From Structured Dropout to Proximal Methods

Guiding the Learning Process in Meaningful Ways

Sebastian Ewert

Machine Listening Lab
Centre for Digital Music
Queen Mary University of London

September 2017

FAST

Queen Mary University of London

EPSRC
Engineering and Physical Sciences Research Council

Sebastian Ewert

Queen Mary
University of London

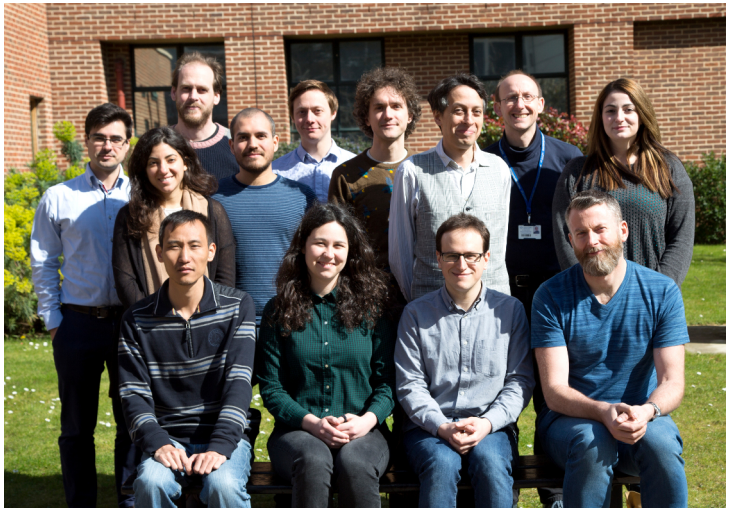
Background: Computer Science and Mathematics

2012: PhD
(Bonn + Max-Planck Informatics)

2015: Lecturer (Assistant Professor) in Signal Processing

2016: Machine Listening Lab

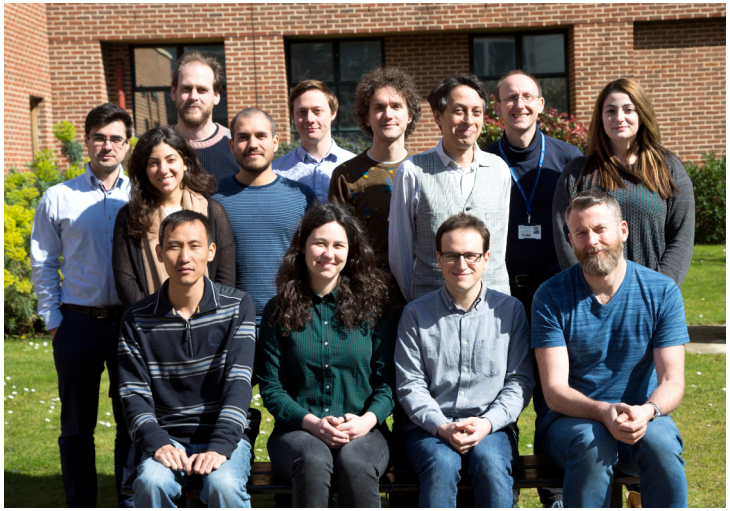
2017: Spotify (Research)



Sebastian Ewert

Queen Mary
University of London

PhD Students



- **Daniel Stoller:** Machine Listening with Limited Annotations (Lyrics Alignment/Transcription/Separation)
- **Delia Fano-Yela:** Proximity-Based Source Sep. (Restoration of Live Rec.)
- **Siying Wang:** Sequence Alignment and Informed Dictionary Learning (Music Tuition)
- **Tim Kirby:** Physical Modelling / Finite Elements Methods (Physical Parameter Estimation)

Research

Machine Listening

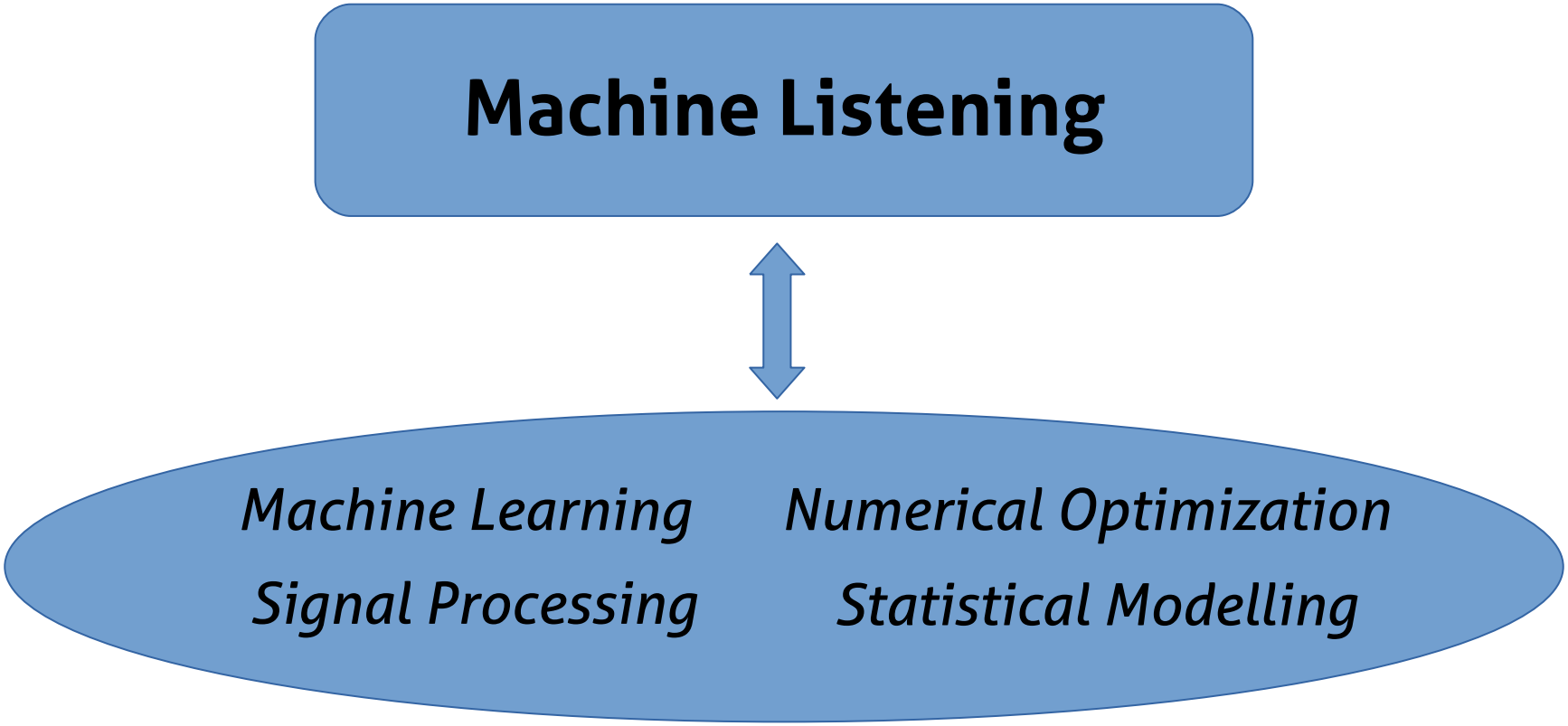
Research

Machine Listening

"How can we teach a computer

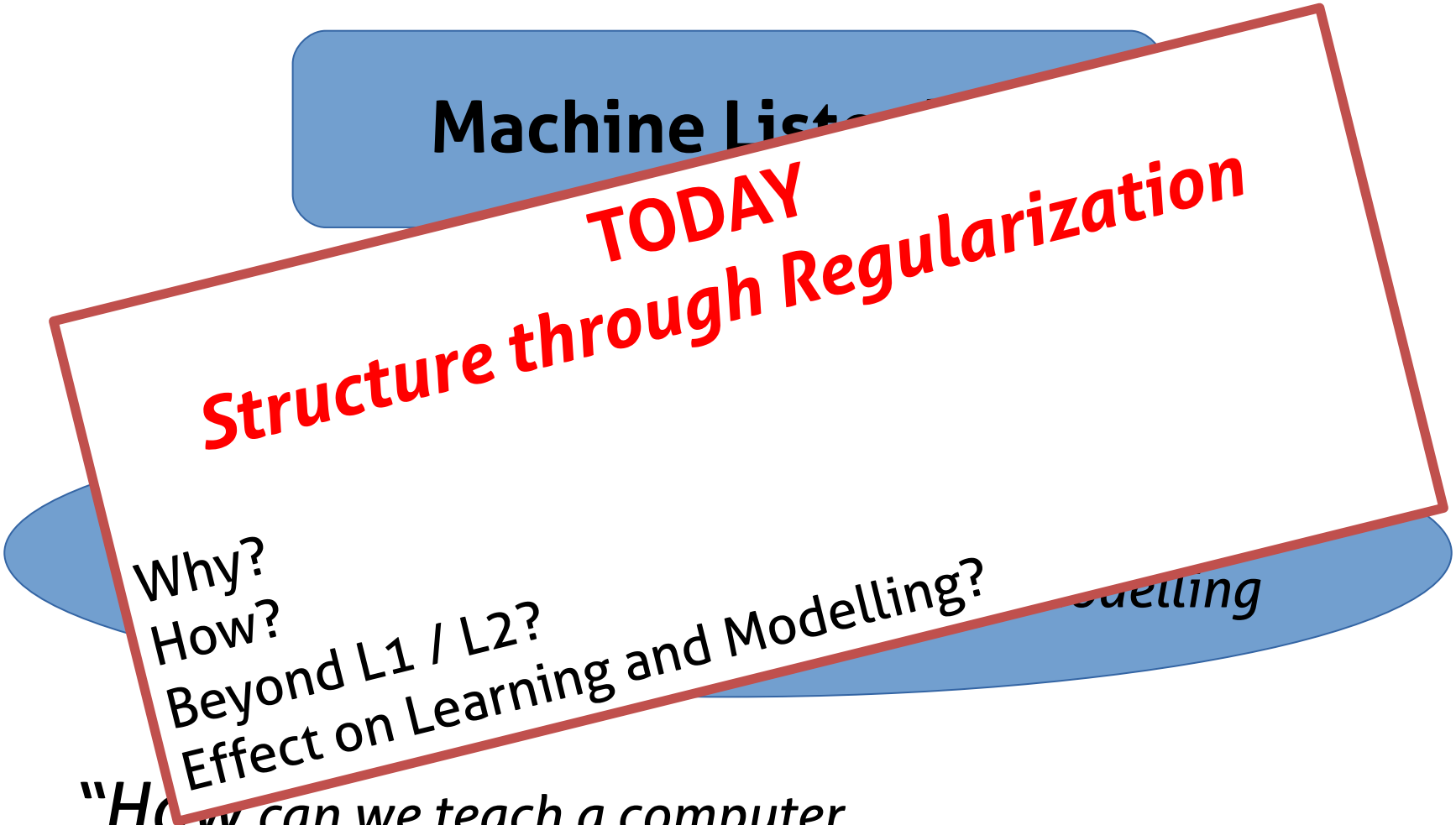
to understand sound the way we do?"

Research



"*How can we teach a computer*
to understand sound the way we do? "

Research



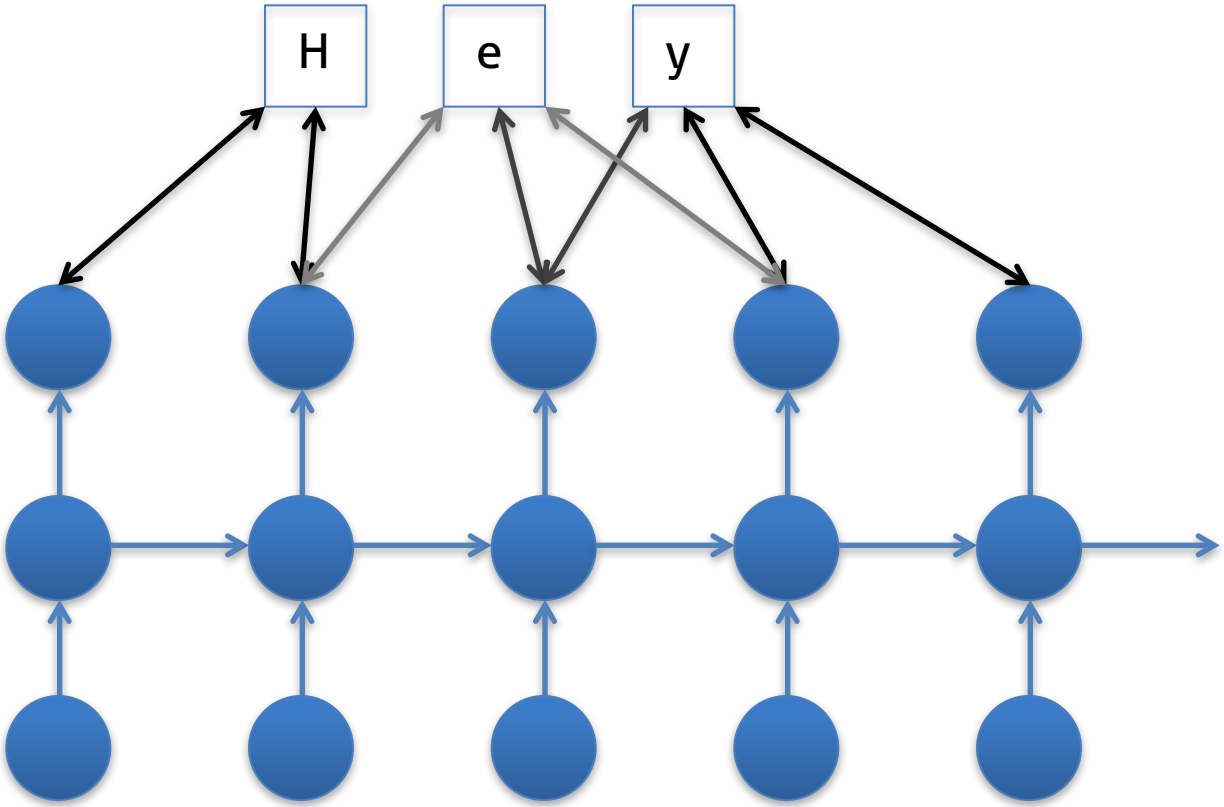
"*How can we teach a computer*

to understand sound the way we do? "

Neural Networks

Project 1: CTC Extensions

Connectionist Temporal Classification (Graves et al [16])

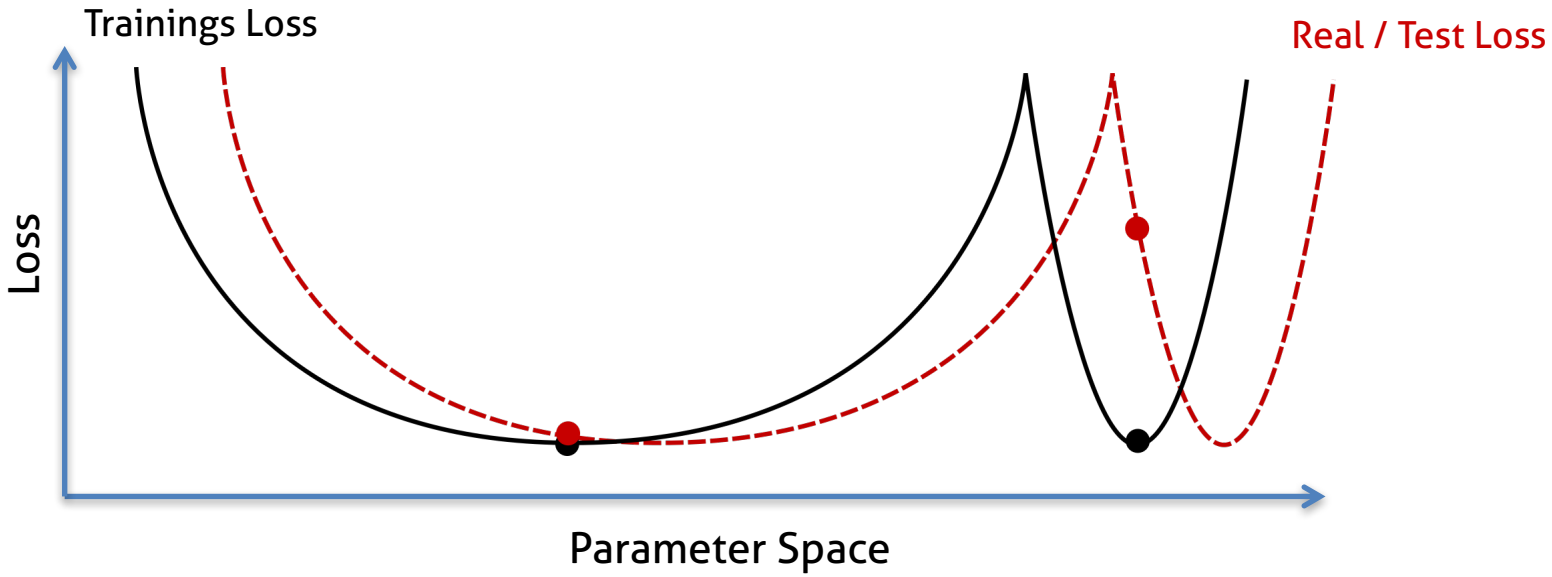


Posterior of specific HMM used as Loss Function for RNN

$\Rightarrow$ Bakis-type HMM!

