



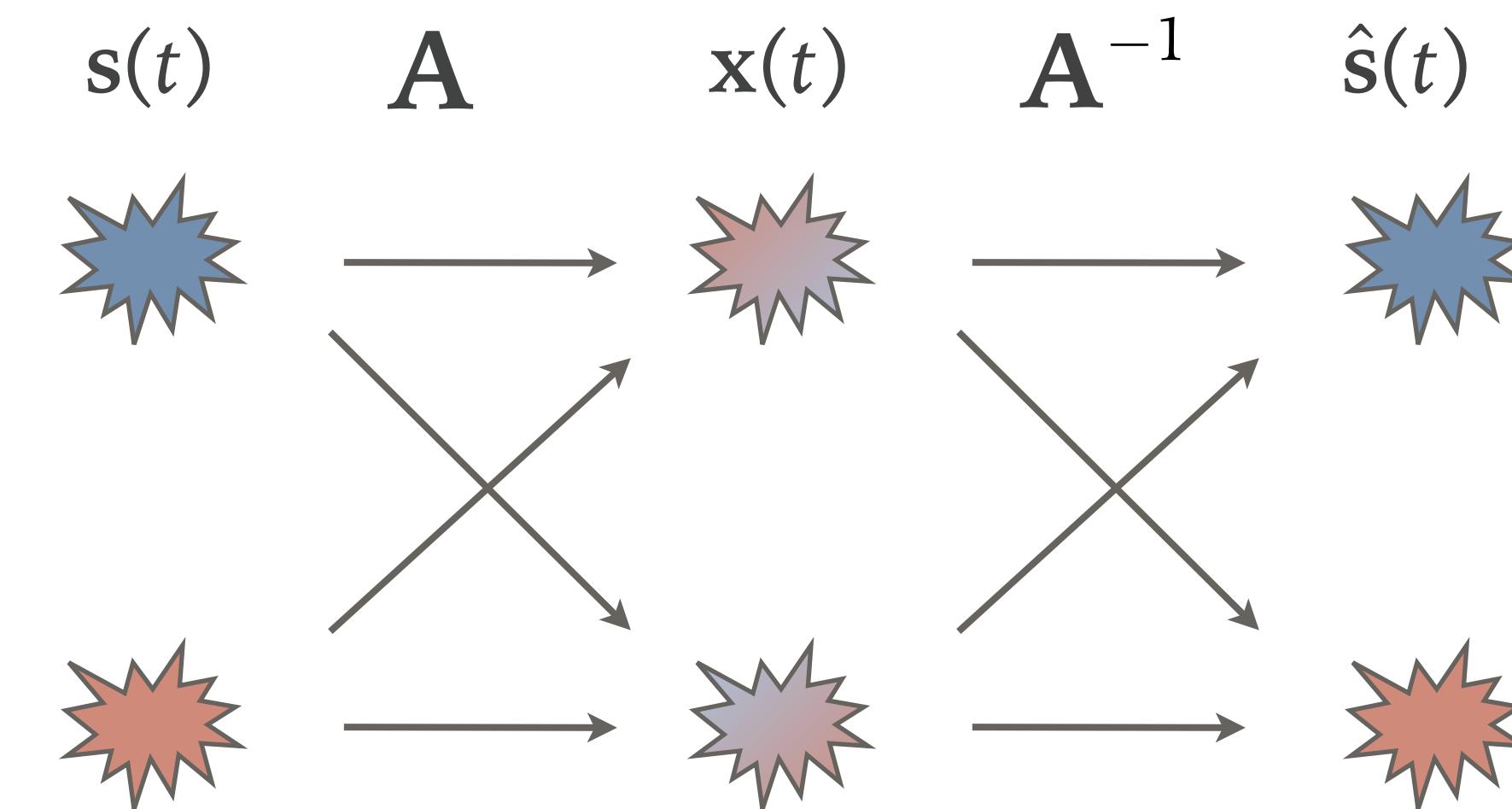
Under-constrained Signal Separation

Today's lecture

- Under-constrained signal separation
- Single-channel signal separation

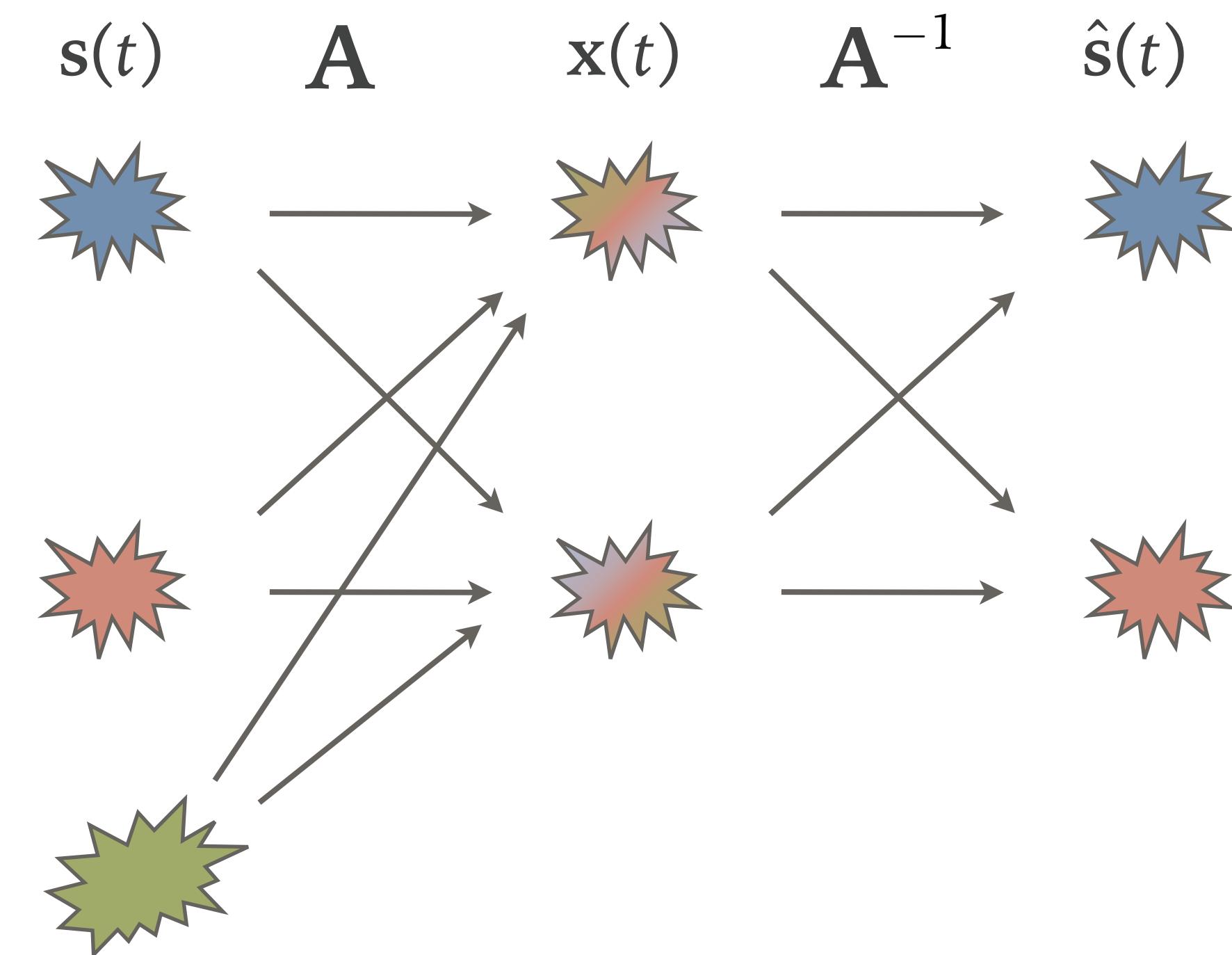
N-in / N-out separation

- Using ICA we can resolve N -sources by using N -sensors



N-in / N-out separation

- More sources create non-invertible mixing
 - A problem!



Under-constrained separation

- When we have fewer sensors than sources

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \cdot \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

- Ill-defined problem

$$\begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix}$$

Straightforward solutions

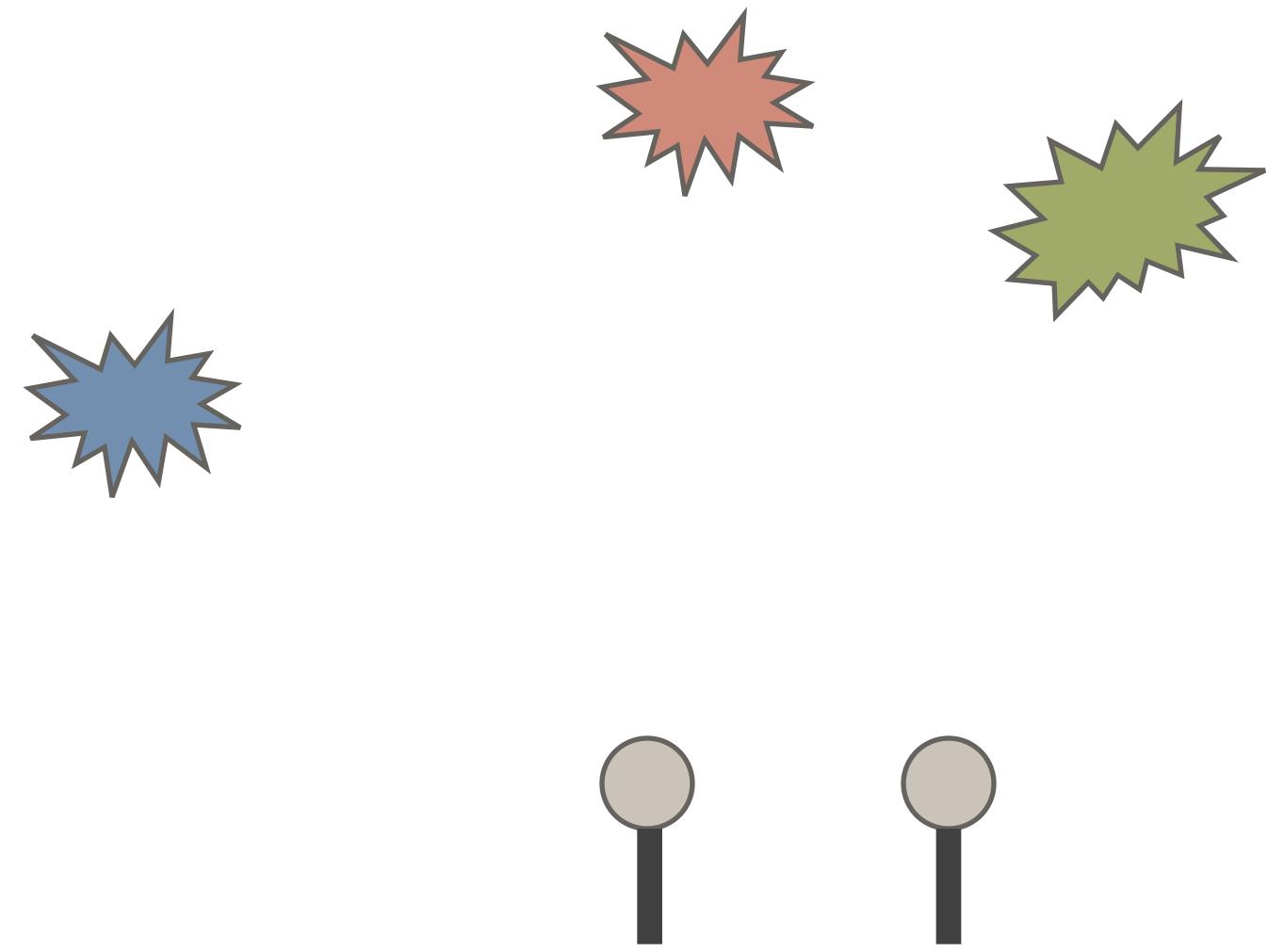
- Beamforming
 - Boosts one signal, but does not separate
- Source properties
 - Known statistics of source characteristics
- Deflation methods
 - Extract one source at a time

But we live in a harsh world ...

- Let's say you only get two sensors
 - Sensors are expensive!
- What can we do to resolve mixtures now?

Using the audio case again

- Put two microphones in a room
 - More than two sources present
- What do we do now?

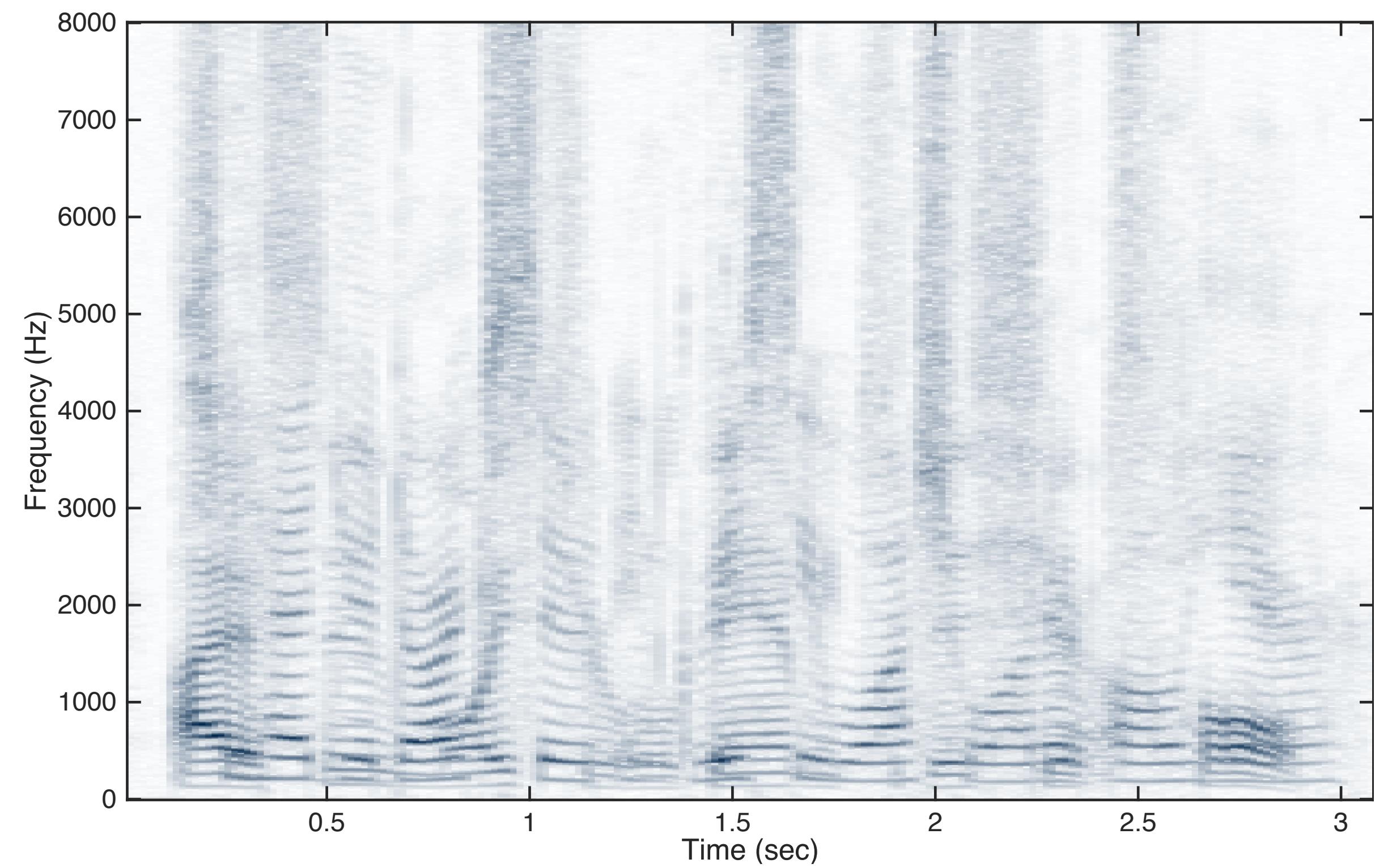


An alternative way to unmix

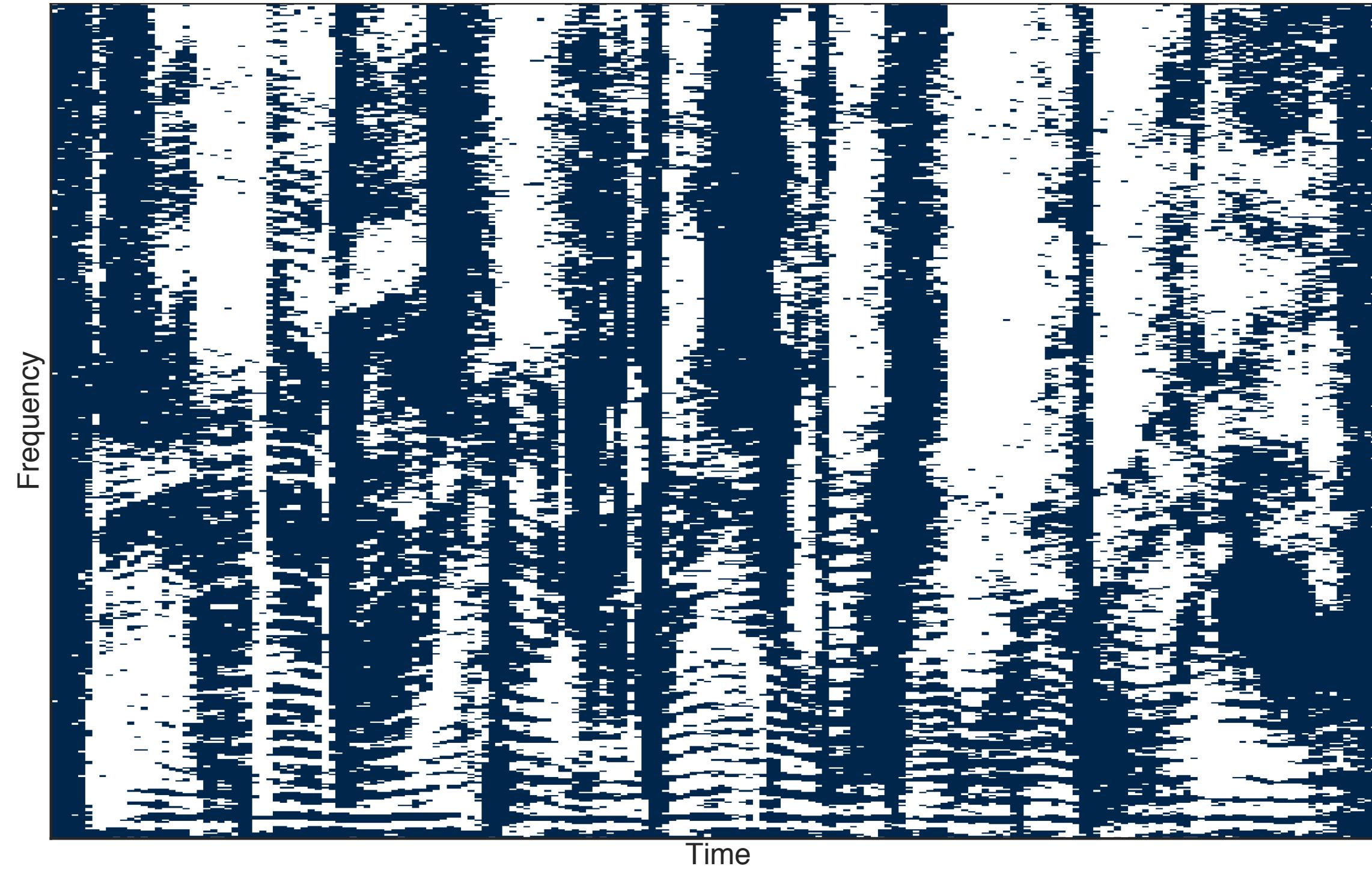
- With ICA we try to undo the mixing
 - We invert the mixing process
 - Tough and tedious for complex mixtures
- Masking alternative
 - Using binary masks on spectrograms we can isolate desired sources
 - Proper masks don't derive from mixing conditions
 - Convulsive mixing, etc. are not an issue

Masking

- Two simultaneous sources
- What are the chances the two source spectrograms coincide at any pixel?
- We can pick only certain “pixels” from the spectrogram to get each source

