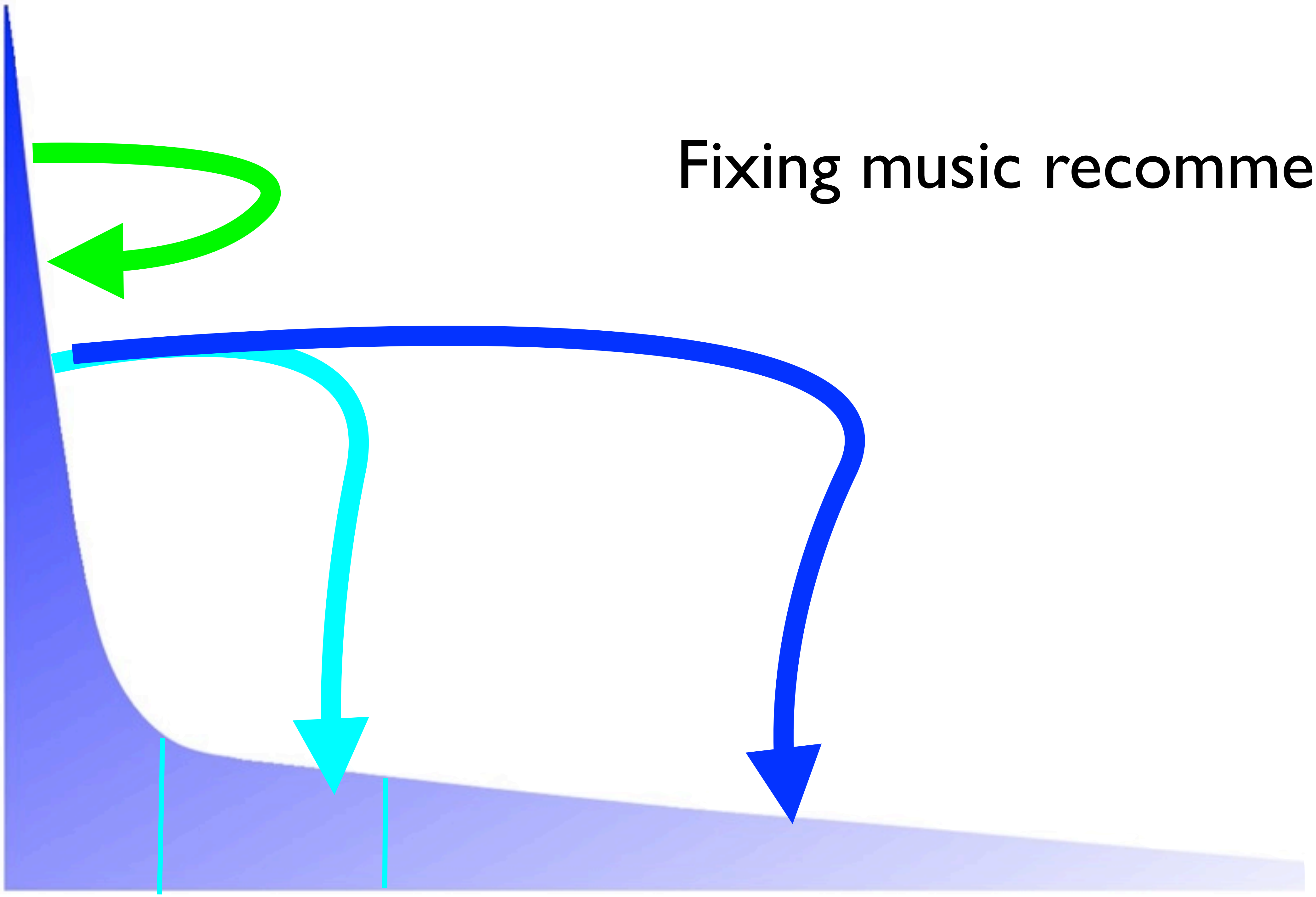
★★★★★ **unique and funny**, January 14, 2009

By [B. Helm "celticboy10"](#) (new orleans) - [See all my reviews](#)

REAL NAME™



Help! My iPod thinks I'm emo



Fixing music recommendation

Fixing music recommendation

Eliminating popularity bias and feedback loops

Adam Paul Tim Eric Jim Brian
Aaron Peter Chris Liz Yury Todd
Bethe Kirk Erik Jason Tristan Sue Jen Todd
Cari Scotty Erik Jason

Fixing music recommendation

Semantic-based recommendation

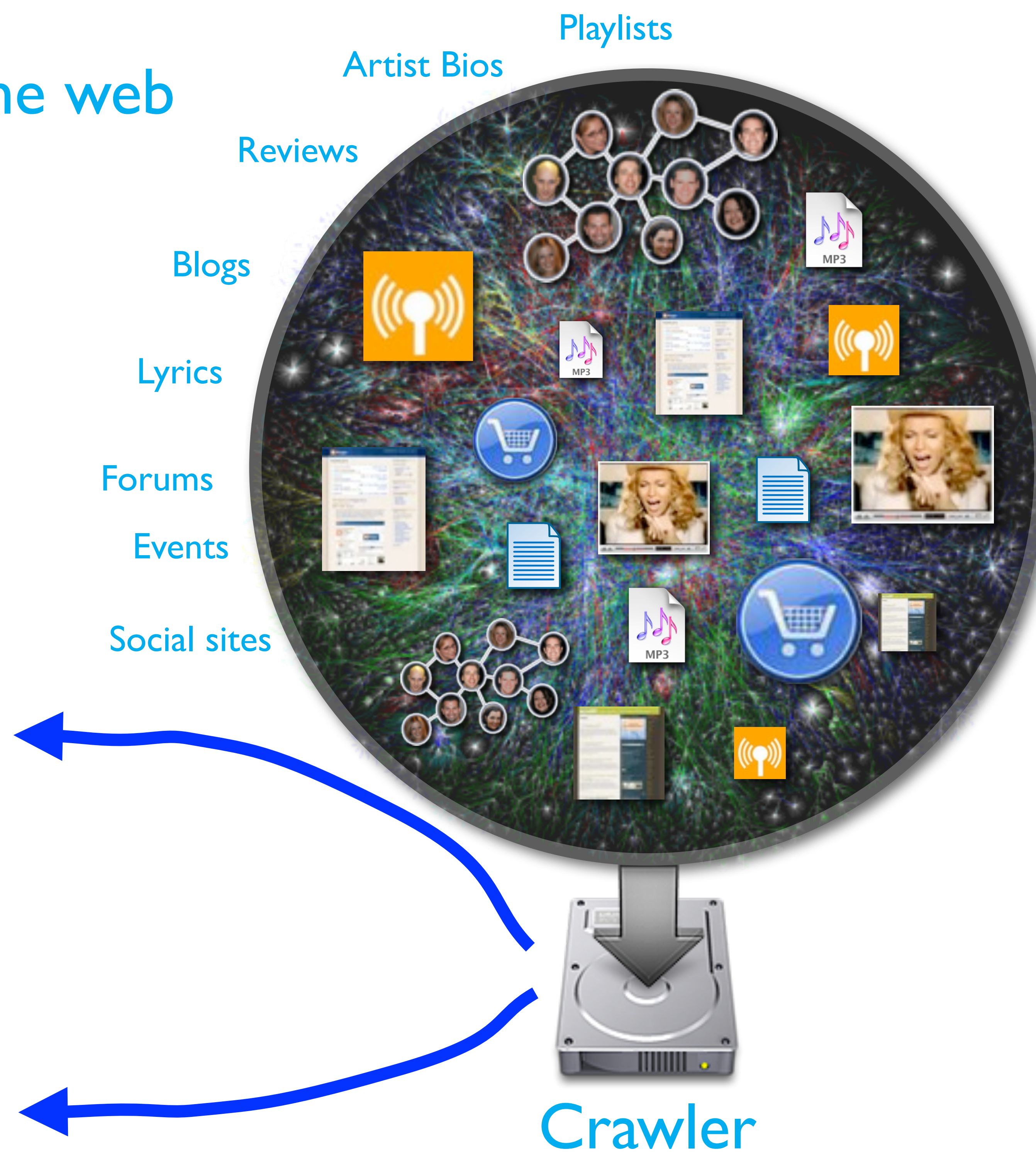
pop legend dance diva sexy
american guilty pleasure
90s teen pop 00s rnb pop
rock rock singer-songwriter dance-pop soul
emo hot alternative 90s

Fixing music recommendation

Where does this information come from?

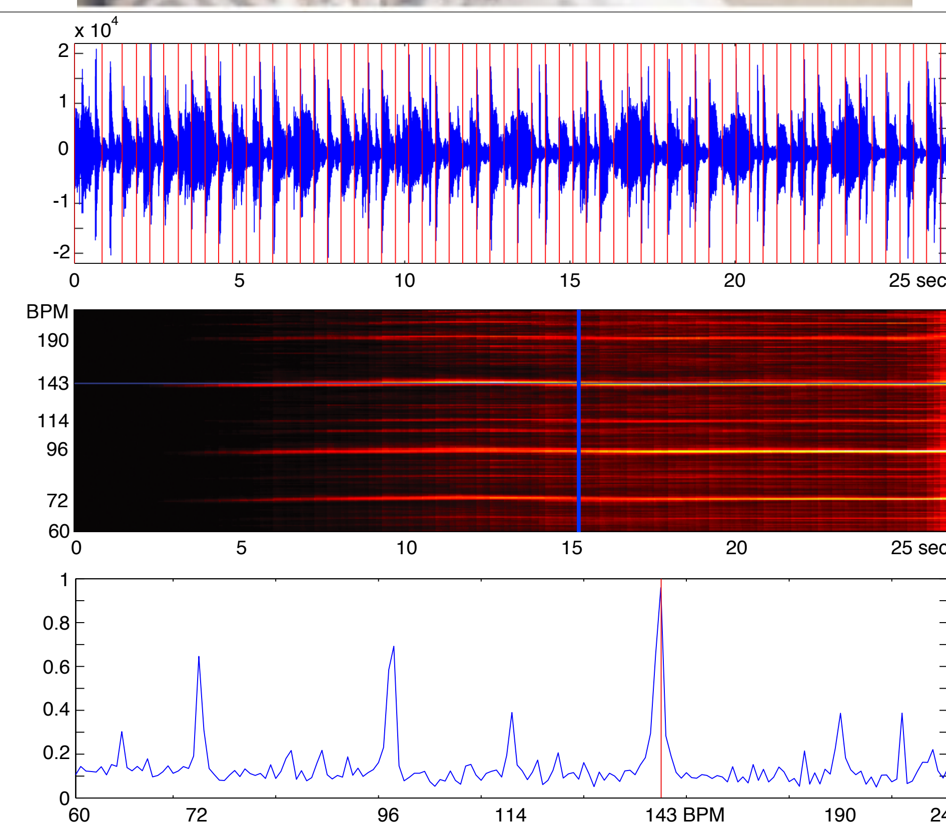
pop legend dance diva sexy american guilty pleasure 90s teen pop 00s rnb pop rock rock singer-songwriter dance-pop soul emo hot alternative 90s

female fronted metal dark rock alternative goth metal metal goth rock emo gothic dark gothic rock heavy metal gothic metal hard rock melodic metal symphonic metal rock metal pop nu



Fixing music recommendation

Content-based recommendation

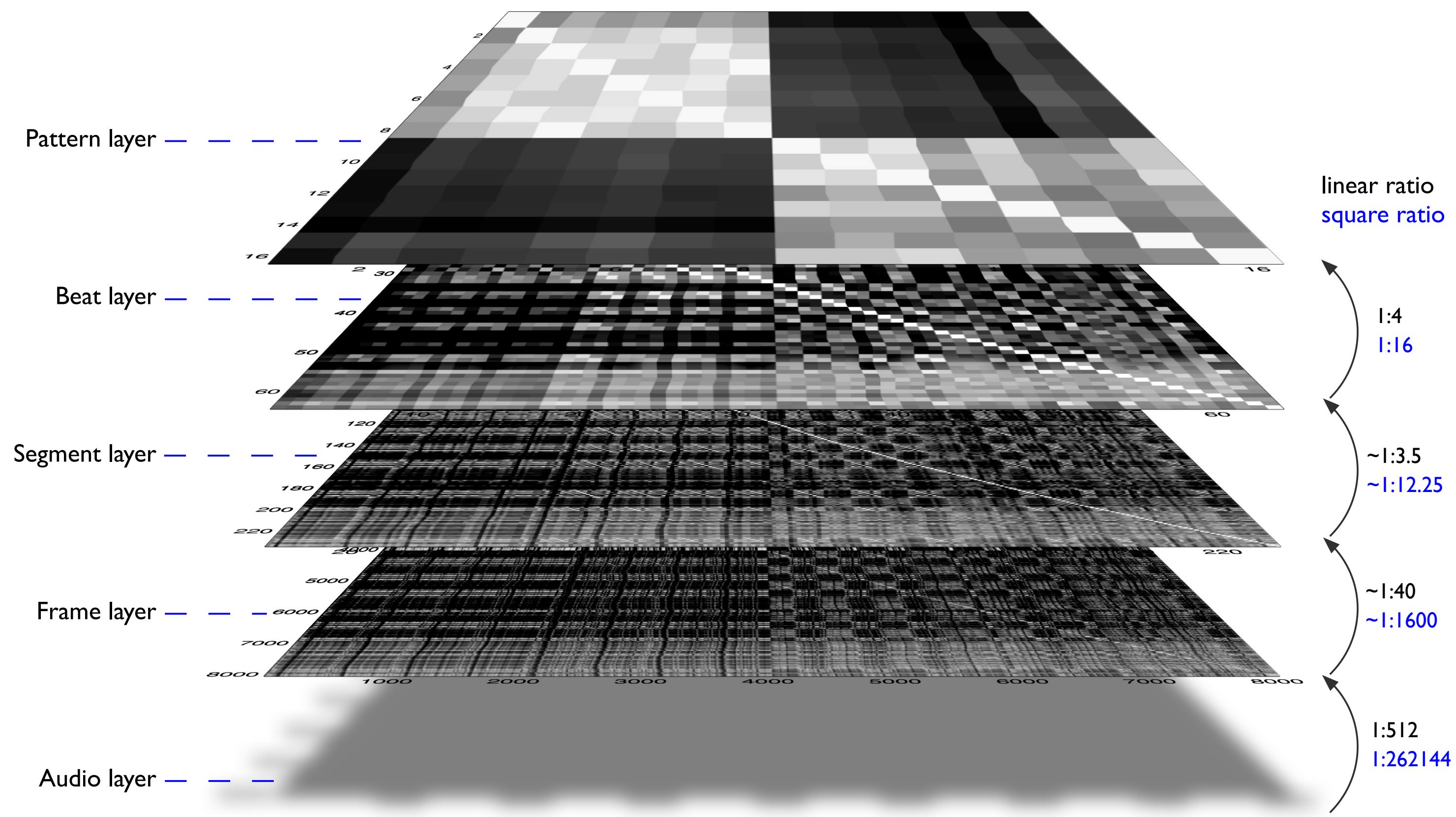


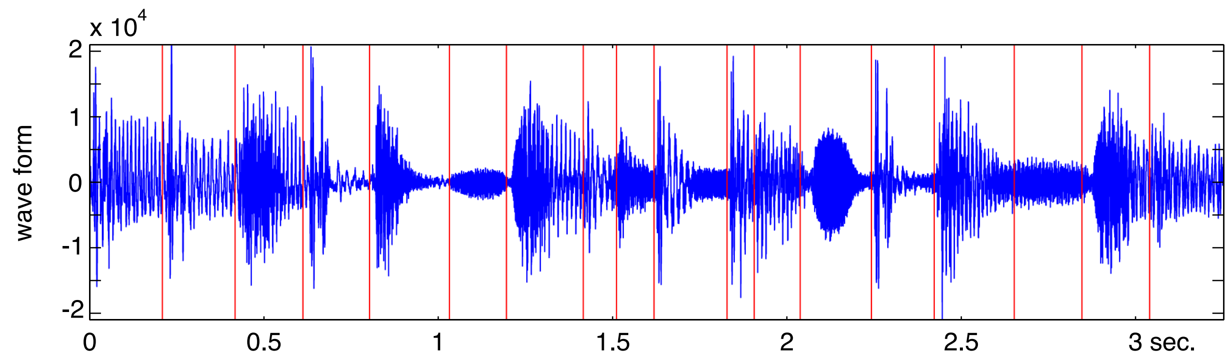
Content-based recommendation

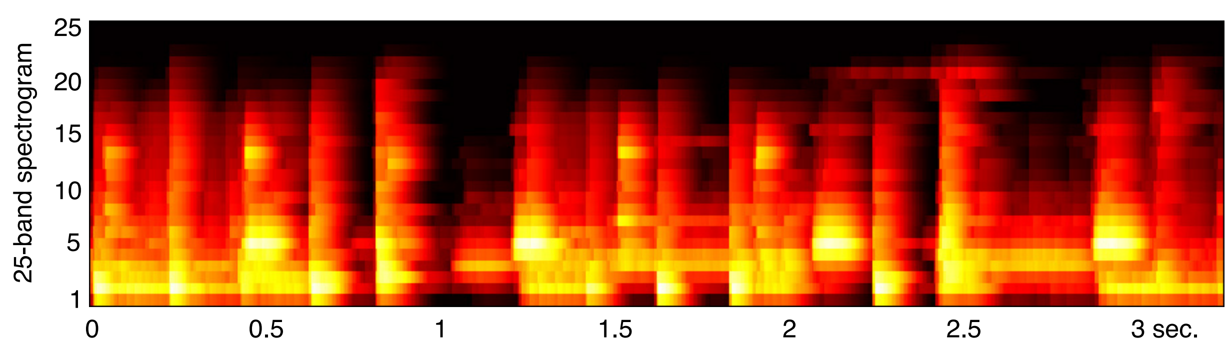
Using machines to listen to music

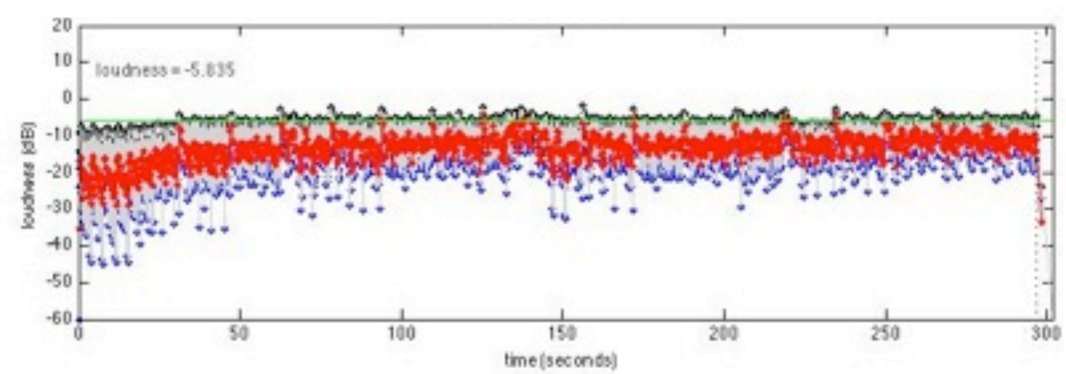
Perceptual features audio:

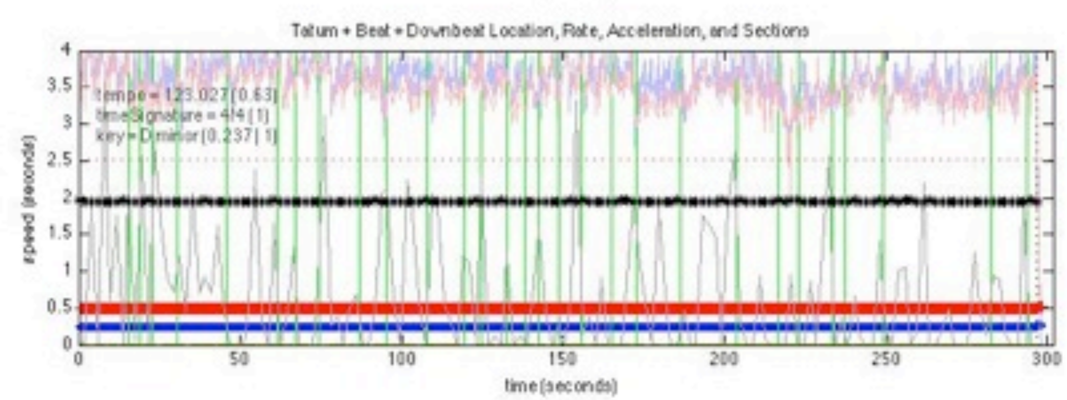
- time signature / tempo
- key/ mode
- timbre
- pitch
- loudness
- structure











Hybrid Recommendation

The best of all worlds

listener data

user history | preference | editorial

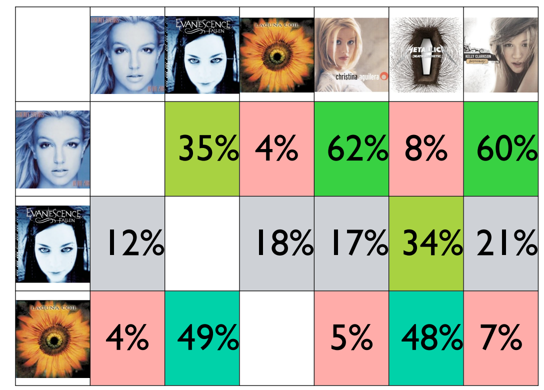
audio data

audio analysis

web data

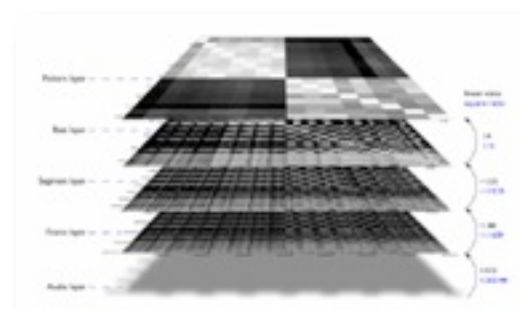
cultural analysis

Hybrid Recommender



		35%	4%	62%	8%	60%
	12%		18%	17%	34%	21%
	4%	49%		5%	48%	7%

collaborative filterer



content-based

adj Term	K-L bits	np Term	K-L bits
aggressive	0.0034	reverb	0.0064
softer	0.0030	the noise	0.0051
synthetic	0.0029	new wave	0.0039
punk	0.0024	elvis costello	0.0036
sleepy	0.0022	the mud	0.0032
funky	0.0020	his guitar	0.0029
noisy	0.0020	guitar bass and drums	0.0027
angular	0.0016	instrumentals	0.0021
acoustic	0.0015	melancholy	0.0020
romantic	0.0014	three chords	0.0019

semantic-based

recommendations

artist

track

user

Counterpoint: Ignore content. Look at users instead

- Malcolm Slaney (Yahoo)
- Using content *hurts* performance.
- Evidence: Netflix competition.

An email exchange on Music-IR

[MUSIC-IR] Recommendations using Music Content Data Inbox | X

Malcolm Slaney to music-ir show details 12:35 AM (15 hours ago) Reply

English > French **Translate message**

I had a number of conversations with people at ISMIR about the use of music content data to improve recommendations. I don't remember who I was talking to, but I thought it was worthwhile to update people with the final outcome.

The winning entries for the Netflix (movie) recommendation contest did NOT use any content data. The names of the movies in the Netflix dataset were known, and many people did try to use content data at the start of the competition. But the final winning entry did NOT include any measures of the content. The winning entry included lots of other data, but not content.

A blog posting talking about this result is available at
http://pragmatictheory.blogspot.com/2008/08/you-want-truth-you-cant-handle-truth.html

It is worth noting that just because the movie people couldn't benefit from content data, it doesn't mean there isn't value in music-content data. Three minutes of music is easier to summarize than 120 minutes of movie. But, I personally wouldn't bet against the accumulated wisdom of the Netflix competitors. :-)

Food for thought...

- Malcolm

P.S. A very readable explanation of what it took to (almost) win is online at
http://www.research.att.com/%7Evolinsky/papers/ieeecomputer.pdf
(The algorithms of the final solution are similar, but involve lots of boosting and many more types of underlying regressors....)

Jeremy Pickens Actually, leaving aside the content vs. machine learni 2:56 AM (13 hours ago)

Jeremy Pickens --- On Mon, 11/16/09, Jeremy Pickens <hostxeng@y: 2:58 AM (13 hours ago)

Matt Hoffman What, so content-based methods don't involve machine l 8:35 AM (7 hours ago)

Paul Lamere Malcom and all: Interesting, but note that the Netflix prize 8:58 AM (7 hours ago)

xavier@amatriain.net (Thanks for the plug Jeremy, wouldn't have done 8:06 AM (8 hours ago)

Douglas Eck I agree. Machine learning is everywhere. Bow down to you 9:11 AM (7 hours ago)

Brian Whitman to music-ir show details 8:51 AM (7 hours ago) Reply

English > French Translate message

On Mon, Nov 16, 2009 at 12:35 AM, Malcolm Slaney <malcolm@ieee.org> wrote:

> I had a number of conversations with people at ISMIR about the use of music content data to improve recommendations. I don't remember who I was talking to, but I thought it was worthwhile to update people with the final outcome.
>
> The winning entries for the Netflix (movie) recommendation contest did NOT use any content data. The names of the movies in the Netflix dataset were known, and many people did try to use content data at the start of the competition. But the final winning entry did NOT include any measures of the content. The winning entry included lots of other data, but not content.

As I understood it every single movie in the Netflix prize had usage data associated with it. And there was only 17,000 movies or so listed.

So let's assume a Netflix 2 (or 3) prize that better maps to what music recommenders are actually up against in the real world -- a list of 6-10 million movie titles with about half of them having no metadata or usage data at all. A good 2 million of them have chinese character set titles with no other metadata. What then?

Tristan Jehan — Since the Netflix contest data never changed, it is likely — 10:24 AM (6 hours ago)

Douglas TURNBULL — (A preemptive apology for this research self-plug — 10:30 AM (6 hours ago)

Matt Hoffman — Overfitting is certainly something to be careful about, bu — 10:56 AM (5 hours ago)

xavier@amatriain.net — You are right about how the prize was structure — 10:59 AM (5 hours ago)

Malcolm Slaney to music-ir — show details 11:34 AM (5 hours ago) — Reply

English > French — Translate message

On Nov 16, 2009, at 6:11 AM, one otherwise VERY smart Music-IR researcher wrote :-)

> I believe any content based method can be improved using good collaborative data and vice versa.

The Netflix competition suggests that statement is wrong. A pure machine-learning approach says that all information is good. But the evidence in the Netflix competition says that the content-based signals tried so far added more noise than signal :-(

Paul Lamere and Brian Whitman point out that the cold-start problem can probably benefit from content data. That is a good point about a hard problem.

But there are other ways to solve the cold-start problem, without using the content, based on machine learning and exploration. The paper below talks about how to solve the cold-start problem by using exploration. New news stories for the Yahoo front page enter the system more often than new music releases, and news stories take less time to consume than music. One solution is to try the content out on a small fraction of the users and see who likes it.

http://research.yahoo.com/pub/2963

(I'm sorry, the paper isn't online yet.. send me email and I can send you a preprint.)

I don't want to say that content measures will never help recommendation systems. But so far the large-scale evidence (i.e. the Netflix competition) says content doesn't help. I think a paper that wants to suggest otherwise will have a much harder time establishing that the approach is interesting. Millions of users seem to be smarter than FFTs :-)

Bryan Pardo to music-ir show details 12:01 PM (4 hours ago) Reply

English > French **Translate message**

Hi all,

I can't resist throwing in my $.02 on this issue.   Let me cast this in a different light:

Imagine doing low-level frame-based feature measurement of stories broadcast on the US's National Public Radio. Not speech-to-text. I mean things like mfcc, chroma, spectral centroid, RMS amplitude. You know. The features we all use.   Now try differentiating between interesting stories and uninteresting ones. You might be able to tell different radio shows (Cartalk vs This American Life) based on talker identification using the mfccs. To that extent you might be successful in finding good things to recommend, but separating the INTERESTING Cartalk episode from the UNINTERESTING one? Not likely.

Why not? Because simple frame-based features don't capture the structures that we process to decide what makes something interesting. Not in speech. Not in music. So of course collaborative filtering works better. The filters (people) are actually basing their ratings on attention to what at least one human cares about in the signal.

My belief is that content-based recommendation systems will only reach the next level when they can identify things about a recording like this:

The singing style is similar to Tom Waits
This is an up-tempo polka
The lyrical content is about puppies
The instruments are all jugs (bottles you blow in).
The song structure is blues

When we content-based have features like that, we're finally talking in terms that might (in conjunction with meta data like record label, year recorded, etc) give meaningful recommendations.

Bryan Pardo

J. Stephen Downie I am with BP on this. Cheers, Stephen ************** 12:12 PM (4 hours ago)

Gert Lanckriet Having looked at the content-based versus collaborativ 12:22 PM (4 hours ago)

Jeremy Pickens Sorry, let me clarify: -Content-based machine learning 1:34 PM (3 hours ago)

Tristan Jehan to music-ir show details 1:52 PM (2 hours ago) Reply

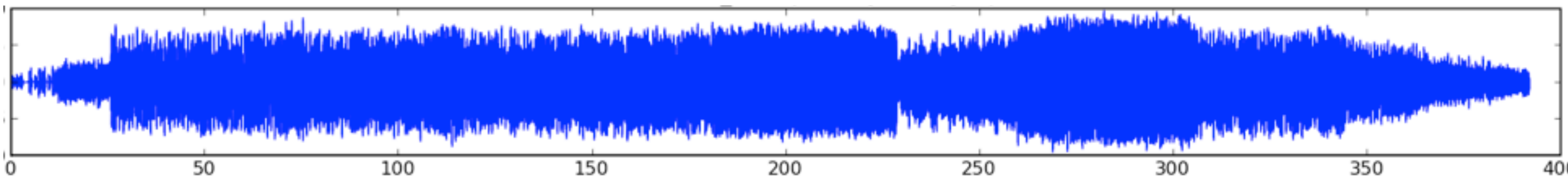
English > French Translate message

CF has limitations by design. Content-based similarity has limitations by the quality of the analysis and the combining of features: it's only a matter of time.

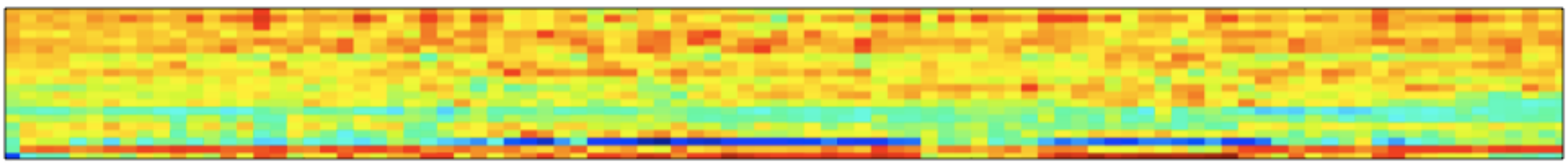
T

Anatomy of an Autotagger

Our approach: content-based music recommendation



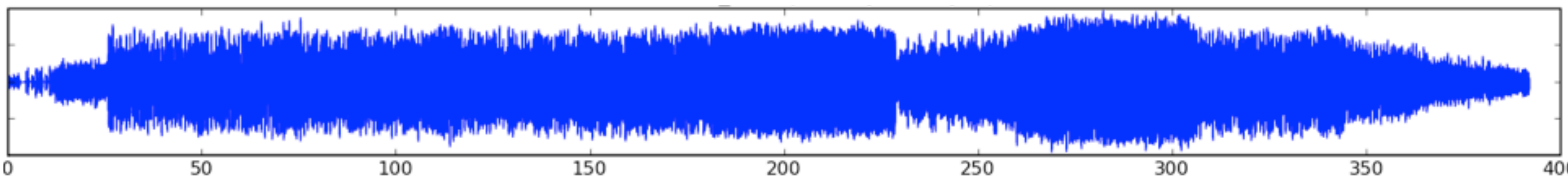
Acoustic feature extraction



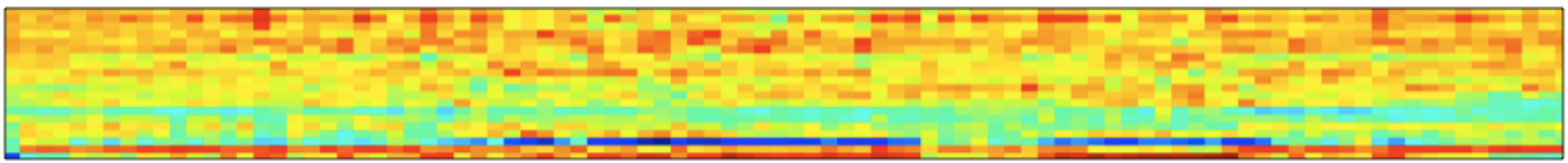
Machine Learning

"I hear 1970s glam rock. It's David Bowie, but with a harder punk edge, like the Clash, but wearing platform shoes and silk jumpsuits."

Our approach: content-based music recommendation



Acoustic feature extraction



Machine Learning

0.74 80s,
0.71 glam
0.69 70s

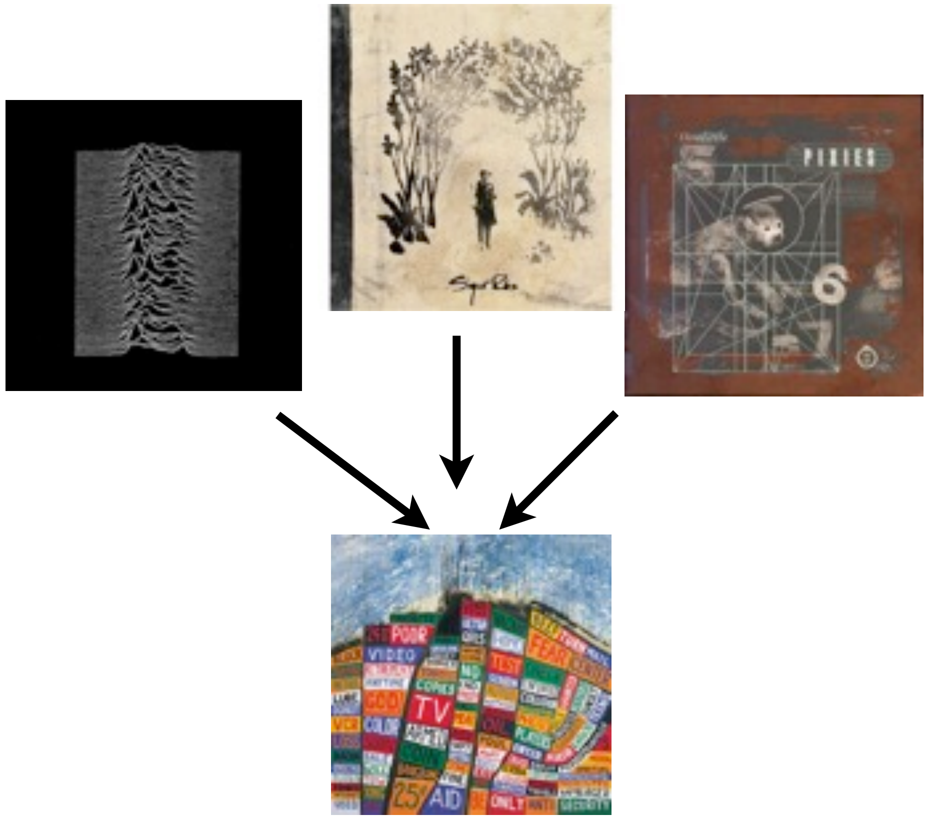
0.68 classic_rock
0.67 england
0.65 english

0.65 proto-punk
0.64 new_wave
0.64 glam_rock

A more realizable goal: generate tag clouds useful for annotation and retrieval

Recommendation from tags

- Annotate all tracks using Autotagger model.
- Use TF-IDF normalization to downweight overused words.
- Cosine distance over word vectors for simliarity.
- Combine autotag signal with other signals:
  - Social tags,
  - Explicit user preferences,
  - Implicit user preferences (skips, long plays)
  - Similarity among users, etc.