Project 2: Proximal Methods for NNs

Nocedal et al [14] (relating to Goodfellow et al [15]):
"*Sharpness* of local minima in deep nets correlates negatively with *generalizability*"

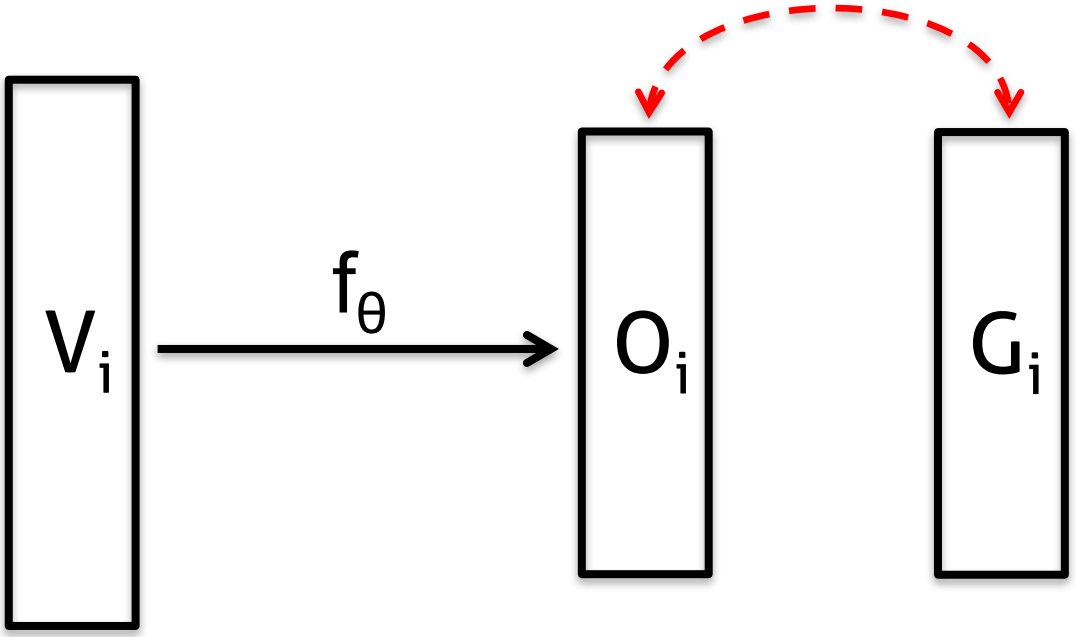


Project 3: Structure in Generative Models

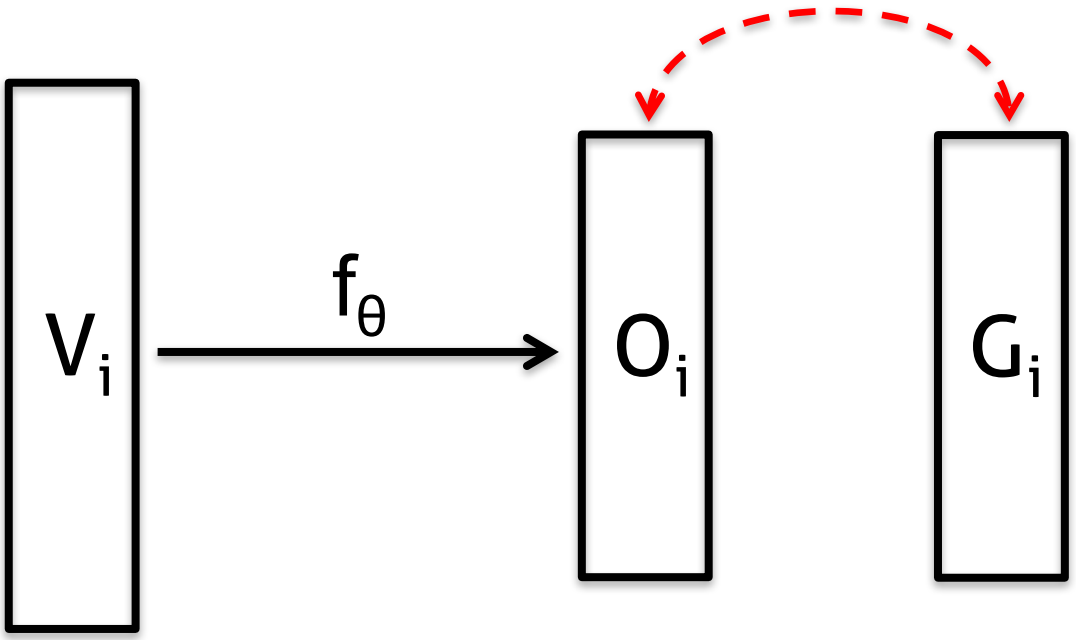
Project 3: Structure in Generative Models

What does *Structure in Generative Models* mean?

Supervised Learning

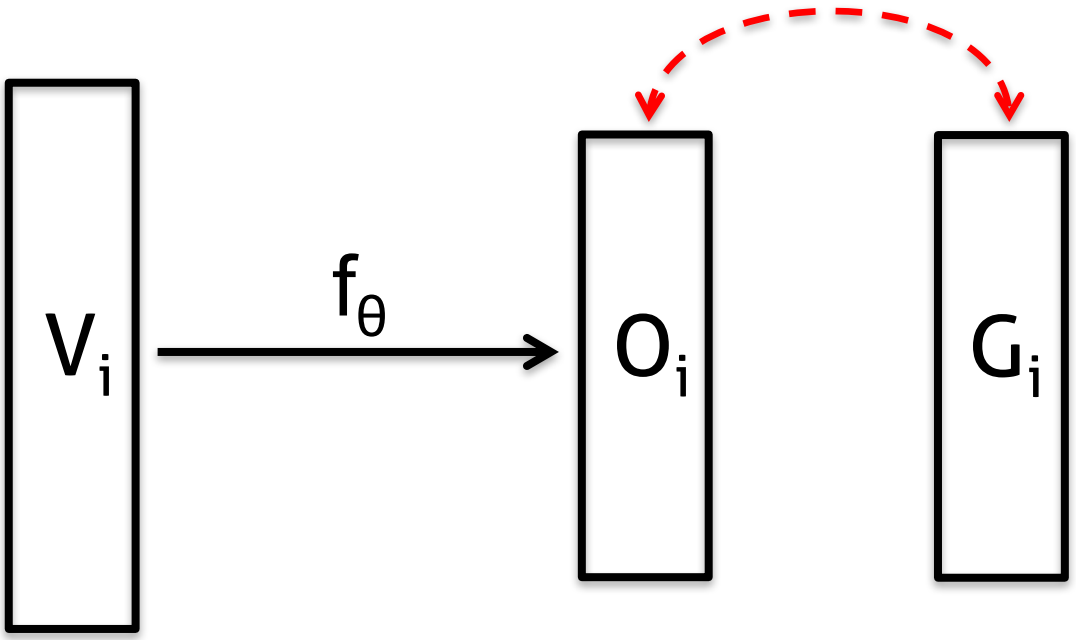


Supervised Learning



Works Well

Supervised Learning



Works Well

Problem: Groundtruth Creation Expensive

Weak Labels

"*Car*"

Example: One label for entire clip

Weak Labels

"*Car*"

Example: One label for entire clip

Weak Labels

"*Car*"

Example: One label for entire clip

Weak Labels

"*Car*"

Example: One label for entire clip

Weak Labels

"*Car*"

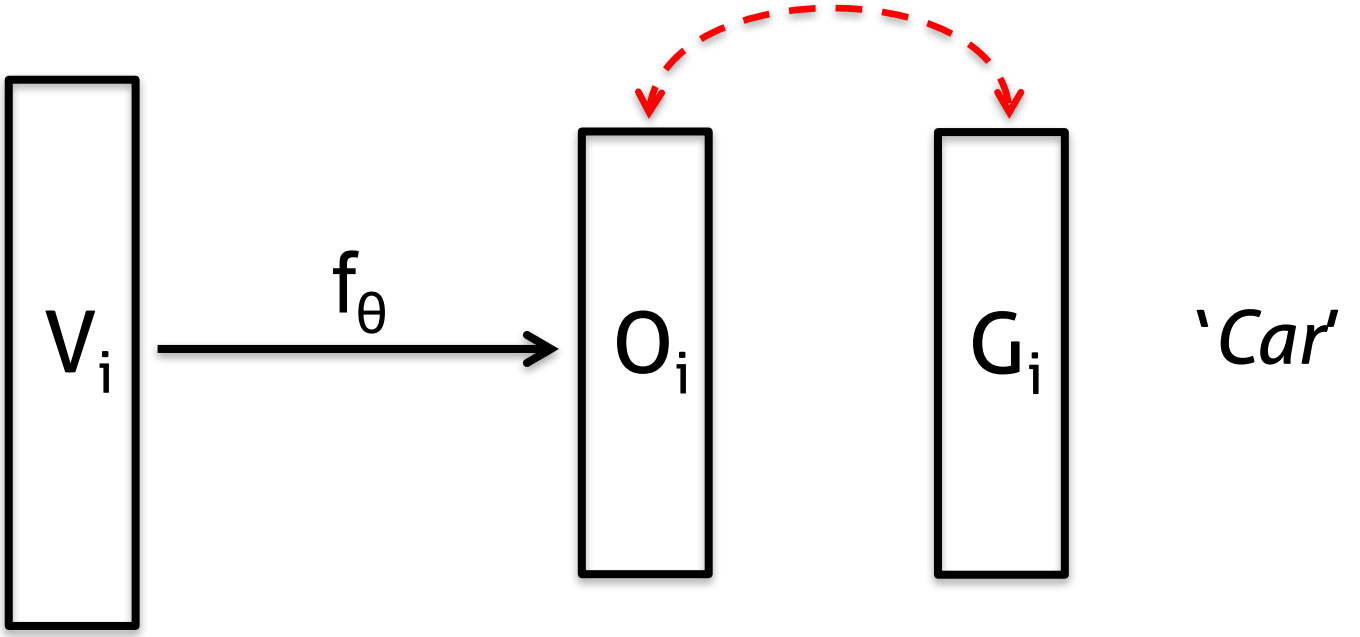
Example: One label for entire clip

Weak Labels

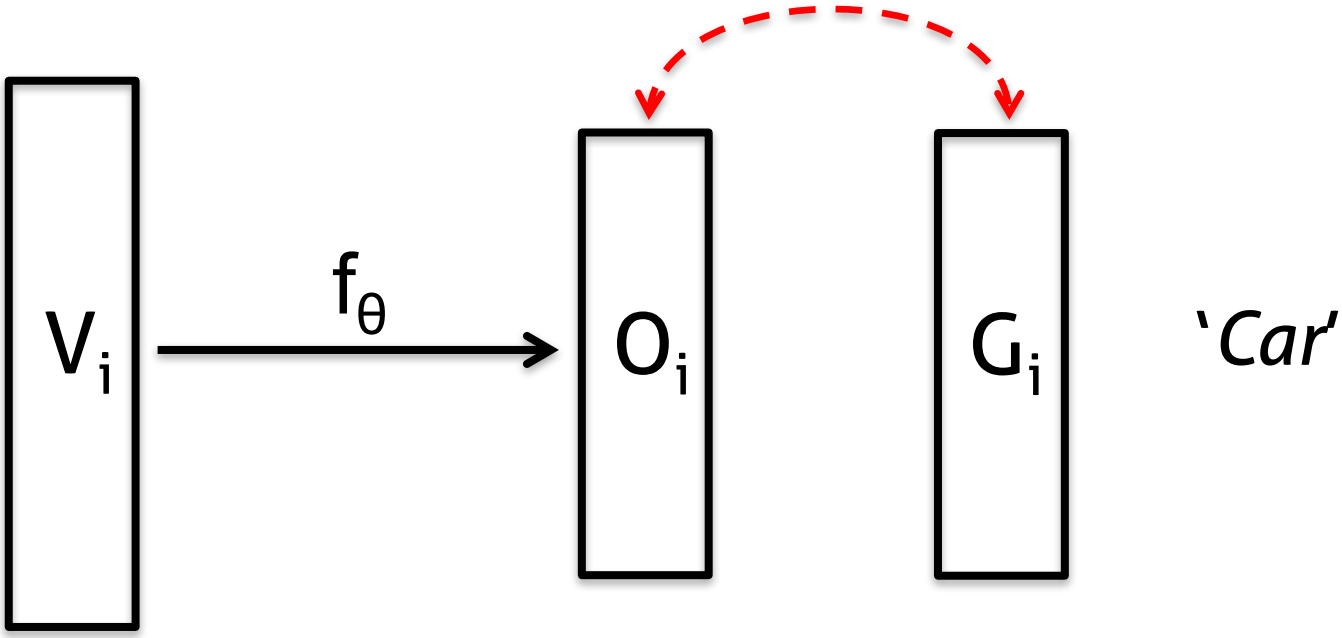
"*Car*"

Example: One label for entire clip

Supervised Learning



Supervised Learning

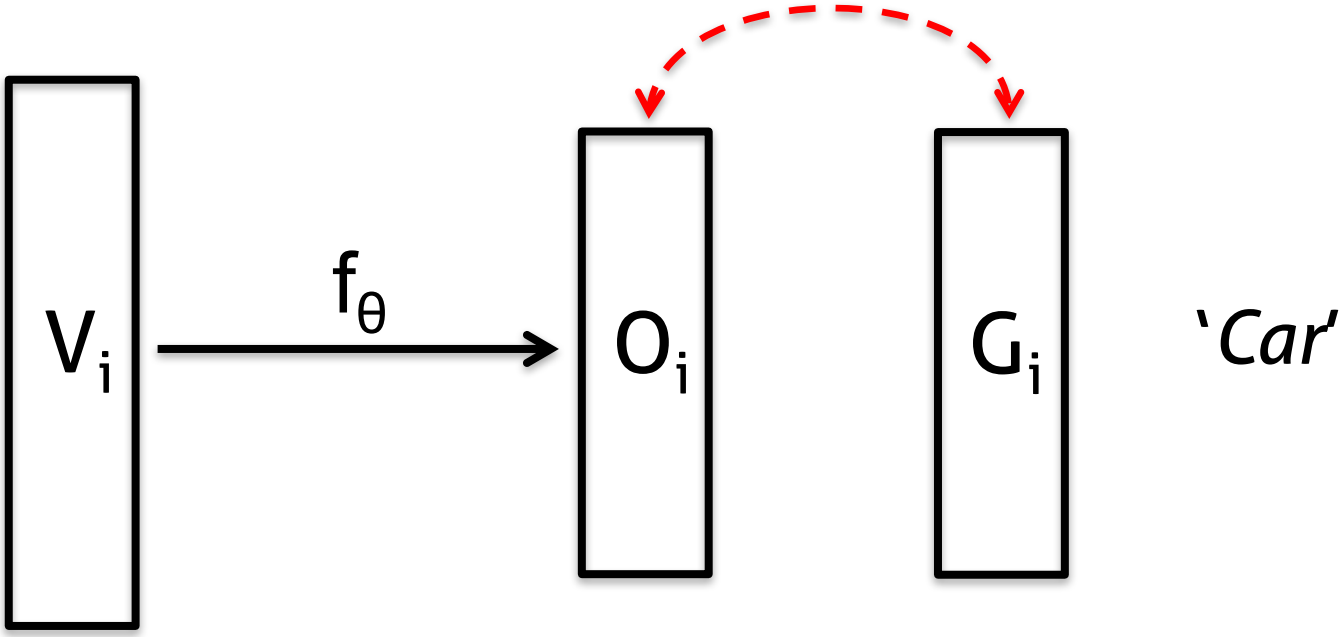


Idea

1.) Train naively with all frames in instance set to 'car'

2.) Re-label all frames clearly classified as 'no-car' and iterate

Supervised Learning



Idea

1.) Train naively with all frames in instance set to 'car'

2.) Re-label all frames clearly classified as 'no-car' and iterate

Problem: No real gain in information

Idea

"How can we take the uncertainty in the labels into account?"

Idea

"How can we take the uncertainty in the labels into account?"

Difficult with supervised learning (needs clear target)

Idea

"How can we take the uncertainty in the labels into account?"

Difficult with supervised learning (needs clear target)

Idea

1. Treat weak label training fundamentally as *unsupervised learning*

Idea

"How can we take the uncertainty in the labels into account?"

Difficult with supervised learning (needs clear target)

Idea

1. Treat weak label training fundamentally as *unsupervised learning*
2. Use *weak labels as guidance* to encourage structure in learned representations

Idea

"How can we take the uncertainty in the labels into account?"

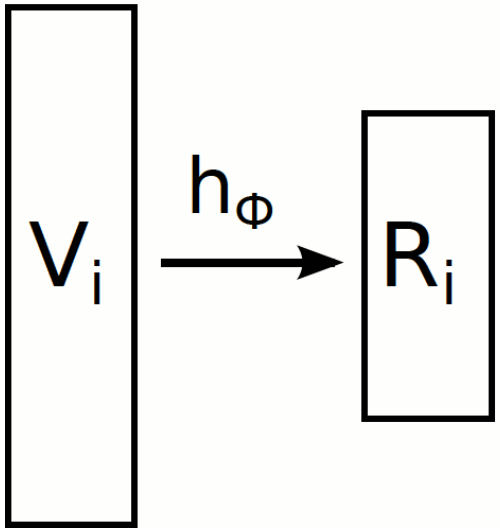
Difficult with supervised learning (needs clear target)

Idea

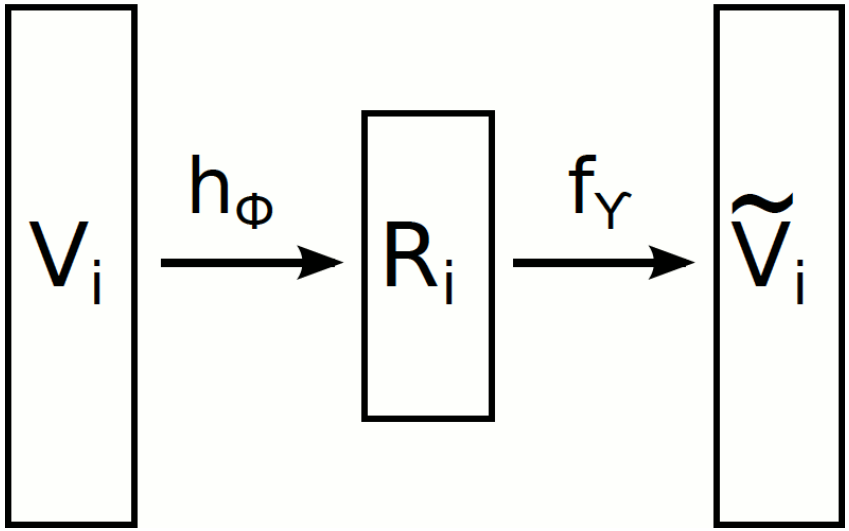
1. Treat weak label training fundamentally as *unsupervised learning*
2. Use *weak labels as guidance* to encourage structure in learned representations

... but how?