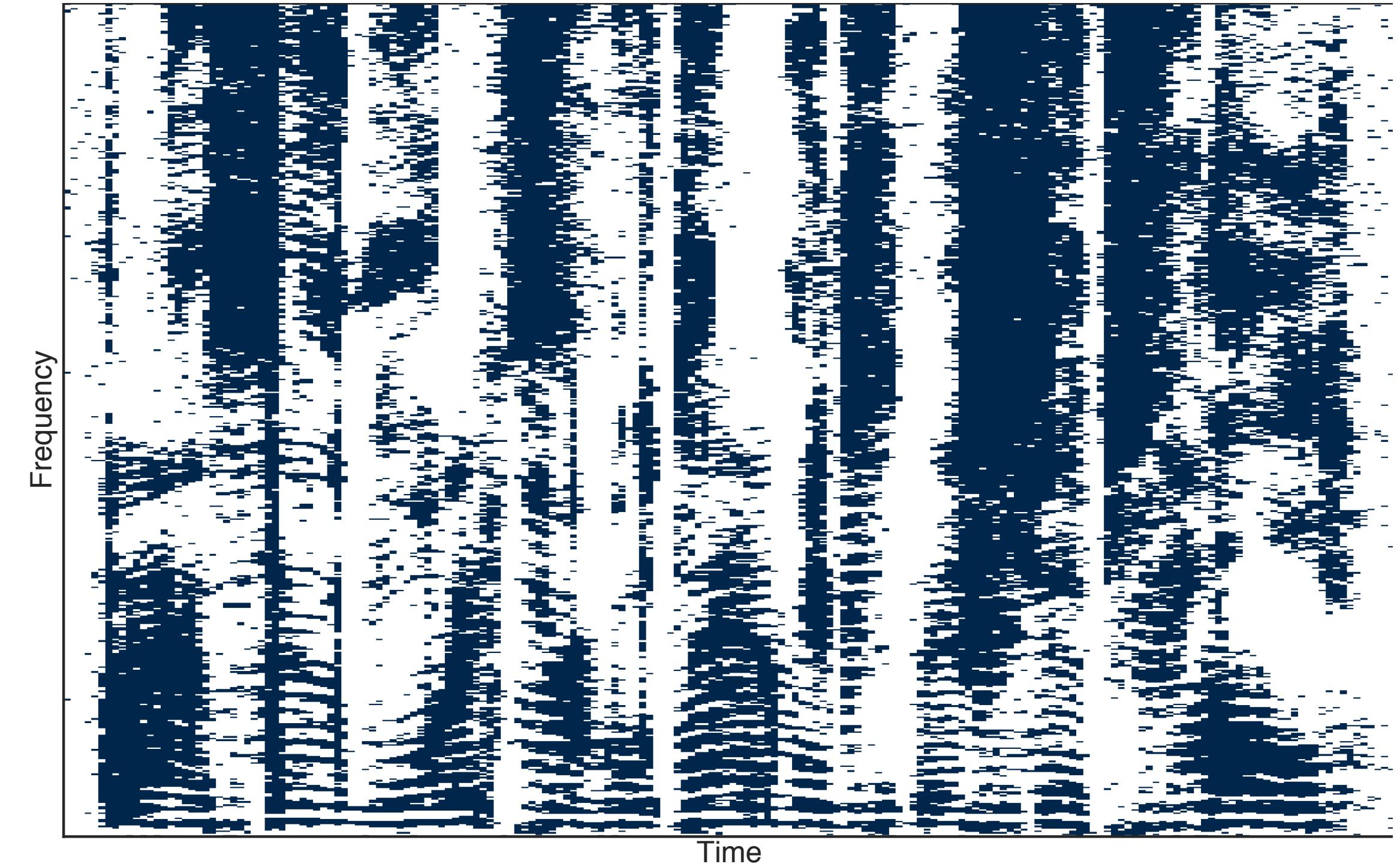


Mask examples



Mask for which:

$$|F_{source1}(f, t)| > |F_{source2}(f, t)|$$

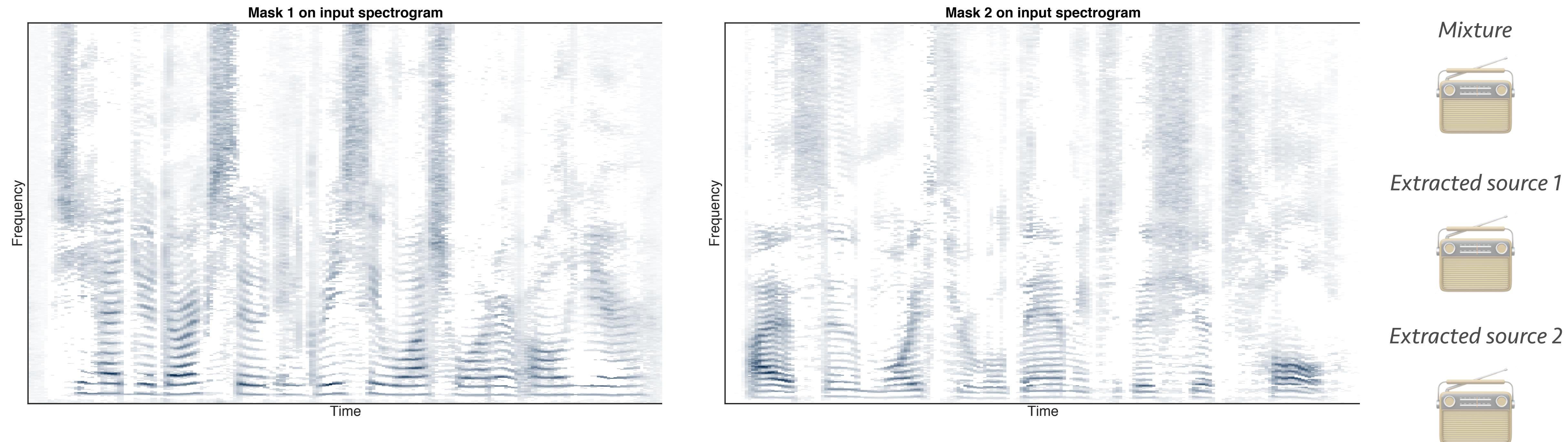


Mask for which:

$$|F_{source1}(f, t)| < |F_{source2}(f, t)|$$

Masks applied on mixture

- When applied on mixture spectrogram, masks produce good approximations of the two source spectrograms

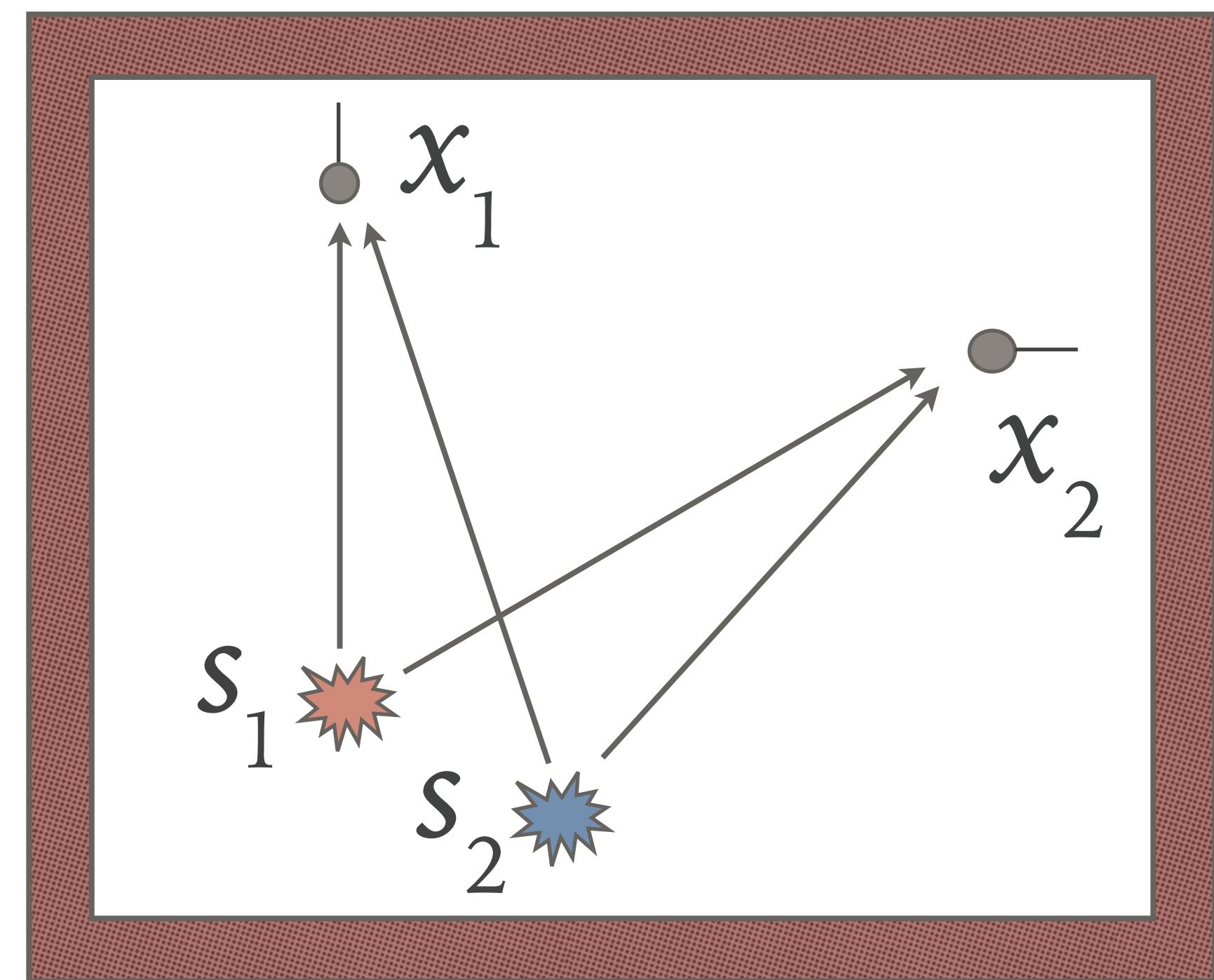


How do we get the proper masks?

- Spatial information can help us discover the appropriate source masks
- Time-frequency cells indicating the same direction form each binary mask
- A plus: No constraint on number of sources and sensors!

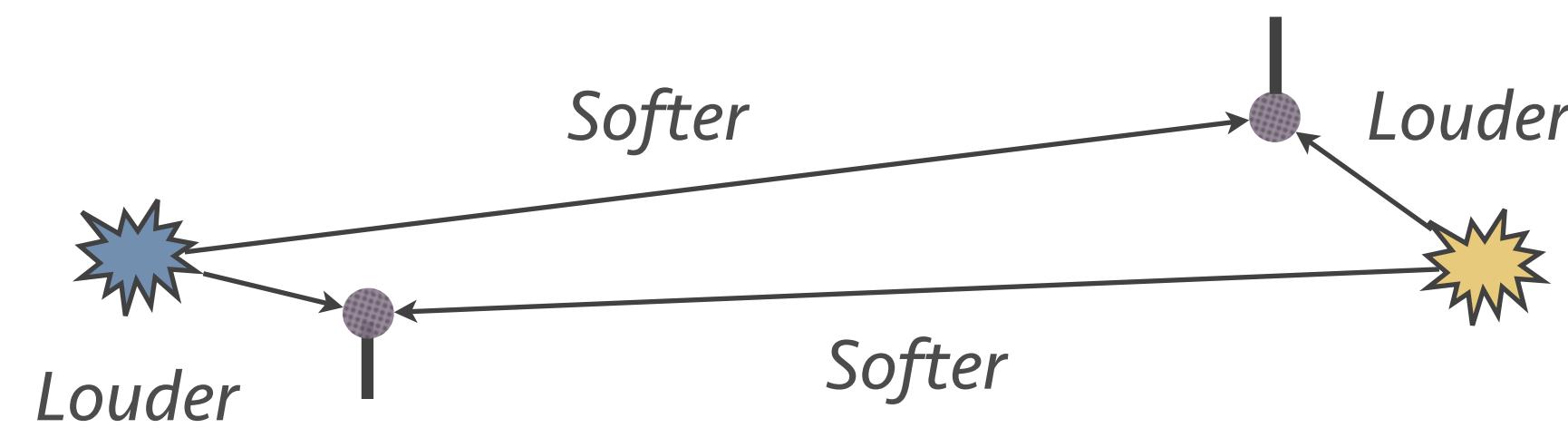
Spatial cues for masks

- Gather spatial statistics for each time-frequency point
 - Amplitude ratios
 - Phase differences
- Cluster cells with similar statistics to form masks
 - Each location would have its own set of features



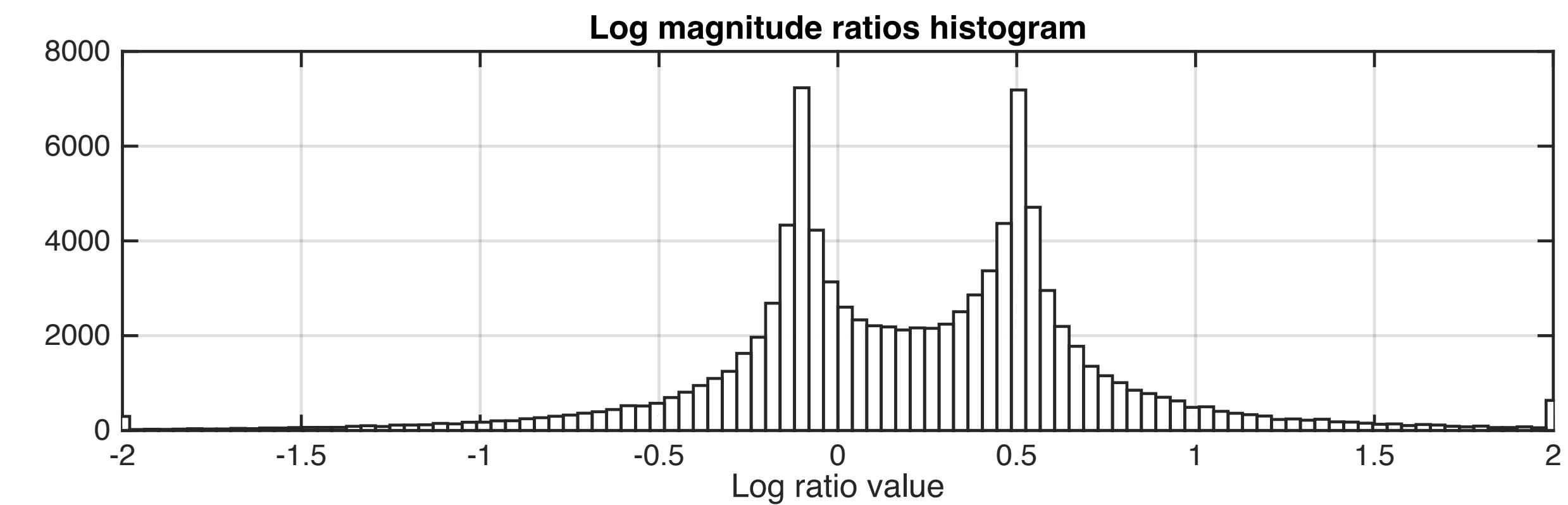
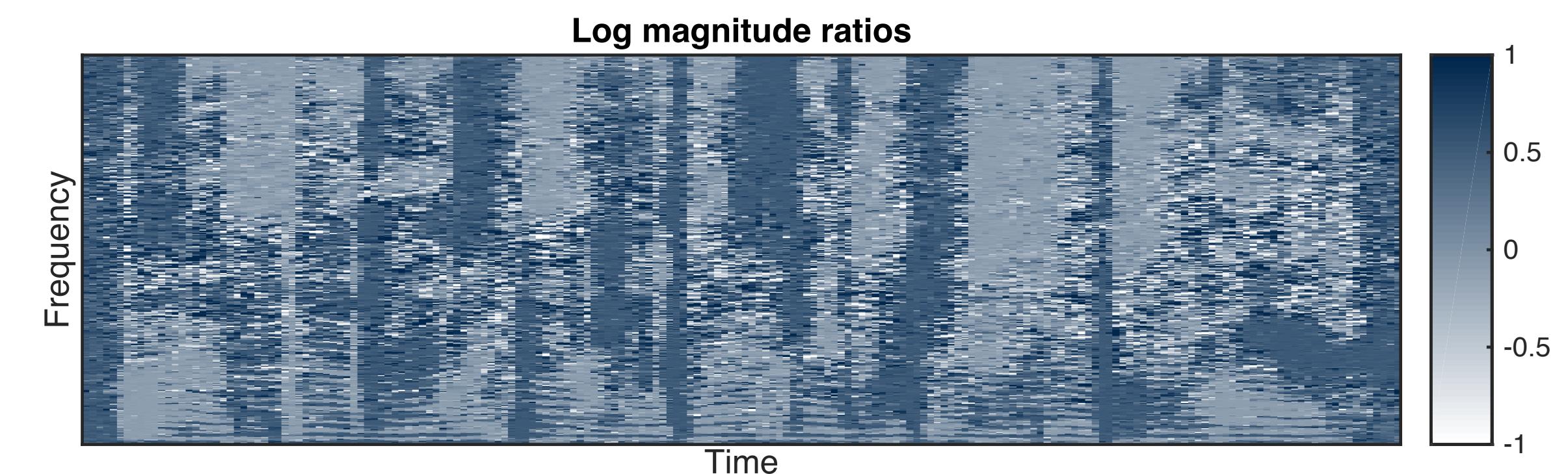
Magnitude ratios as spatial features

- Time/frequency bin ratios between the sensors will be different depending on a source's position



- We extract these as:

$$\alpha(\tau, \omega) = \log \left\| \frac{F_2(\tau, \omega)}{F_1(\tau, \omega)} \right\|$$



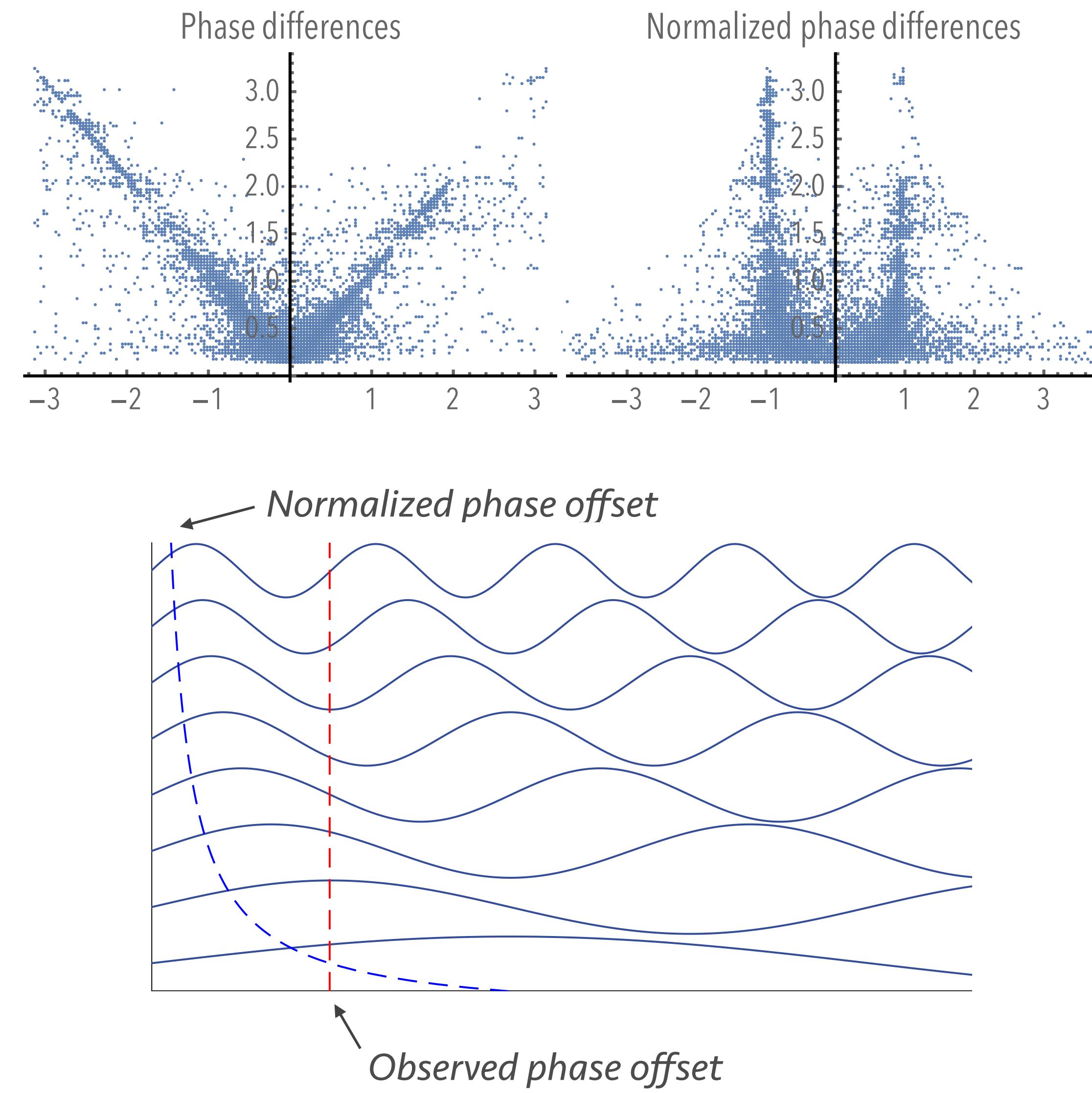
Normalized phase differences

- Use per-bin phase differences
 - But they depend on frequency

- Instead we can normalize:

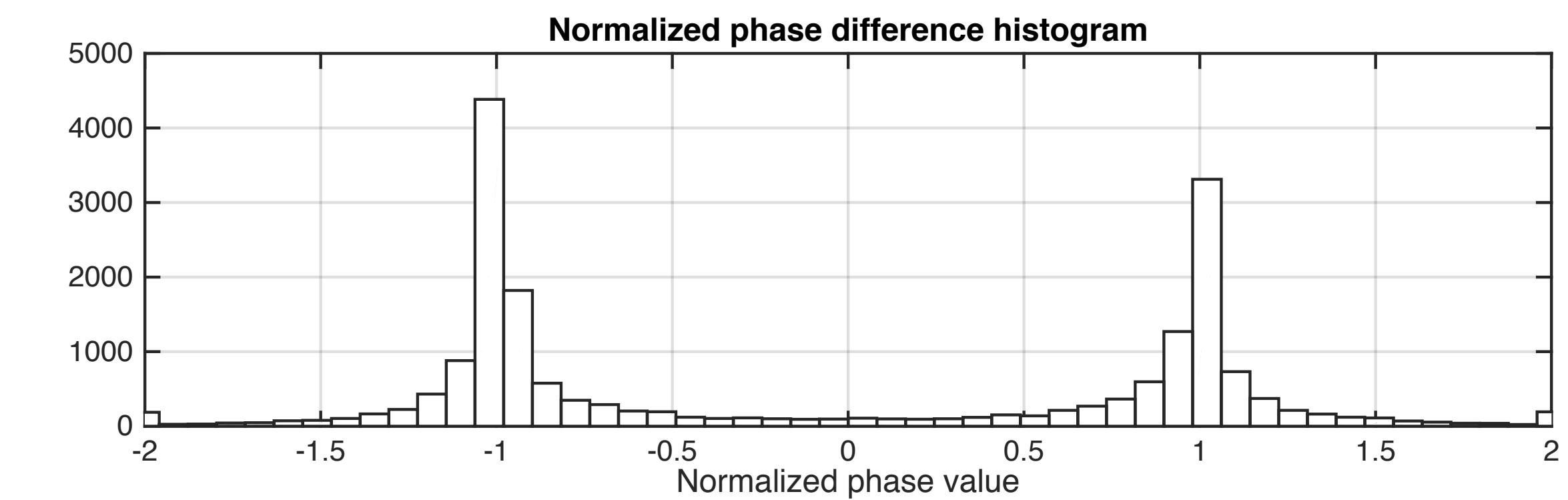
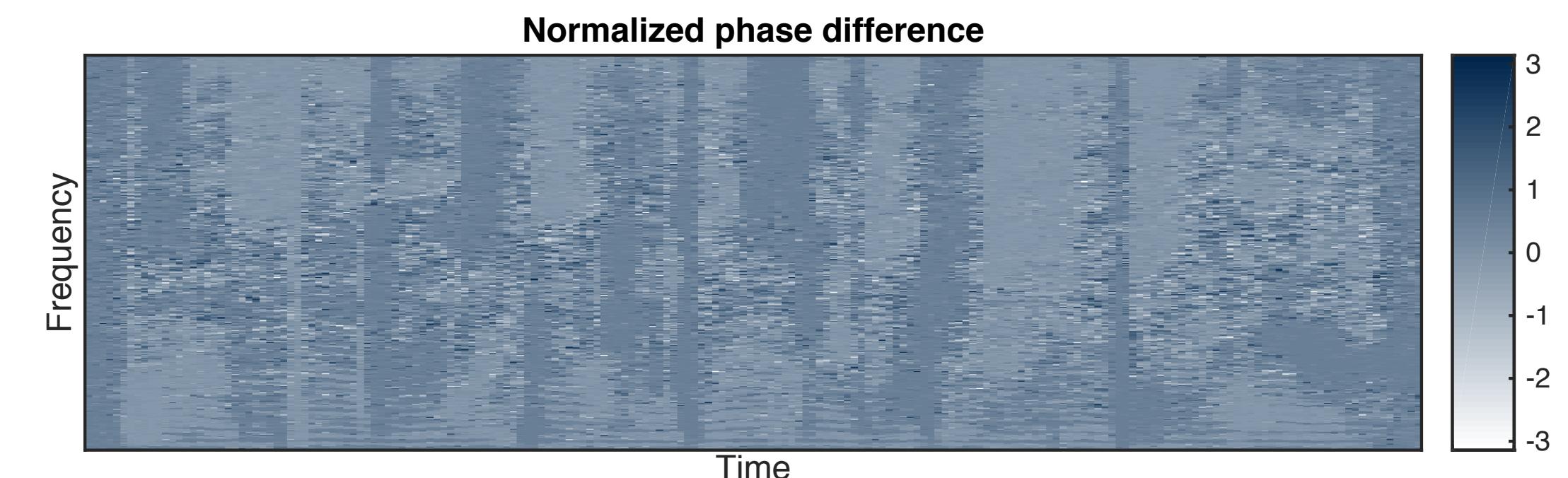
$$\delta(\tau, \omega) = \frac{1}{\omega} (\angle F_2(\tau, \omega) - \angle F_1(\tau, \omega))$$

- Which gives us a “delay” value
- Only works if the delay is less than a sample though!



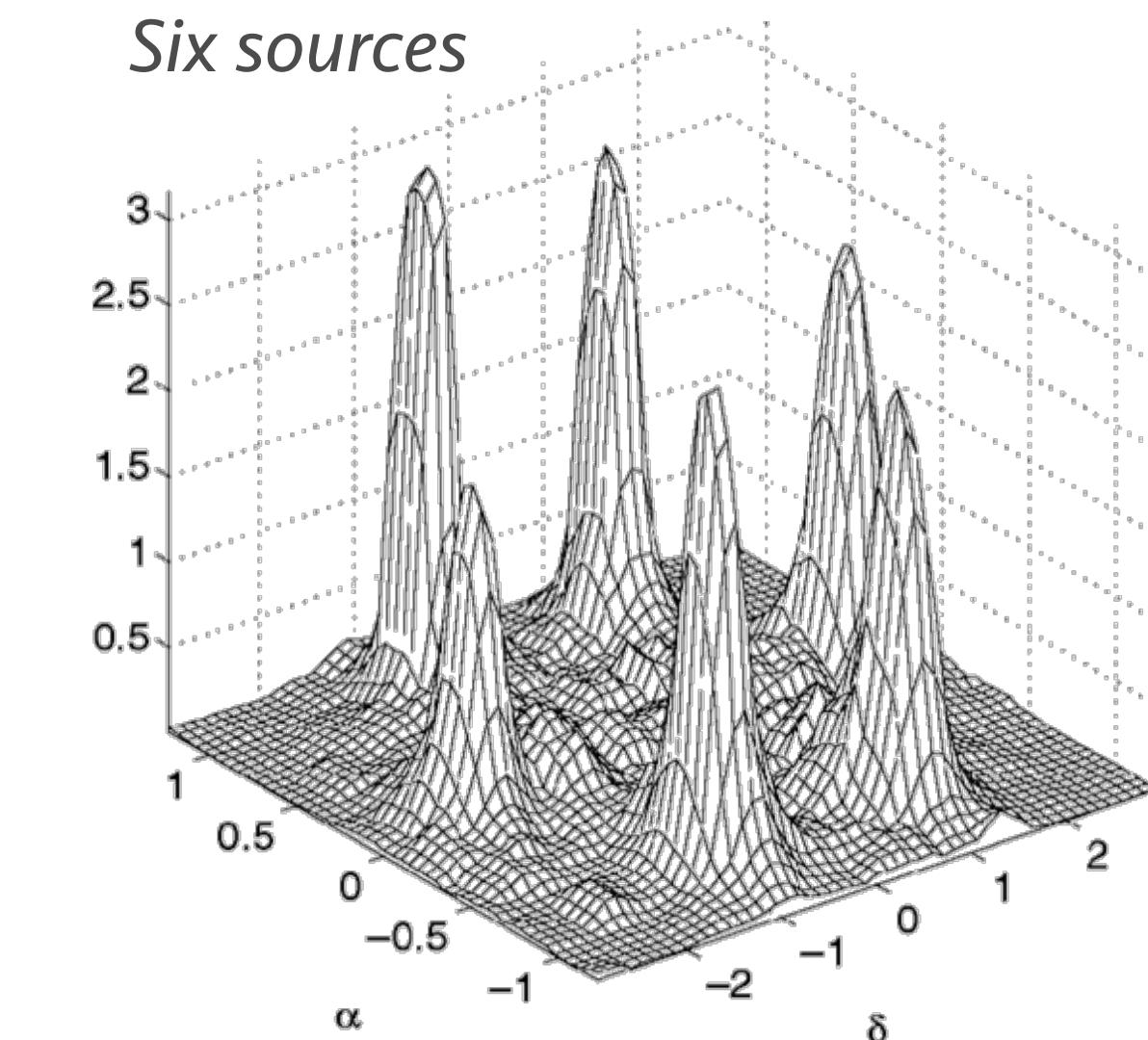
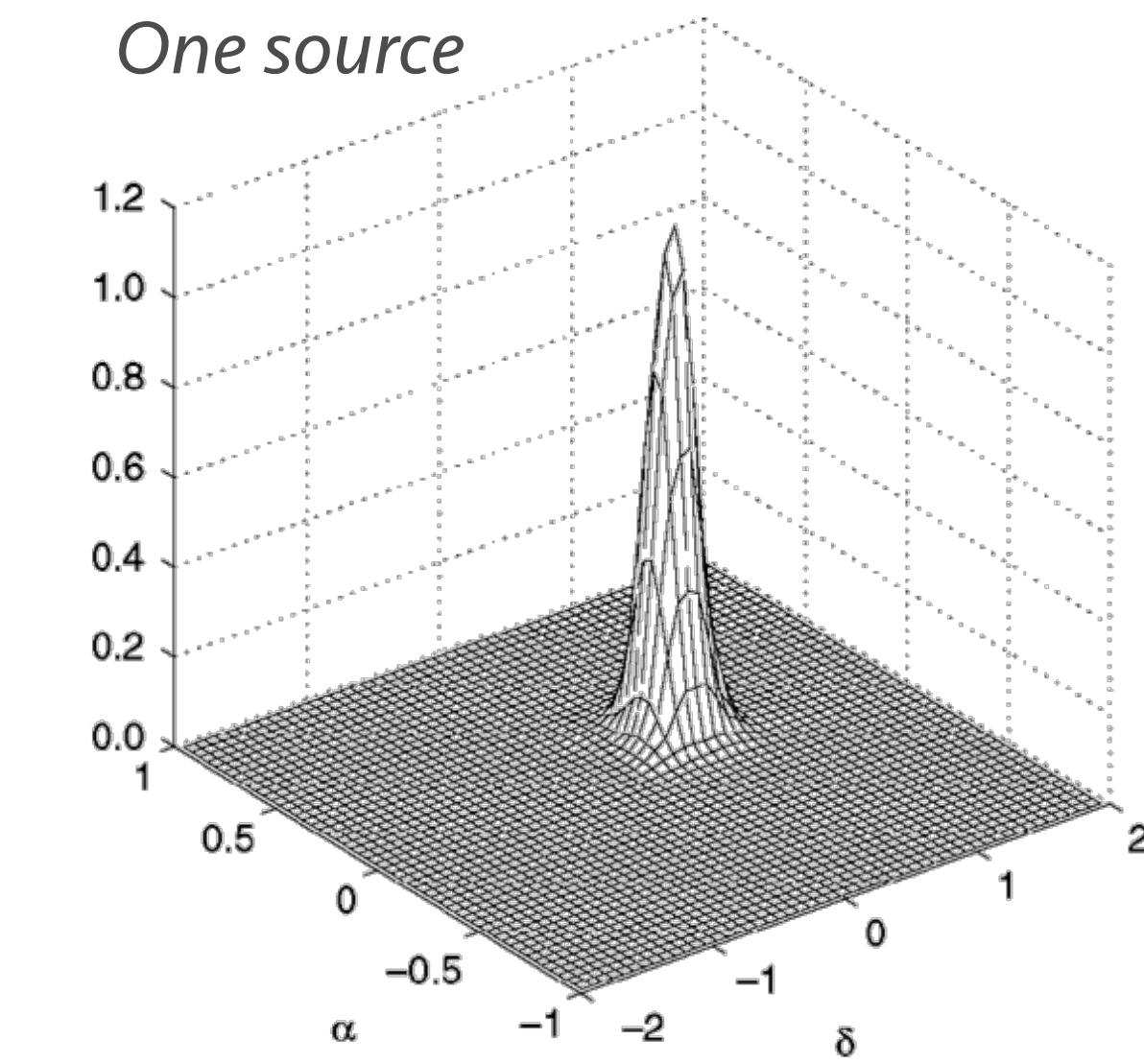
Normalized phases as spatial features

- Each spatial location will generate a unique value
- We will get clear peaks for each location's contribution



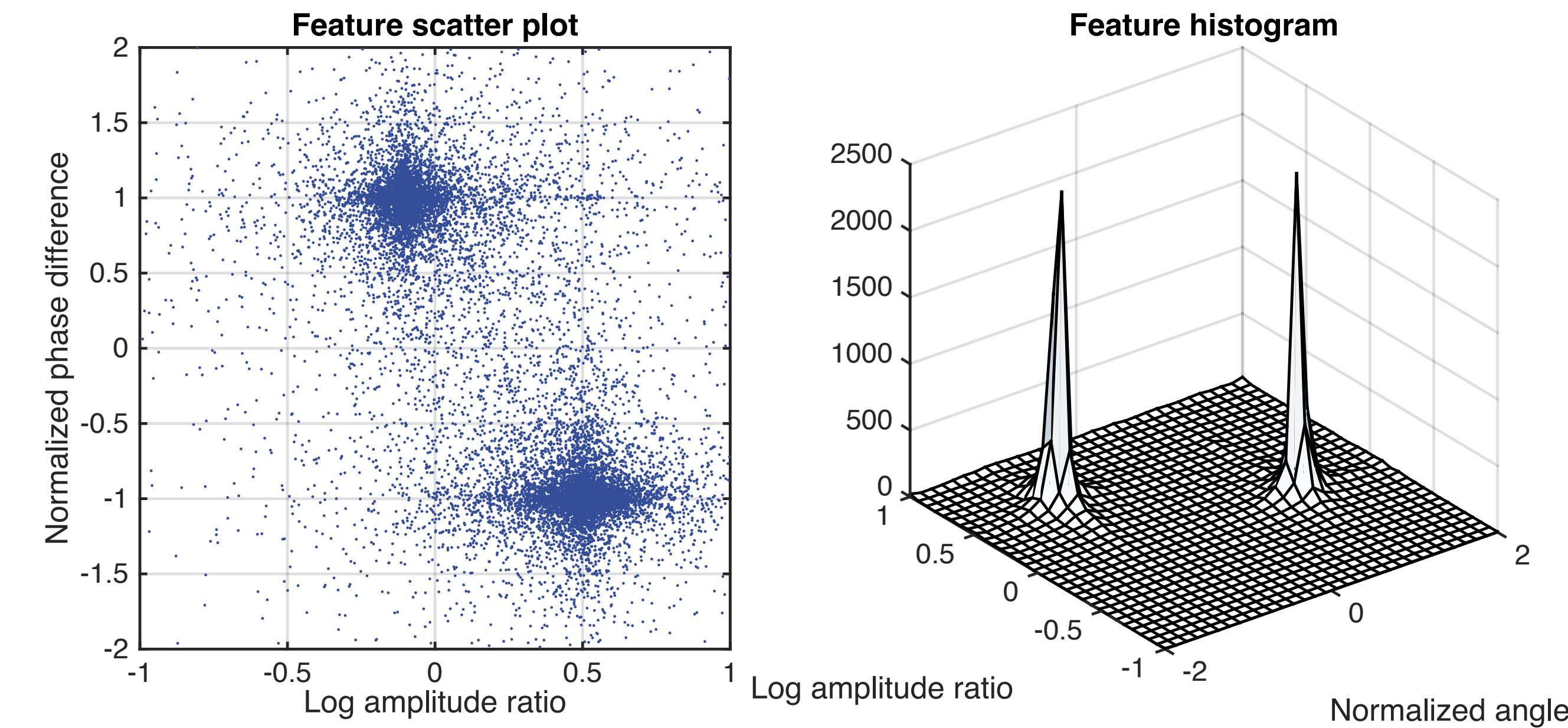
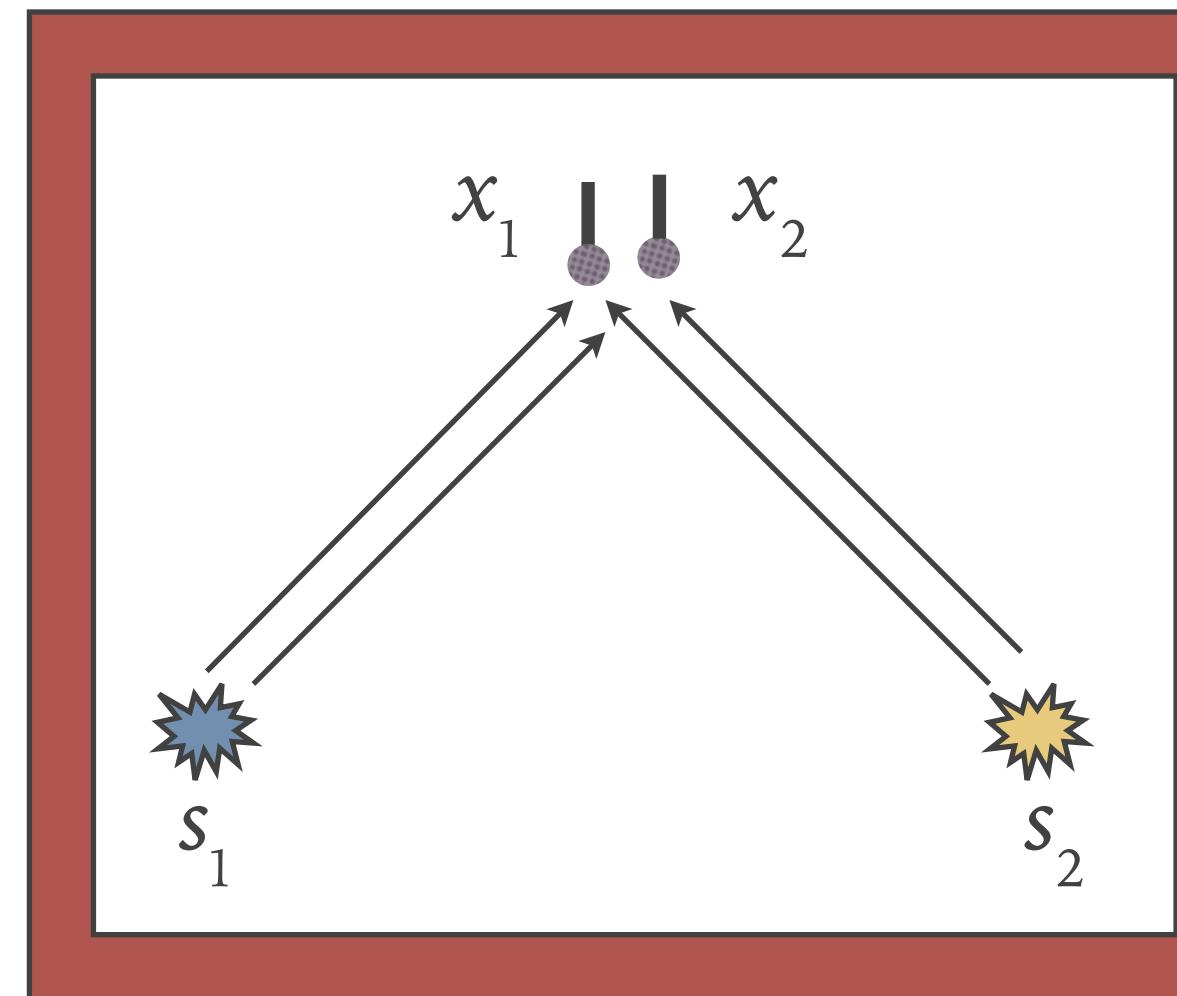
Clustering the location statistics