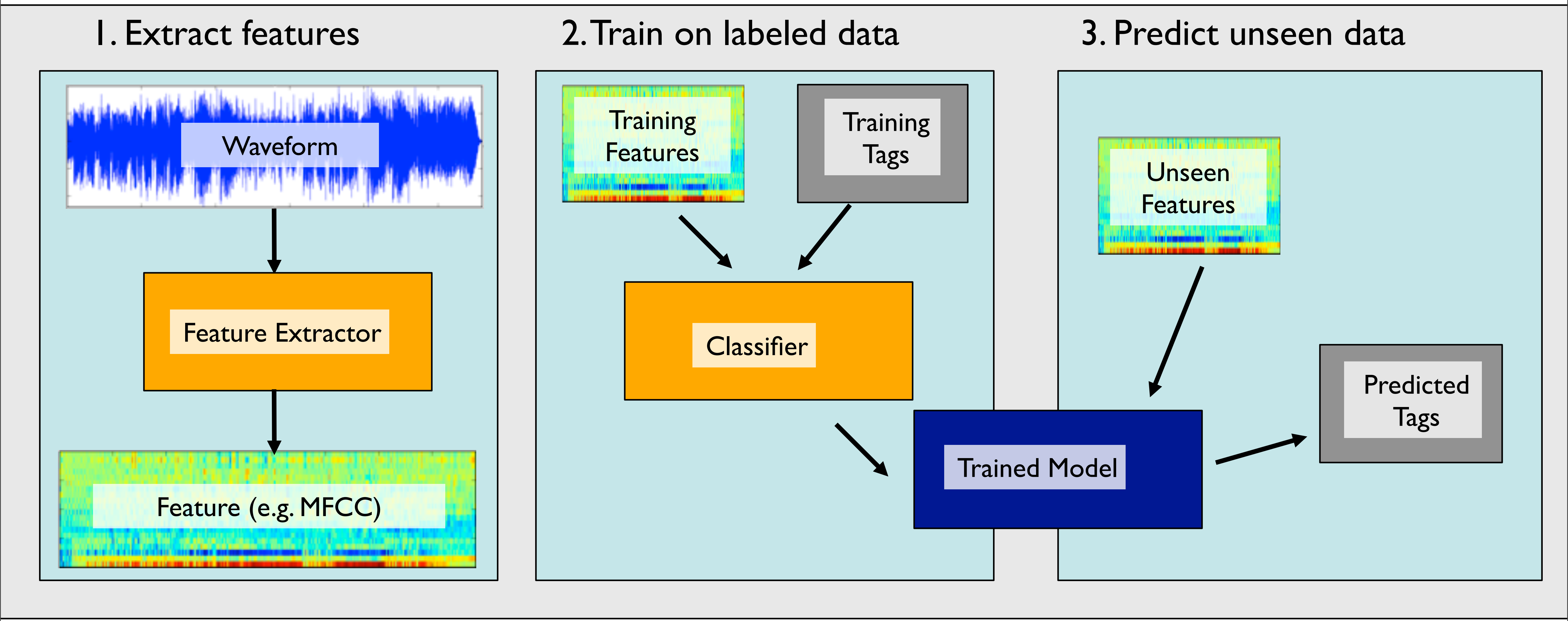


ML challenges and previous approaches

- Challenges
  - What features to use?
  - What machine learning algorithm to use?
  - How to scale to huge datasets?

- ML approaches (tag, genre and artist prediction):
  - SVM (Ellis & Mandel 2006)
  - Decision Trees (West, 2005)
  - Nearest Neighbors (Palmpalk, 2005)
  - Hierarchical Mixture Models (Turnbull et al, 2009)
  - AdaBoost / FilterBoost (our work)

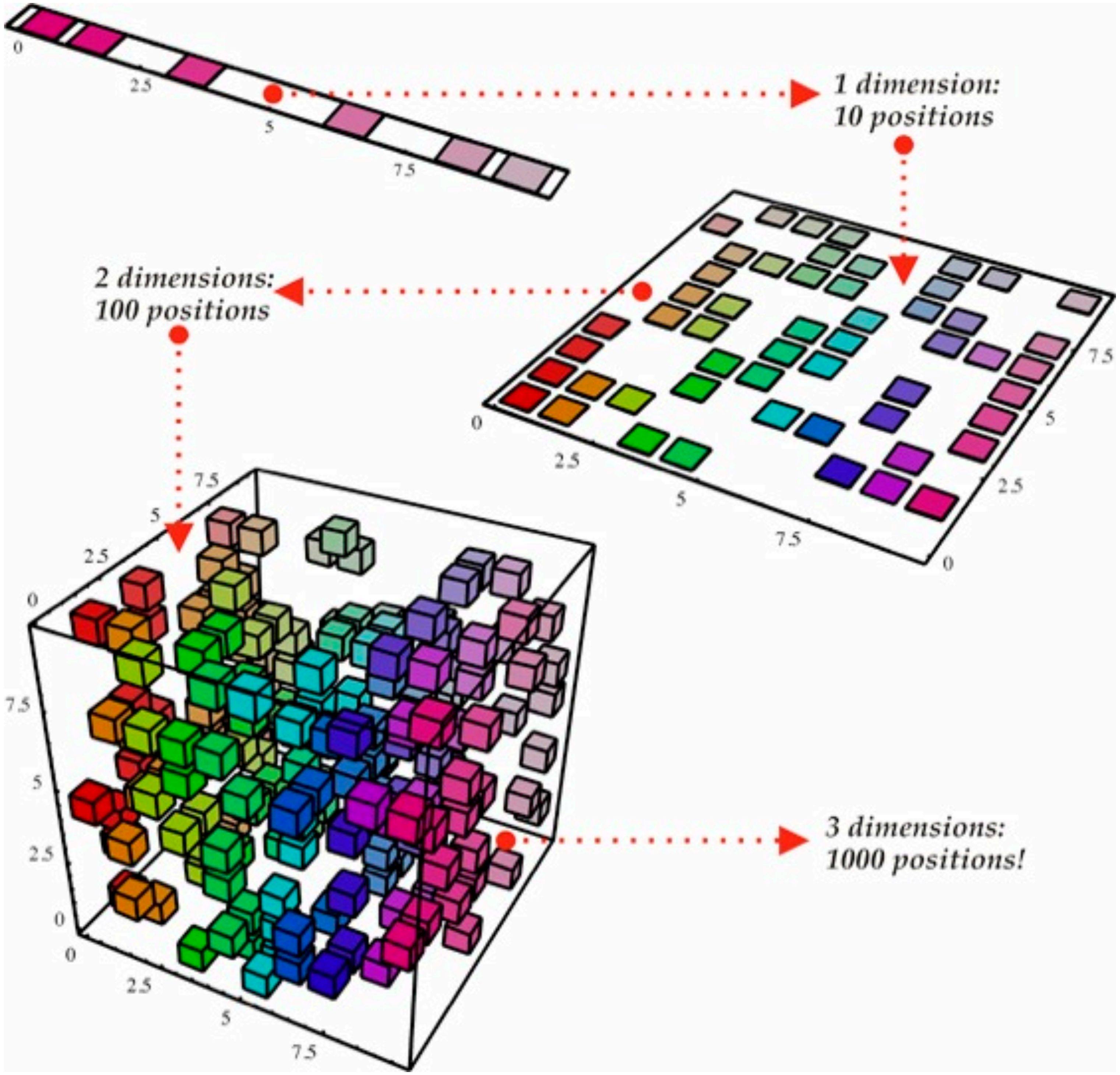
One autotagging pipeline



Audio Feature Demos

Curse of dimensionality

- A 3min stereo CD-quality audio sequence contains 254,016,000 bits (44100 * 2 * 60 * 3 * 16)
- Number of possible unique bit configurations for 3min songs : $2^{254,016,000}$
- We need to process
  >100K audio files for lab work
  >1M for commercial work

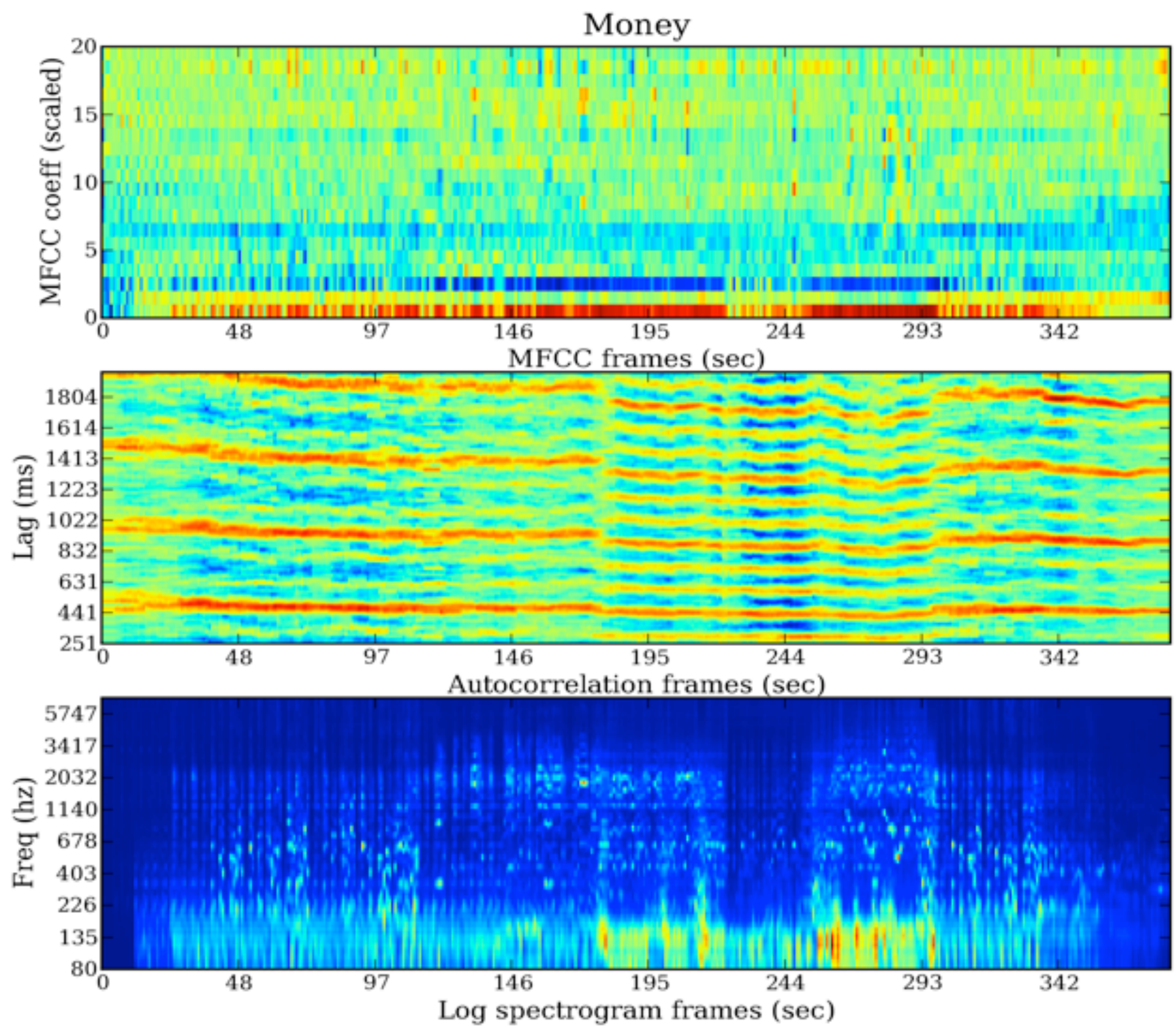


Representing different musical attributes

MFCC
Timbre / instrumentation

Autocorrelation
Temporal structure
(rhythm, meter)

Spectrogram
Pitch, melody

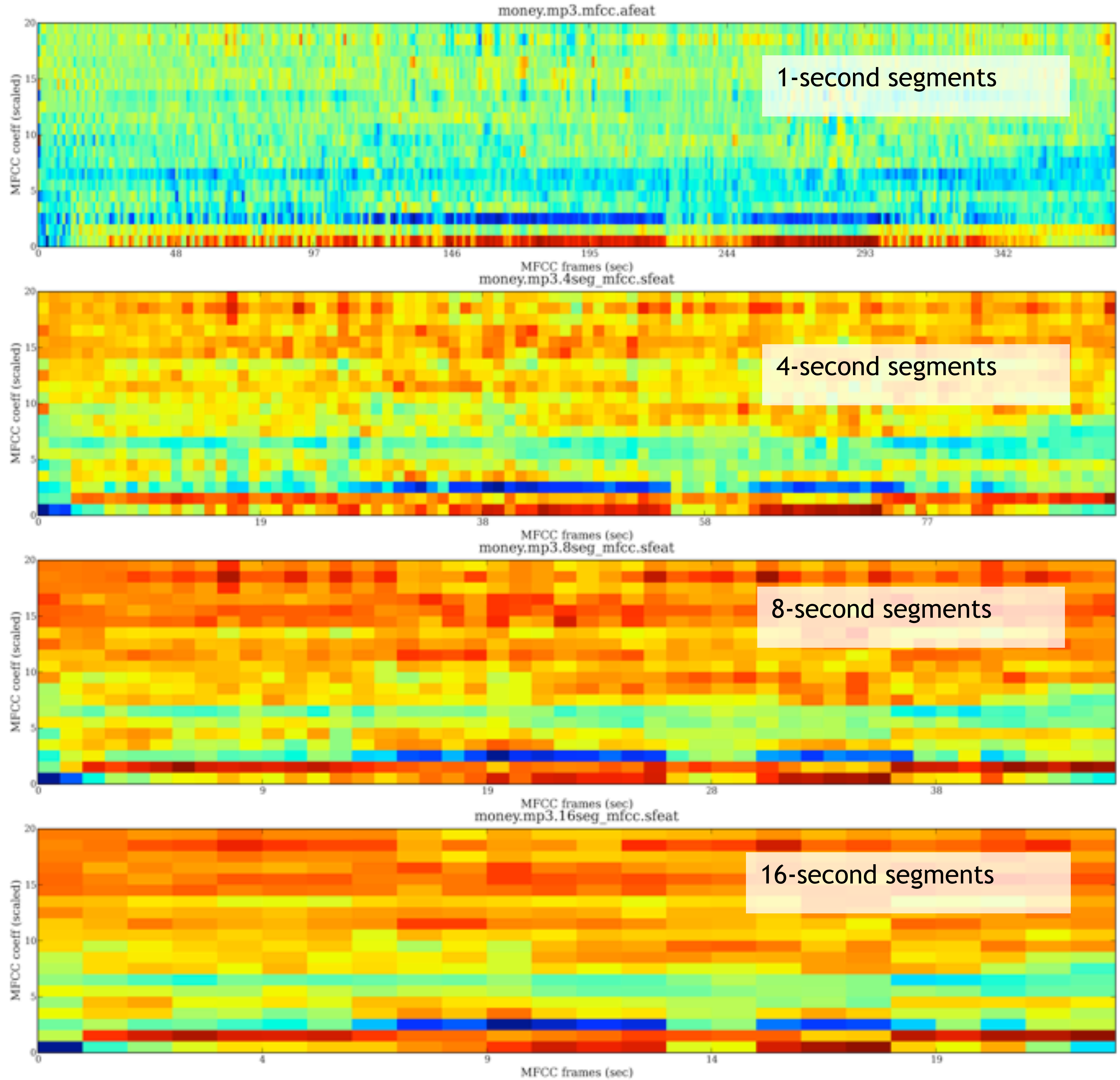


Pink Floyd "Money"

Aggregate Features

- Aggregate chunks of feature frames into longer-timescale segments
- Vote over these larger segments.
- Question: What is the best segment size?
- One answer: 3-5 seconds (Bergstra et.al.)

Pink Floyd "Money" MFCCs



Sparse coding techniques

- Example: K-Means Analysis.
- Simpler than (but similar to) a Gaussian Mixture Model

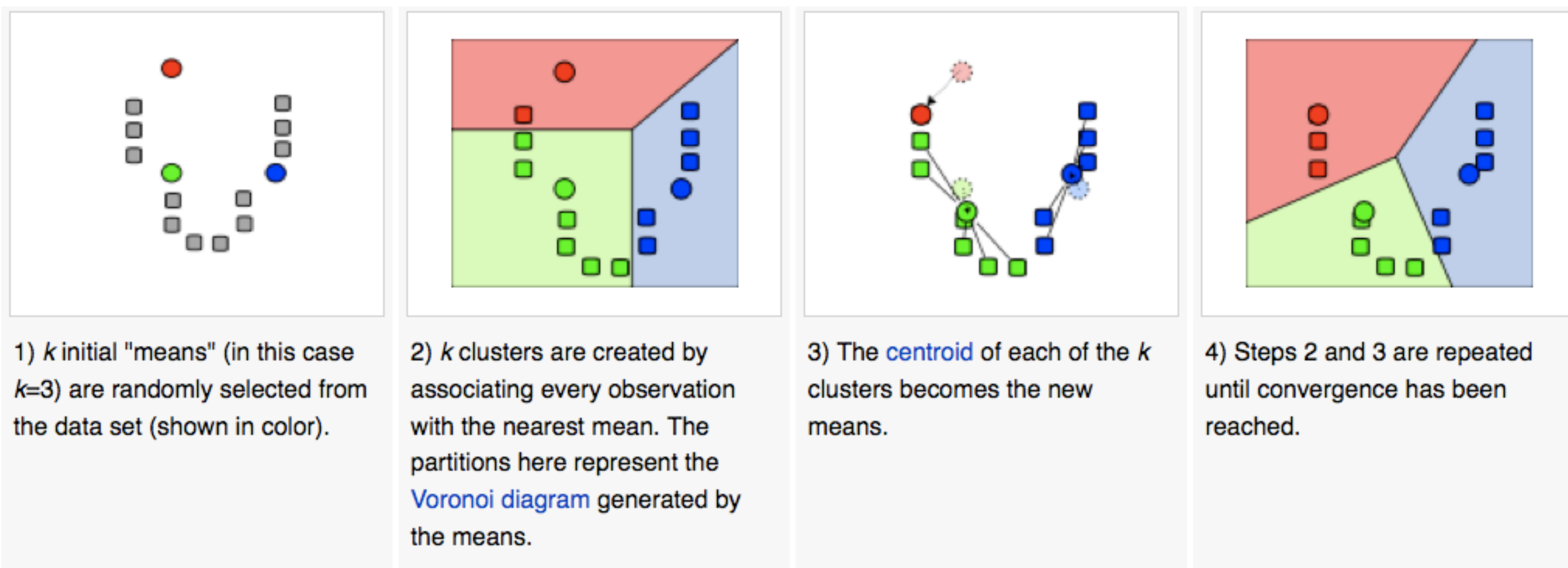
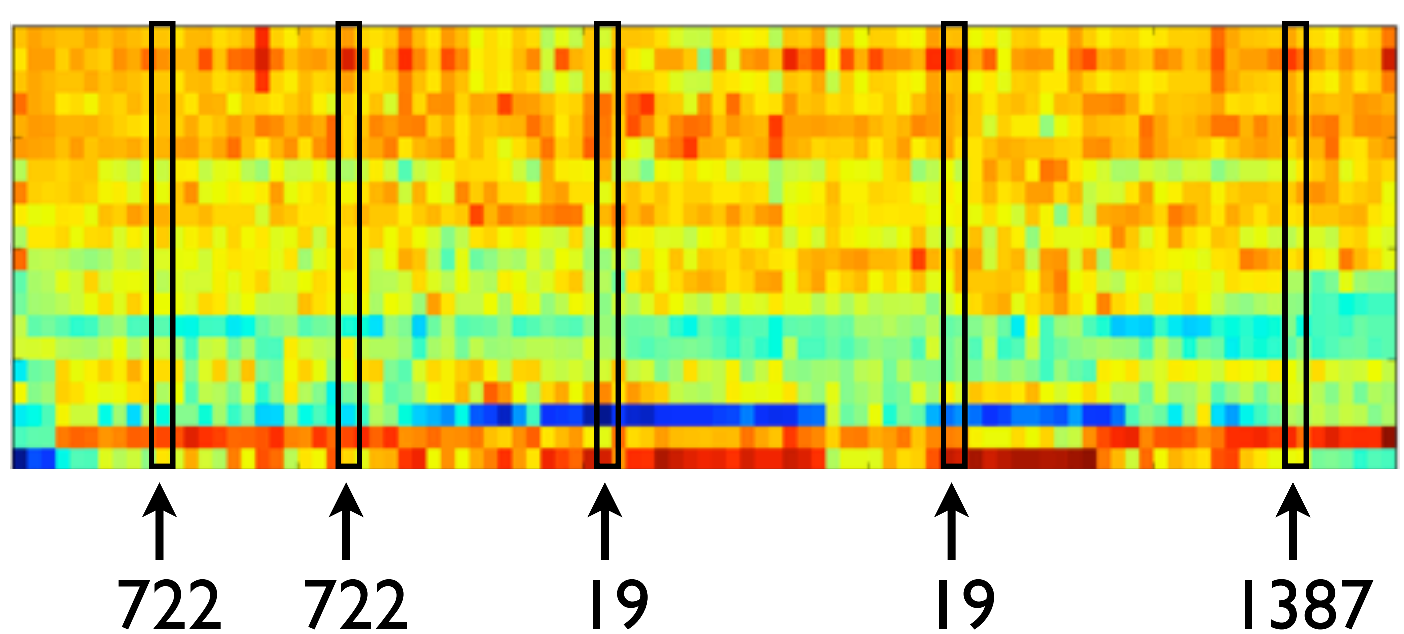