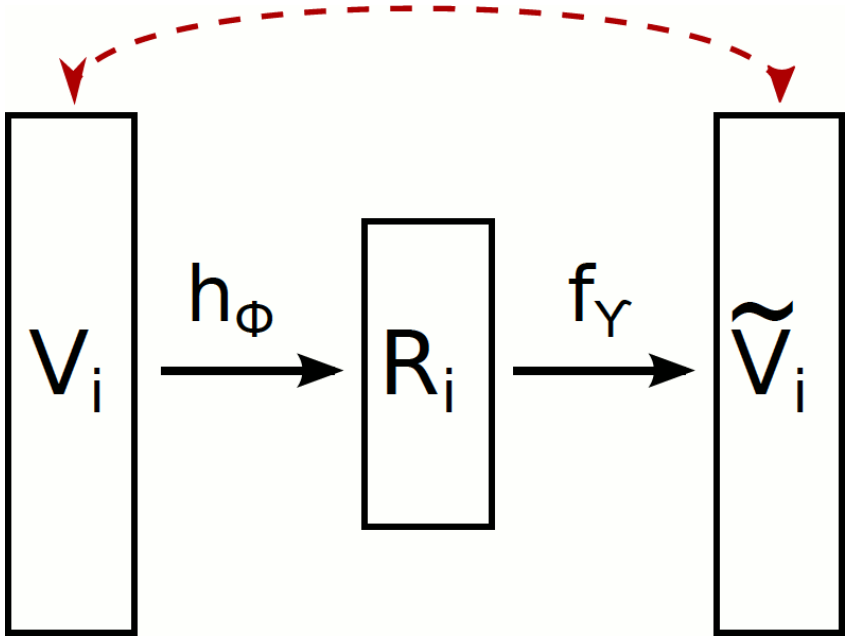
Autoencoder



Autoencoder

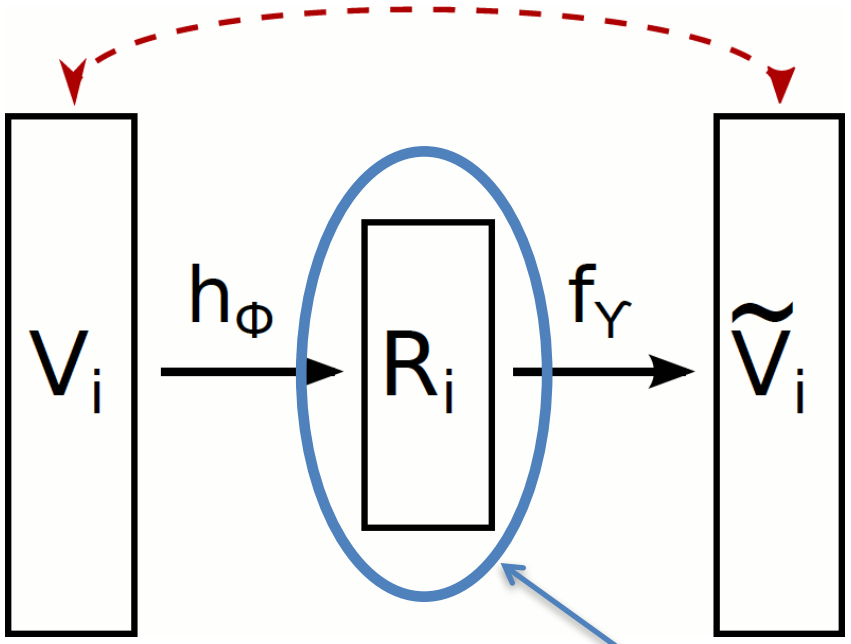


Autoencoder



Autoencoder

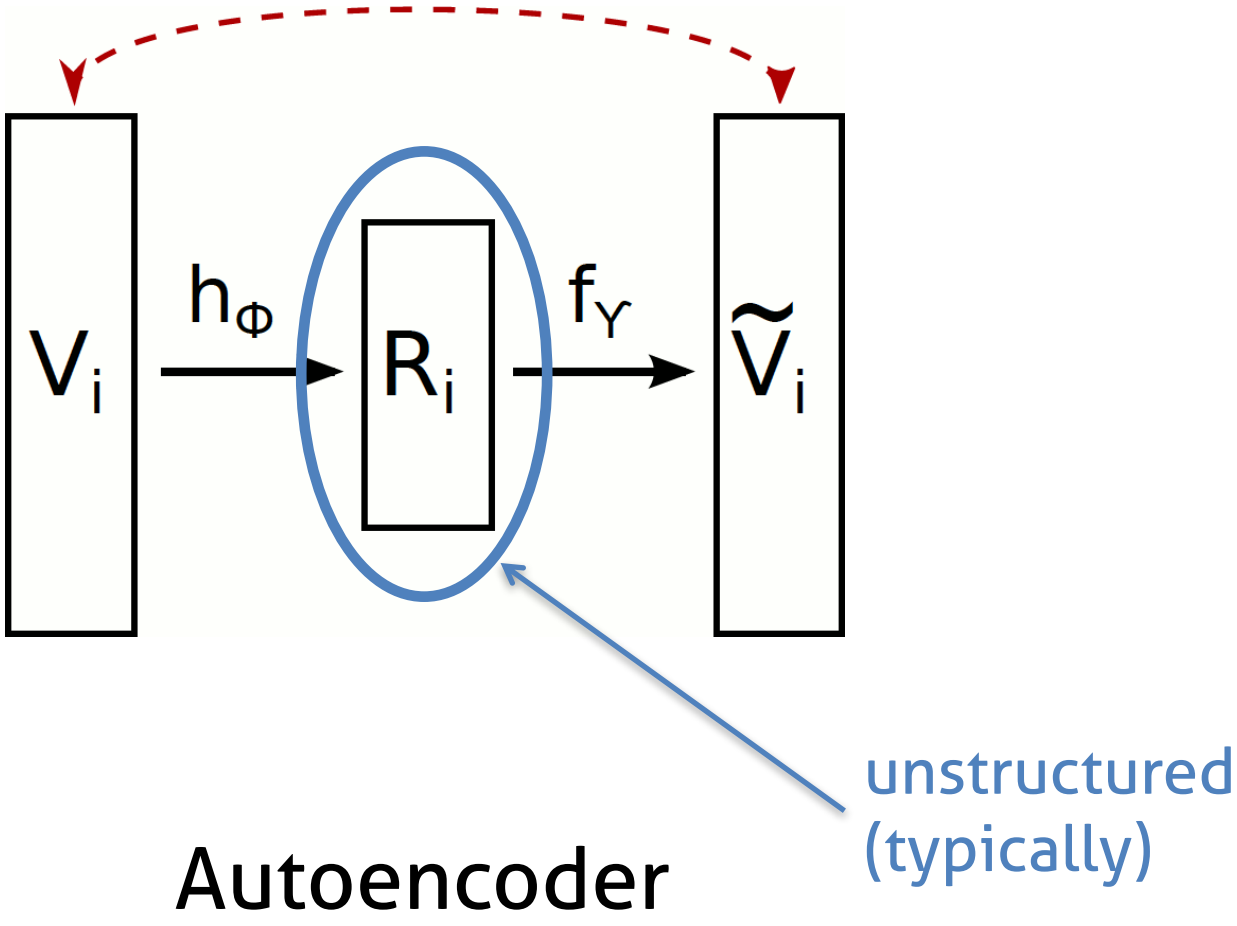
Autoencoder



unstructured
(typically)

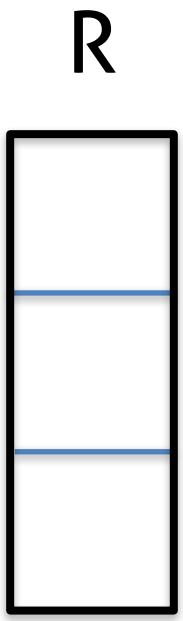
Autoencoder

Autoencoder



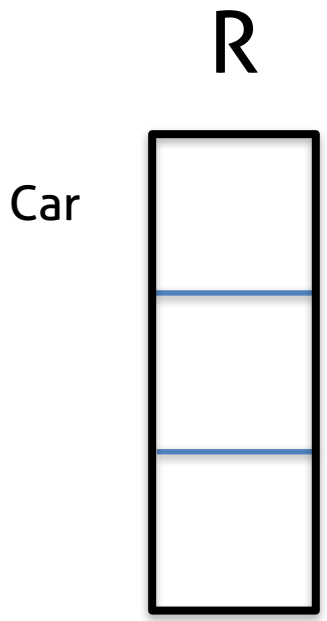
Idea: Use weak labels to encourage structure

Association of Units



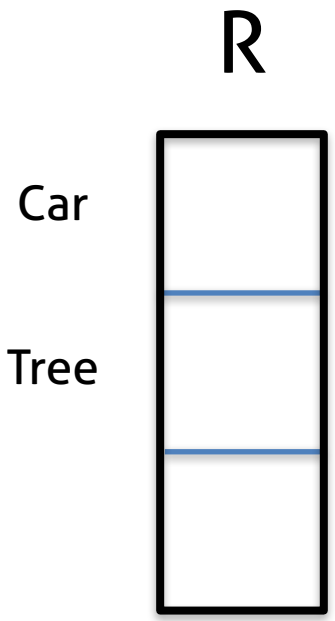
We associate each unit in R
with a certain class

Association of Units



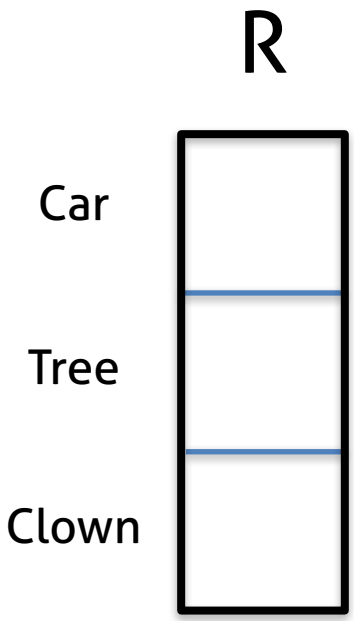
We associate each unit in R
with a certain class

Association of Units



We associate each unit in R with a certain class

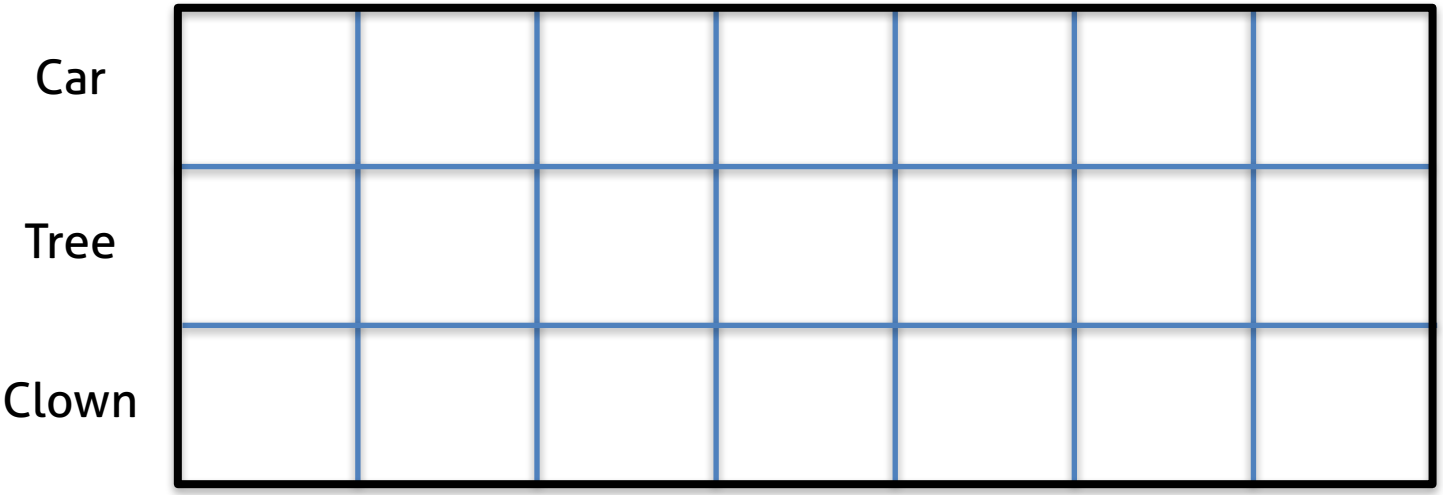
Association of Units



We associate each unit in R with a certain class

Label Encoding

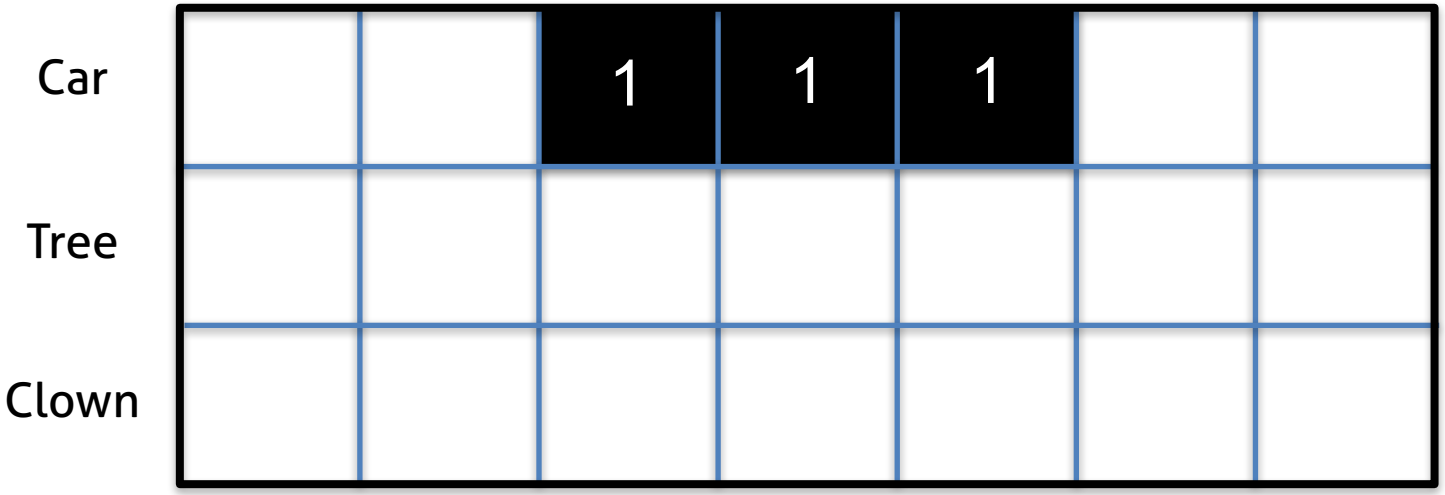
Label Matrix **S**



Weak Labels

Label Encoding

Label Matrix **S**

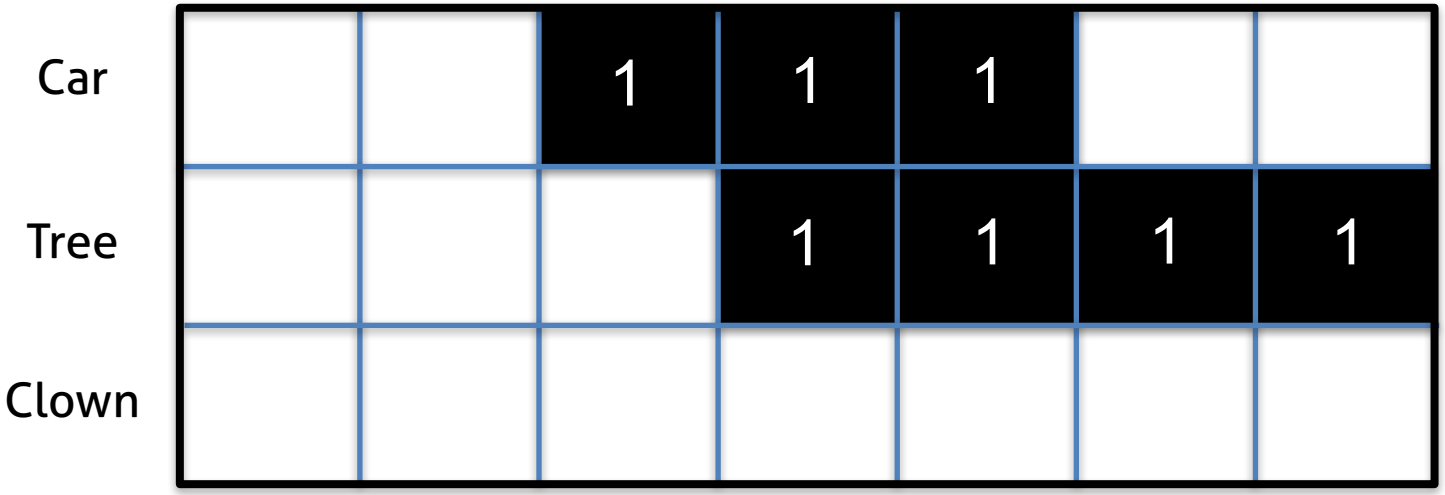


Weak Labels

1. Car is maybe active in frames 3-5

Label Encoding

Label Matrix **S**

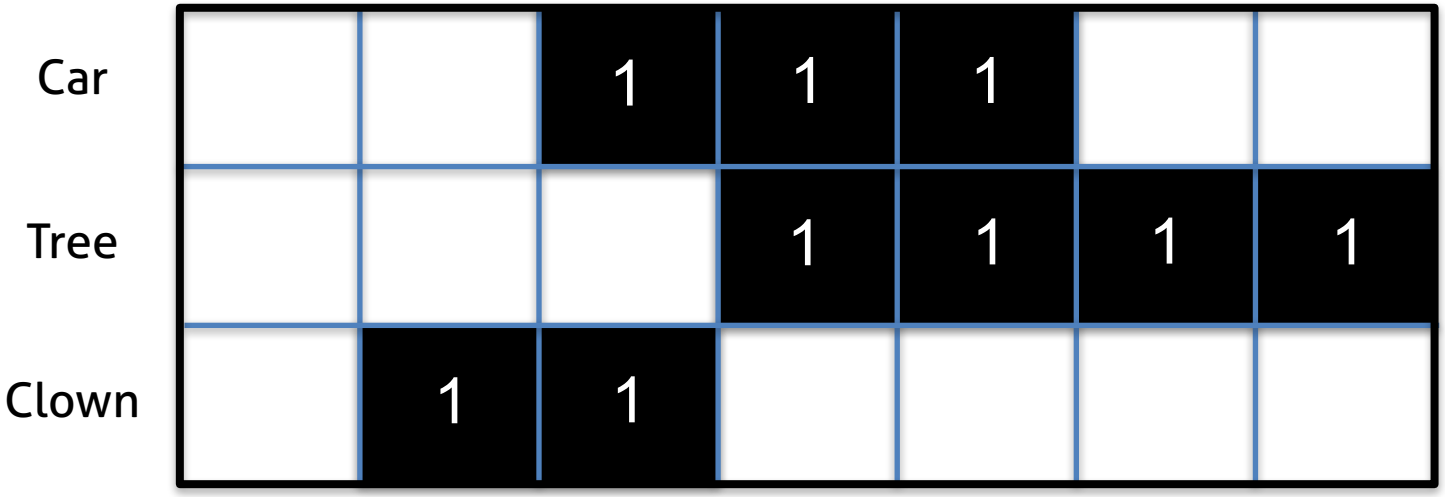


Weak Labels

1. Car is maybe active in frames 3-5
2. Tree is maybe active in frame 4-7

Label Encoding

Label Matrix **S**



Weak Labels

1. Car is maybe active in frames 3-5
2. Tree is maybe active in frame 4-7
3. Clown is active from 2-3

Label Encoding

Label Matrix **S**

Car	0	0	1	1	1	0	0
Tree	0	0	0	1	1	1	1
Clown	0	1	1	0	0	0	0

Weak Labels

1. Car is maybe active in frames 3-5
2. Tree is maybe active in frame 4-7
3. Clown is active from 2-3

Label Encoding

Label Matrix **S**

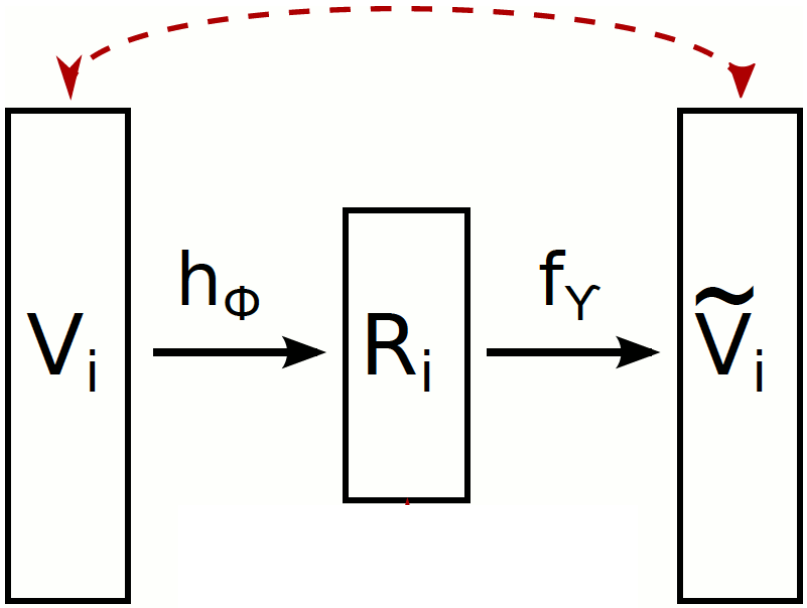
Car	0	0	1	1	1	0	0
Tree	0	0	0	1	1	1	1
Clown	0	1	1	0	0	0	0