- Histogram all measurements in the joint parameter space
- Peaks are spatially separate sources
 - N sources will result in N peaks
- Each peak corresponds to a location
 - We can now group time-frequency “pixels” according to the peaks
 - Each mask is made out of the time-freq “pixels” which are closest to a mode



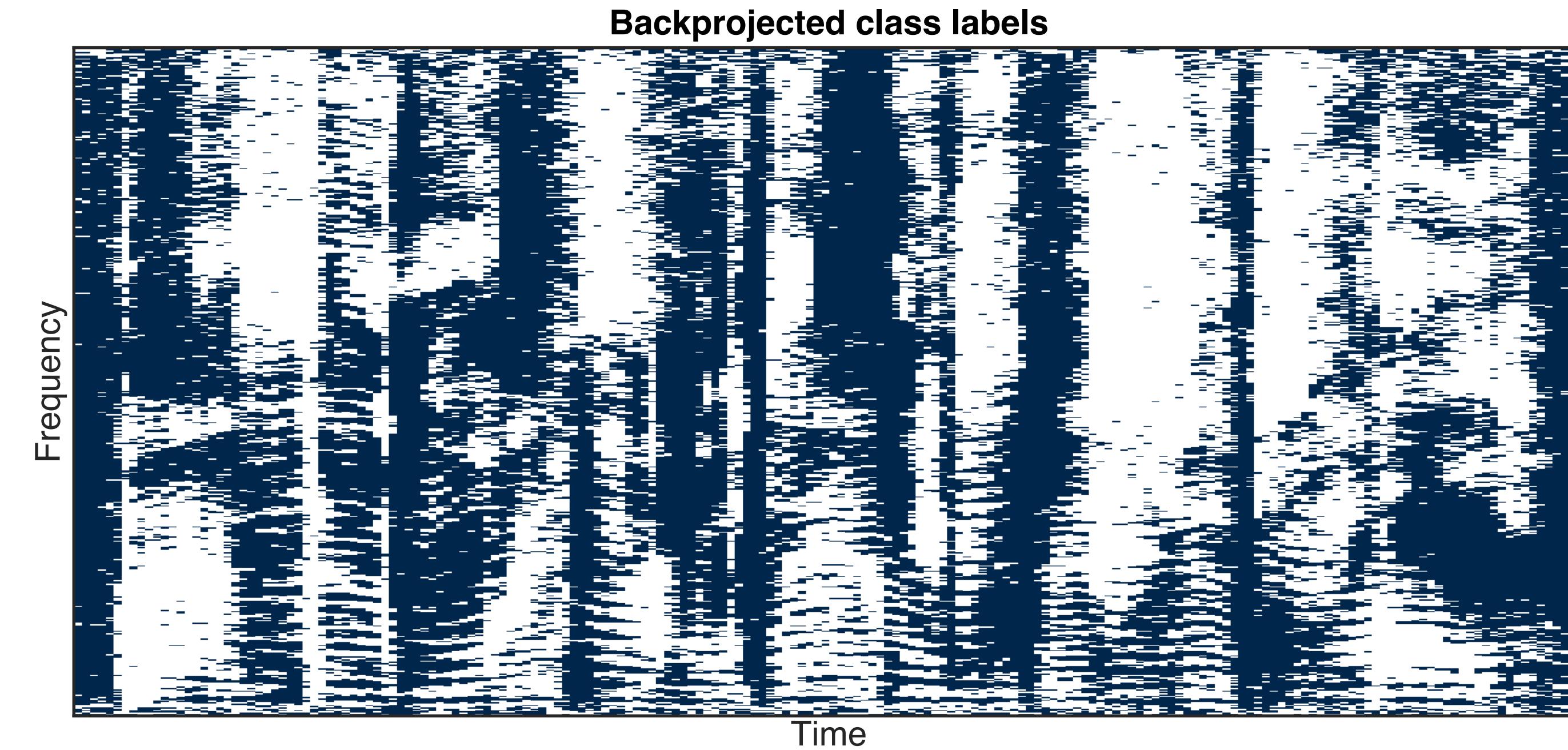
2 microphone example

- Closely spaced mics, symmetric sources
 - Each source creates a cluster with a unique center



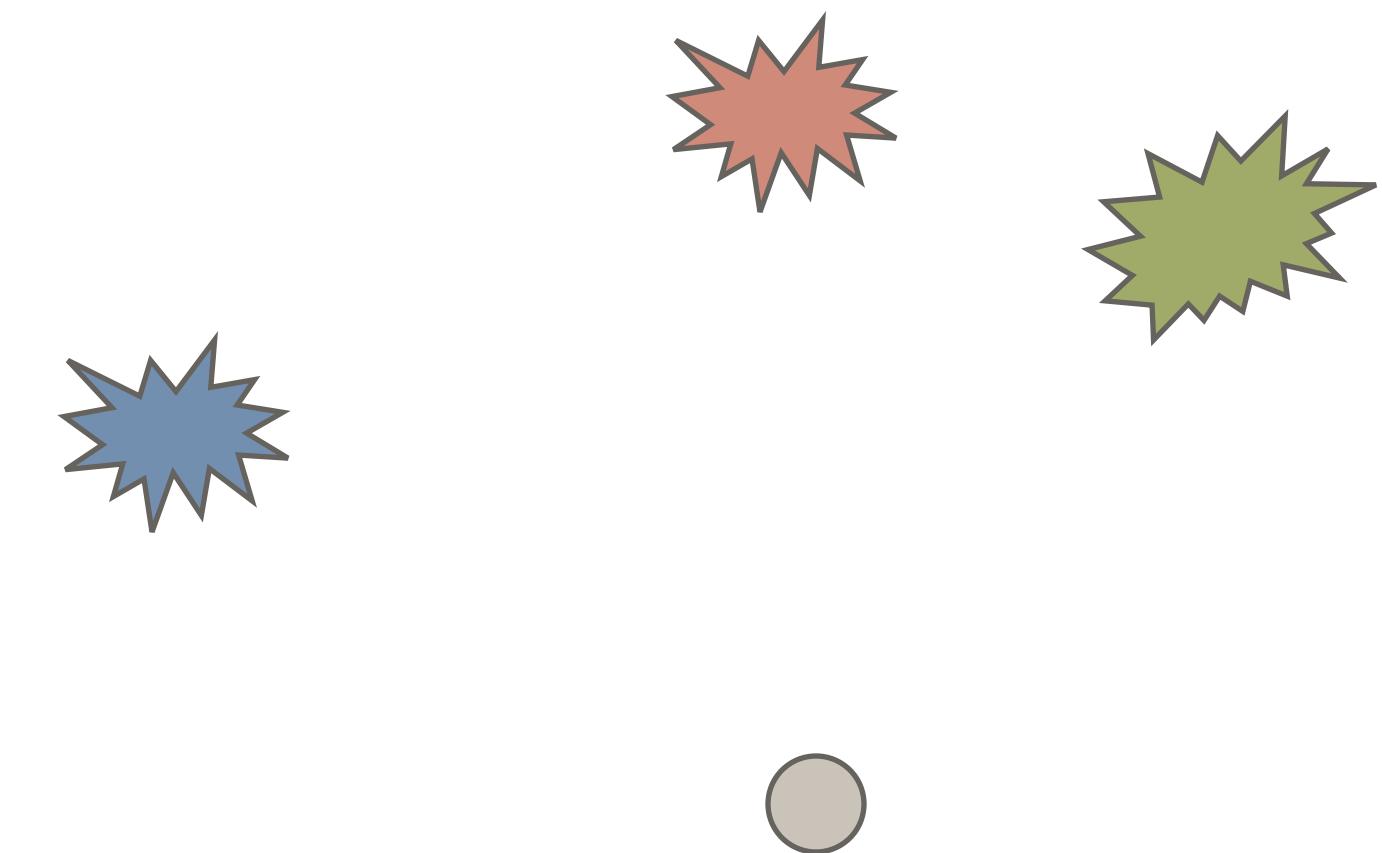
Back-projecting to the spectrograms

- Use the cluster labels to assign each point to a source
 - Place back into spectrogram and you get the desired binary mask

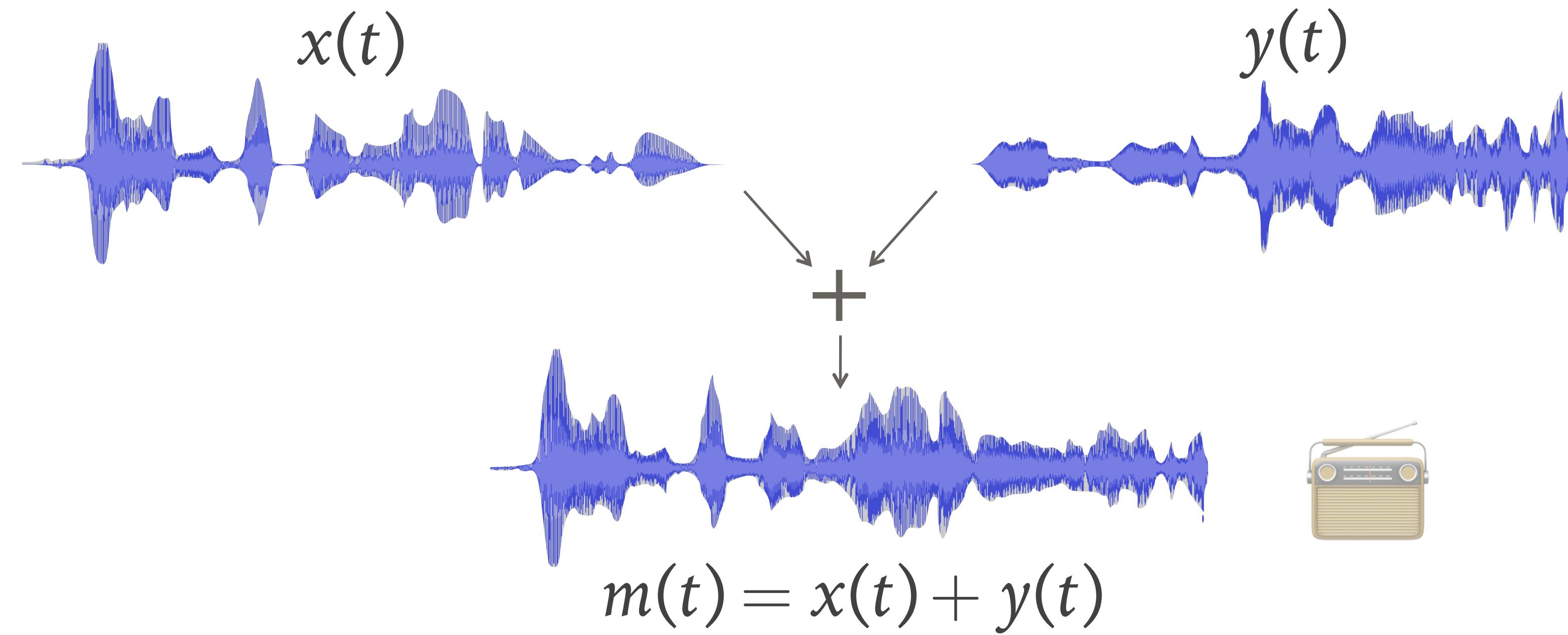


But sensors are so expensive ...

- What if we don't have an array at all?
 - One sensor – multiple inputs case

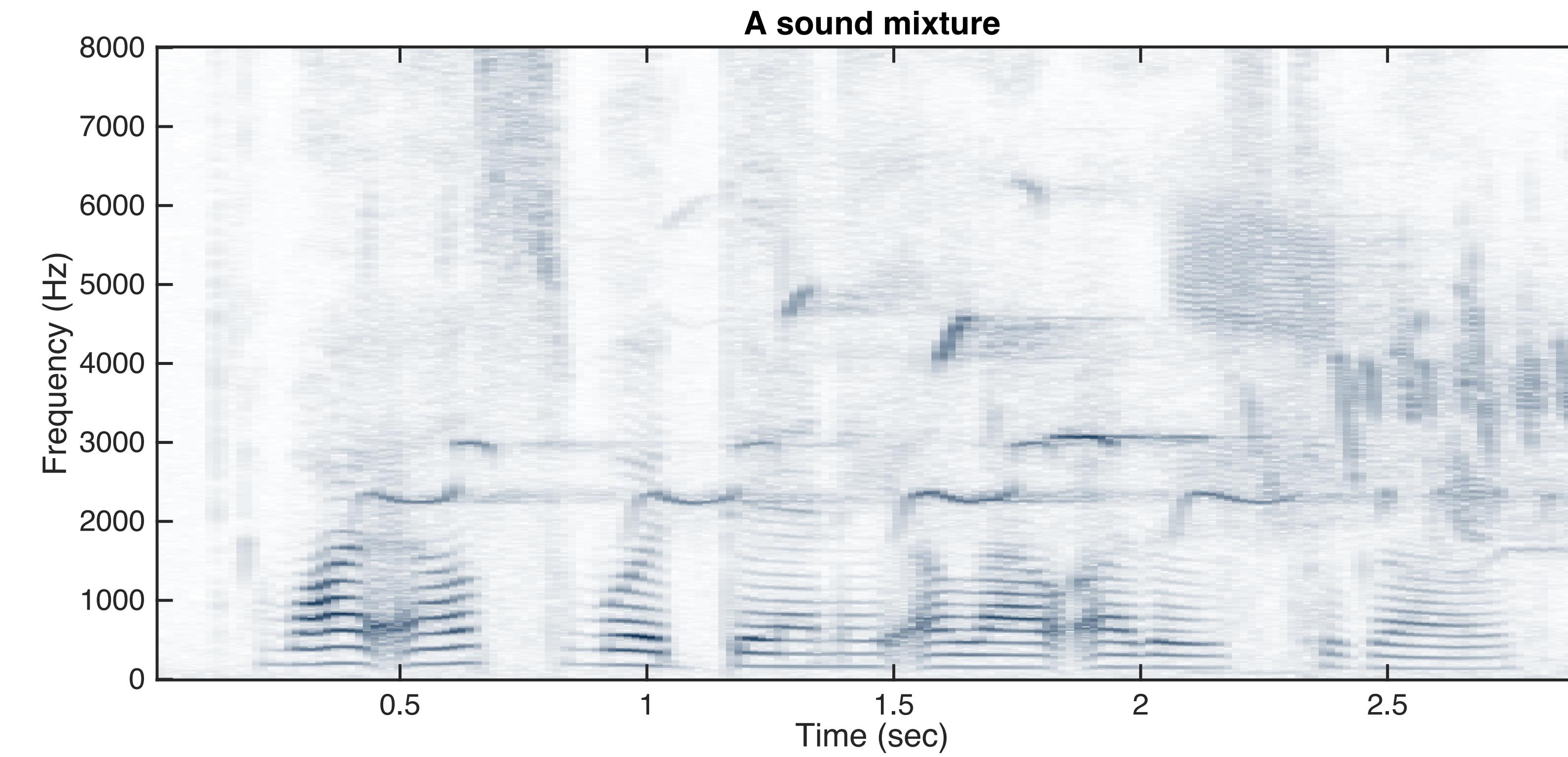


Defining the problem



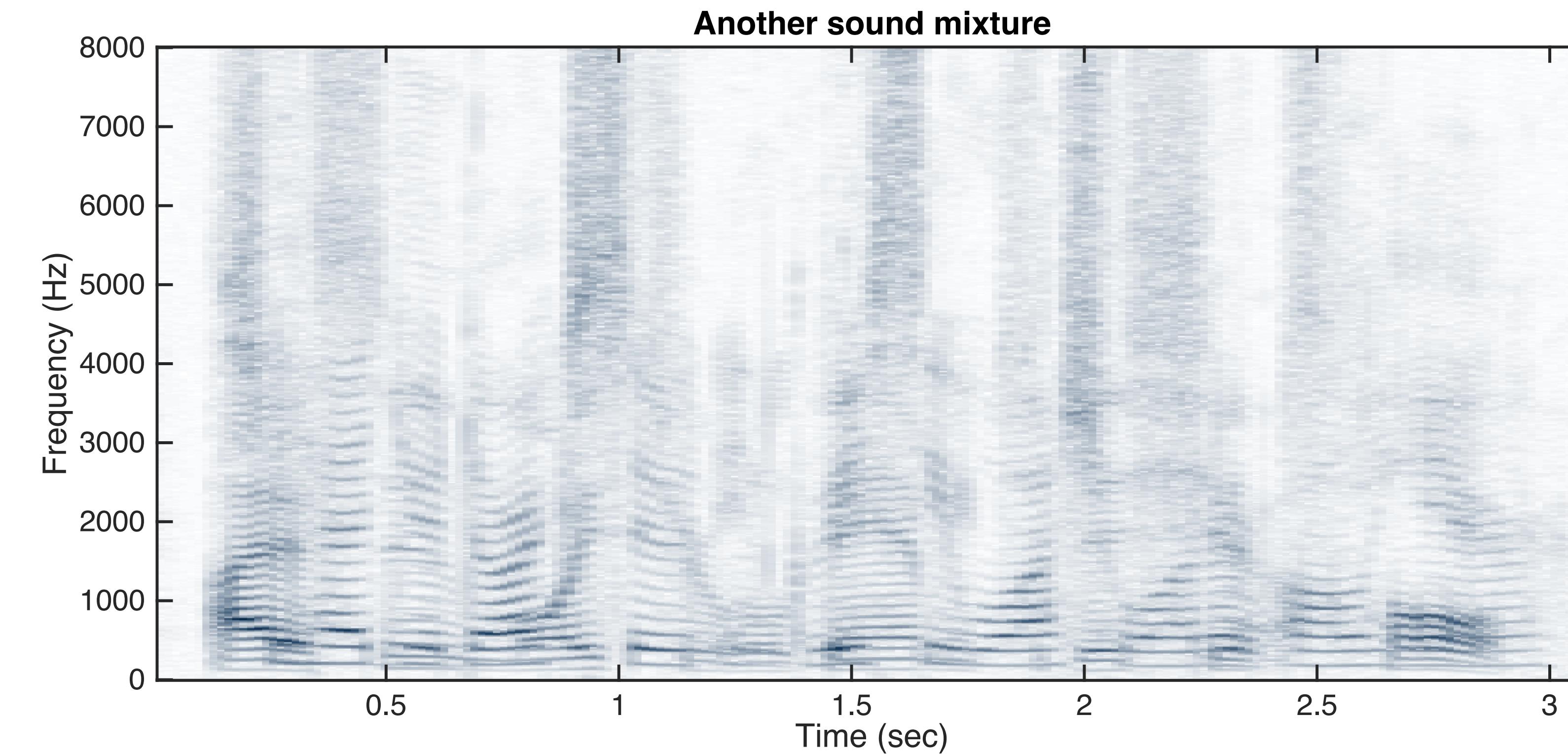
- Another ill-defined problem!
 - “Single-channel source separation”

The name of the game



- Finding signal priors to perform separation
 - School a: Perceptually-minded approaches
 - School b: Statistical approaches