Illustration of K-means from wikipedia

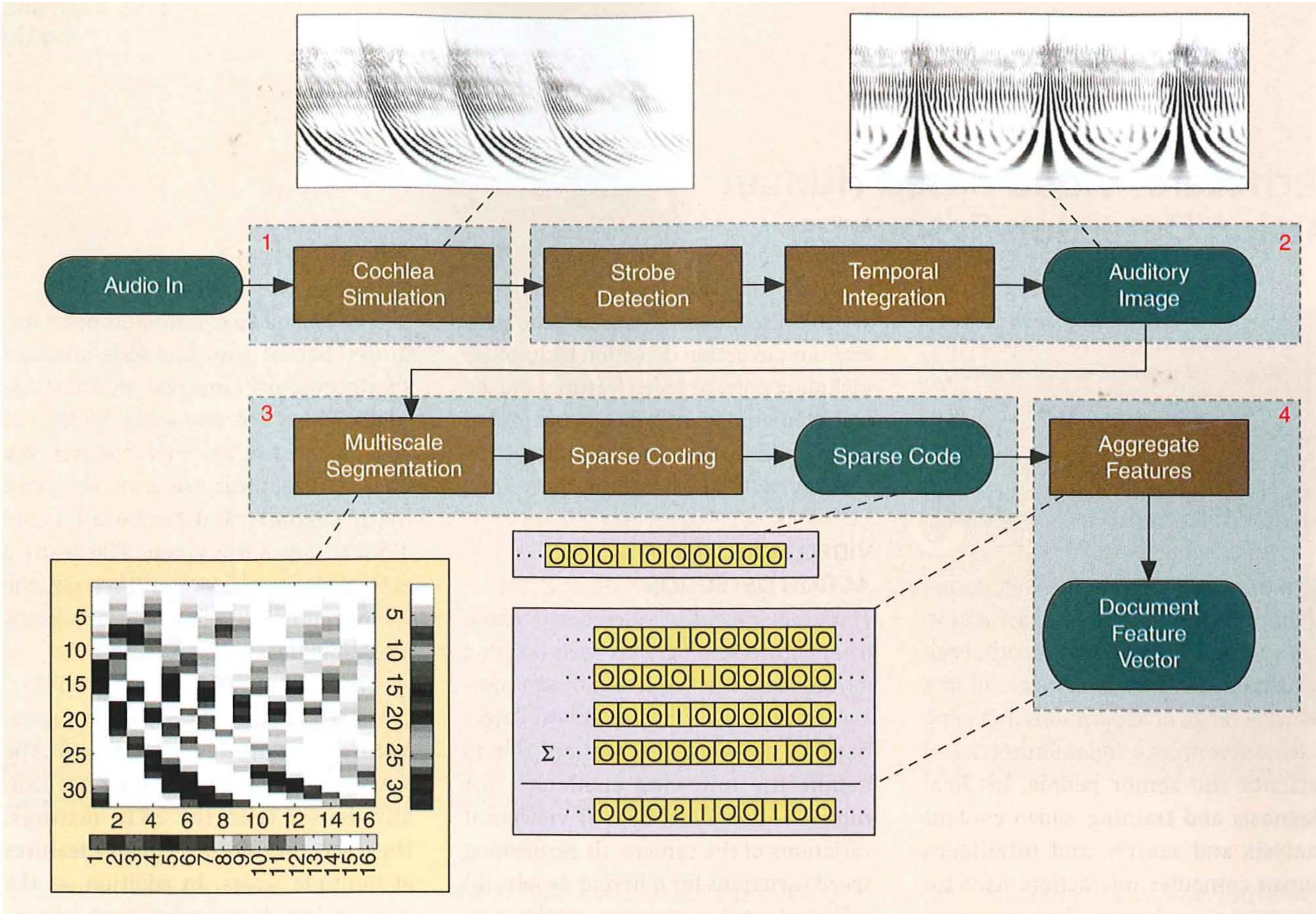
Sparse coding techniques.

- Performed k-means on MFCCs
- K=3000 / 20,000 30sec audio files
- Used to build sparse representation of audio (Bengio et al; Google)
- Song represented as a sparse histogram of frame centroids. *Extremely* sparse.
- Motivation: sparse document similarity approaches. Can a single MFCC frame function as a concept? Song is a histogram of concepts.



Representation:
[19=2, 722=2, 1387=1]

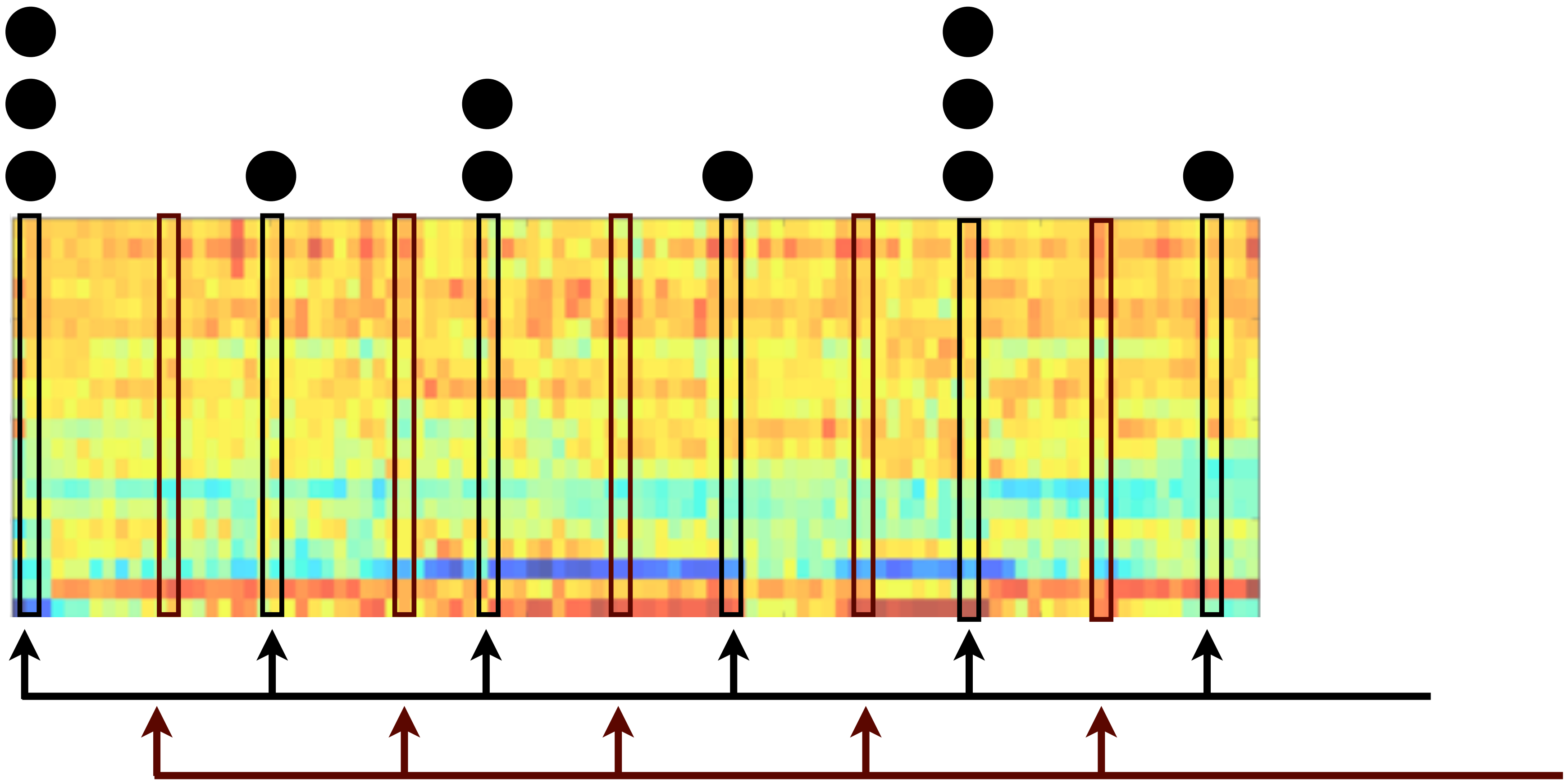
A more complete (and complex) example



[FIG4] Generating sparse codes from an "audio document," in four steps: 1) cochlea simulation, 2) stabilized auditory image creation, 3) sparse coding by vector quantization of multiscale patches, and 4) aggregation into a "bag of features" representation of the entire audio document. Steps 3 and 4 here correspond to the feature extraction module in the four-module system structure. To the fourth module, a PAMIR-based learning and retrieval system, this entire diagram represents a front end providing abstract sparse features for audio document characterization.

From "Machine Hearing: An Emerging Field"
Richard F. Lyon. IEEE Sig. Proc. Mag. Sept. 2010.

Beat-based aggregation



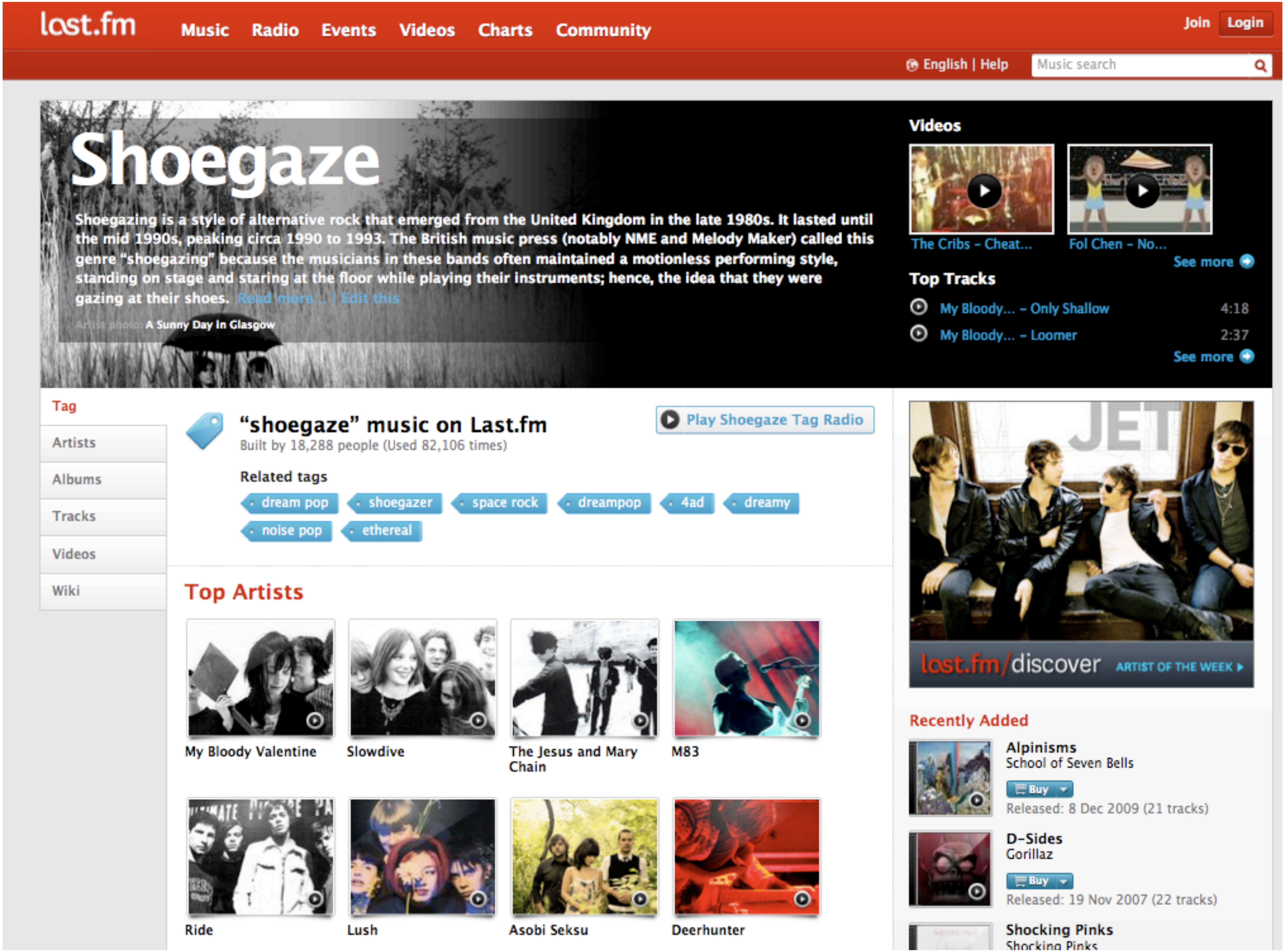
Cheap to compute and popular (e.g. Dan Ellis cover song detector).

Training data

Data source: Last.FM

- Social tags obtained via data mining (Last.fm AudioScrobbler API)
- Identified 350 most popular tags
- Mined tags and tag frequencies for nearly 100,000s artist from Last.FM
- Genre, mood, instrumentation account for 77% of tags

Tag Type	Frequency	Examples
Genre	68%	heavy metal, punk
Locale	12%	French, Seattle, NYC
Mood	5%	chill, party
Opinion	4%	love, favorite
Instrumentation	4%	piano, female vocal
Style	3%	political, humor
Misc	3%	Coldplay, composers
Personal	1%	seen live, I own it


last.fm
Music
Radio
Events
Videos
Charts
Community
Join
Login
English | Help
Music search
Shoegaze
Shoegazing is a style of alternative rock that emerged from the United Kingdom in the late 1980s. It lasted until the mid 1990s, peaking circa 1990 to 1993. The British music press (notably NME and Melody Maker) called this genre "shoegazing" because the musicians in these bands often maintained a motionless performing style, standing on stage and staring at the floor while playing their instruments; hence, the idea that they were gazing at their shoes. Read more... | Edit this
Artist photo: A Sunny Day In Glasgow
Videos
The Cribs – Cheat...
Fol Chen – No...
See more
Top Tracks
My Bloody... – Only Shallow 4:18
My Bloody... – Loomer 2:37
See more
Tag
Artists
Albums
Tracks
Videos
Wiki
"shoegaze" music on Last.fm
Built by 18,288 people (Used 82,106 times)
Play Shoegaze Tag Radio
Related tags
dream pop
shoegazer
space rock
dreampop
4ad
dreamy
noise pop
ethereal
Top Artists
My Bloody Valentine
Slowdive
The Jesus and Mary Chain
M83
Ride
Lush
Asobi Seksu
Deerhunter
JET
last.fm/discover ARTIST OF THE WEEK
Recently Added
Alpinisms
School of Seven Bells
Buy
Released: 8 Dec 2009 (21 tracks)
D-Sides
Gorillaz
Buy
Released: 19 Nov 2007 (22 tracks)
Shocking Pinks
Shocking Pinks

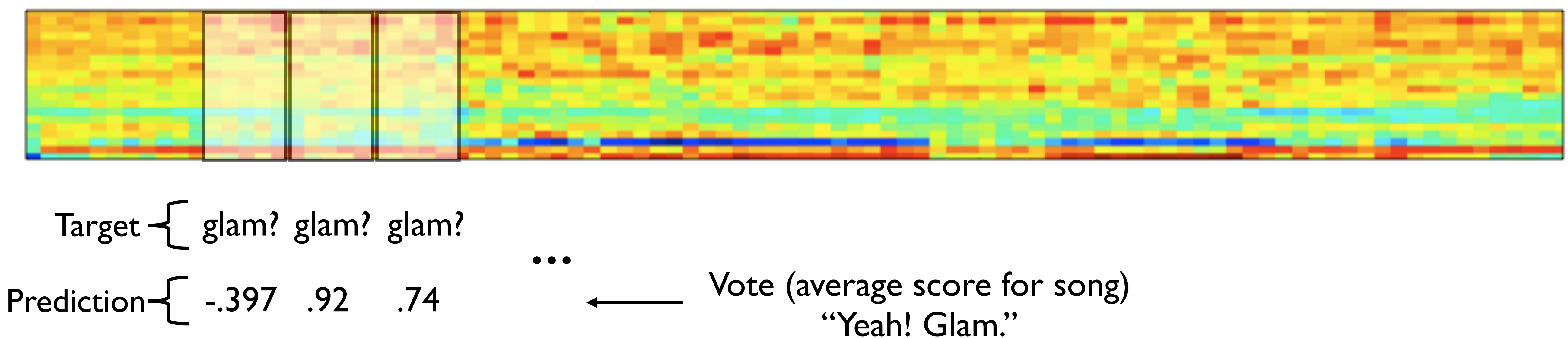
Constructing datasets

- Built list of 350 most popular tags
- Generate classification targets for each tag:
  - All songs by top 10 artists for a tag used as positive examples
  - All songs by next 200 artists ignored (uncertain)
  - All remaining songs treated as negative examples
- Matched songs to audio collection and extracted features from audio.