Weak Labels

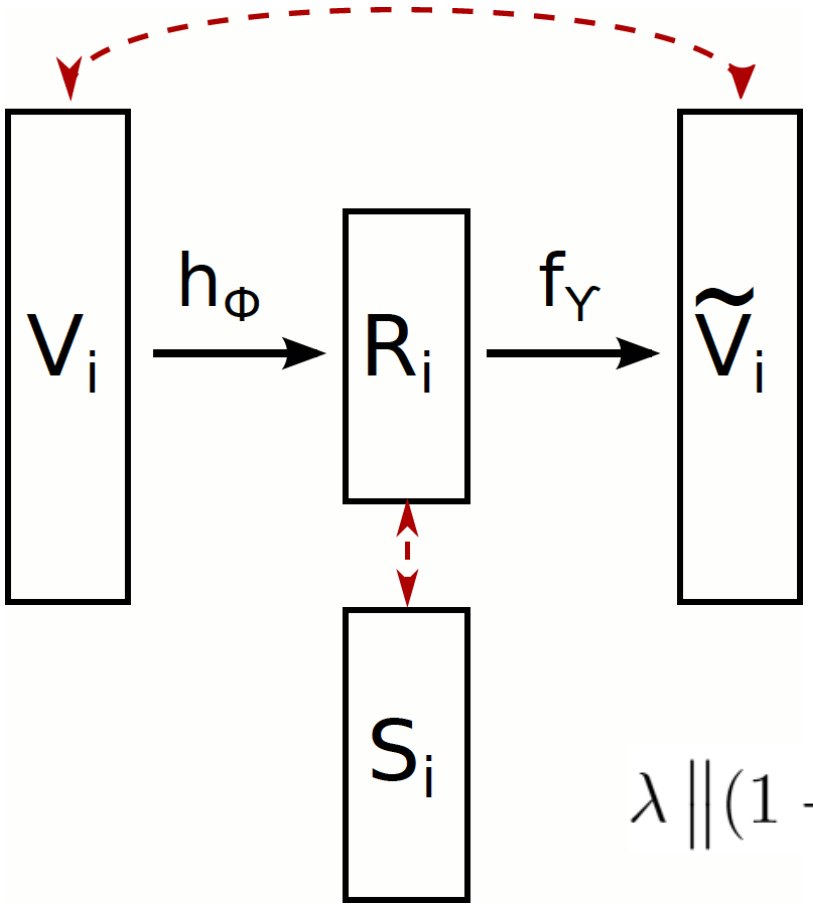
1. Car is maybe active in frames 3-5
2. Tree is maybe active in frame 4-7
3. Clown is active from 2-3