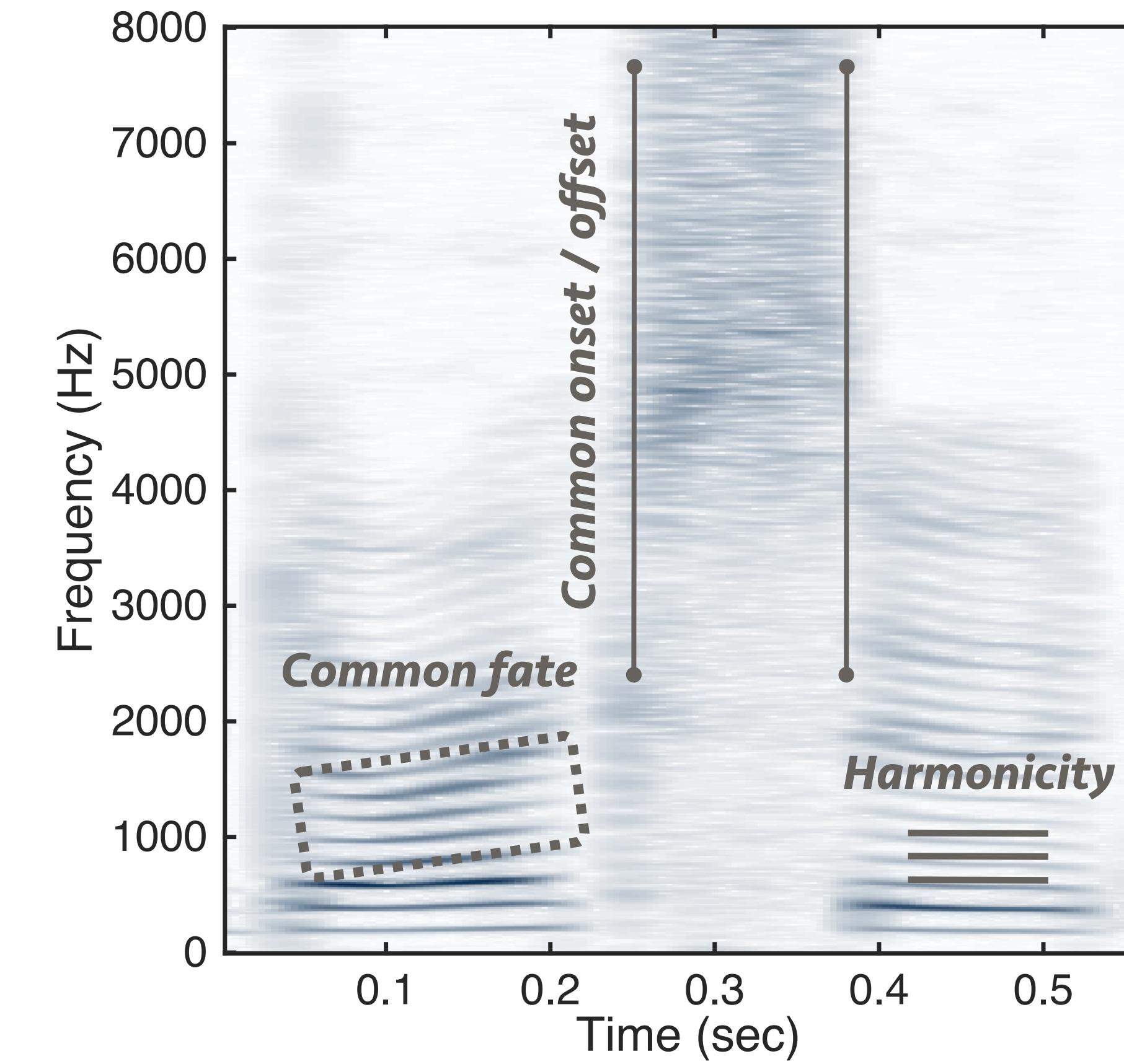
The name of the game



- Finding signal priors to perform separation
 - School a: Perceptually-minded approaches
 - School b: Statistical approaches

Perceptual approaches

- “Computational Auditory Scene Analysis”
 - Driven by psychoacoustic experiments



Some (general) statistical approaches

- Approaches with general source assumptions
 - Lee and Jang
 - ICA dictionaries of time waveforms
 - Reyes, Jojic and Ellis
 - Graphical model on TF distributions
 - Lagrange, et al.
 - Normalized cuts
 - Bach and Jordan
 - Spectral clustering for perceptual grouping
- Things aren't great ...

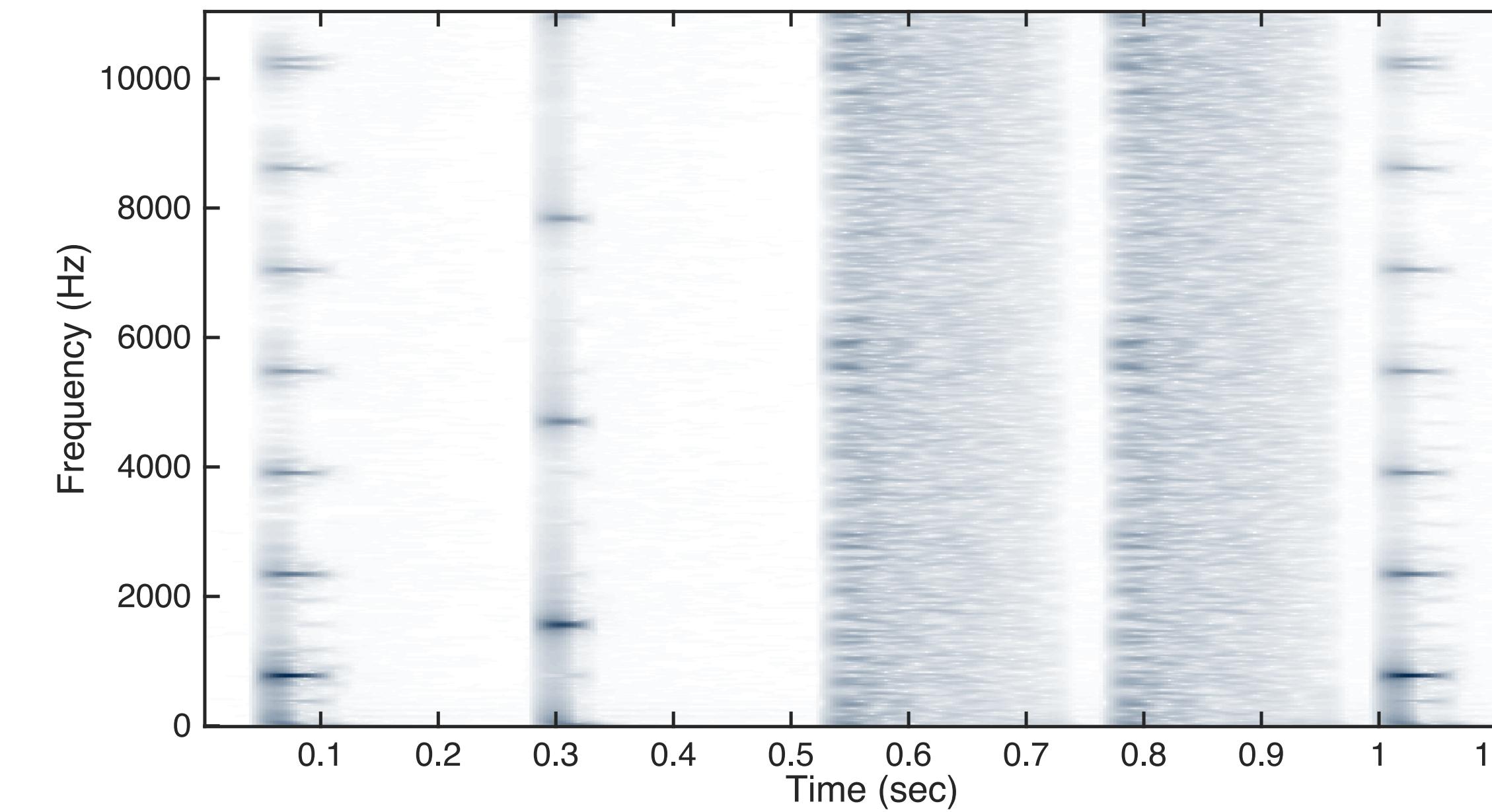


Forgoing unsupervised methods

- It is hard to define source structure
 - We should learn it instead
- Supervised source separation
 - Use training data as hints on what you want

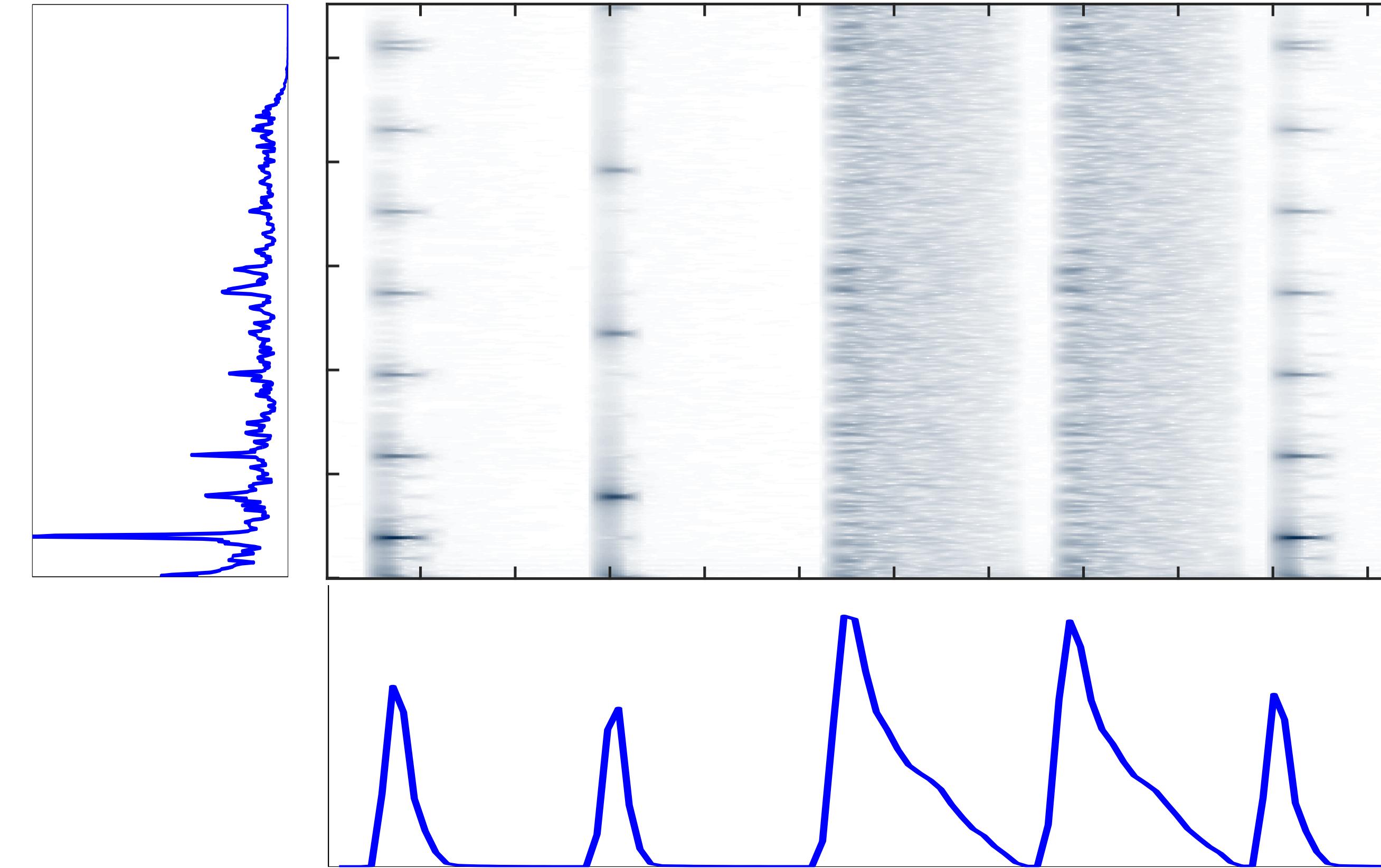
Learning models of sounds

- Starting with a sound having simple elements



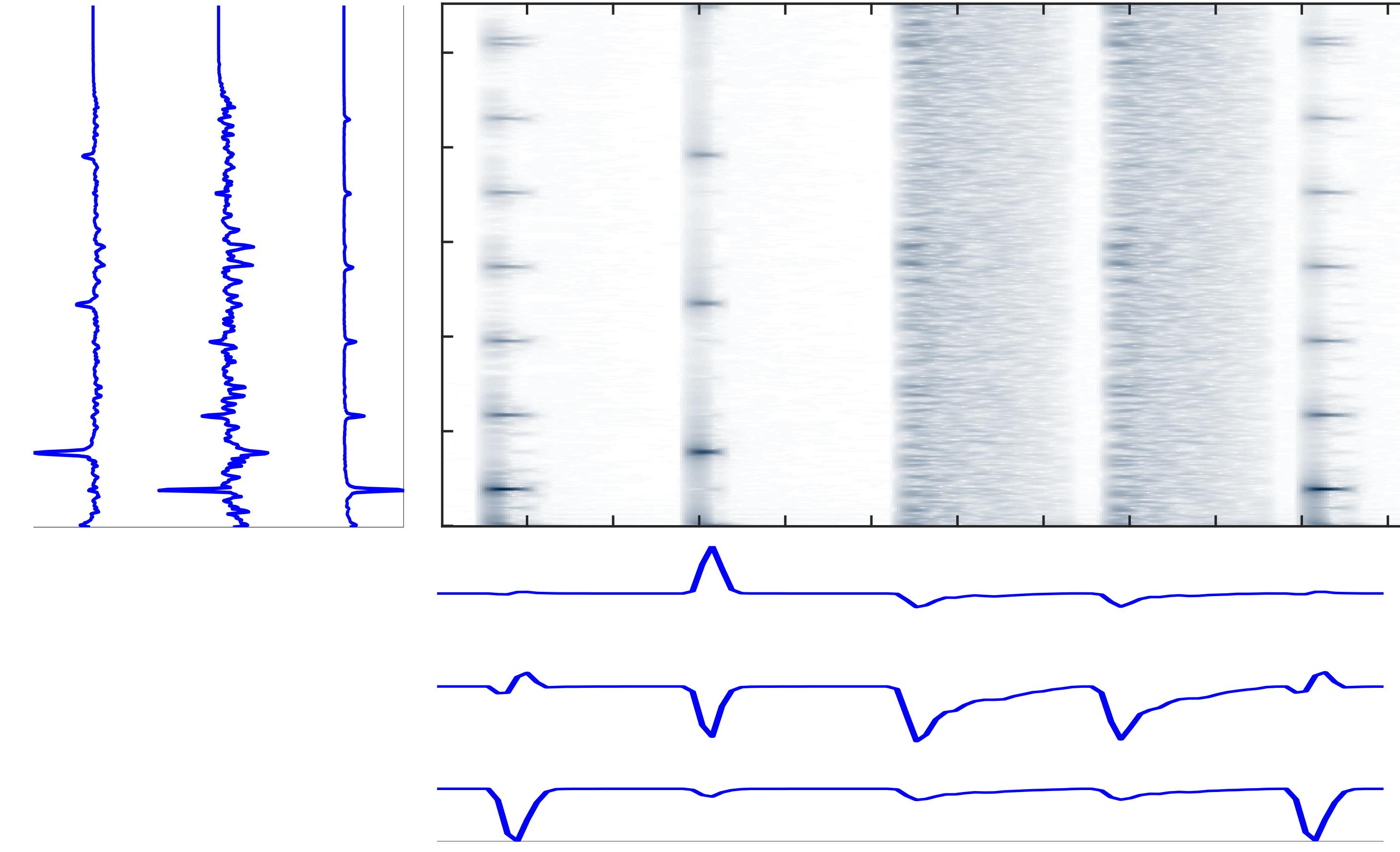
A simple factorization model

- Factor input as: $F = w \cdot h$



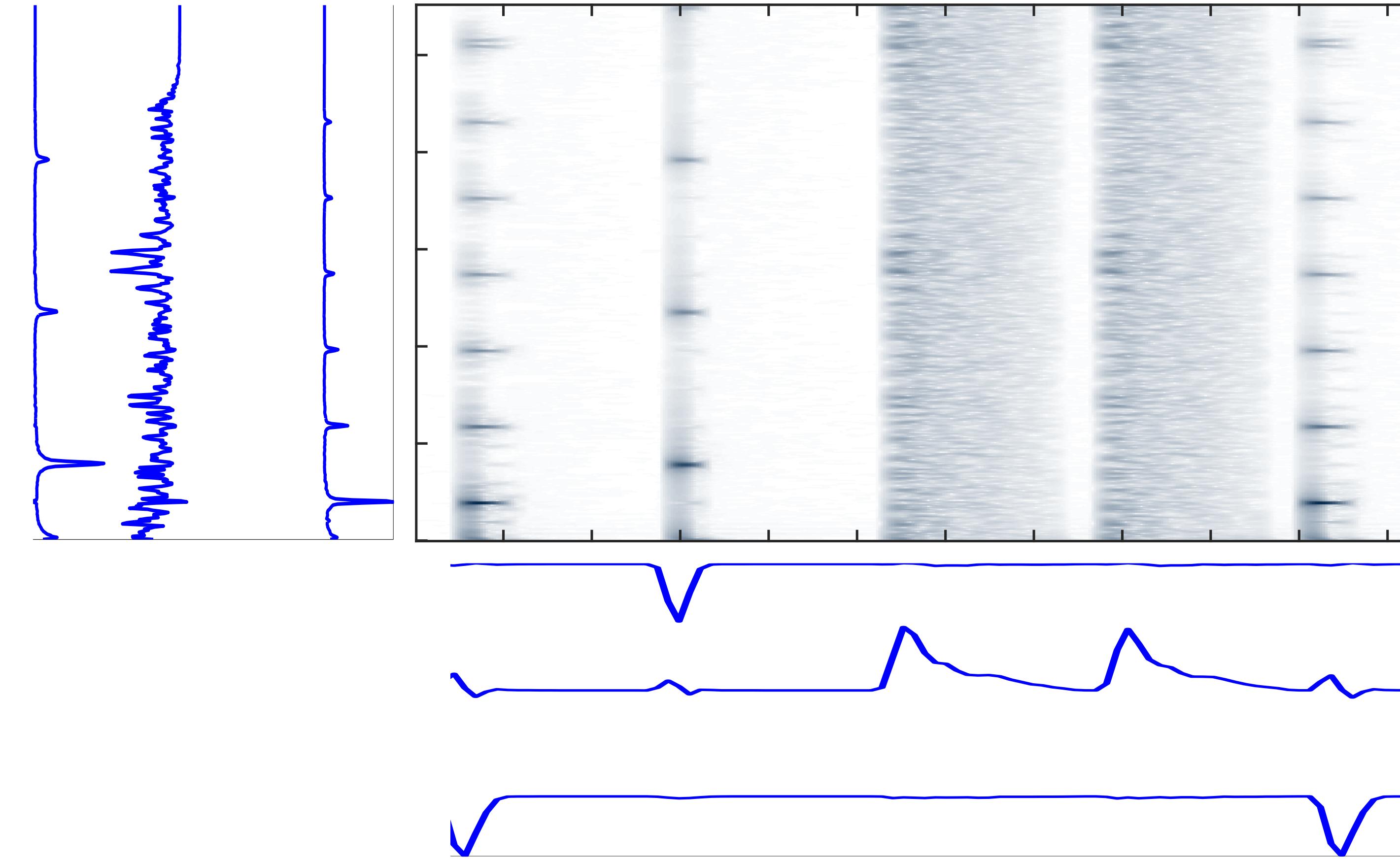
Upping the rank

- Use PCA instead: $F = W \cdot H$



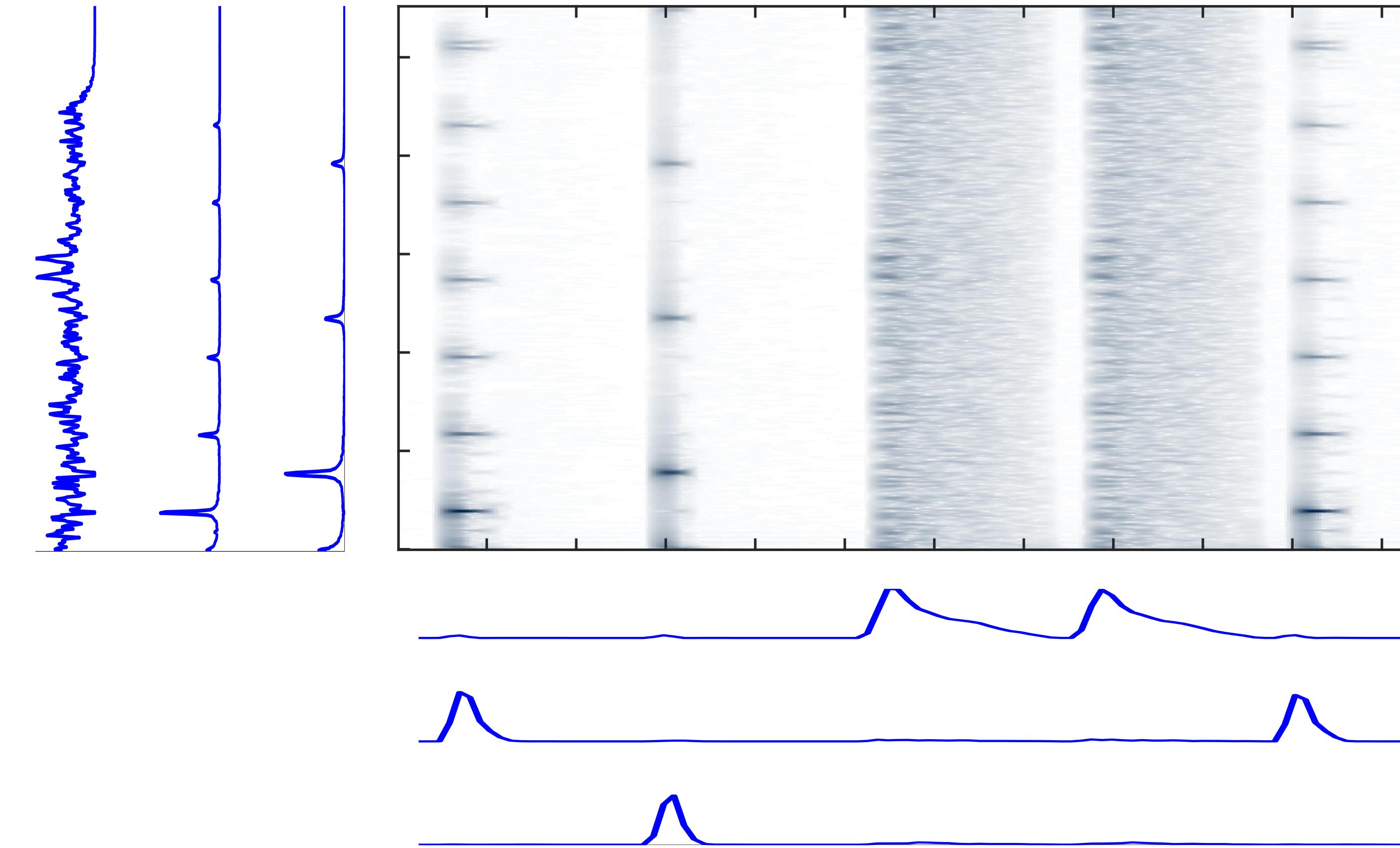
Trying something else

- Use ICA instead: $F = W \cdot H$



One last try ...