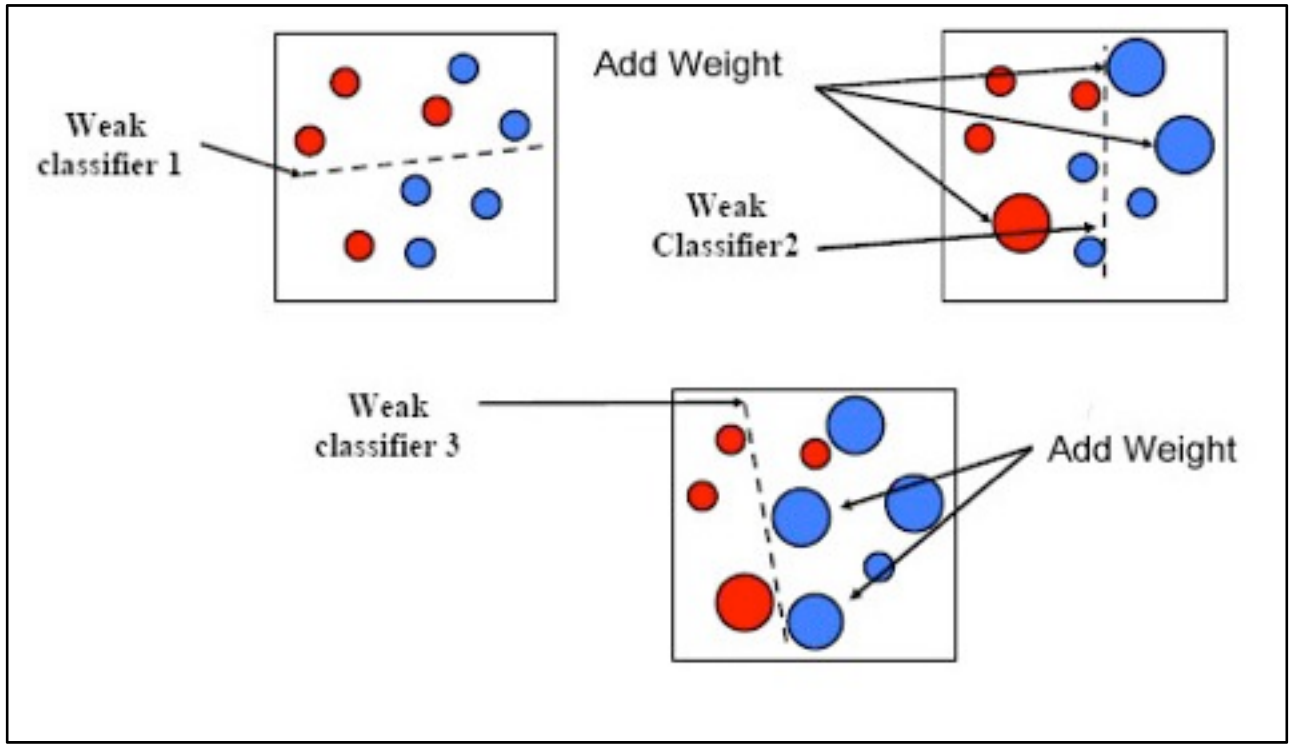
Learning details and results

Voting over blocks of features

- MFCCs calculated over timescale where audio should be steady-state (~100ms)
- MFCCs aggregated into 3 to 5sec blocks (mean, std, covariance)
- Train segments (columns) individually; all on same song-level label
- Integrate predictions over song (vote) to choose winner



Classifier



- Used AdaBoost ensemble learner (Freund & Schapire 1995)
- Basic idea:
  1) Search for best weak learner in set of learners
  2) Add it to list of active learners (store its weight and confidence)
  3) Reweight data to avoid wasting resources on points already classified
- Builds smart classifier from weighted linear combination of relatively-stupid "weak learners"
- Feature selection based on minimization of empirical error

Principle of Adaboost

- Three cobblers with their wits combined equal Zhuge Liang the master mind.
- Failure is the mother of success

$$F(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x) + \alpha_3 f_3(x) + \ldots$$

Strong classifier

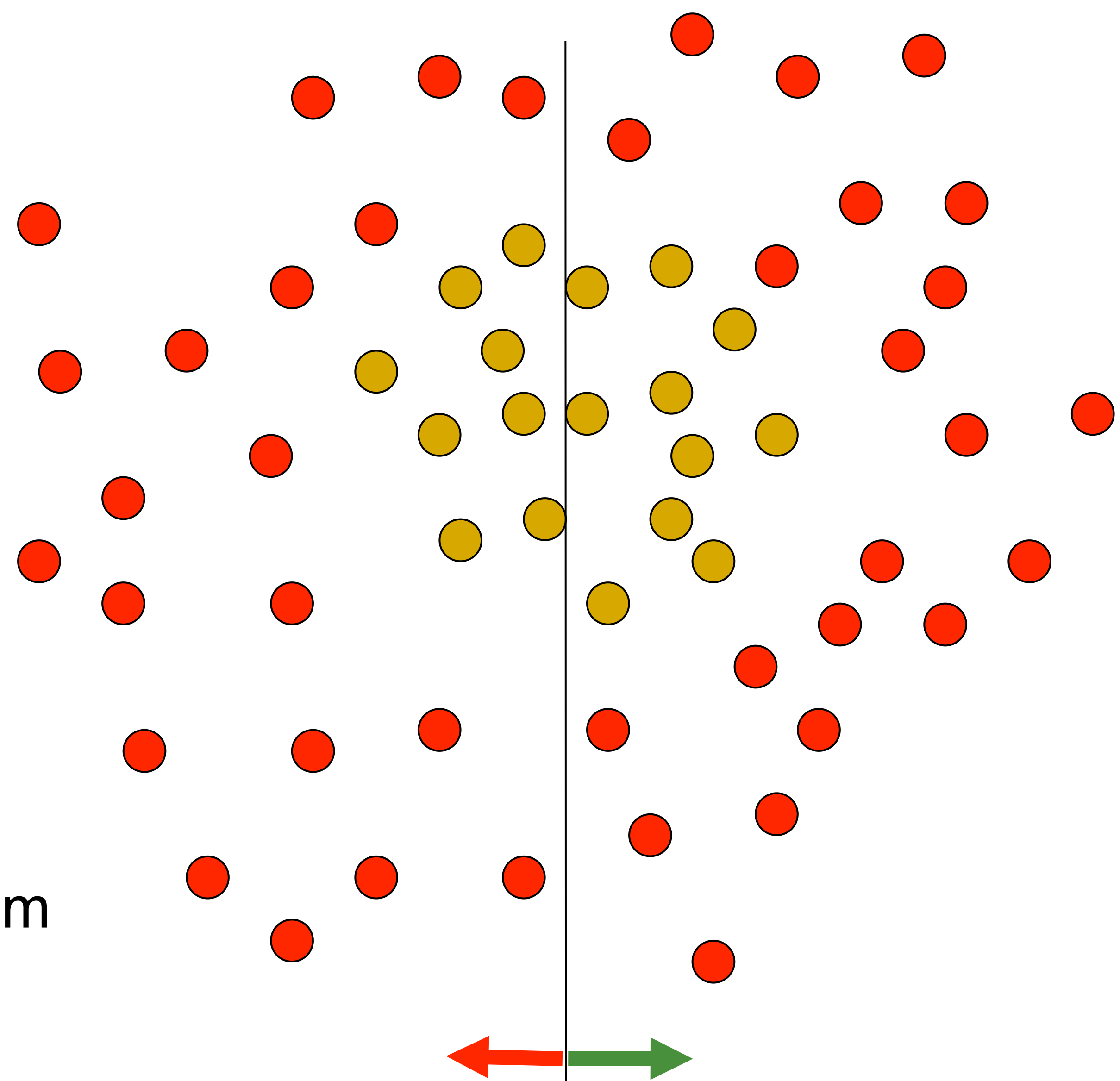
Weak classifier

Weight

Features vector

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)

Toy Example – taken from **Antonio Torralba @MIT**



Each data point has

a class label:

$$y_t = \begin{cases} +1 \; (\ ) \\ -1 \; (\ ) \end{cases}$$

and a weight:

$w_t = 1$

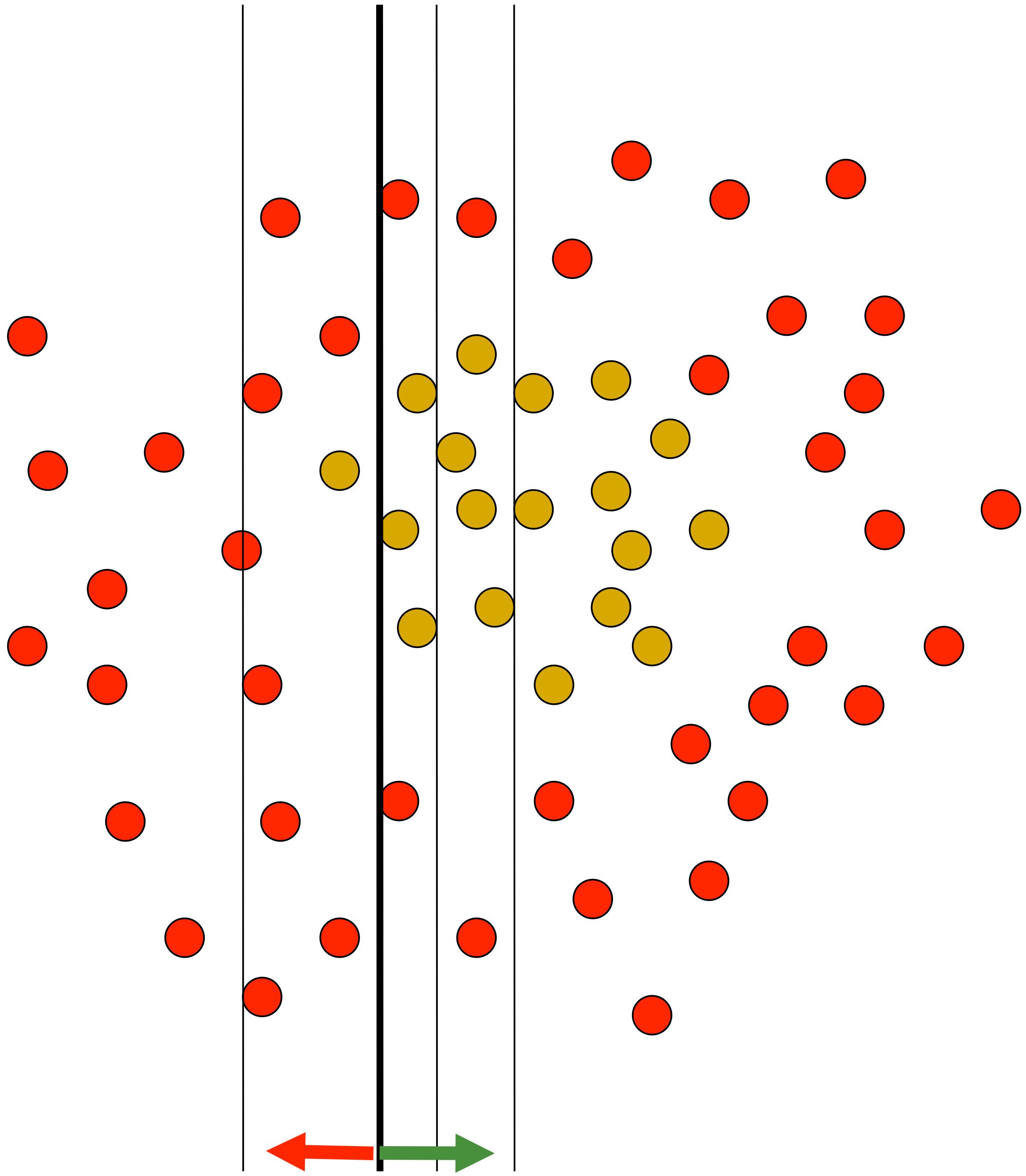
Weak learners from
the family of lines

h => p(error) = 0.5 it is at chance

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)

Toy example

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)



This one seems to be the best

Each data point has a class label:

$$y_t = \begin{cases} +1\ (\ ) \\ -1\ (\ ) \end{cases}$$

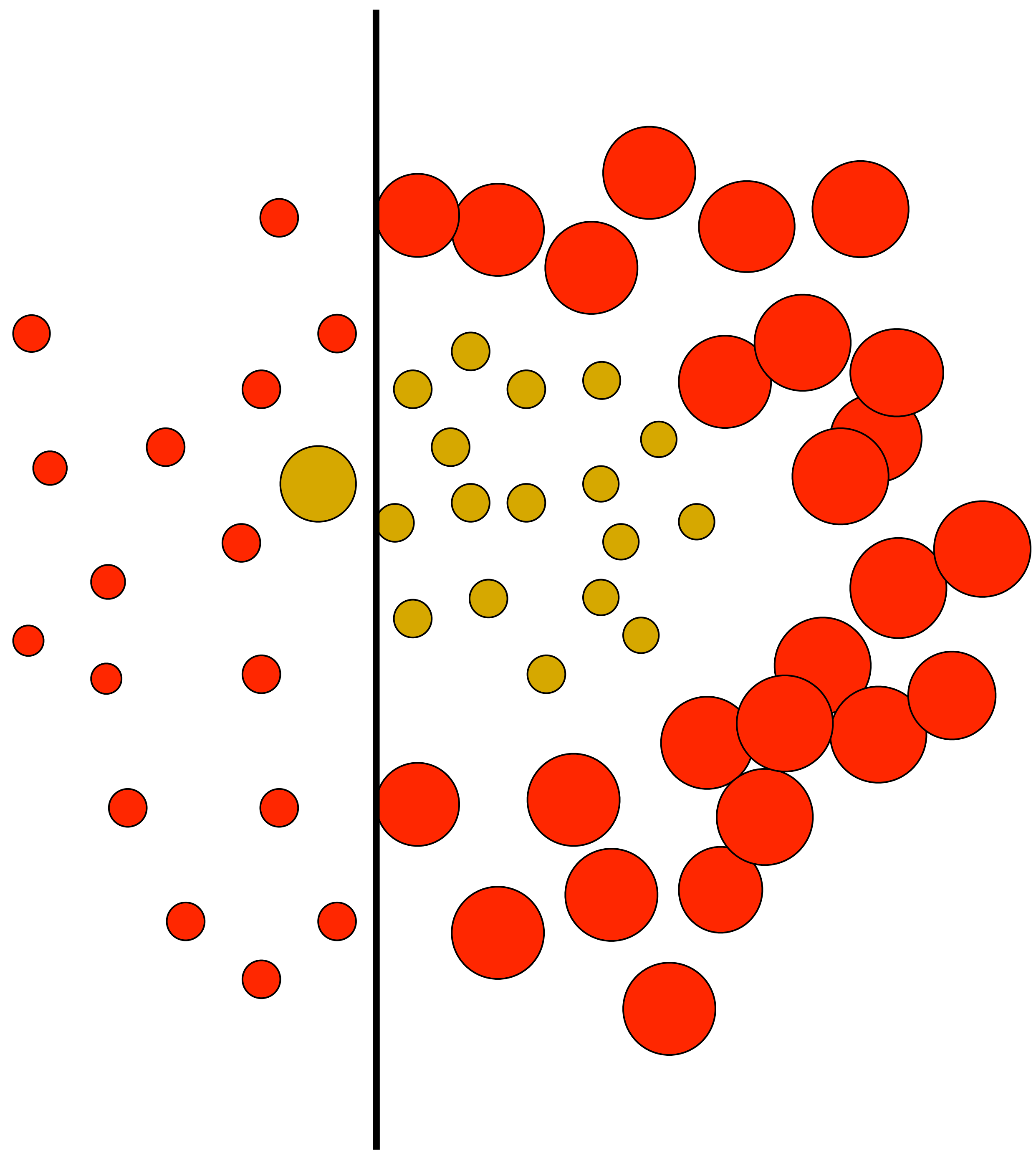
and a weight:

$w_t = 1$

This is a '**weak classifier**': It performs slightly better than chance.

Toy example

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)



Each data point has

a class label:

$$y_t = \begin{cases} +1 \ (\ ) \\ -1 \ (\ ) \end{cases}$$

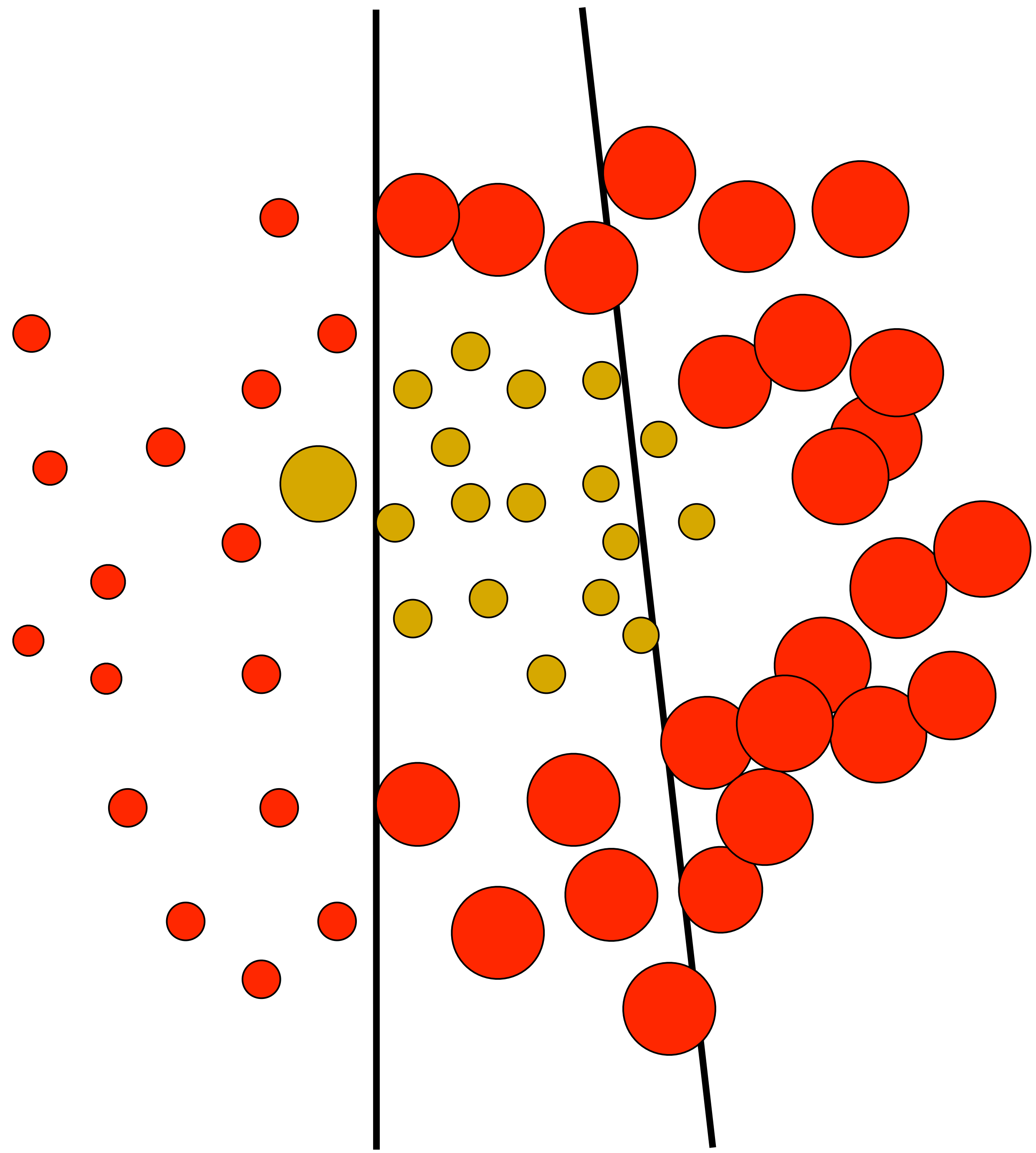
We update the weights:

$$w_t \leftarrow w_t \exp\{-y_t H_t\}$$

We set a new problem for which the previous weak classifier performs at chance again