S encodes *Label Information* in binary form

Guided Unsupervised Learning



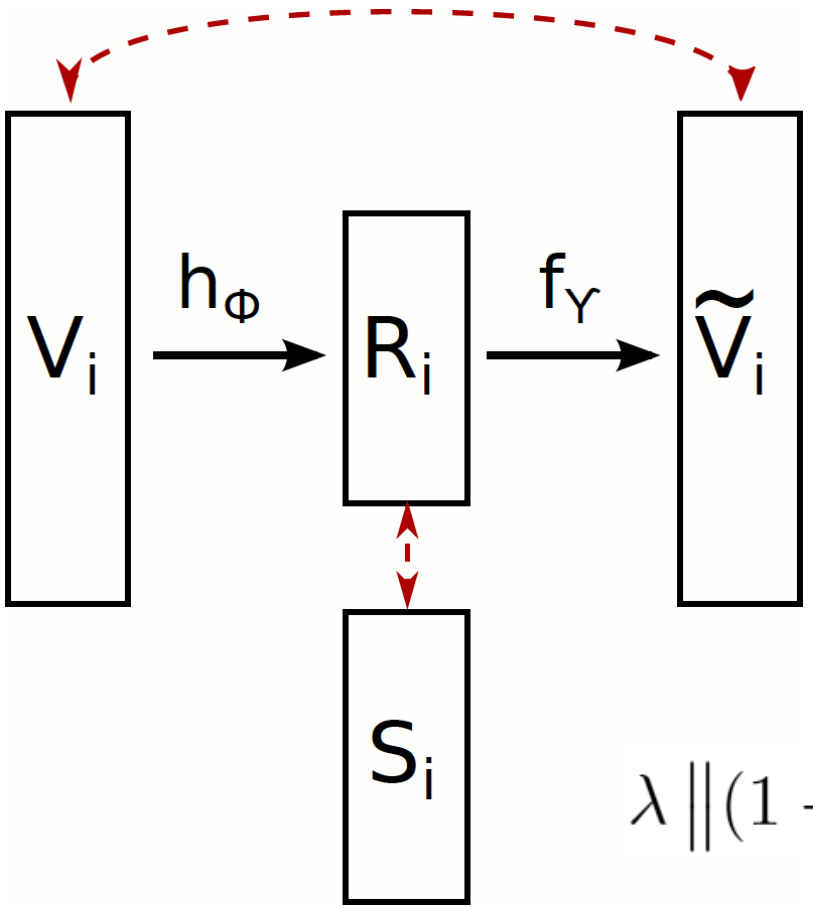
Guided Unsupervised Learning



Activity Penalties

$$\lambda \left\| (1 - S_i) \odot h_\Phi(V_i) \right\|_2^2$$

Guided Unsupervised Learning

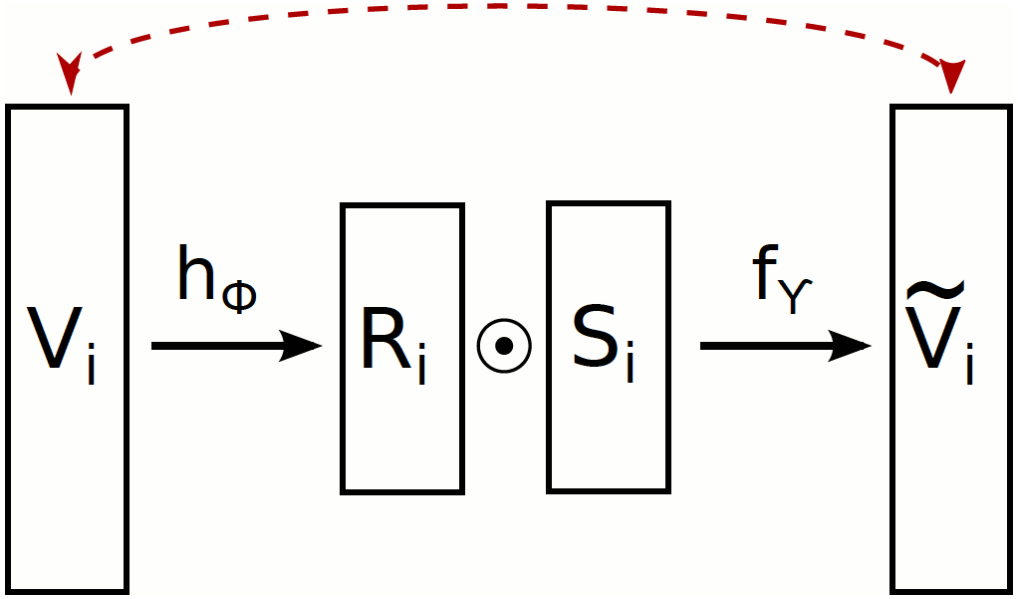


Activity Penalties

$$\lambda \left\| (1 - S_i) \odot h_\Phi(V_i) \right\|_2^2$$

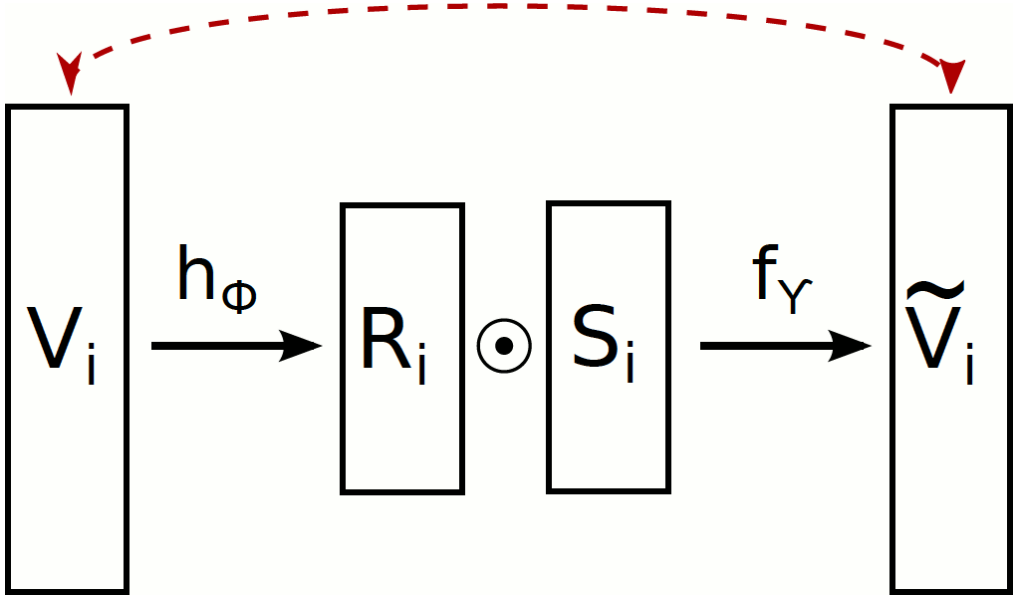
Problem: Difficult to train

Guided Unsupervised Learning



Structured Dropout

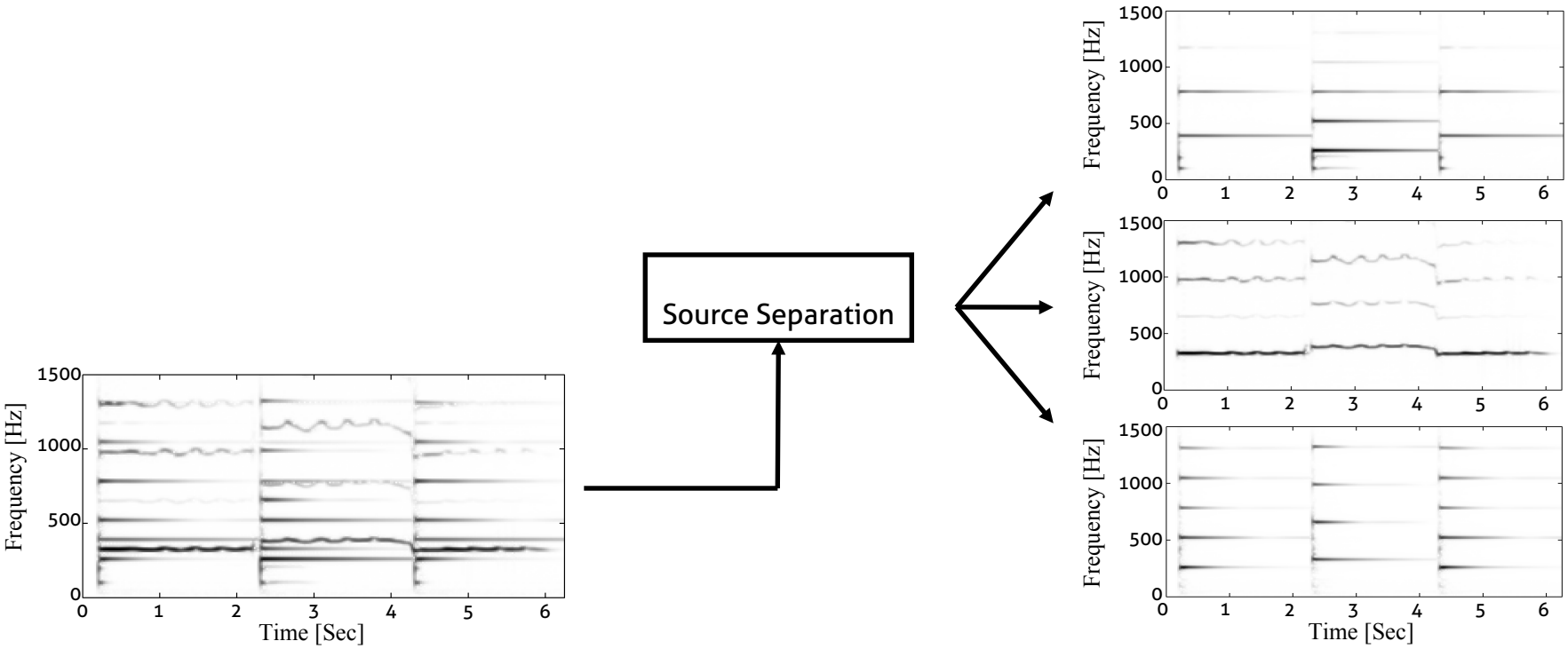
Guided Unsupervised Learning



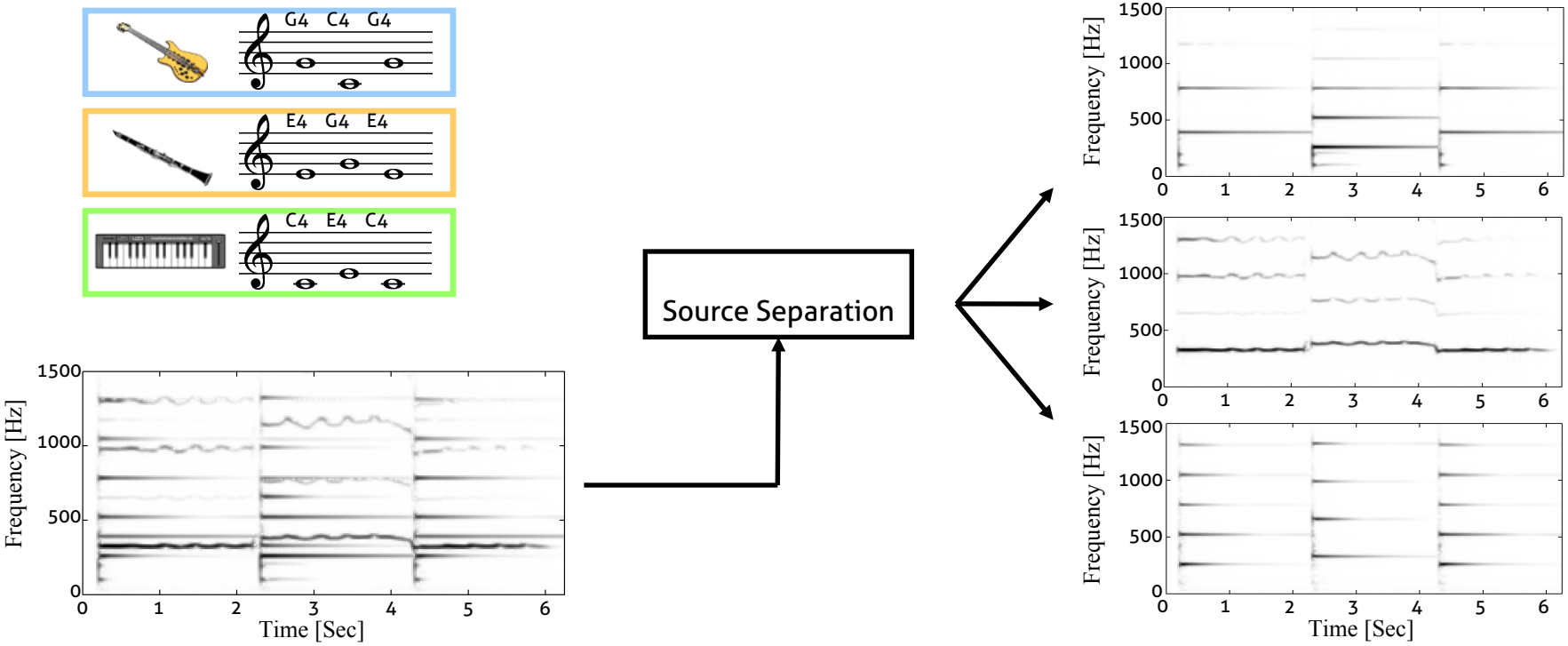
Structured Dropout

Faster convergence

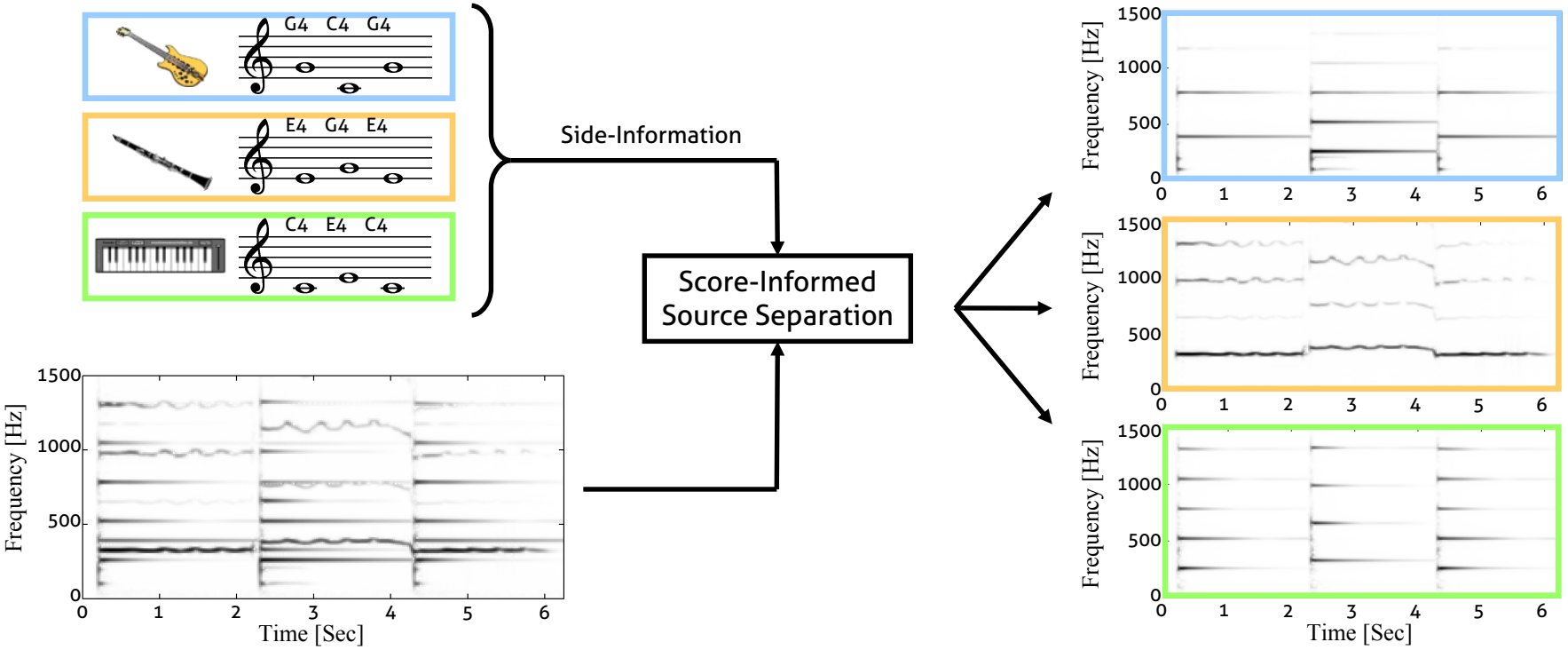
Score-Informed Source Separation



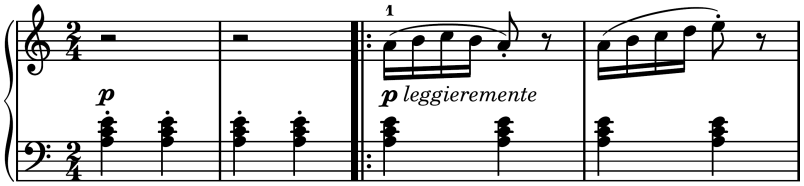
Score-Informed Source Separation

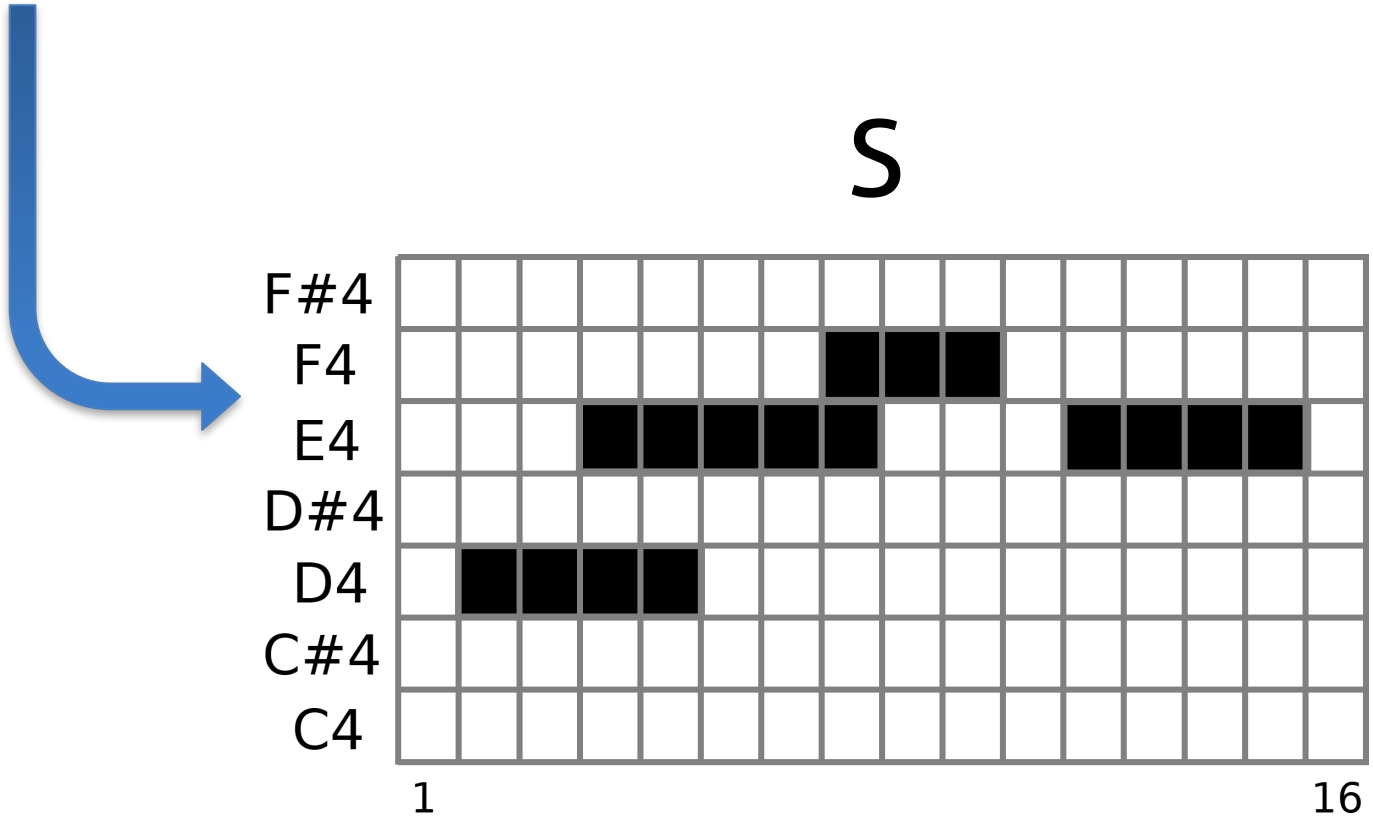


Score-Informed Source Separation



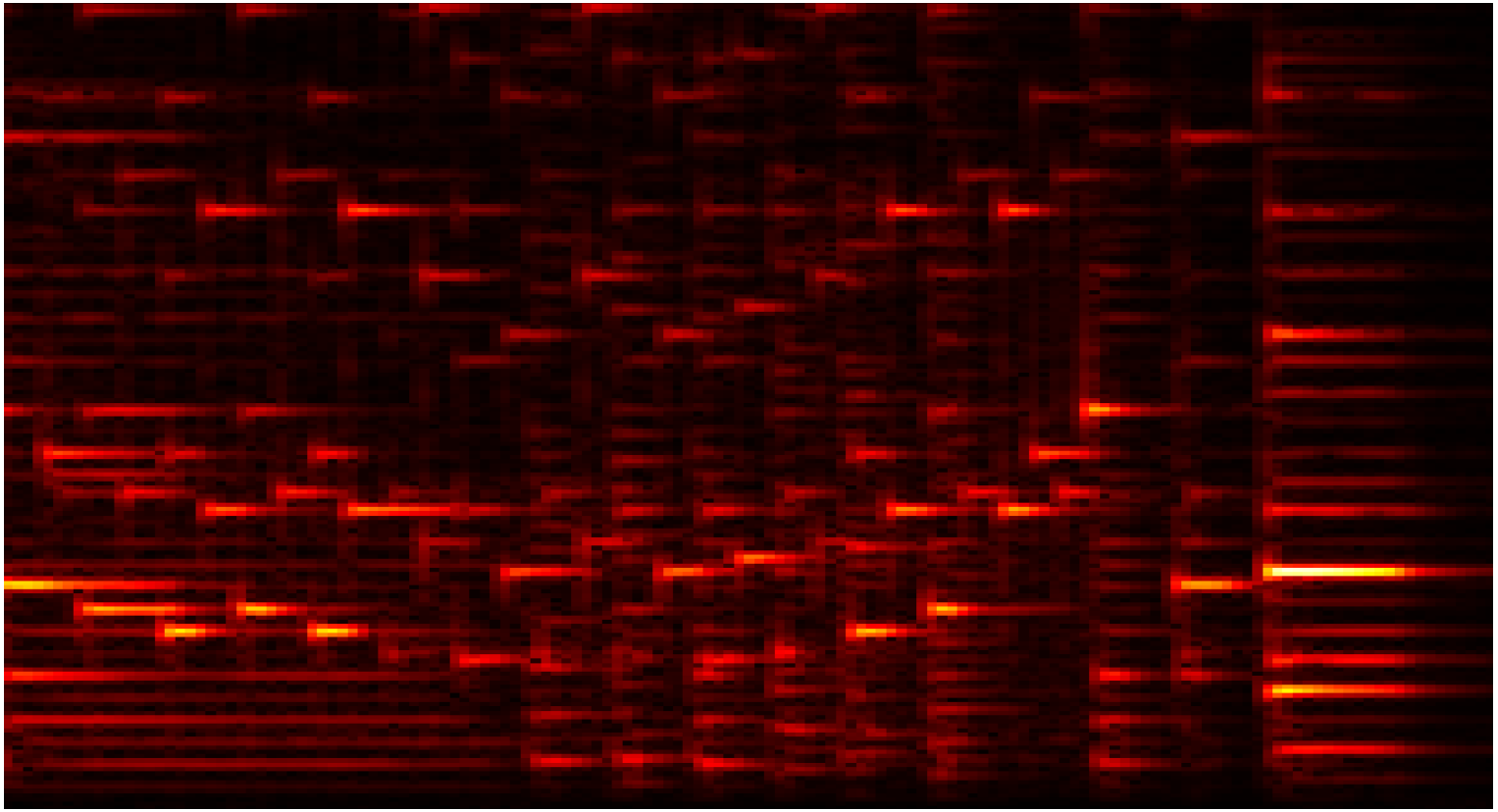
Informed Source Separation with Deep Nets





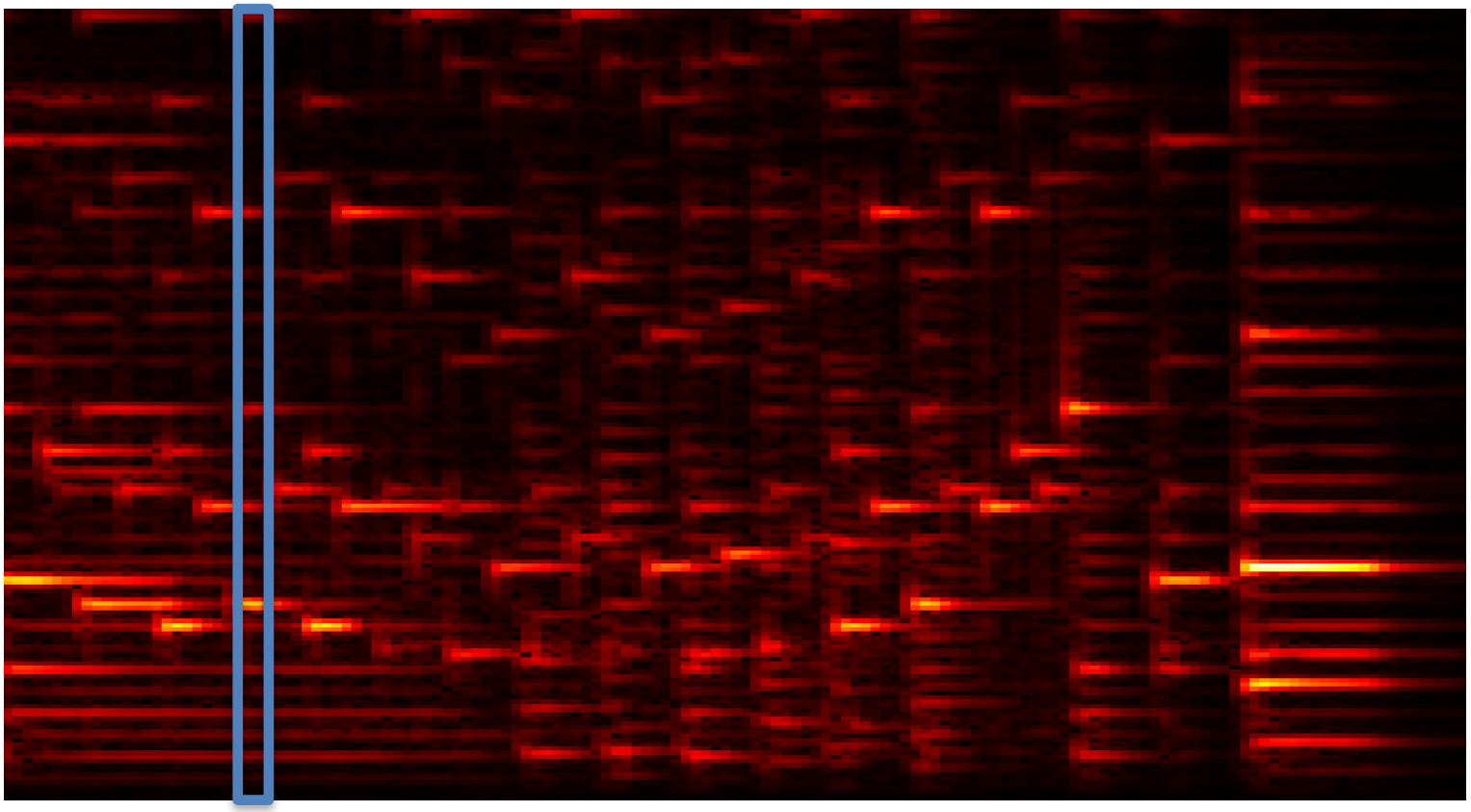
Informed Source Separation with Deep Nets

V



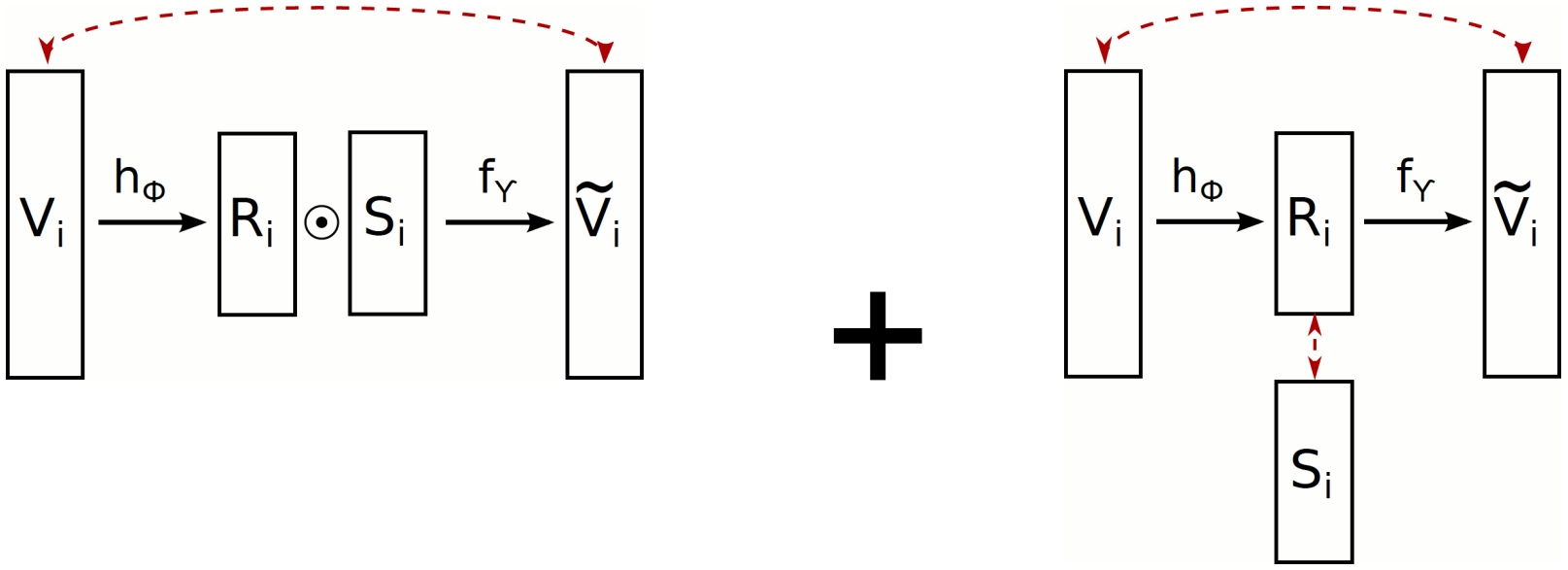
Informed Source Separation with Deep Nets

V



V_i

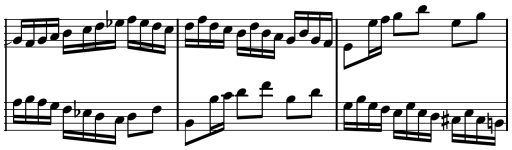
Training

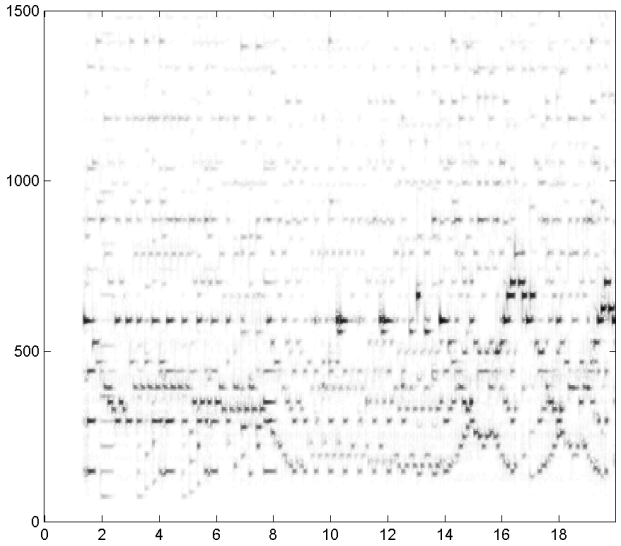


Structured Dropout

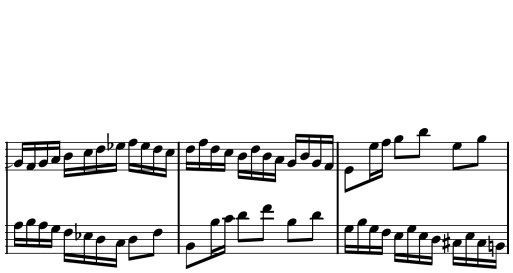
Activity Penalties

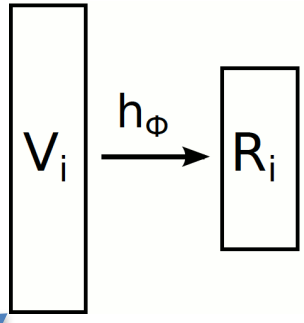
Informed Source Separation with Deep Nets

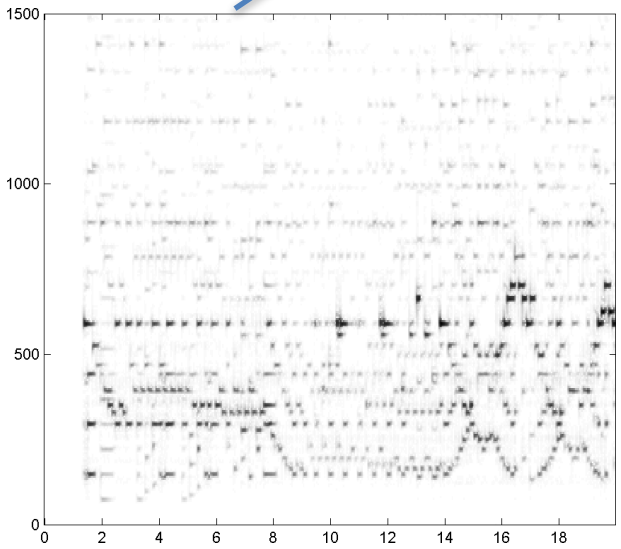




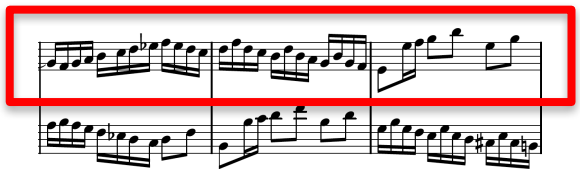
Informed Source Separation with Deep Nets