- Use NMF instead: $F = W \cdot H$



Interpreting the model

- Rank- R model returns R components

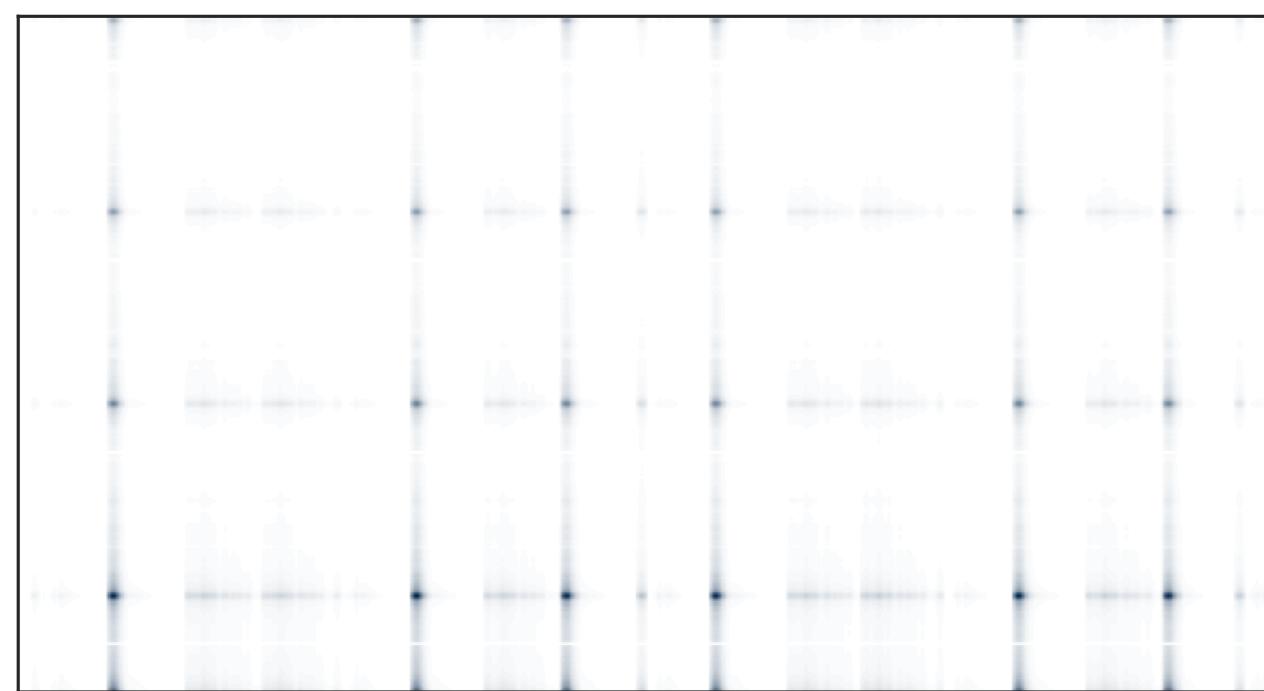
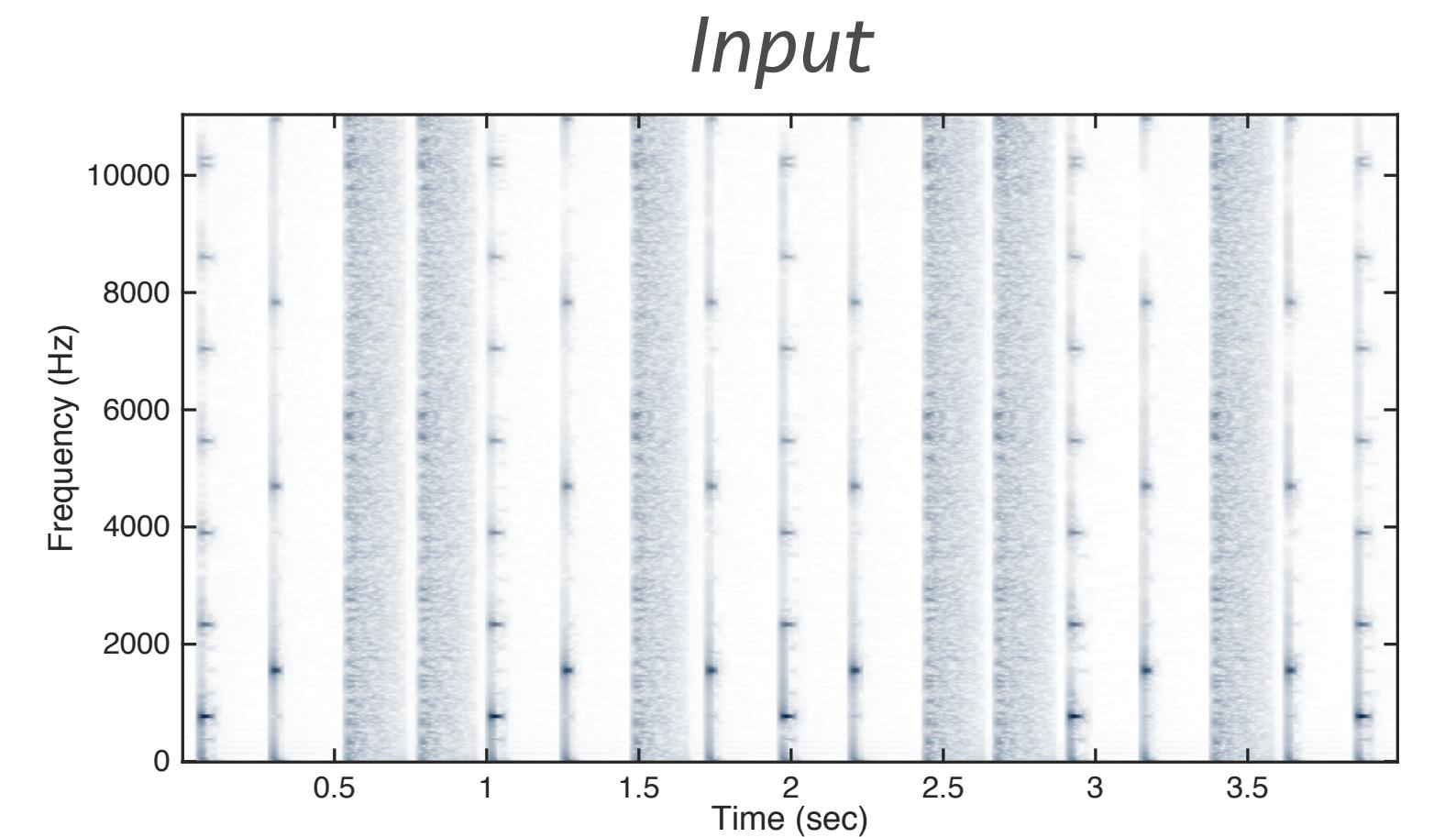
$$\mathbf{F} = \mathbf{W} \cdot \mathbf{H}$$

$$\mathbf{F} \in \mathbb{R}^{M \times N, \geq 0}, \mathbf{W} \in \mathbb{R}^{M \times R, \geq 0}, \mathbf{H} \in \mathbb{R}^{R \times N, \geq 0}$$

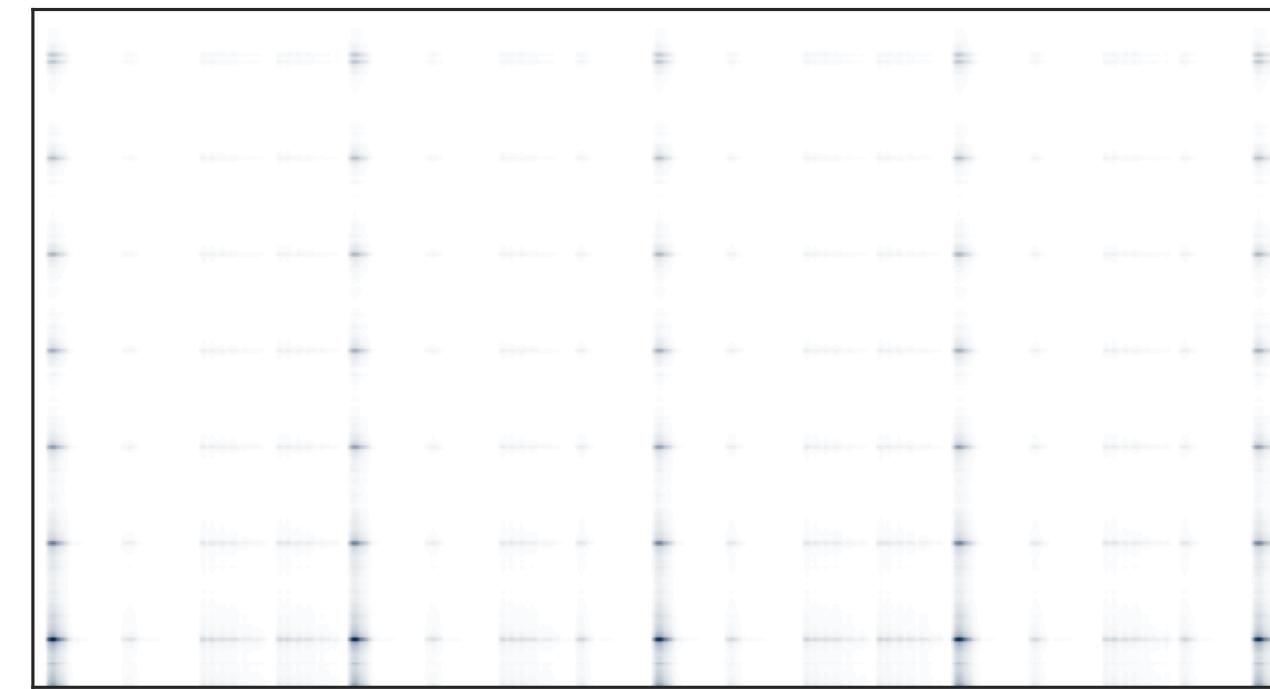
- \mathbf{W} matrix holds spectral templates
 - vertical structure
- \mathbf{H} matrix their time activations
 - horizontal structure
- Or vice-versa!

Element-wise reconstructions

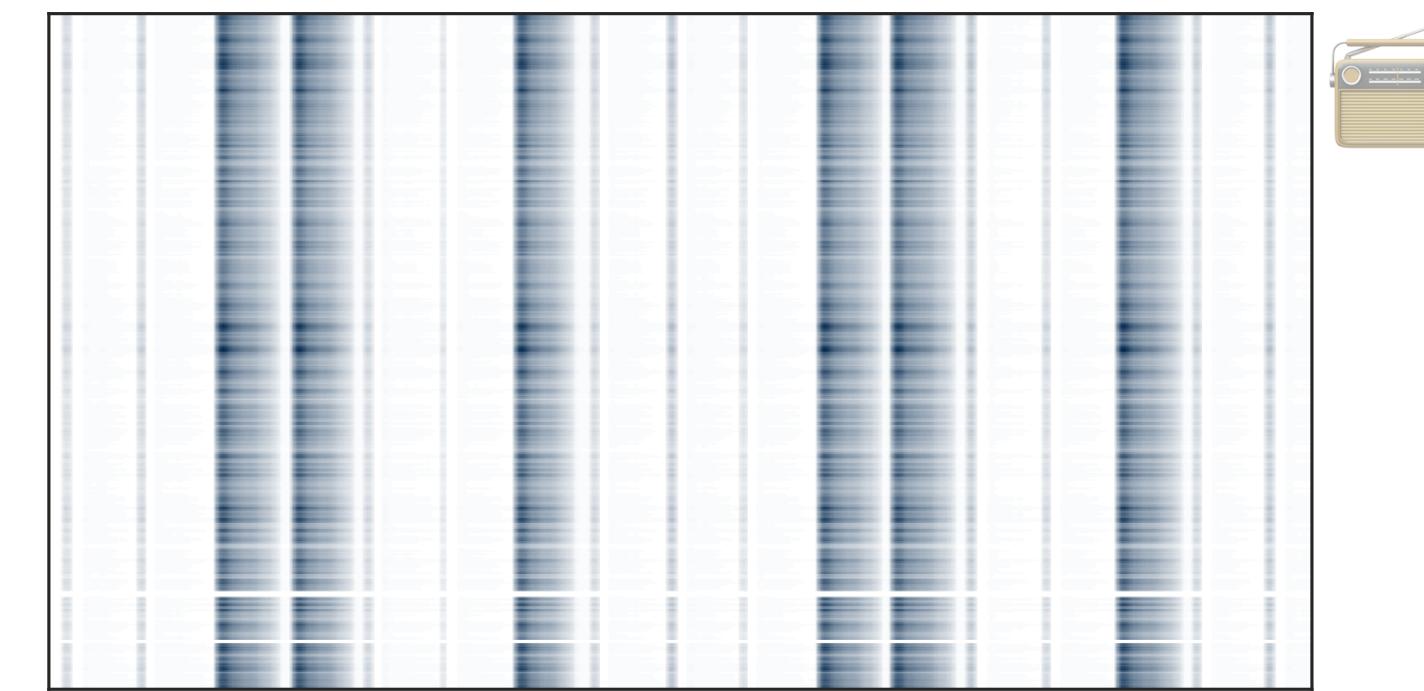
- Each component latches to a “subsound”
 - and can then be isolated!