Toy example

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)



Each data point has

a class label:

$$y_t = \begin{cases} +1 \; (\;) \\ -1 \; (\;) \end{cases}$$

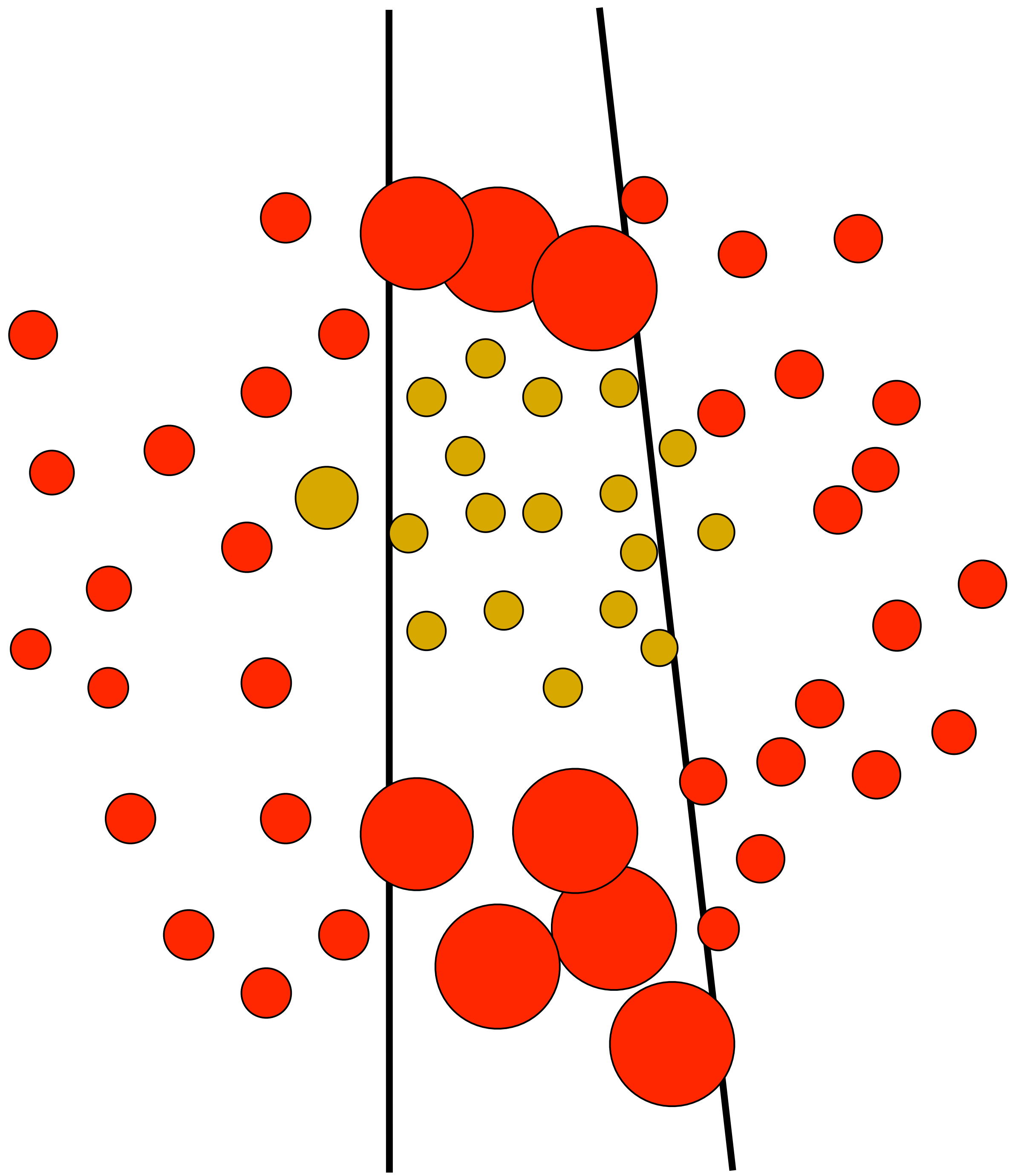
We update the weights:

$$w_t \leftarrow w_t \exp\{-y_t H_t\}$$

We set a new problem for which the previous weak classifier performs at chance again

Toy example

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)



Each data point has

a class label:

$$y_t = \begin{cases} +1 \; (\;) \\ -1 \; (\;) \end{cases}$$

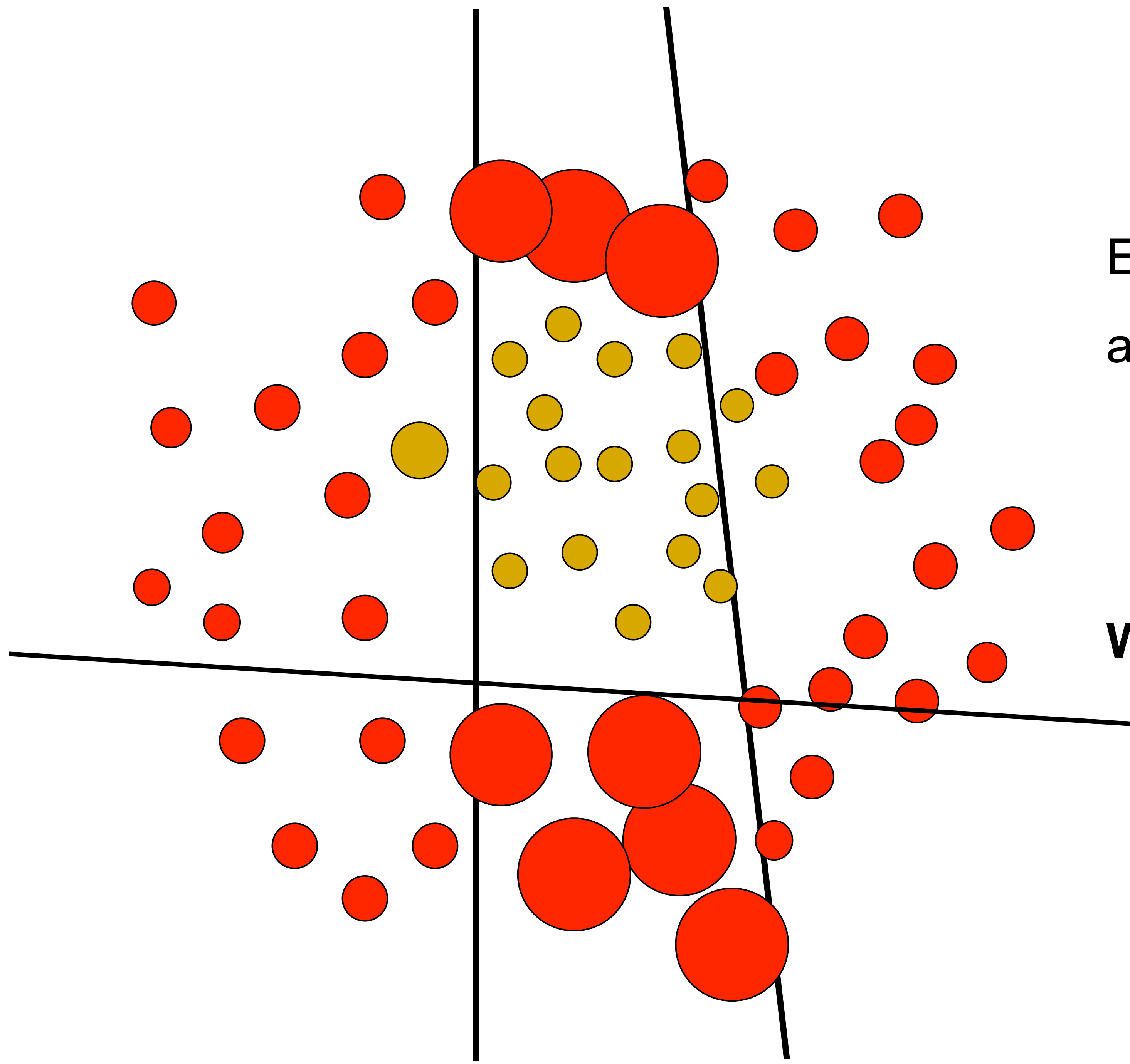
We update the weights:

$$w_t \leftarrow w_t \exp\{-y_t H_t\}$$

We set a new problem for which the previous weak classifier performs at chance again

Toy example

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)



Each data point has

a class label:

$$y_t = \begin{cases} +1 \; (\;) \\ -1 \; (\;) \end{cases}$$

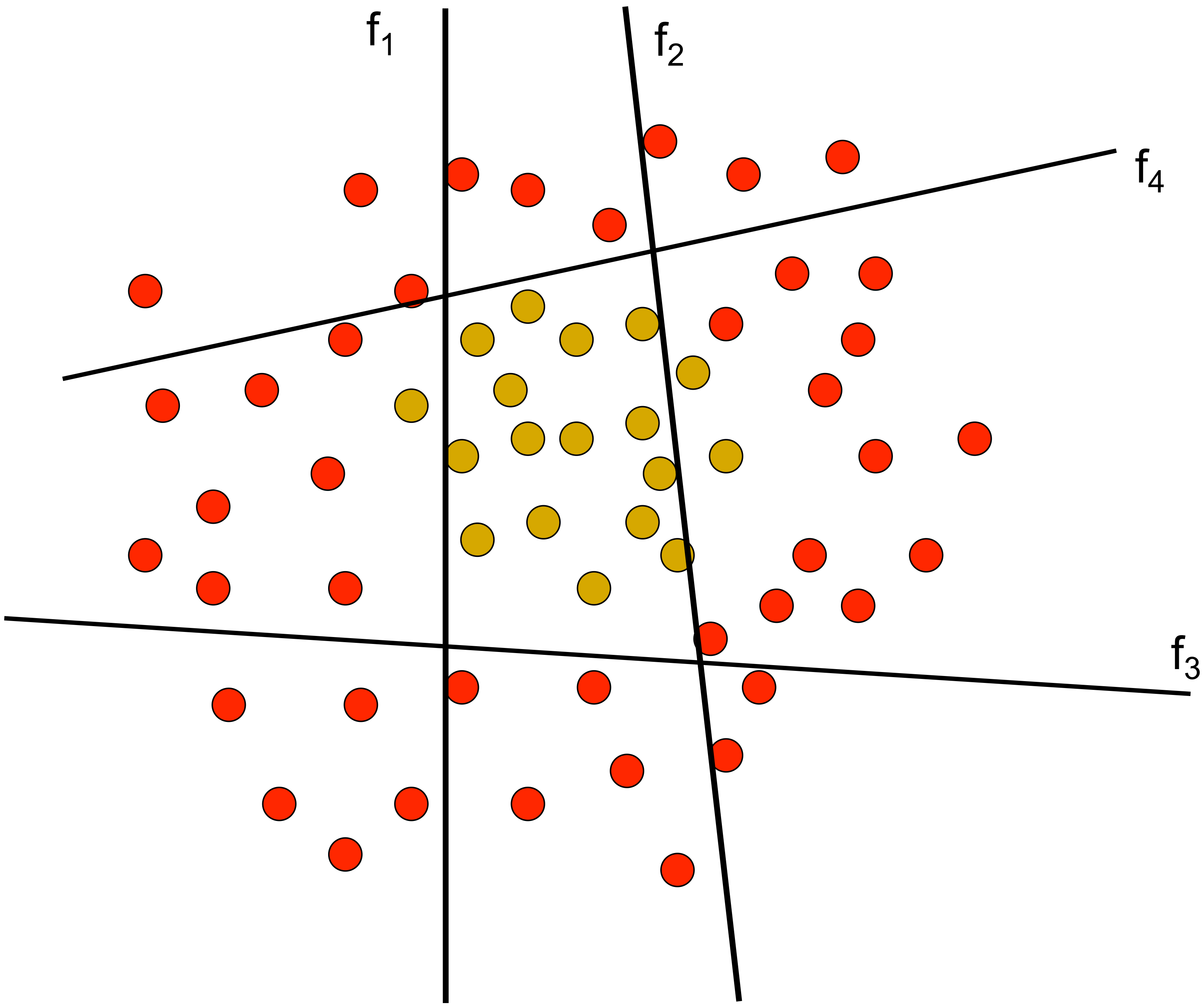
We update the weights:

$$w_t \leftarrow w_t \exp\{-y_t H_t\}$$

We set a new problem for which the previous weak classifier performs at chance again

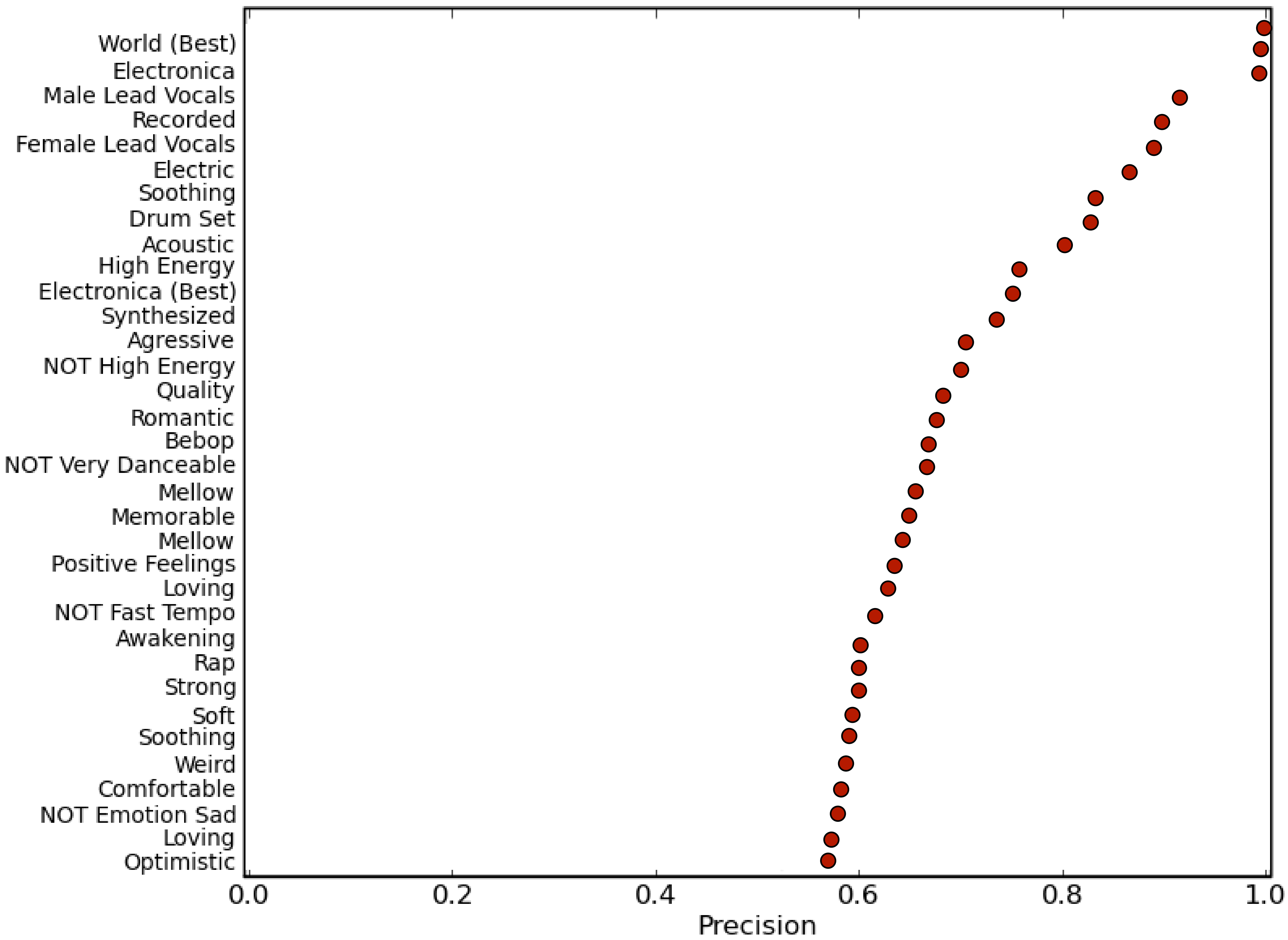
Toy example

From rii.ricoh.com/~liu/homepage/**adaboost**.ppt (Xu and Arun)



The strong (non- linear) classifier is built as the combination of all the weak (linear) classifiers.