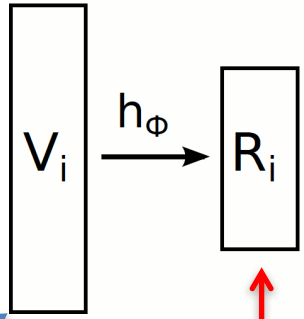




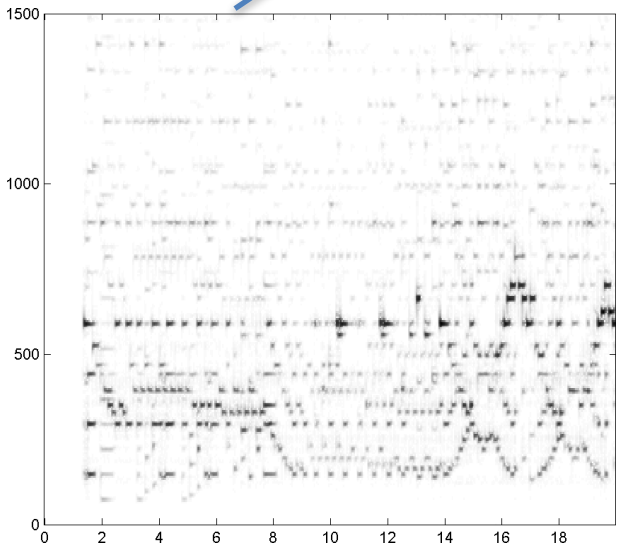


Informed Source Separation with Deep Nets

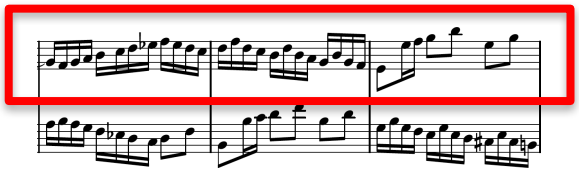


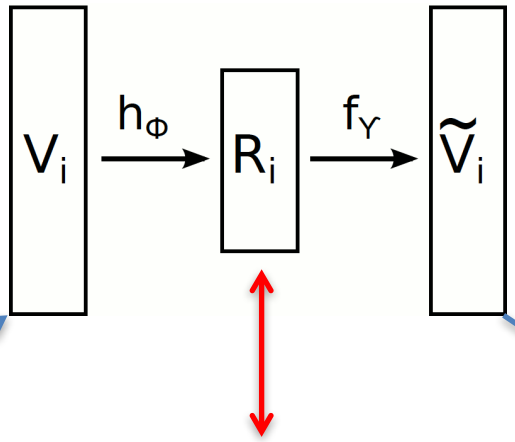


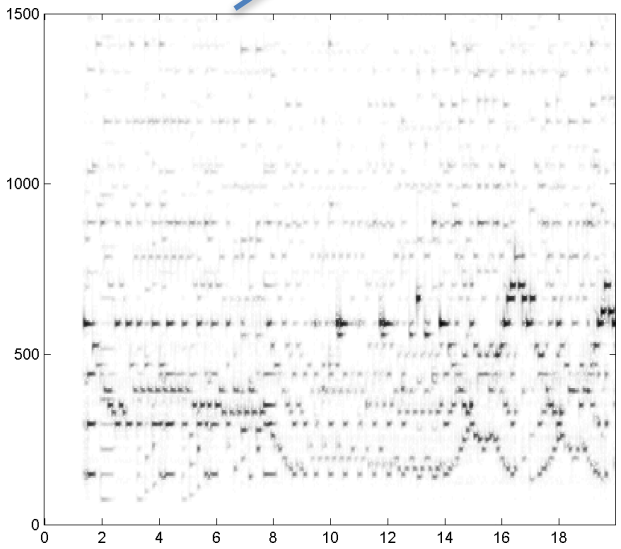
Modified Repr.

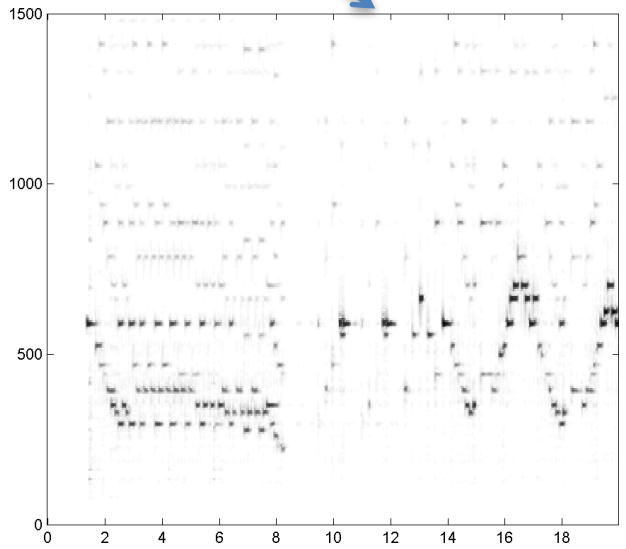


Informed Source Separation with Deep Nets

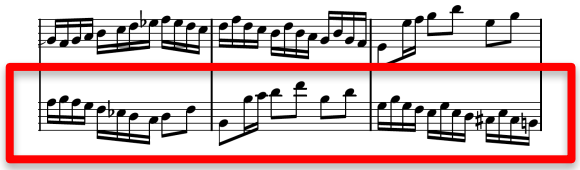


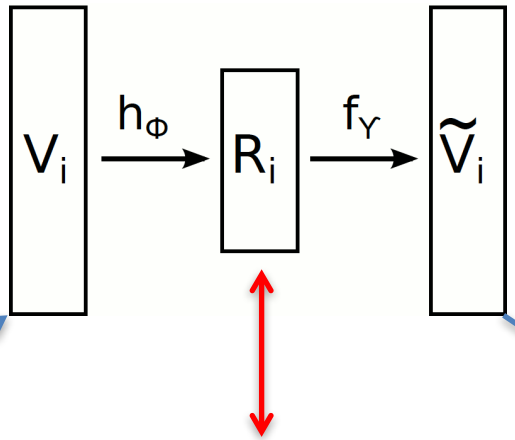


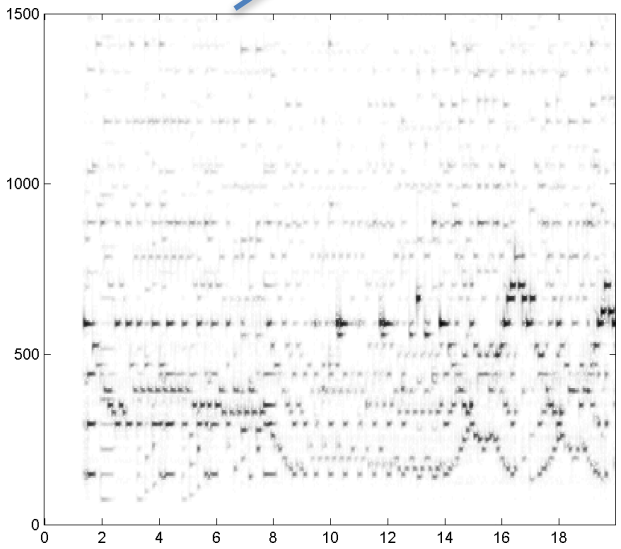


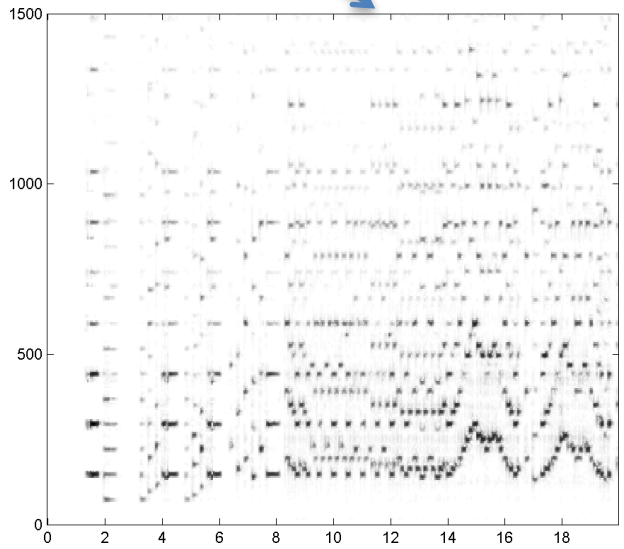


Informed Source Separation with Deep Nets









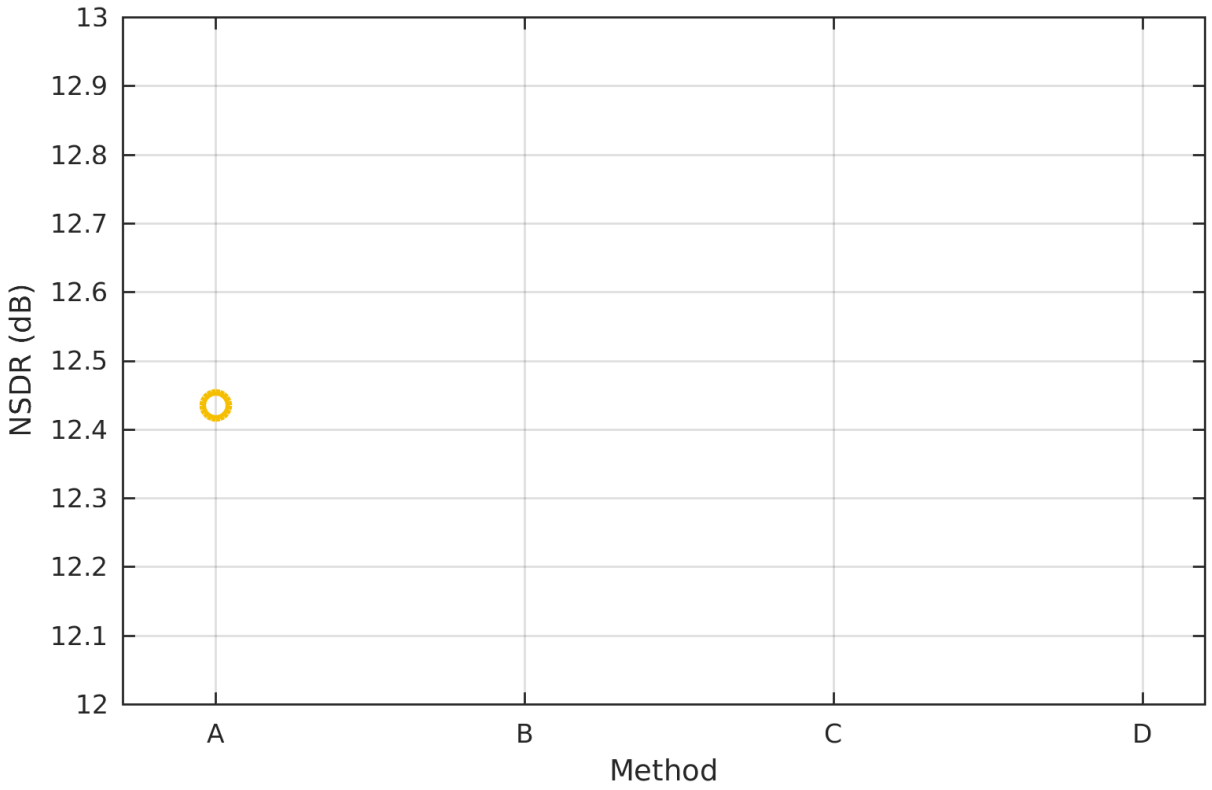
Experiments

- Dataset (taken from [1])
  - 10 MIDI files from Mutopia project
  - Synthesized using Native Instruments VST (multisample)
  - Each recording 30-300 seconds long
  - Notes for left and right synthesized separately
- Task: given mix (as magnitude spectrogram), separate left from right hand notes

Experiments

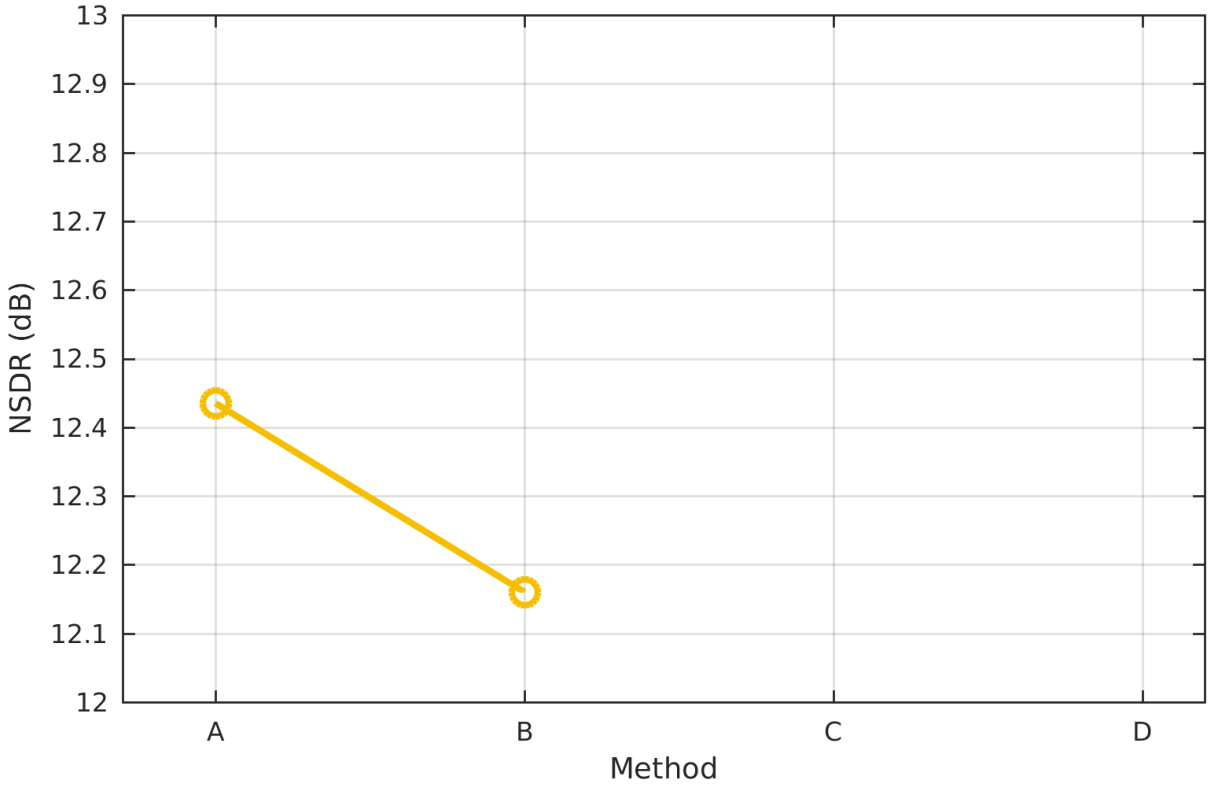
- Networks *h* and f standard feed-forwards DNNs
  - Three layers each
  - 1500 units on each layer
  - Sigmoids activations. ReLUs at representation and output layers
- Optimizer: Adam with decreased step-size (defaults otherwise)
- No Dropout or BatchNorm
- Training: Full-Batch on individual recordings
- Evaluation: BSS Eval Toolkit