$$\mathbf{W}_{:,1} \cdot \mathbf{H}_{1,:}$$

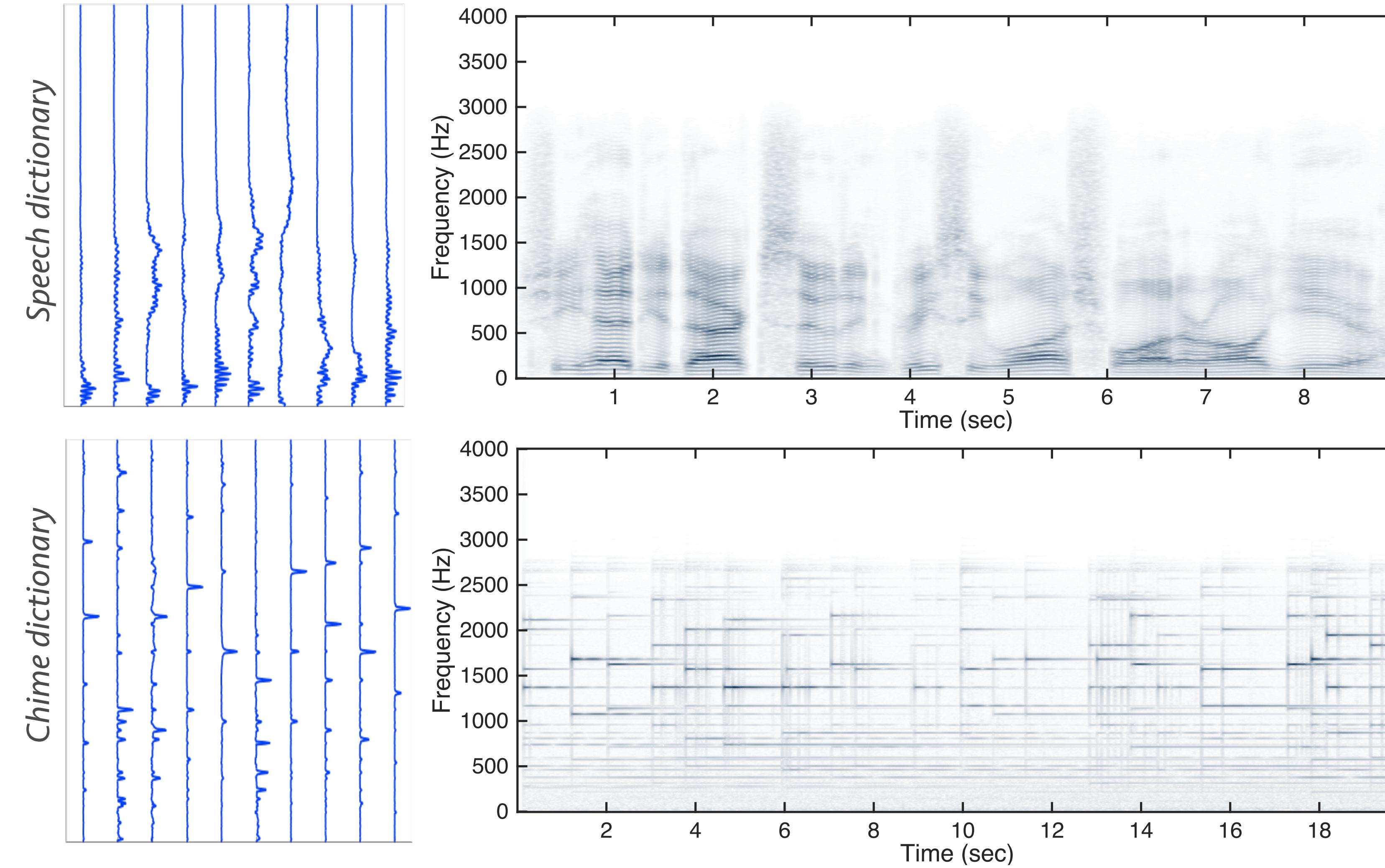


$$\mathbf{W}_{:,2} \cdot \mathbf{H}_{2,:}$$



$$\mathbf{W}_{:,3} \cdot \mathbf{H}_{3,:}$$

Making NMF sound models



Speech recording



Chime recording

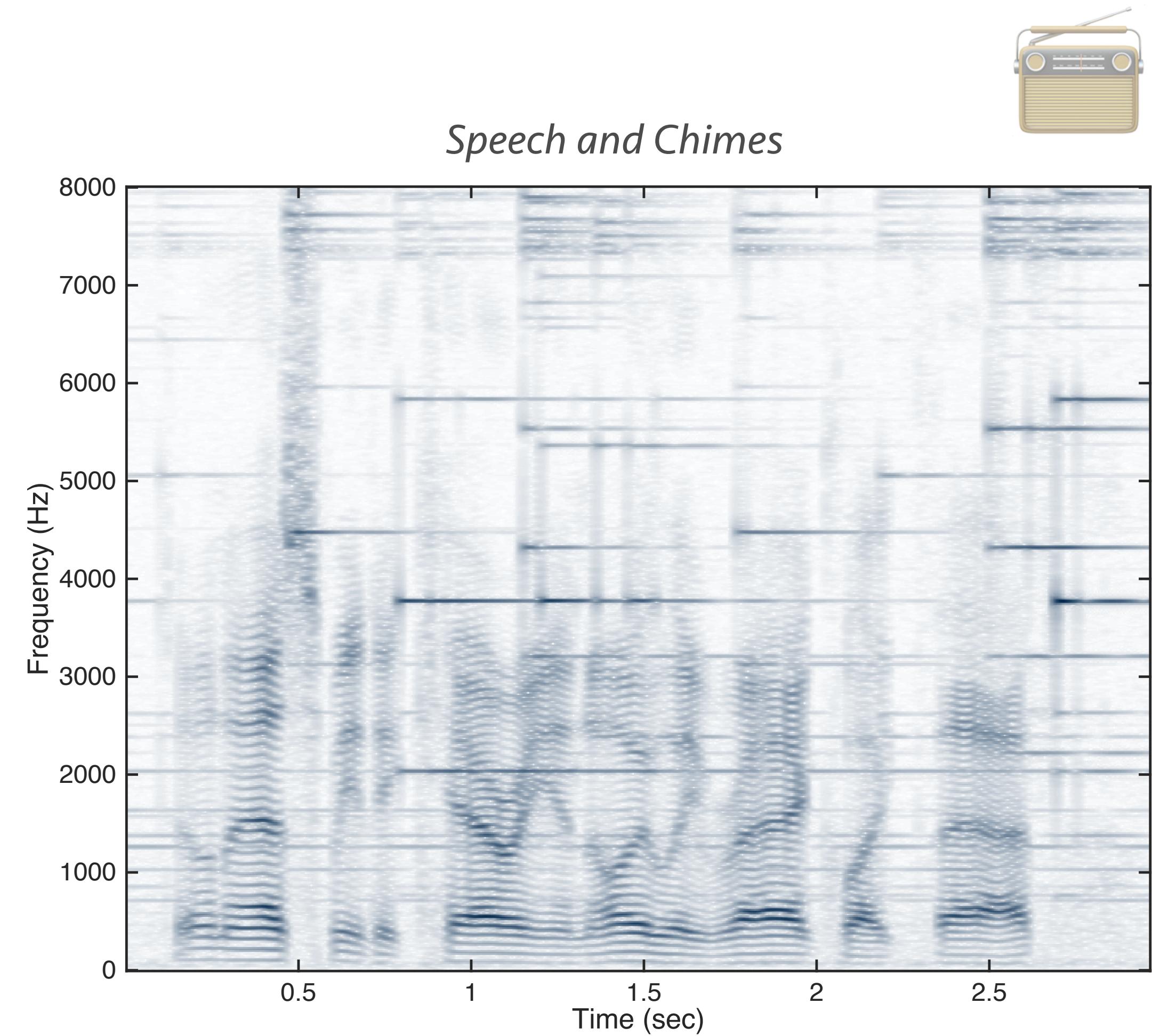
Mixtures of sounds

- Use spectrogram additivity
 - combine models to explain mixture

$$\mathbf{F} = \begin{bmatrix} \mathbf{W}_{chimes} & \mathbf{W}_{speech} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{H}_{chimes} \\ \mathbf{H}_{speech} \end{bmatrix}$$

↙ *Known/fixed* ↙ *Estimated*

- We estimate only the weights
- The known bases claim only the parts that they can fit best

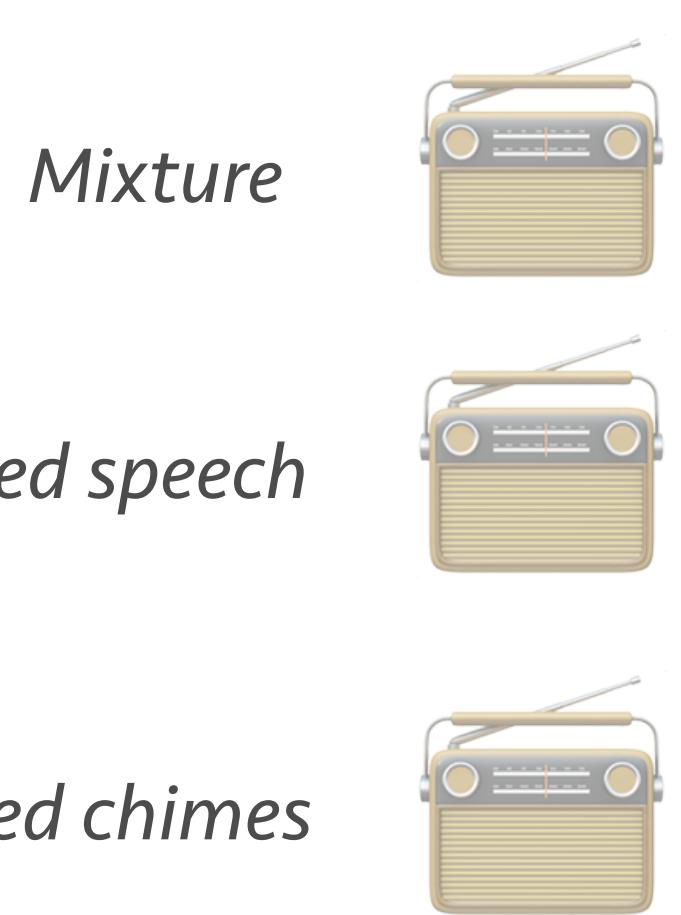


Separation

- Recompose sources individually

$$\mathbf{F}_{speech} = \mathbf{W}_{speech} \cdot \mathbf{H}_{speech}$$

$$\mathbf{F}_{chimes} = \mathbf{W}_{chimes} \cdot \mathbf{H}_{chimes}$$



- And convert spectrograms to time domain
 - Use the phase of the mixture
 - Unlike before this is a soft mask

Two problems

- Sounds in mixture have to be distinct
 - but not my much!

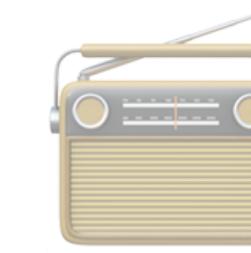
Speech & speech mixture



Extracted speaker 1



Extracted speaker 2



- What are the chances we know all sounds?
 - Usually we know a target or a noise

Separation with unknown sounds

- Same as before, use only one model:

$$\mathbf{F} = \begin{bmatrix} \mathbf{W}_{known} & \mathbf{W}_{unknown} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{H}_{known} \\ \mathbf{H}_{unknown} \end{bmatrix}$$

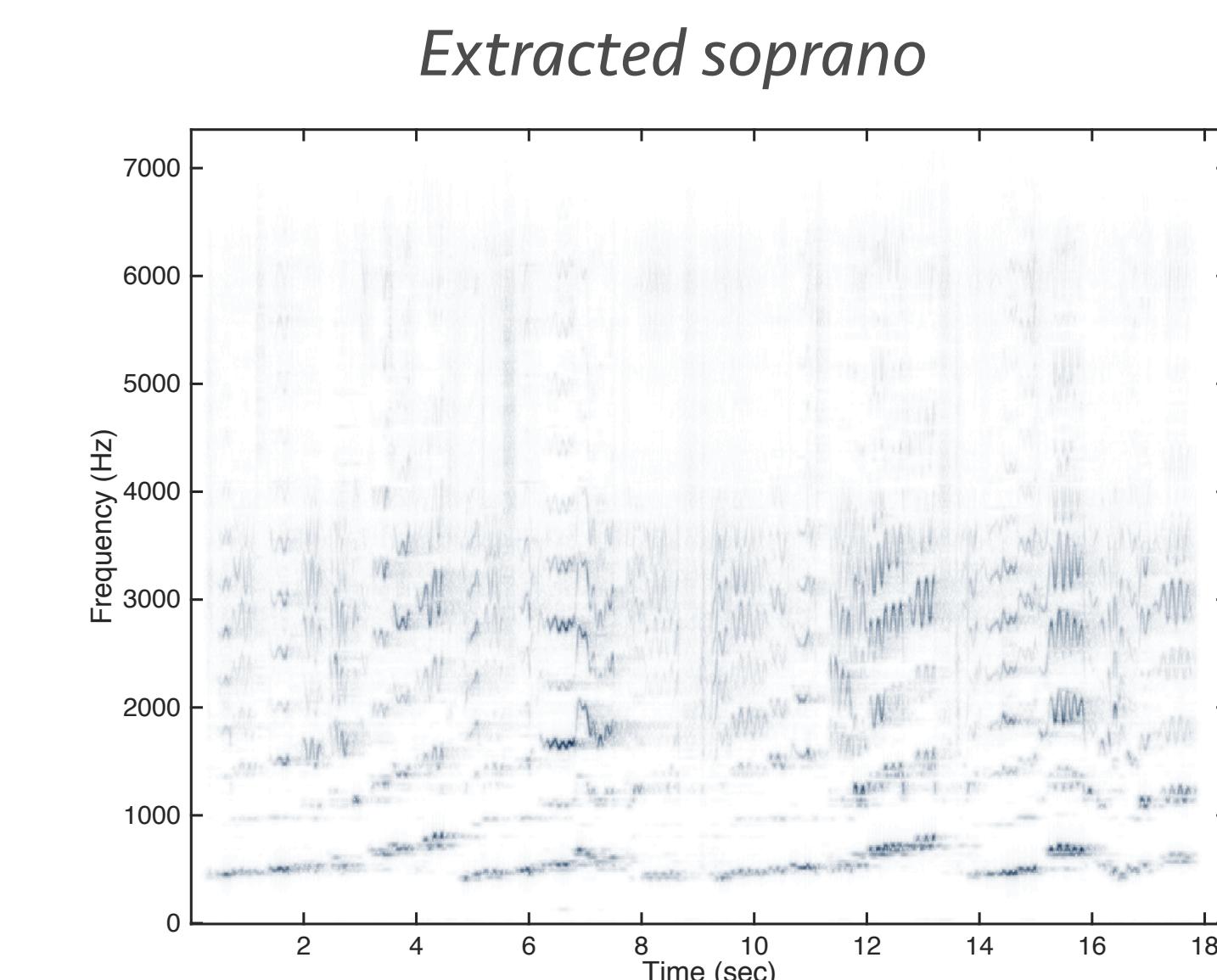
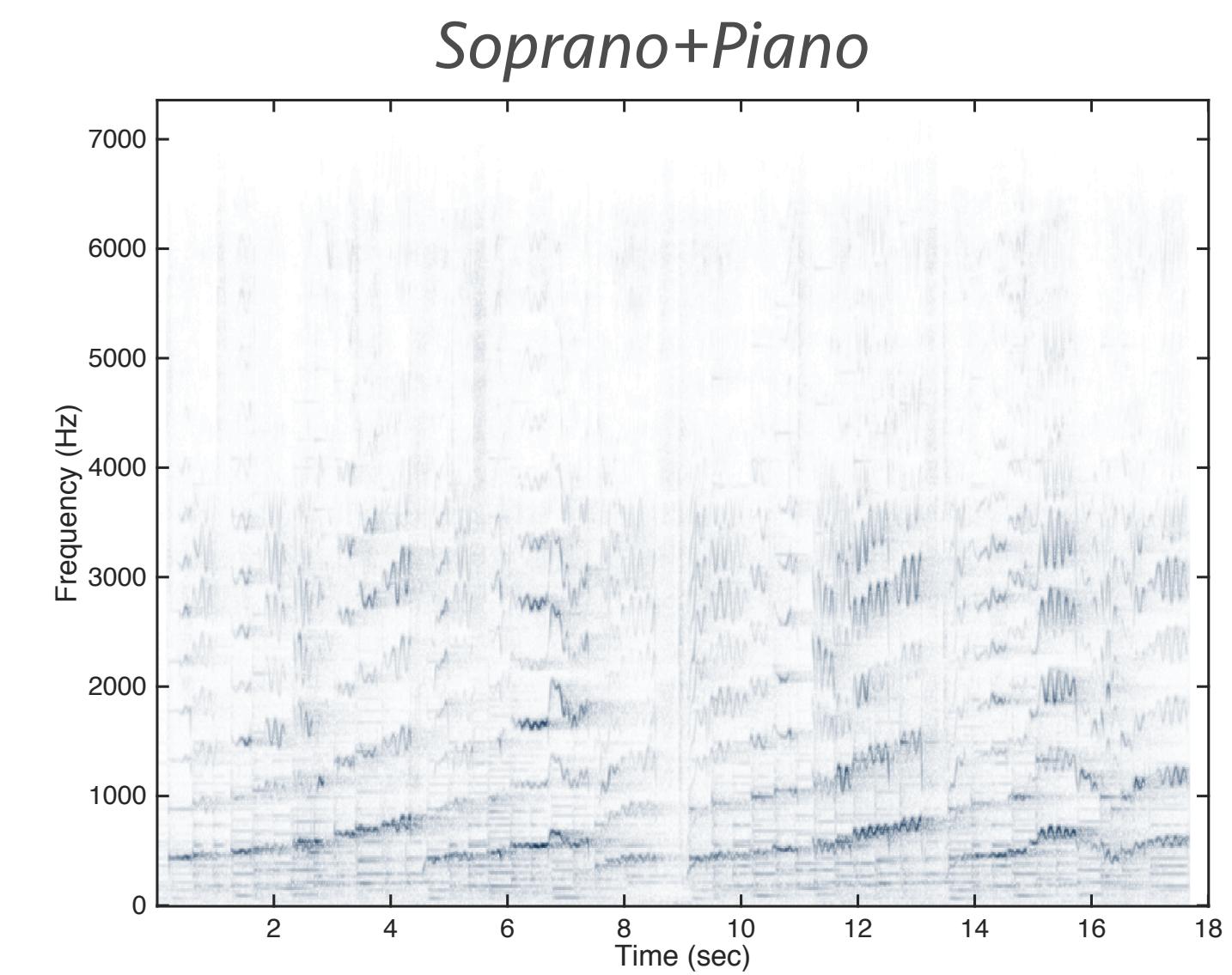
↓ *Known/fixed* ↓ *Estimated*

- Learn weights and unknown bases
 - Unknown bases converge to the unknown parts in the mixture

Soprano & Piano



Extracted soprano



Setting up the problem

- Can be done in two ways
 - Have model of noise, extract extras
 - Have model of target, remove extras
- All cases can be binary
 - What you want vs. the rest
- Can be applied to denoising
 - Can deal with non-stationary noise

Speech + the beauty of mechanics



Wideband noise



Extracted speech



Loosely correlated "noise"

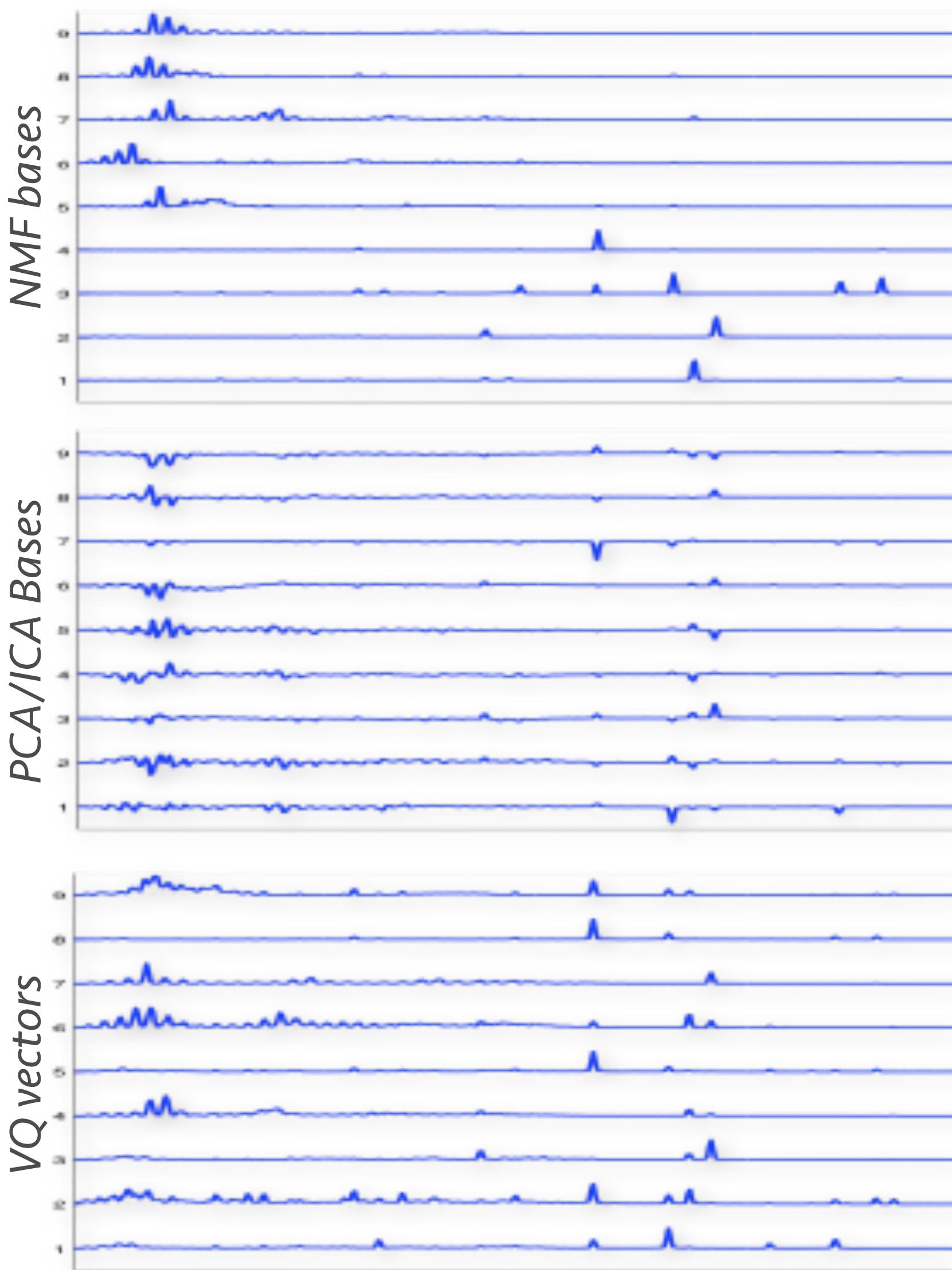


Denoised sound



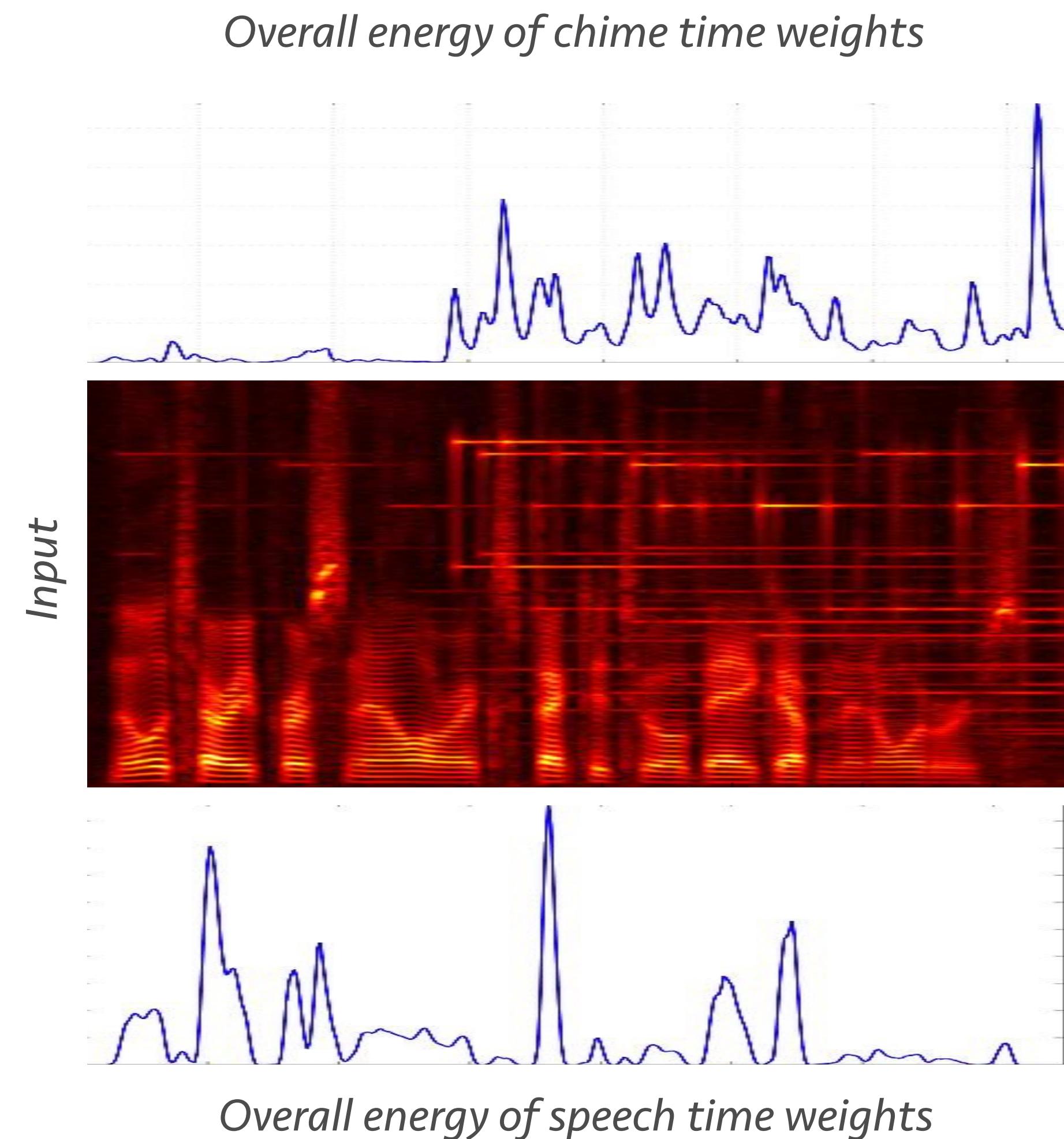
Why this model?