Some autotags sorted by precision



Some tags are learned with high precision ("male lead vocals").
Some are completely unlearnable (e.g. "loving")

Top Tags for Artists (annotation)

Radiohead
0.82 Britrock
0.81 alternative_rock
0.78 alternative
0.76 britpop
0.76 melancholic
0.76 melancholy
0.75 alt_rock
0.73 seen_live
0.73 00s
0.73 Experimental_Rock

Peter Tosh
0.96 roots_reggae
0.94 Rasta
0.93 reggae
0.85 dancehall
0.64 rhythm_and_blues
0.62 funk
0.60 old_school
0.60 soft_rock
0.57 soul
0.55 male

The Who
0.70 rock
0.68 60s
0.67 classic_rock
0.65 power_pop
0.65 Favourites
0.64 good
0.63 us
0.60 hard_rock
0.60 90's
0.59 Aussie

Ella Fitzgerald
0.86 vocal
0.83 jazz
0.82 vocal_jazz
0.69 swing
0.58 trumpet
0.55 breakcore
0.54 oldies
0.53 easy_listening
0.50 saxophone
0.48 Asian

David Bowie
0.74 80s
0.71 glam
0.69 70s
0.68 classic_rock
0.67 england
0.65 english
0.65 proto-punk
0.64 new_wave
0.64 glam_rock
0.62 pop

Douglas Eck
0.74 singer-songwriter
0.67 folk
0.64 blues
0.60 folk_rock
0.57 genius
0.57 mpb (Brazilian pop)
0.56 bluegrass
0.56 indie_folk
0.55 gentle
0.54 americana

Enya
0.92 ethereal
0.88 celtic
0.88 Female_Voices
0.86 relaxing
0.86 relax
0.85 Meditation
0.85 fantasy
0.81 irish
0.76 neofolk
0.76 female

James Brown
0.93 rhythm_and_blues
0.91 soul
0.90 funk
0.79 motown
0.79 funky
0.76 blues
0.68 Rock_and_Roll
0.63 60s
0.63 oldies
0.63 rock_n_roll

Tag top-20 lists : Reggae

1 Max Romeo
2 The Upsetters
3 The Meditations
4 Dillinger
5 Dub Specialists
6 U Roy
7 Johnny Clarke
8 The Twinkle Brothers
9 Bunny Wailer
10 Tapper Zukie
11 Bob Marley & The Wailers
12 Leroy Brown
13 Lee "Scratch" Perry
14 The Wailers
15 Sly & Robbie
16 U Brown
17 Poet & The Roots
18 Big Youth
19 Ranking Trevor
20 Jah Lloyd

List from website last.fm

- Bob Marley
- Bob Marley & The Wailers
- Sublime
- Manu Chao
- Sean Paul
- UB40
- Gentleman
- Matisyahu
- Shaggy
- Rihanna
- Seeed
- Damian Marley
- 5'nizza
- Wyclef Jean
- Lee "Scratch" Perry
- Sizzla
- The Police
- 311
- Toots and The Maytals
- Peter Tosh

Tag top-20 lists : Shoegaze

1 R.A.V.A.G.E.
2 Catherine Wheel
3 Electroluminescent
4 My Bloody Valentine
5 Keith Fullerton Whitman
6 Dan Gardopee
7 Ulrich Schnauss
8 M83
9 The Jesus and Mary Chain
10 Times New Viking
11 thisquietarmy
12 Pumice
13 Swervedriver
14 Kinski
15 Spiritualized®
16 Readymade
17 Lush
18 SIANspheric
19 Sugar
20 Throwing Muses

List from website last.fm

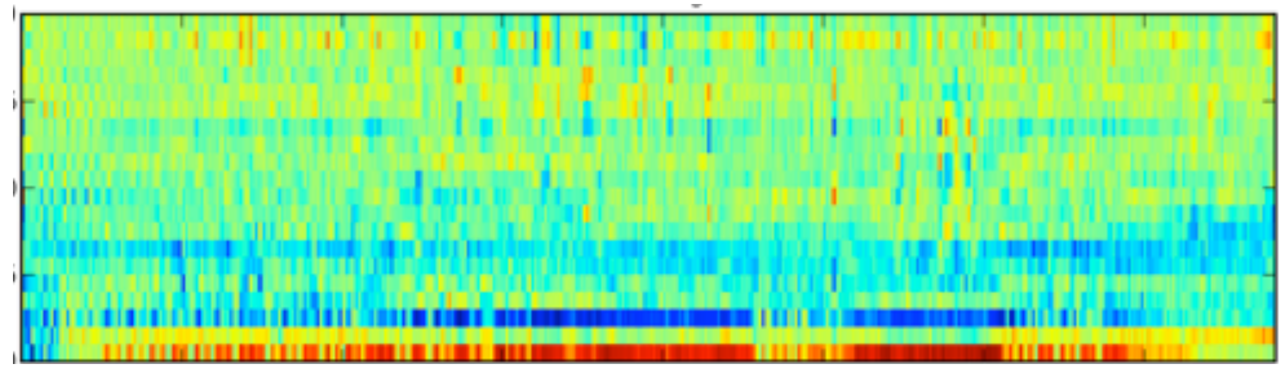
- My Bloody Valentine
- Sigur Rós
- The Jesus and Mary Chain
- M83
- Cocteau Twins
- Slowdive
- Spiritualized
- The Verve
- Black Rebel Motorcycle Club
- The Radio Dept.
- Ride
- The Brian Jonestown Massacre
- Deerhunter
- Yo La Tengo
- Lush
- Mazzy Star
- Spacemen 3
- Asobi Seksu
- Silversun Pickups
- Curve

How do features map onto tags?

Our classifier (AdaBoost) selects features based on their ability to minimize error (automatic feature selection)

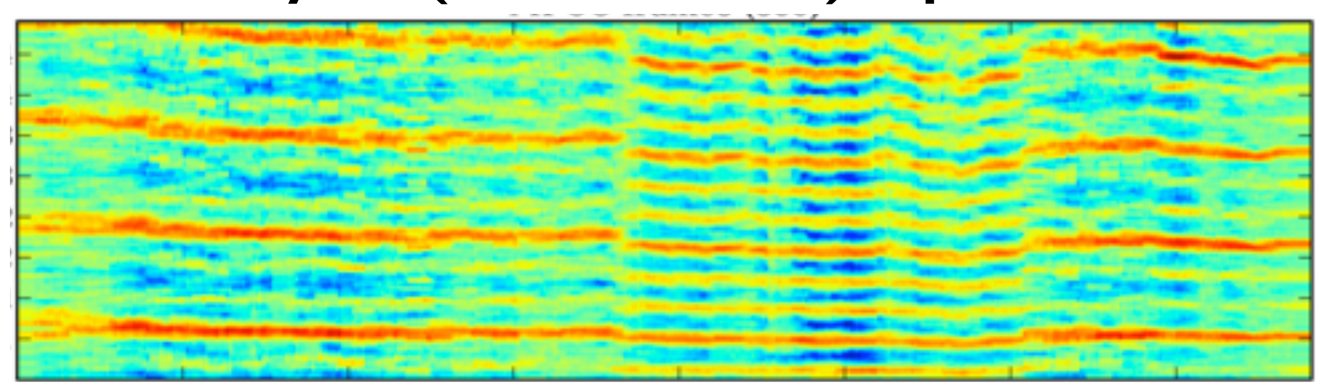
Which features predicted what?

Timbre (mfcc) Top 15



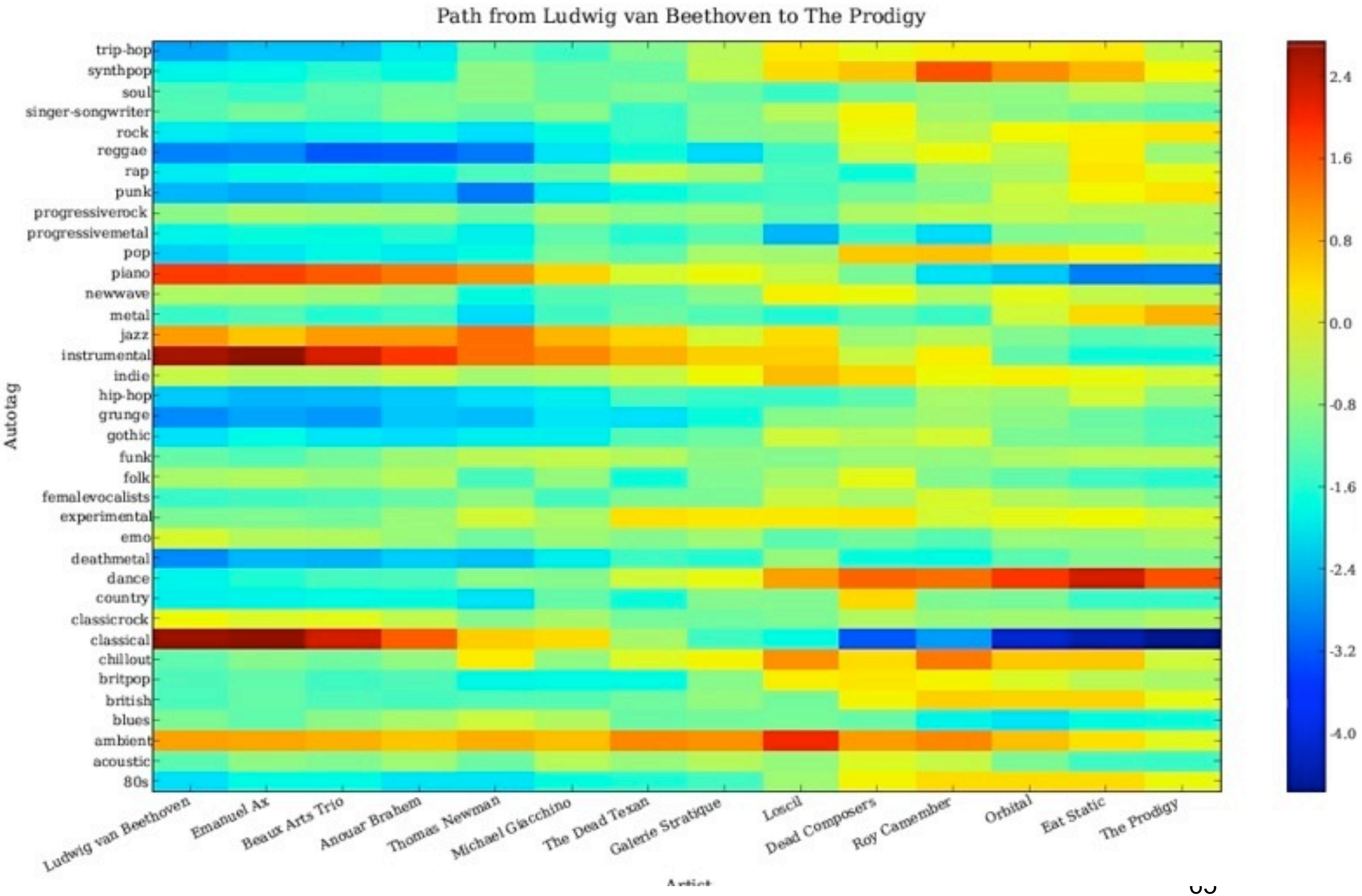
1 anarcho-punk
2 romantic
3 left-wing
4 trumpet
5 Classical
6 Female_Voices
7 free_jazz
8 jazz_fusion
9 celtic
10 composers
11 saxophone
12 jazz
13 Scottish
14 female_vocalists
15 piano

Rhythm (autocorrelation) Top 15



1 eurodance
2 trance
3 progressive_trance
4 psytrance
5 idols
6 Electroclash
7 video_game_music
8 electro_industrial
9 goa
10 synthpop
11 vocal_trance
12 minimal_techno
13 big_beat
14 House
15 electropop

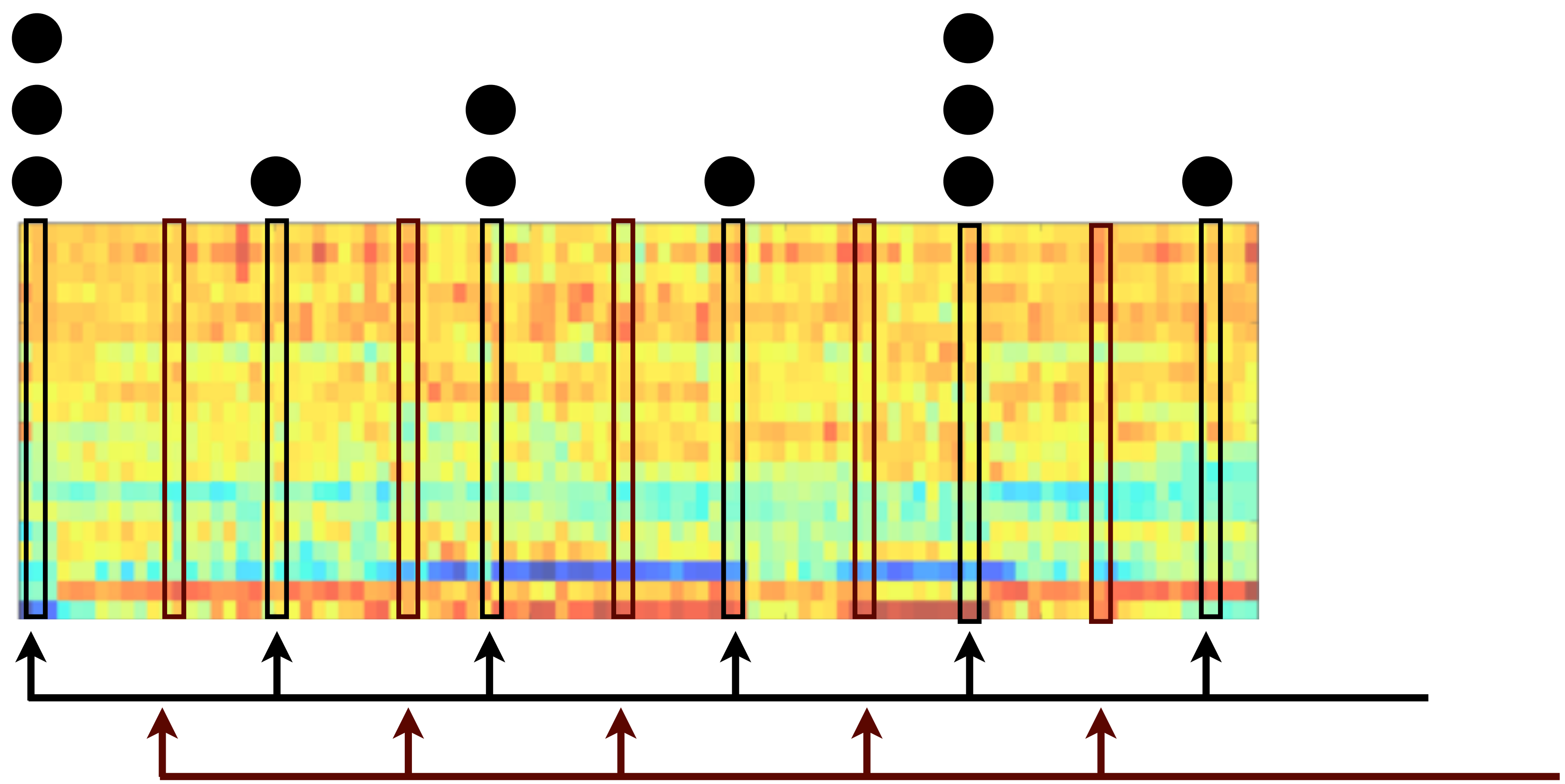
Moving from one artist to another



Expressive timing and dynamics

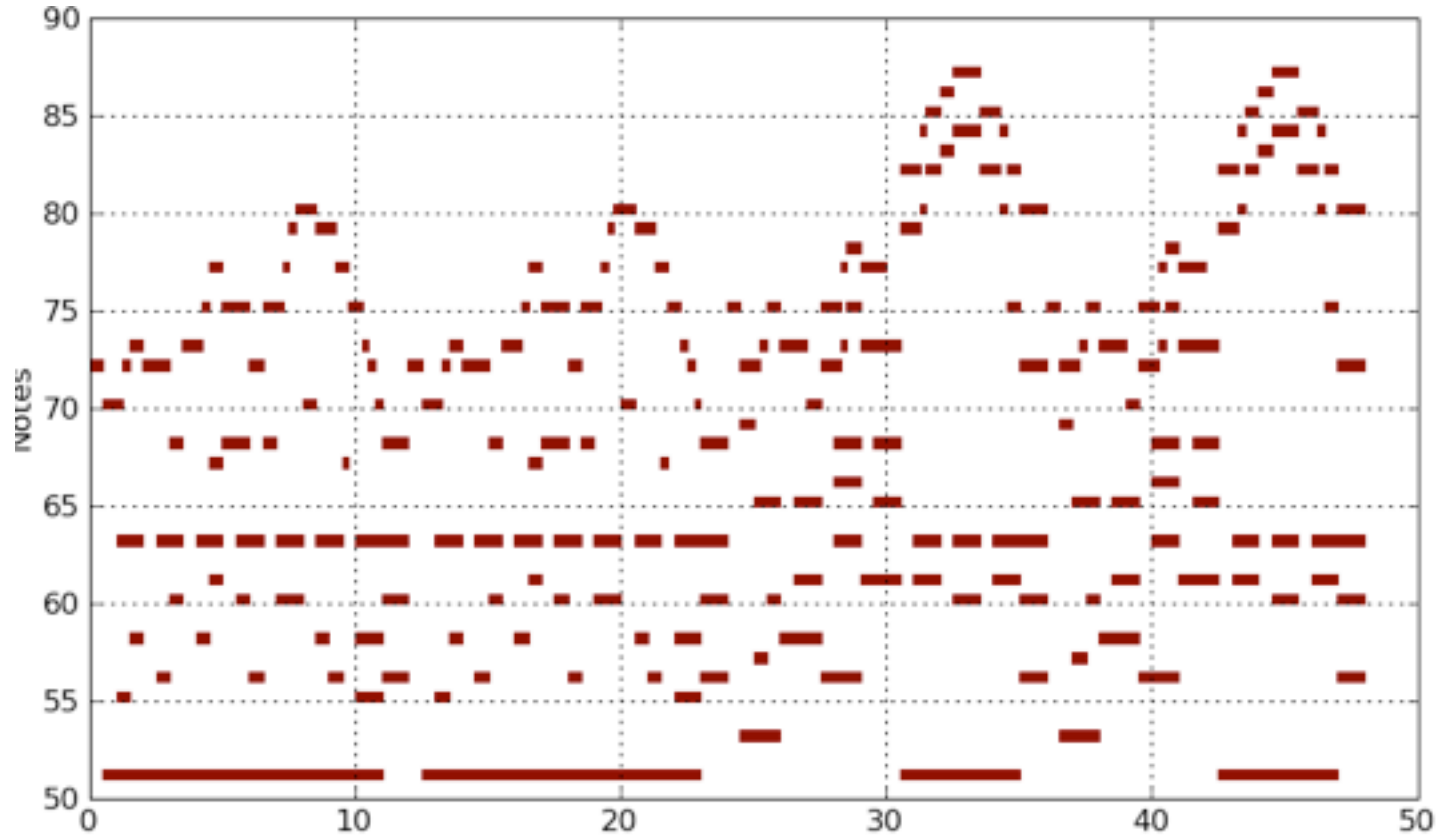
Audio detour: multi-timescale learning

- Future Research
- Chopping up a song into 200ms frames and mixing up those frames seems a pretty bad idea
- Localize long-timescale structure using meter/beat
- Features aligned to beat, measure, phrase of music