Experiments



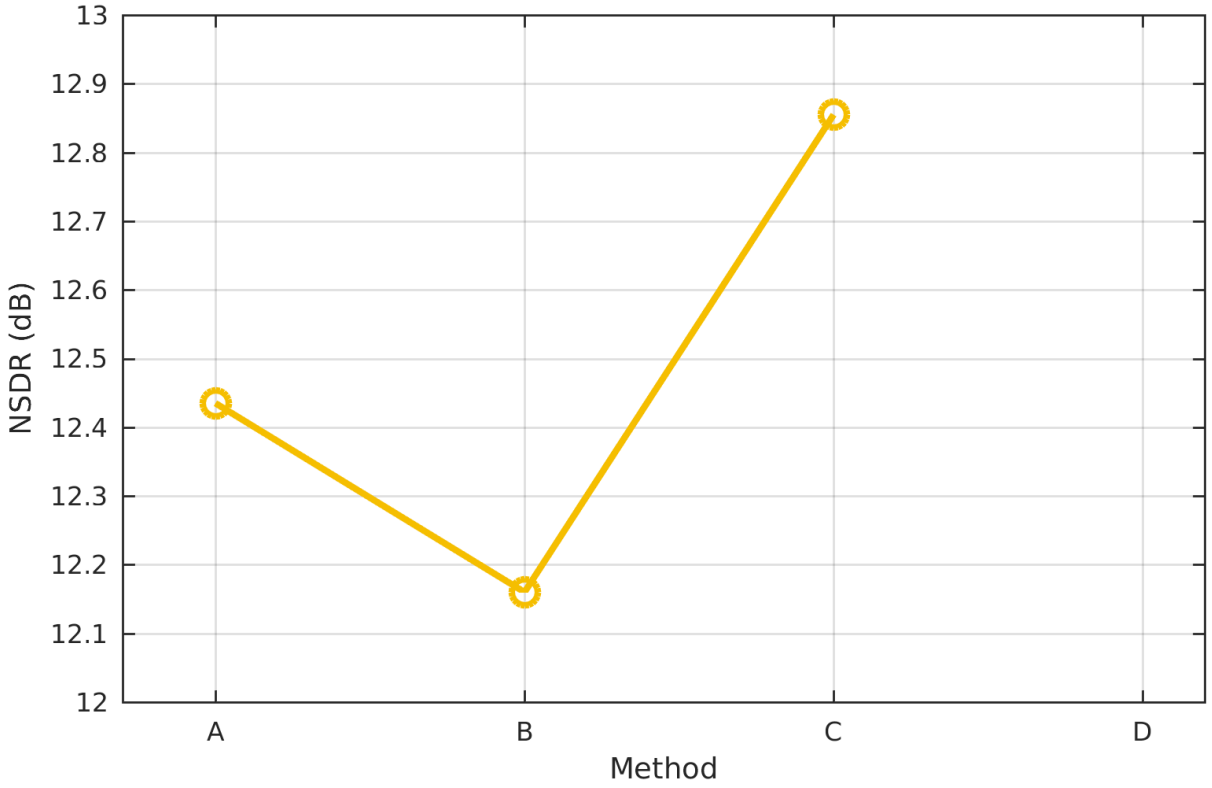
A: NMF baseline

Experiments



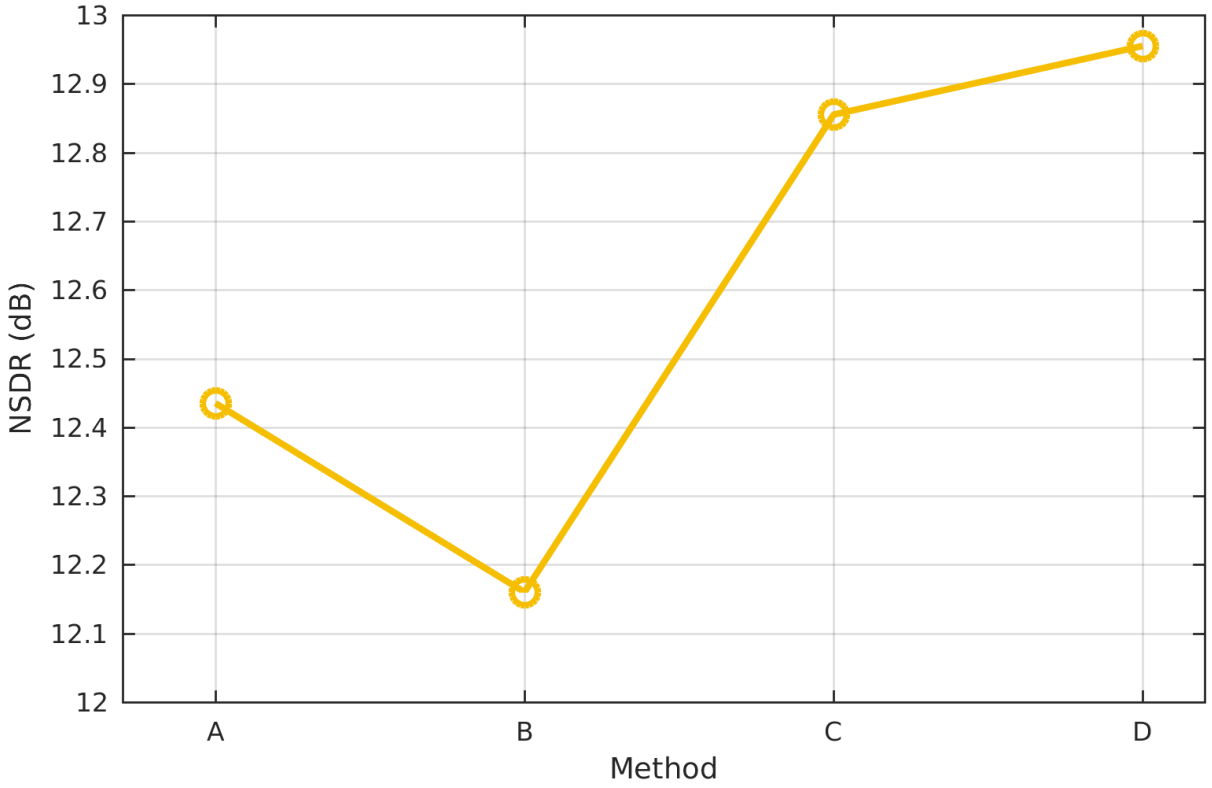
A: NMF baseline
B: Proposed Method

Experiments



A: NMF baseline
B: Proposed Method
C: Proposed + Non. Neg. Weights

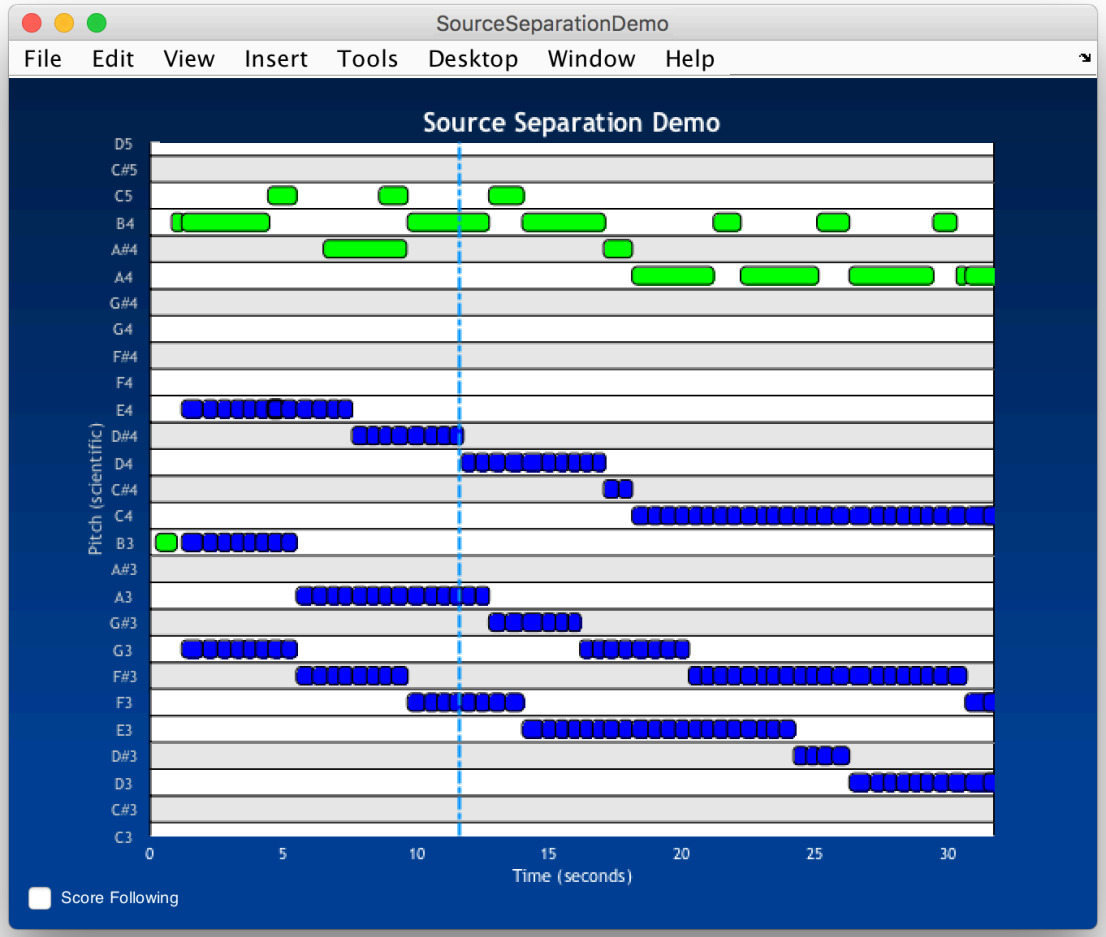
Experiments



A: NMF baseline
B: Proposed Method

C: Proposed + Non. Neg. Weights
D: Proposed + Non. Neg. Weights + Temporal Context

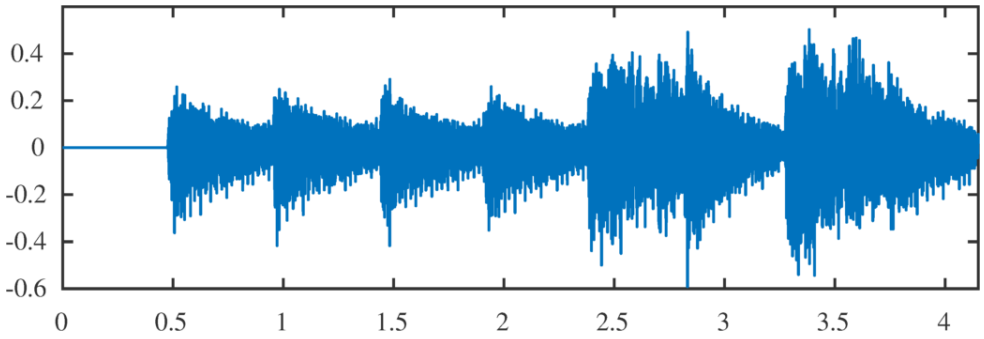
Informed Source Separation with Deep Nets

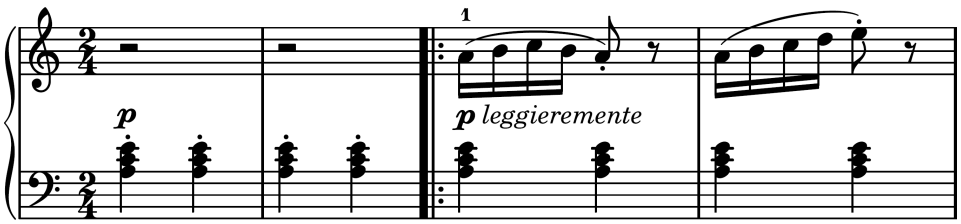


Interface: Jonathan Driedger, Harald Grohganz, Thomas Prätzlich

On the Importance of *Appropriate* Regularization in Difficult Optimization Problems

Example: Music Transcription





Music Transcription – Instrument Dependent

Speaker-Dependent
Speech Recognition

Calibration data for
speaker available

Music Transcription – Instrument Dependent

Speaker-Dependent
Speech Recognition

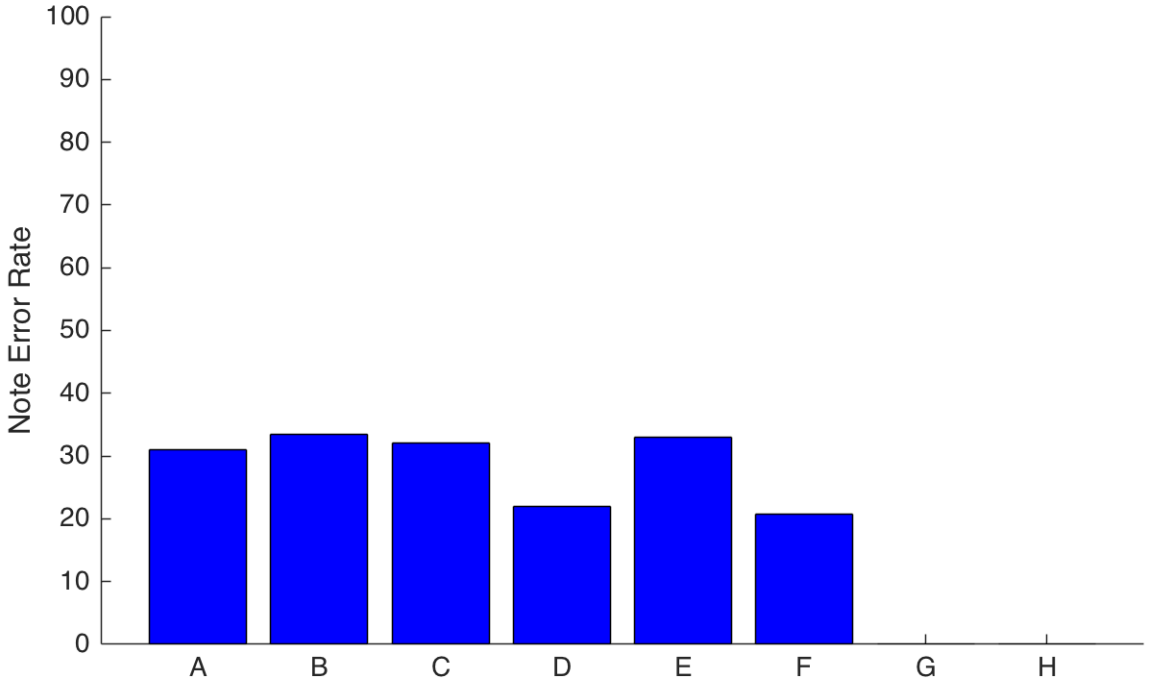
Calibration data for
speaker available

Instrument-Dependent
Music Transcription

Calibration data for
Instrument available

Music Transcription – State of the Art

Instrument Dependent Transcription (Piano)



A: Vincent et al [4] (Harmonic Decomposition)
B: Benetos et al [5] (Shift Invariant NMF)
C: Boeck et al [7] (LSTM)
D: O'Hanlon/Plumbley [8] (Group Sparsity)
E: Sigtia et al [6] (Conv-Net / LSTM Hybrid)
F: Boeck et al [9] (Conv-Net / F-Anova)
G:
H:

(*) Evaluation methodologies and thus numbers not always comparable (take with a grain of salt)