- We need to measure the presence of something
 - Therefore our domain is inherently non-negative
- PCA, ICA, etc don't work
 - The use of cross-cancellation gives nonsensical results
- VQ/K-means is not additive
 - Can't model mixtures
- NMF is best at this



Measuring presence

- Recognition in mixtures
 - We can't use classifiers!
 - No hard answer
 - Not even a soft one ...
- Measuring source presence
 - Observe source weights
 - Deduce amount of sounds



Sound recognition in mixtures

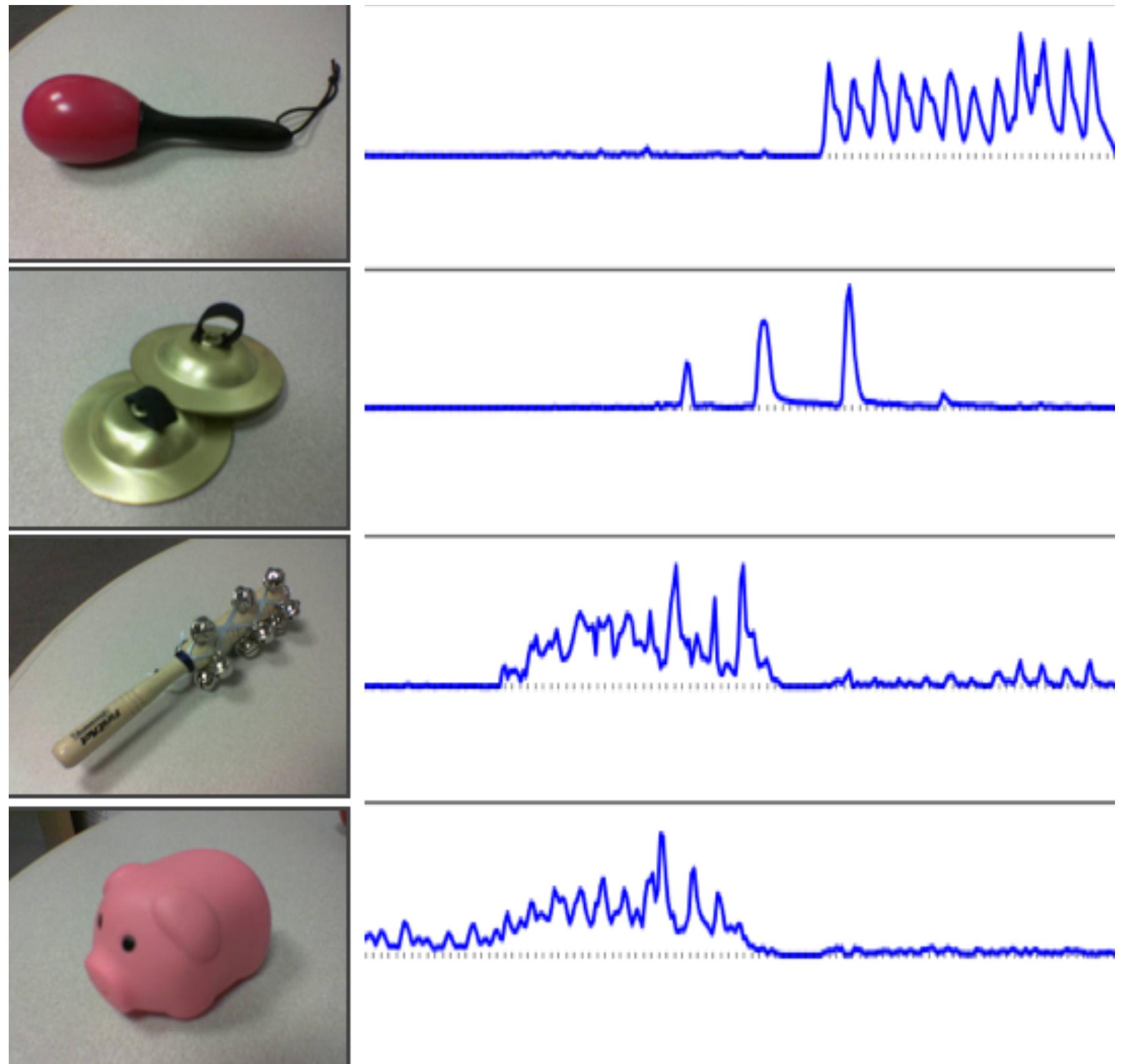


- Use known models to estimate presence of these sounds in a mixture

$$\mathbf{F} = \begin{bmatrix} \mathbf{W}_{shaker} & \mathbf{W}_{cymbals} & \mathbf{W}_{jingles} & \mathbf{W}_{pig} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{H}_{shaker} \\ \mathbf{H}_{cymbals} \\ \mathbf{H}_{jingles} \\ \mathbf{H}_{pig} \end{bmatrix}$$

Known/fixed
Estimated

- We are explaining the mixture, not doing simple classification

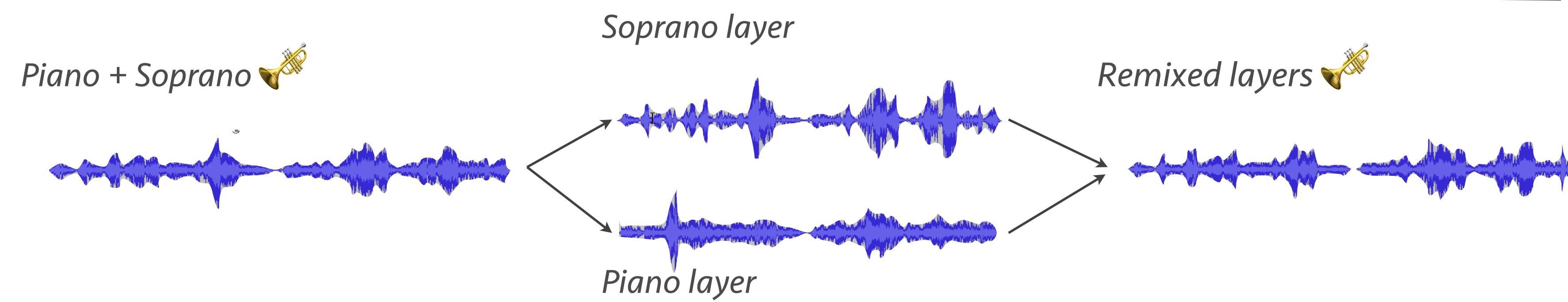
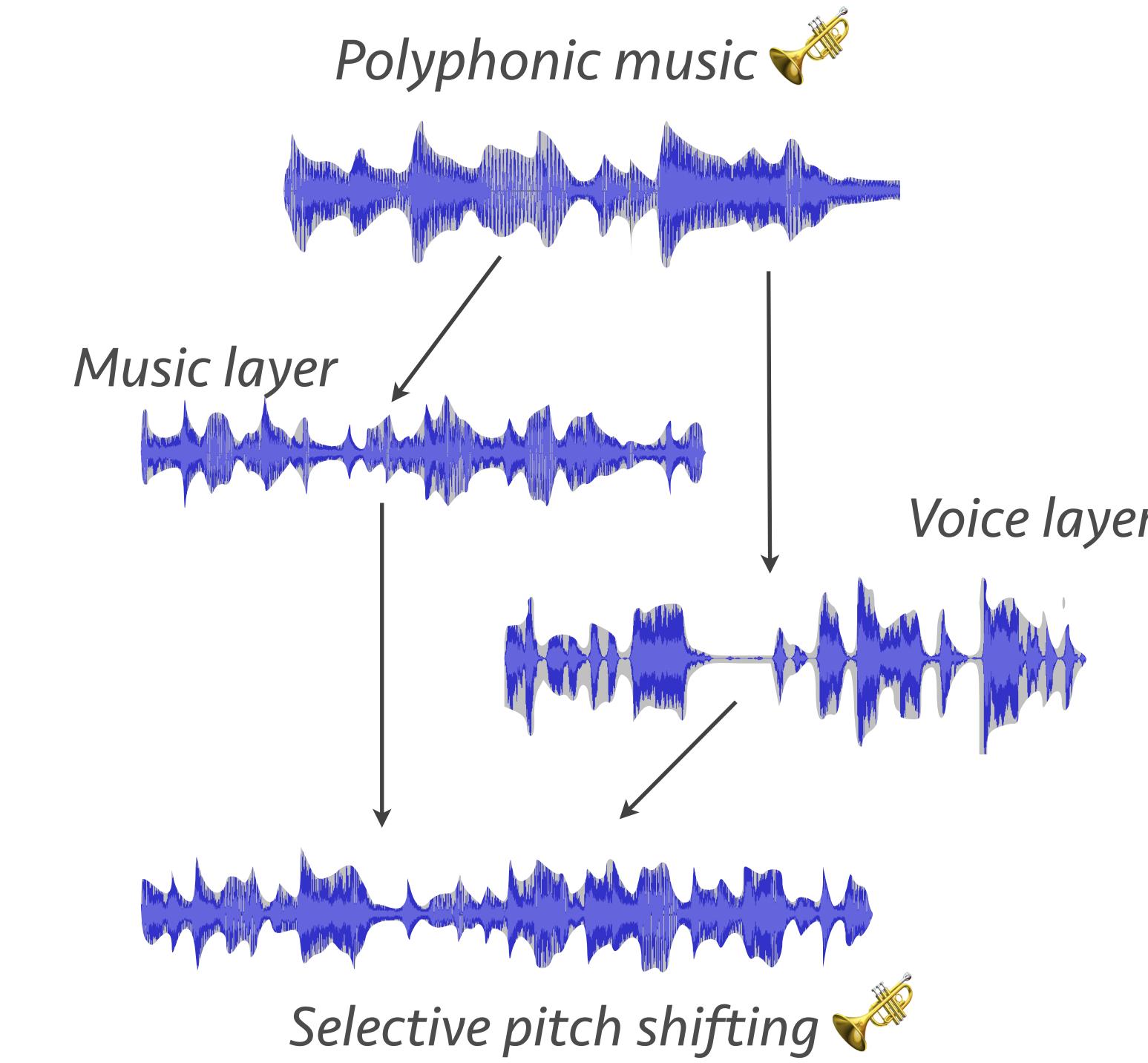
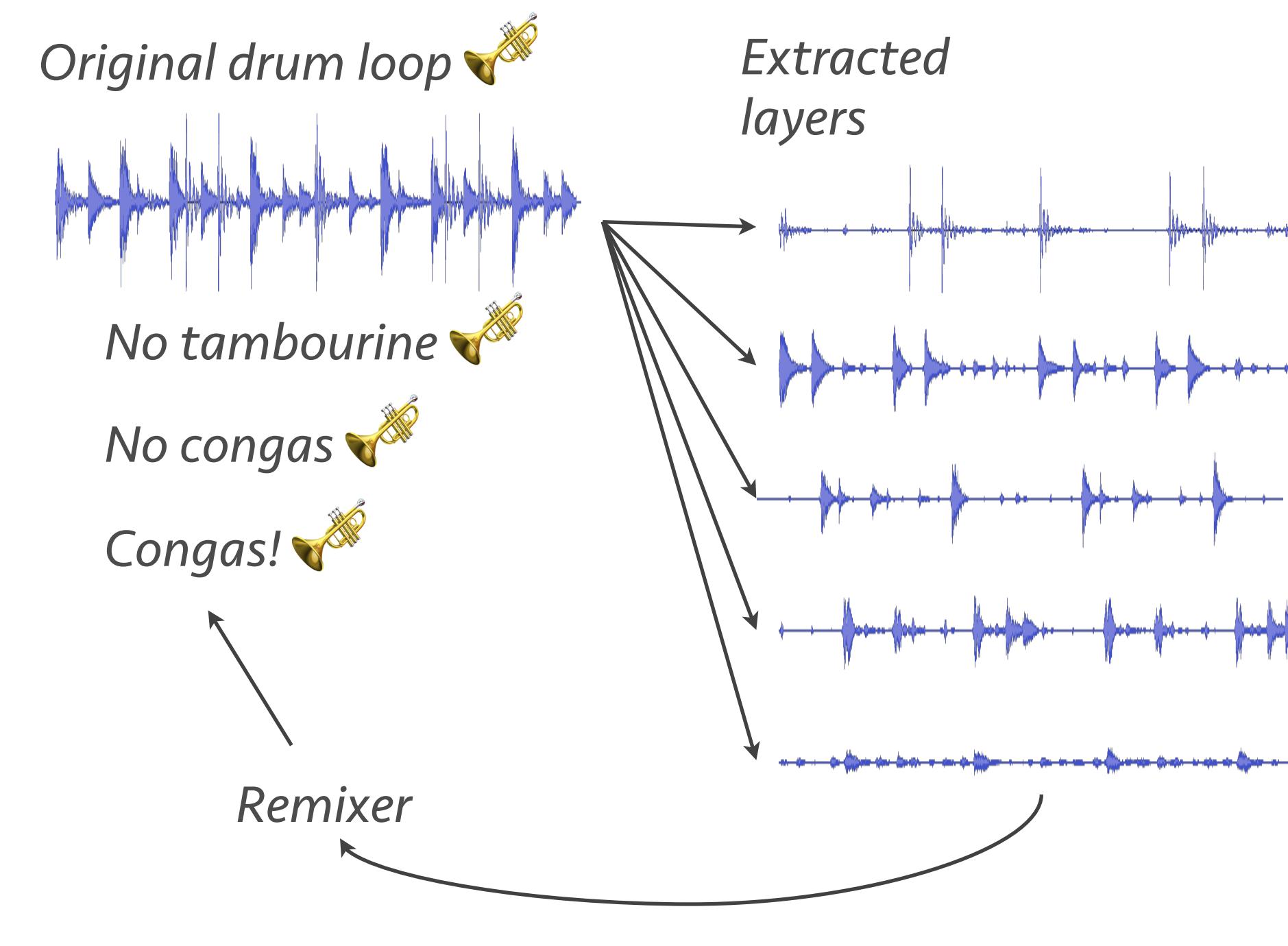


Video Content Analysis

- Detecting sounds in mixtures
 - We learn dictionaries offline and explain the movie soundtrack



Audio layer editing



Using priors

- We can bias the NMF models while learning
- In each iteration we can add a bias to the estimate of the two factors

$$\mathbf{W} = \mathbf{W} + \alpha \mathbf{B}_W$$

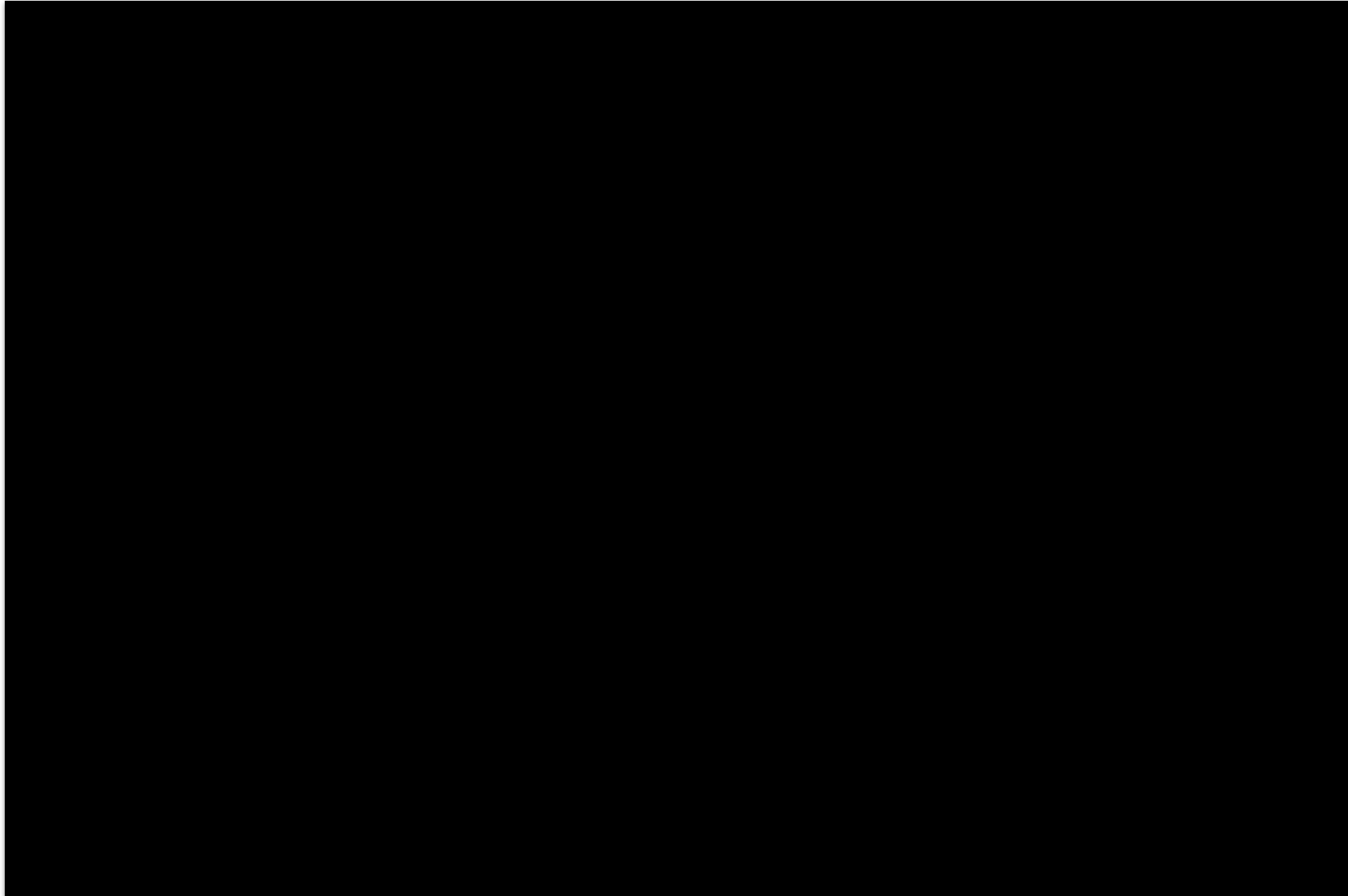
How we want W to be

$$\mathbf{H} = \mathbf{H} + \beta \mathbf{B}_H$$

How we want H to be

- Forces them to assume a specific form
- This allows using user guidance in learning

User-guided sound selection



Recap

- Under-constrained signal separation
 - The DUET algorithm
- Single-channel separation
 - Spectral factorizations

Reading

- The DUET algorithm
 - https://www.researchgate.net/profile/Scott_Rickard/publication/3318963_Blind_Separation_of_Speech_Mixtures_via_Time-Frequency_Masking/links/02bfe51277499a70da000000.pdf
- Spectral factorizations for separation
 - <http://paris.cs.illinois.edu/pubs/smaragdis-FA2005.pdf>
 - <http://paris.cs.illinois.edu/pubs/smaragdis-ica07.pdf>
 - <http://www.merl.com/reports/docs/TR2004-104.pdf>