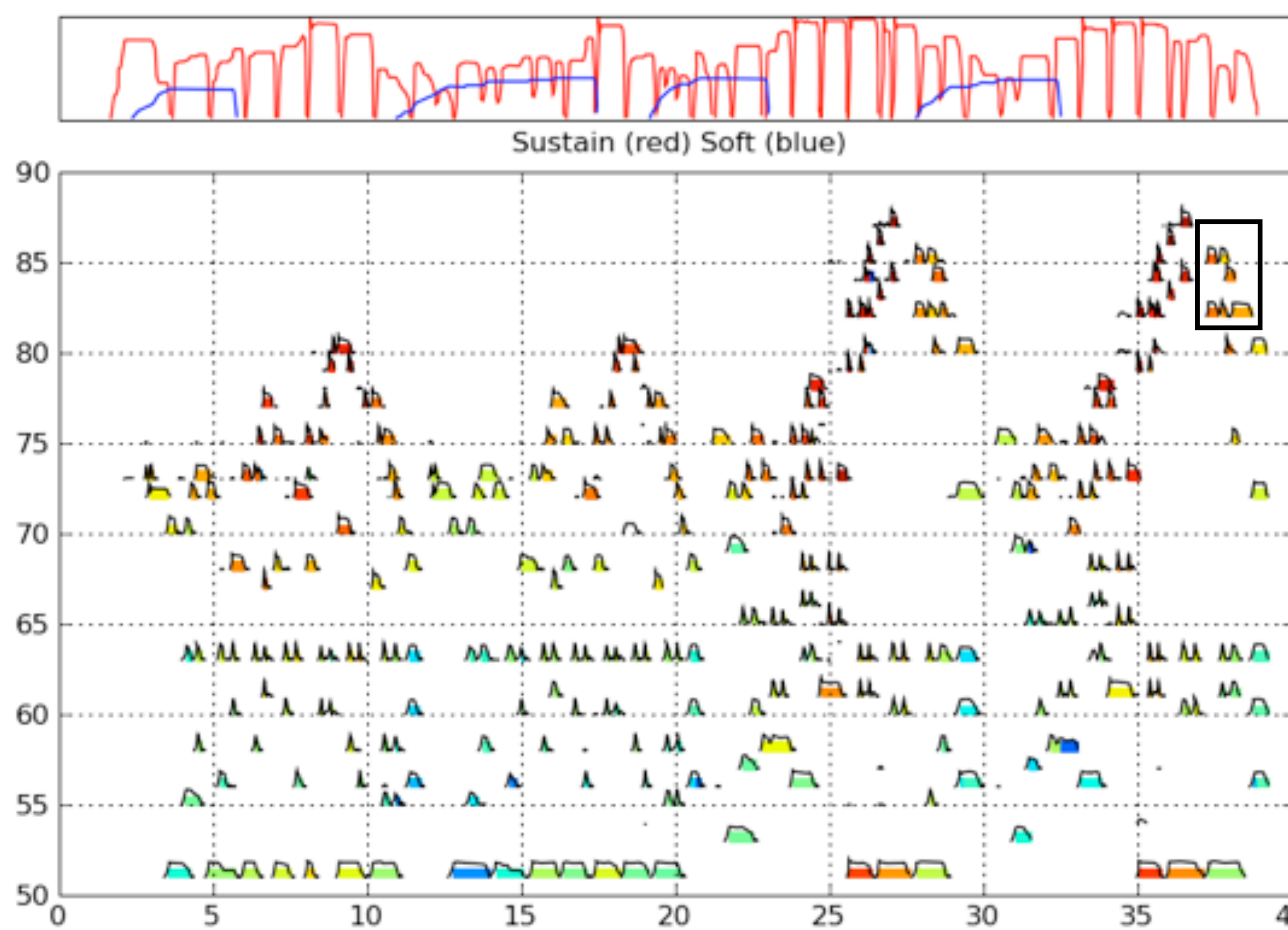
Example:
Chopin Etude
Opus 10 No 3

Bösendorfer example: Schubert Waltz

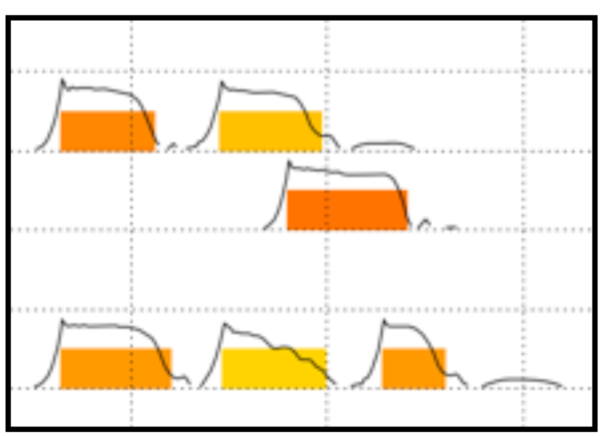


Deadpan
(no expressive timing or dynamics)

Human performance
(Recorded on Bösendorfer ZEUS)



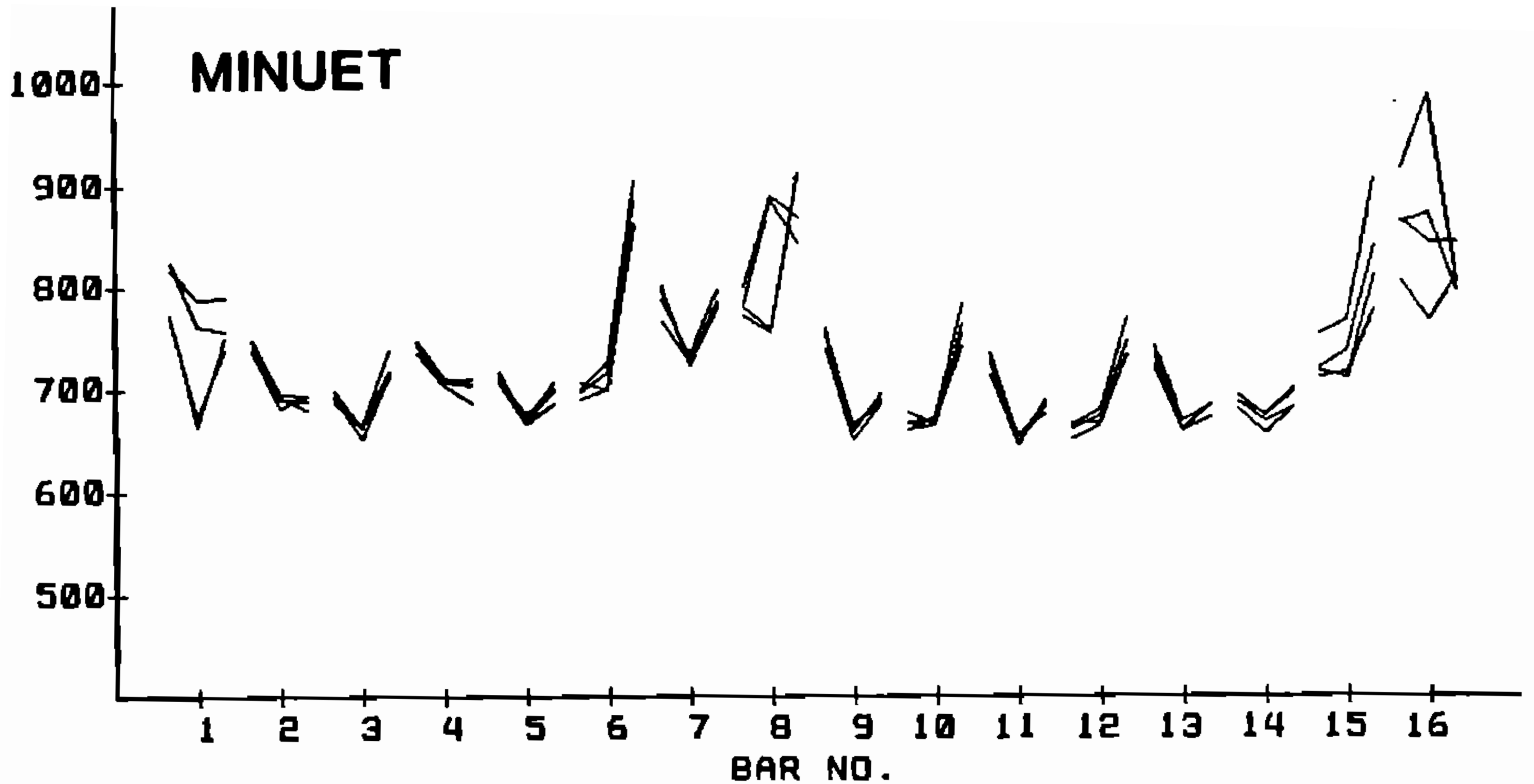
Differences from MIDI:
- timing (onset, length)
- velocity (seen as red)
- pedaling (blue shading)
- key angles (below)



Aside: Meter/Pulse

What can we measure?

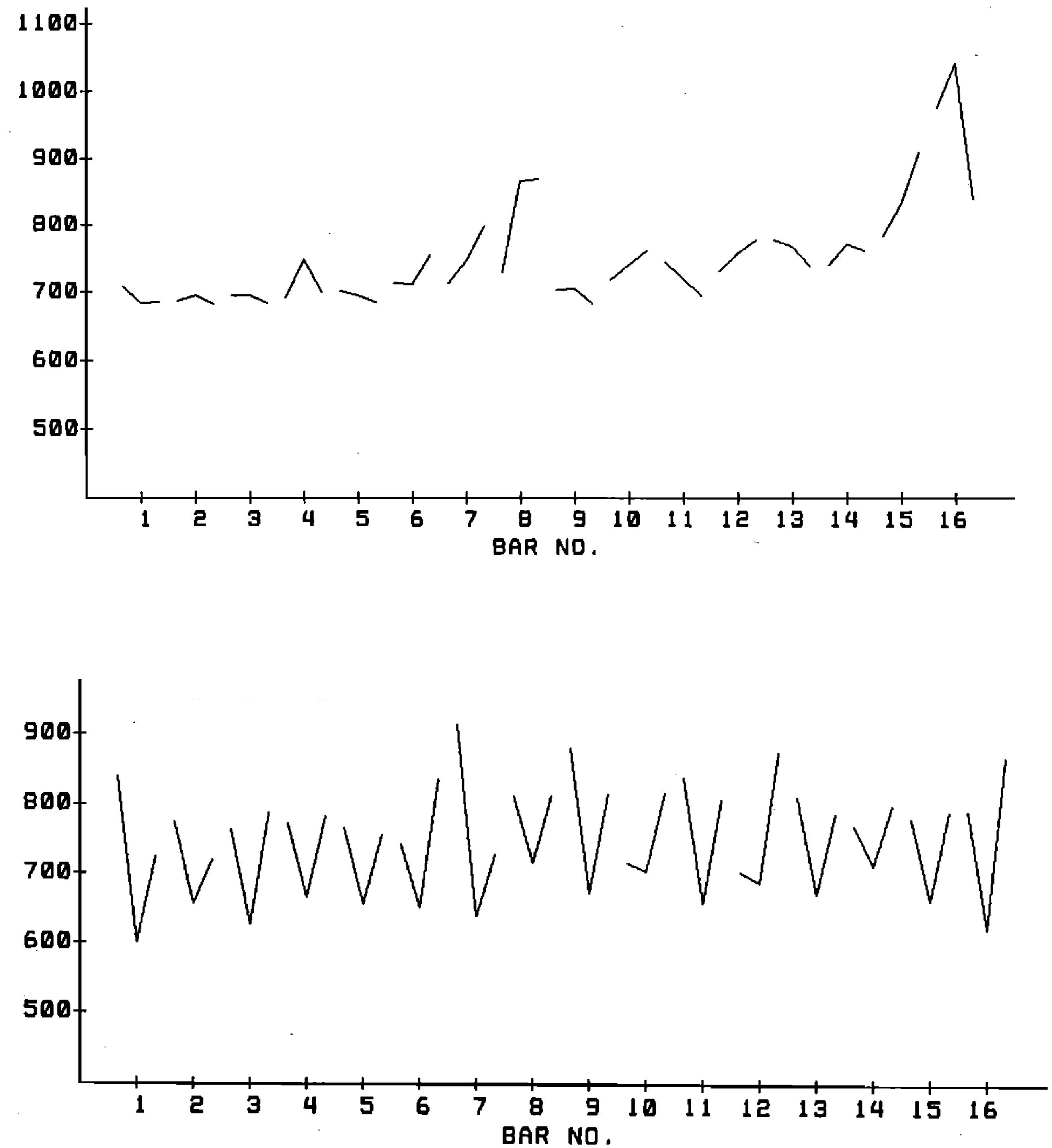
- Repp (1989) measured note IOIs in 19 famous recordings of a Beethoven minuet (Sonata op 31 no 3)



Grand average timing patterns of performances with repeats plotted separately. (From B. Repp "Patterns of expressive timing in performances of a Beethoven minuet by nineteen famous pianists",1990)

What can we measure?

- PCA analysis yields 2 major components
  - Phrase final lengthening
  - Phrase internal variation
- Simply taking mean IOIs yields can yield pleasing performance
- Reconstructing using principal component(s) can yield pleasing performance
- Concluded that timing underlies musical structure



Adapted from Repp (1990)

Experiment: Learn to Perform Schubert Waltzes

- 12 highly trained pianists (performance PhD, University of Montreal Faculty of Music)
- 5 similar waltzes by Schubert; 115 total performances; 38284 notes in all
- Recorded on Bösendorfer ZEUS reproducing imperial grand piano
- Used this data to teach a machine learning model about piano performance

Listen at Stan's Demo....

Training and generation

Training:

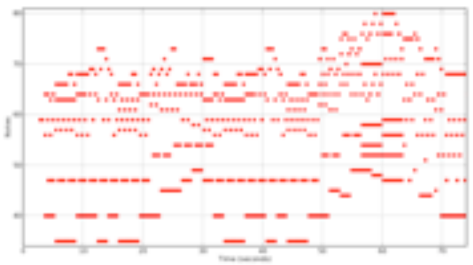
- Train algorithms on 4 pieces using MIDI performances captured from Bösendorfer ZEUS.
- Ensure generalization using out-of-sample data

Generation:

- Predict note velocities, local time deviations and overall tempo deviation for 5th piece
- Generate machine performance as MIDI from predictions
- Record performance from MIDI on Bösendorfer ZEUS

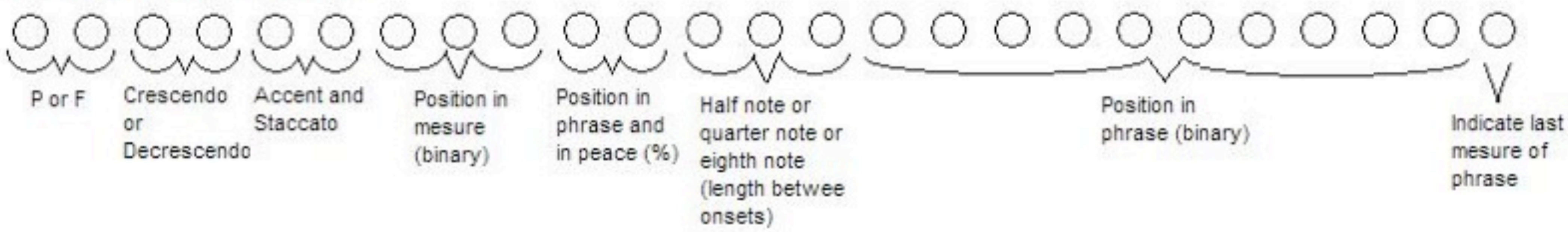
Pianist pedaling was ignored. We generated pedaling from note timing profile. (Future work)

Learning Expressive Timing (Stanislas Lauly)

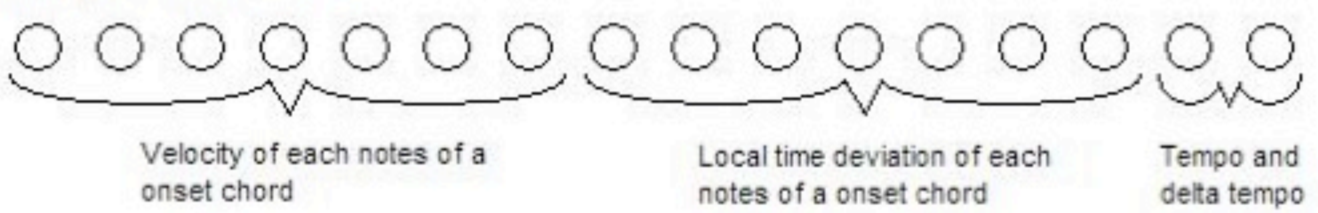


Represent dynamics and timing deviations as input/target vectors

Input (score) at time t :



Target at time t :



Data at time t :

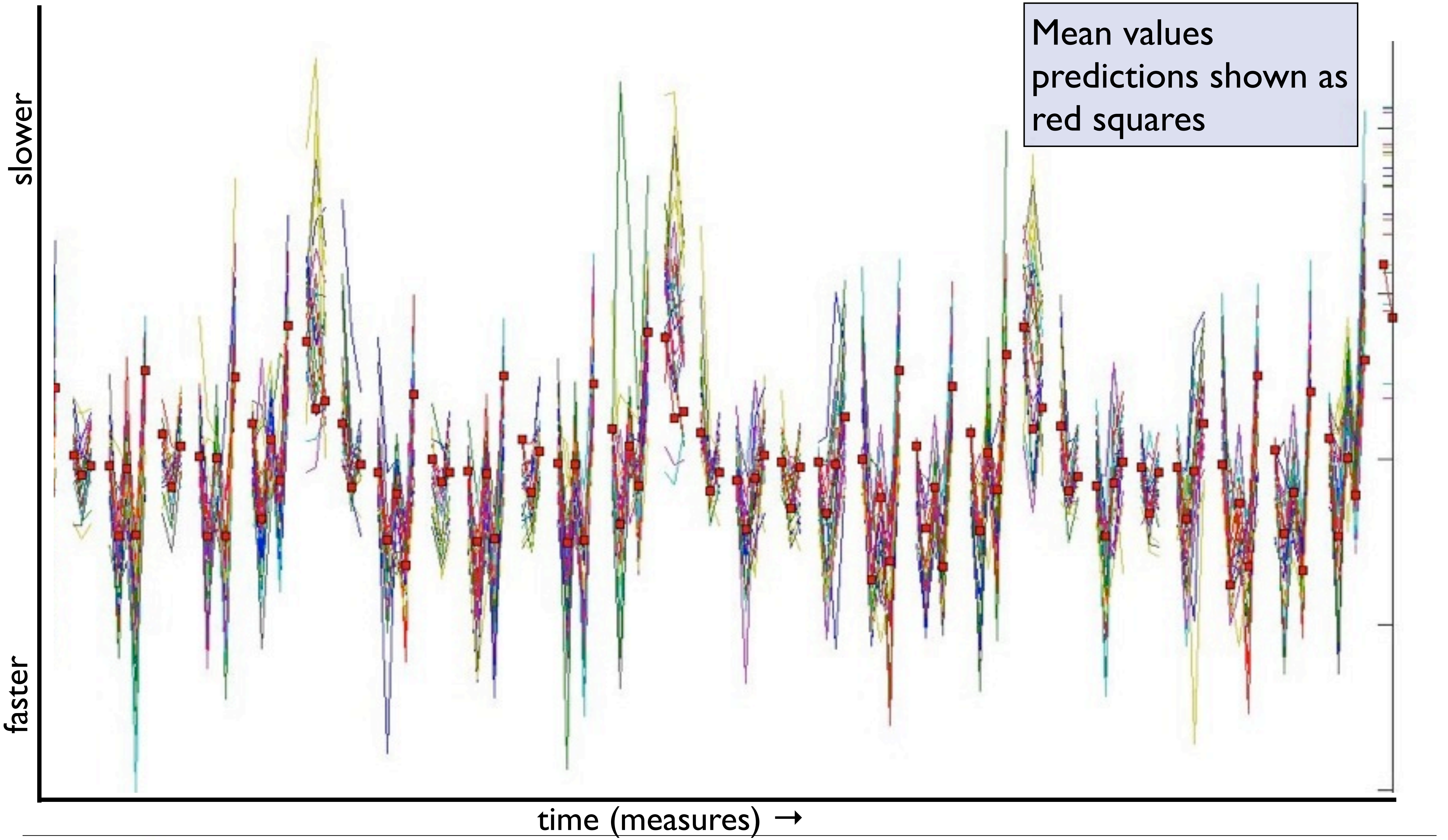
Input

Input t-2, input t-1, input t, input t+1, input t+2, target t-1, target t-2

Target

Target t

Timing deviations for all 20 performances of a single waltz.



Mean timing deviations (blue) versus predicted deviations (red)