Our Idea

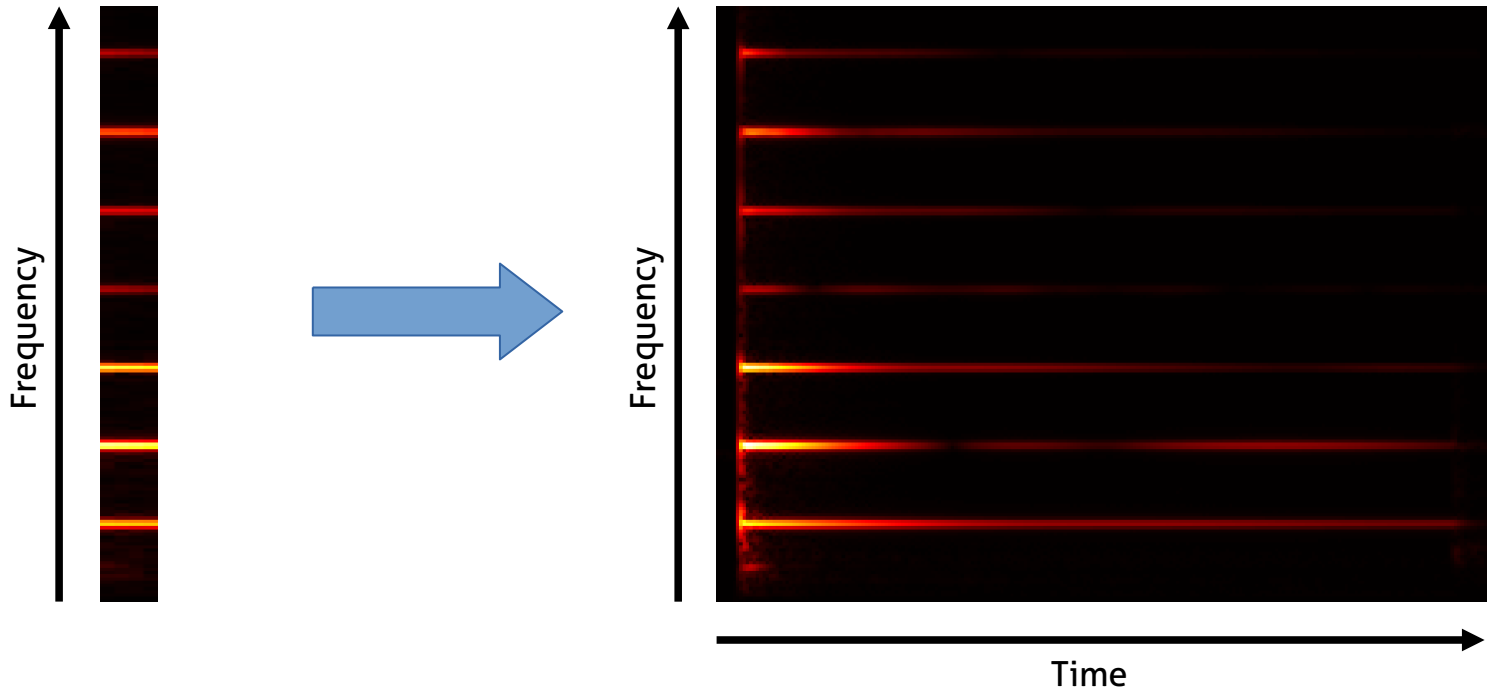
Previous Methods

Note Energy Pattern in Frequency

Our Idea

Previous Methods

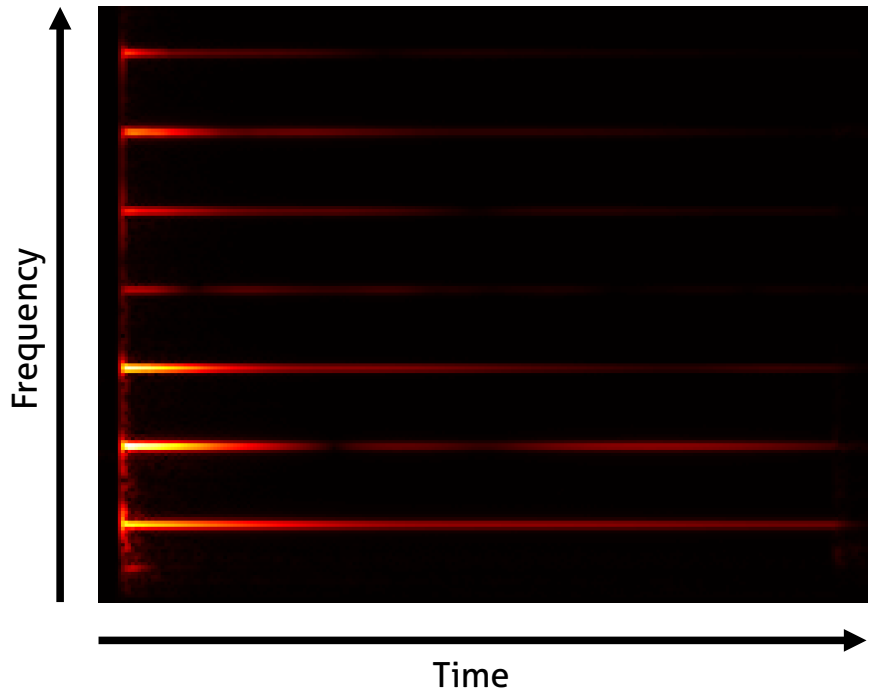
Our Approach



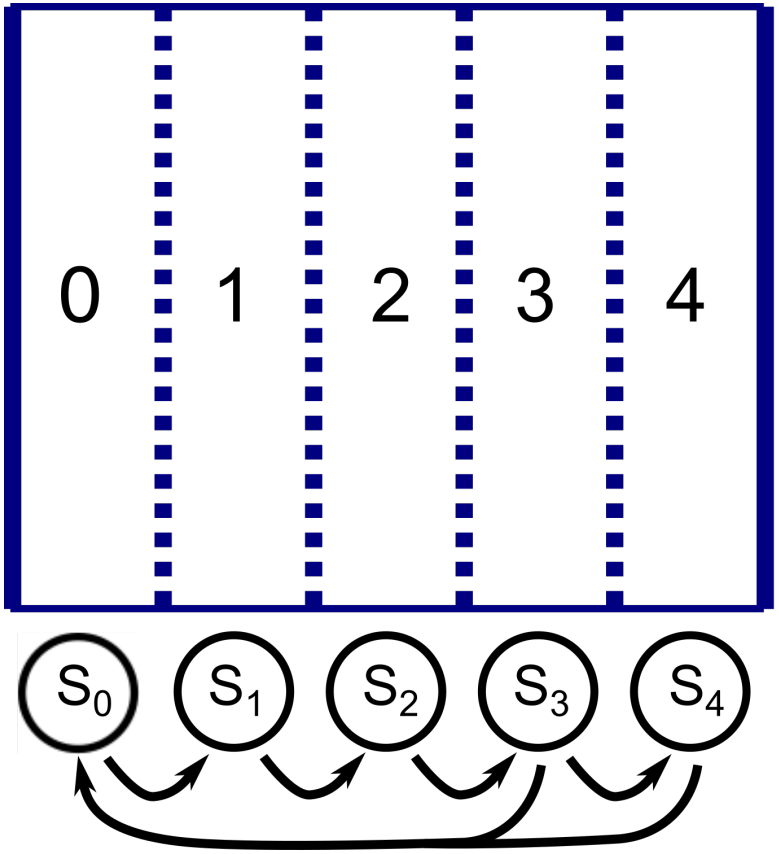
Note Energy Pattern in Frequency

Note Energy Pattern in Time and Frequency

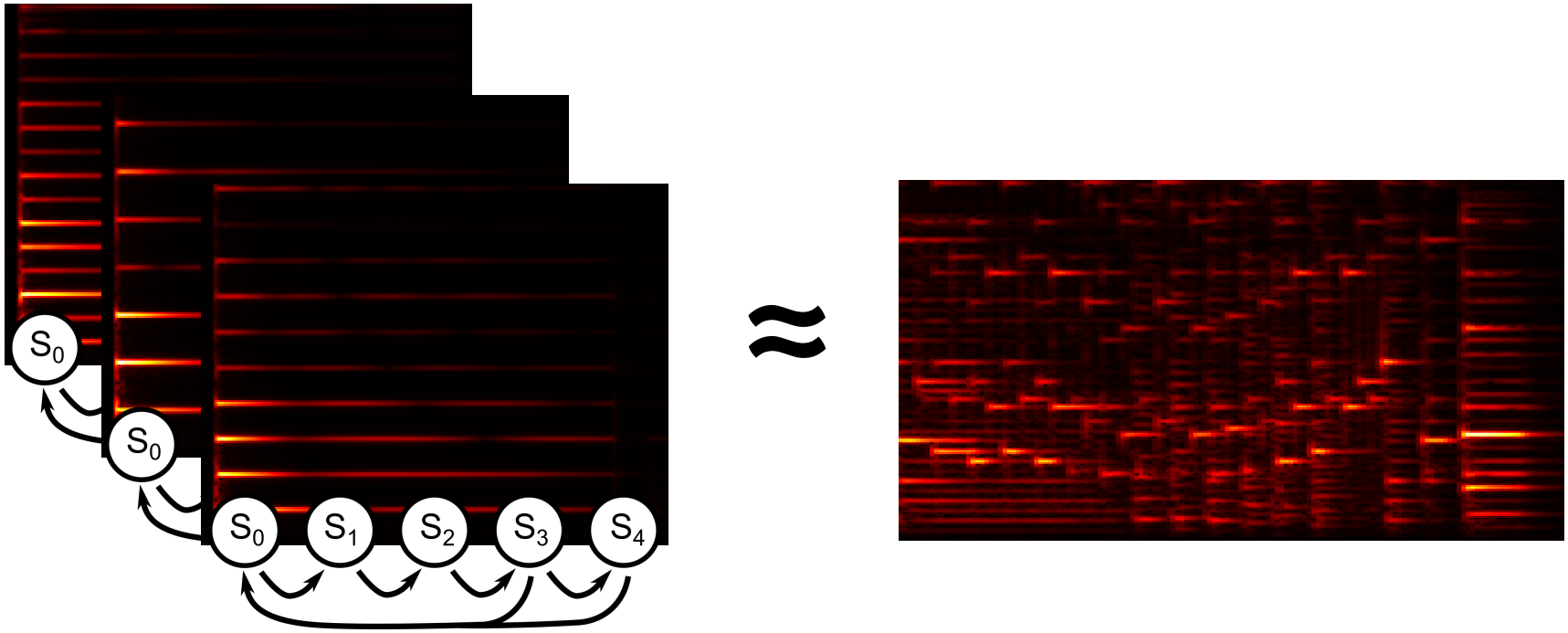
Generative Approach – Markov Condition



Generative Approach – Markov Condition

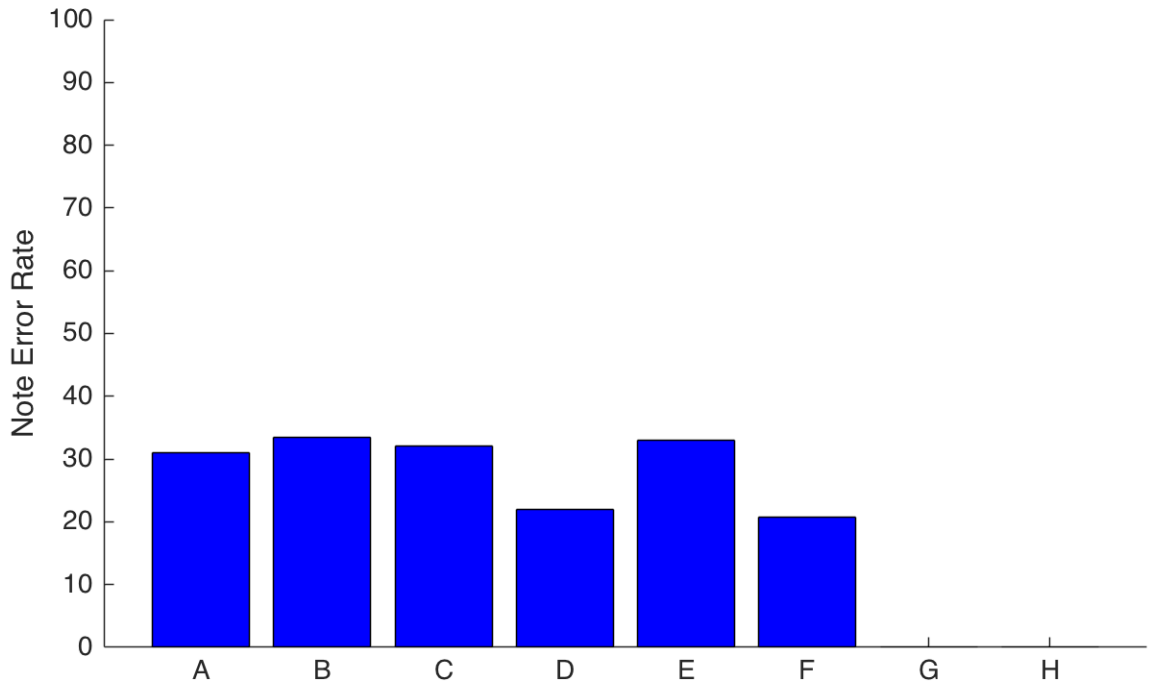


Markov-Constrained Sparse Coding



"What is the most likely combination of these patterns, accounting for the Markov constraints?"

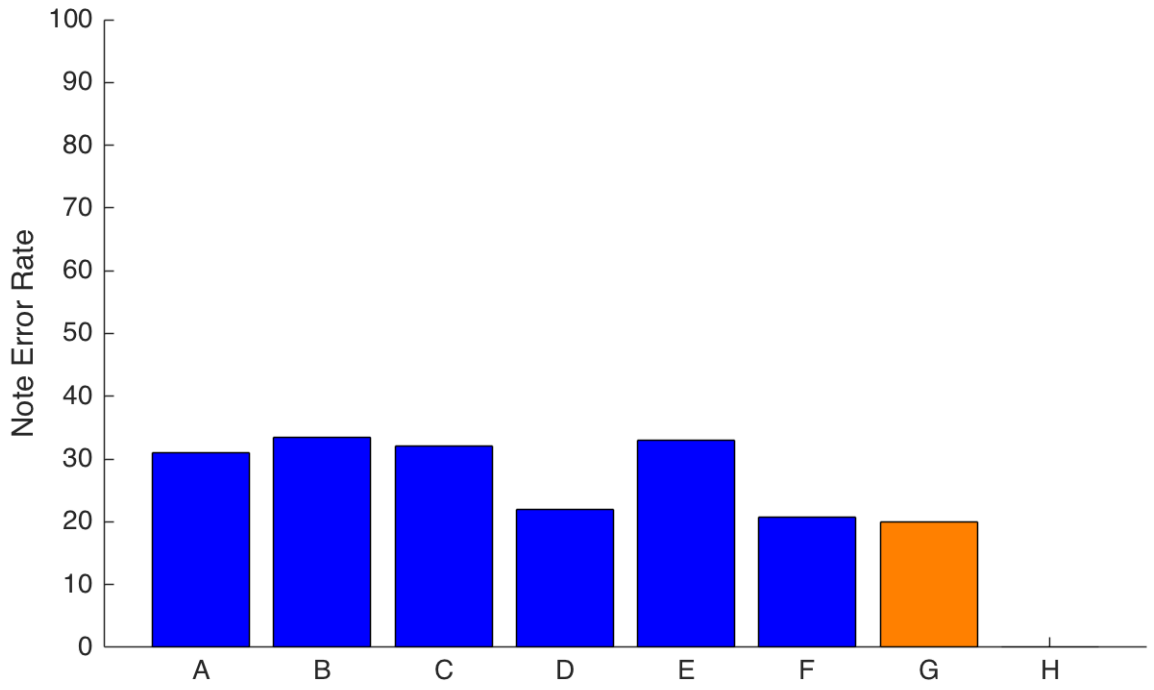
Music Transcription – State of the Art



A: Vincent et al [4] (Harmonic Decomposition)
B: Benetos et al [5] (Shift Invariant NMF)
C: Boeck et al [7] (LSTM)
D: O'Hanlon/Plumbley [8] (Group Sparsity)
E: Sigtia et al [6] (Conv-Net / LSTM Hybrid)
F: Boeck et al [9] (Conv-Net / F-Anova)
G:
H:

(*) Evaluation methodologies and thus numbers not always comparable (take with a grain of salt)

Music Transcription – State of the Art



A: Vincent et al [4] (Harmonic Decomposition)
B: Benetos et al [5] (Shift Invariant NMF)
C: Boeck et al [7] (LSTM)
D: O'Hanlon/Plumbley [8] (Group Sparsity)
E: Sigtia et al [6] (Conv-Net / LSTM Hybrid)
F: Boeck et al [9] (Conv-Net / F-Anova)
G: Ewert et al [10]
H:

(*) Evaluation methodologies and thus numbers not always comparable (take with a grain of salt)

Why so difficult?

Why so difficult?

Reason 1

Experiment: Condition Analysis of Music Transcription (from system analysis perspective)

Result: Even using perfect spectral patterns, the transcription problem has a very high condition (i.e. little noise in observation is strongly amplified in estimates)

Why so difficult?

Reason 1

Experiment: Condition Analysis of Music Transcription (from system analysis perspective)

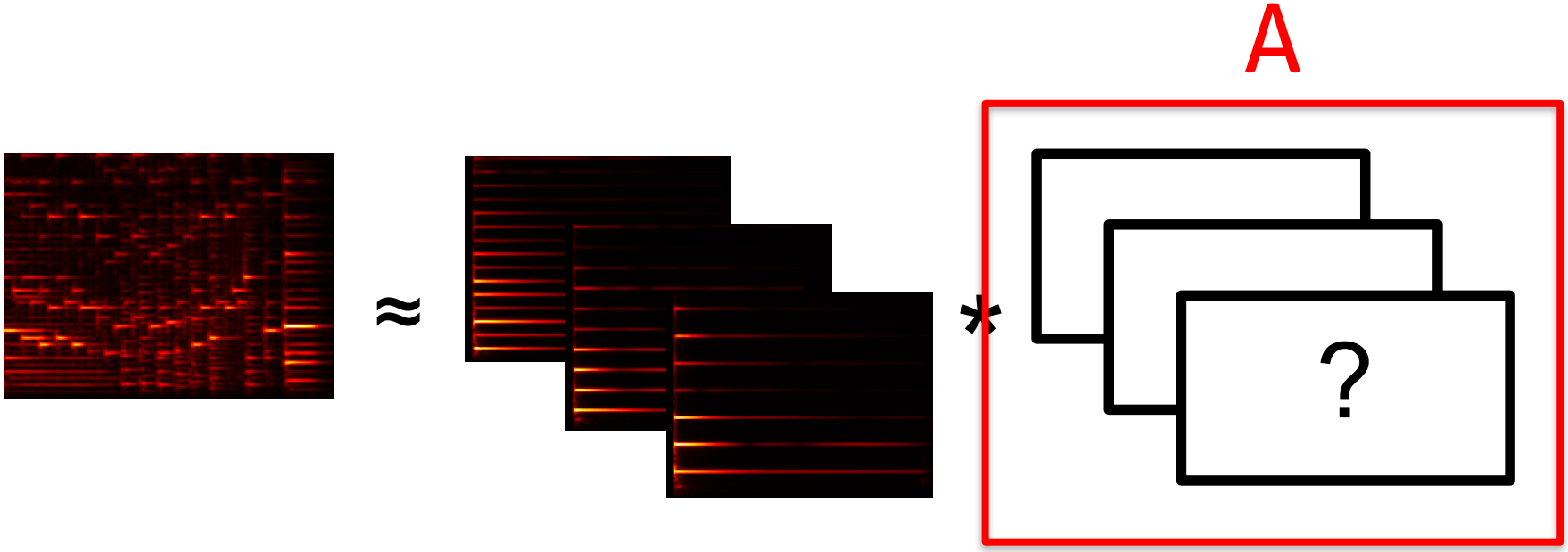
Result: Even using perfect spectral patterns, the transcription problem has a very high condition (i.e. little noise in observation is strongly amplified in estimates)

Reason 2

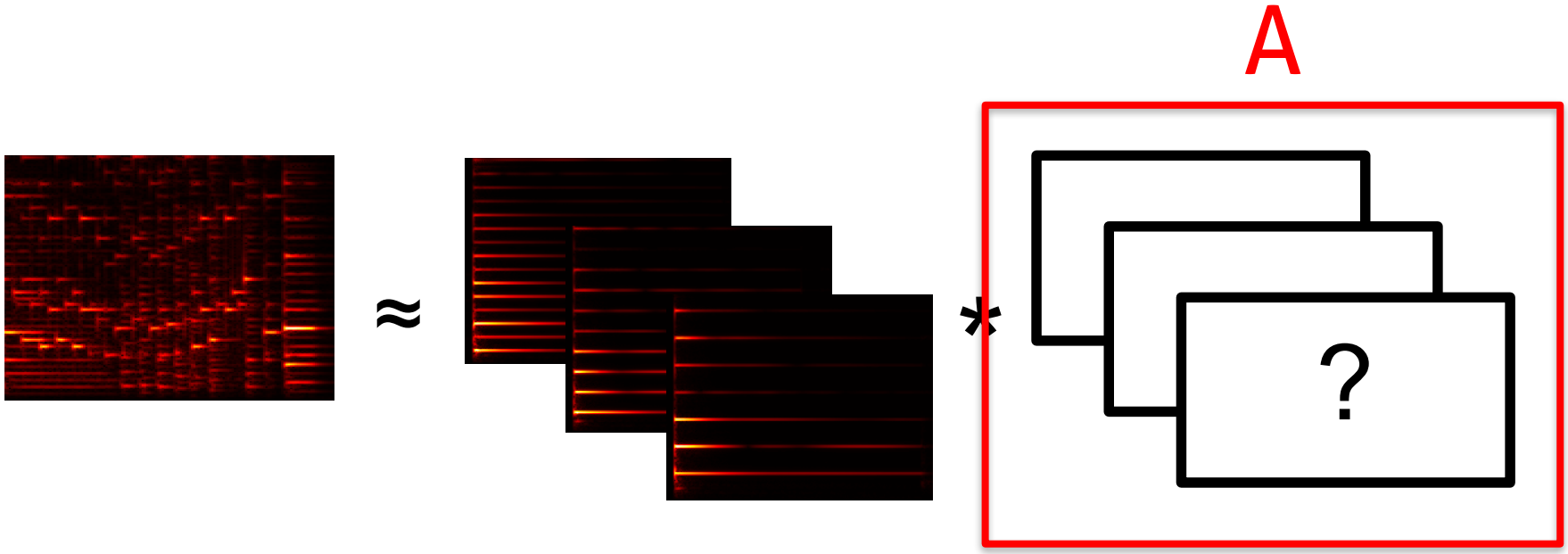
Parameter Estimation for Markov Constraint Model: Many similarities to Factorial HMMs (designed for 2-3 concurrent streams – not 88!)

Proposed parameter estimation: scale up to 88 processes, but too prone to local minima.

Numerics – 2nd Try



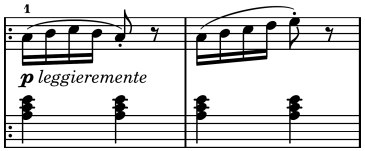
Numerics – 2nd Try

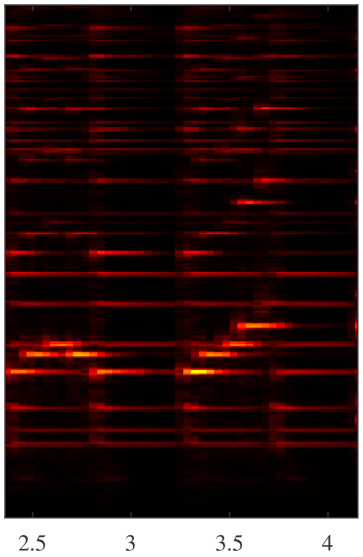


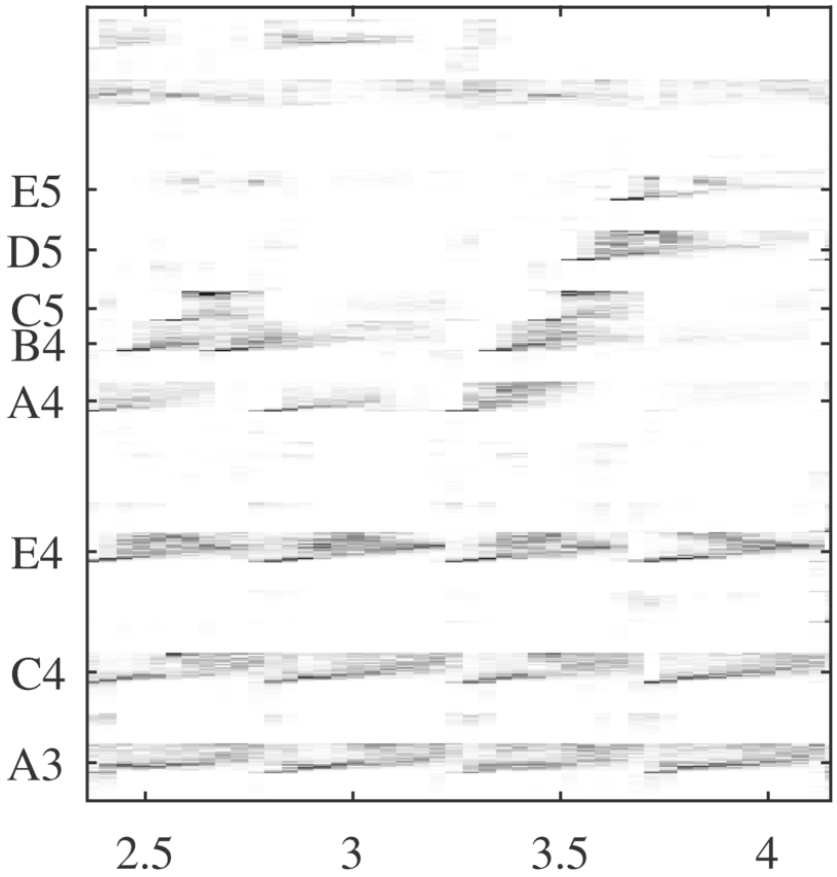
Task

- Everything needs to be soft
- Everything should be as convex as possible
- Encourage, don't enforce

Parameter Estimation – Regularizers





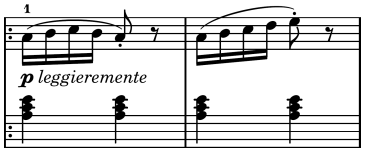


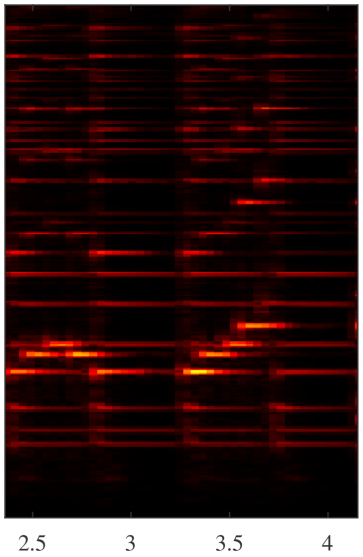
Generalized Kullback-Leibler

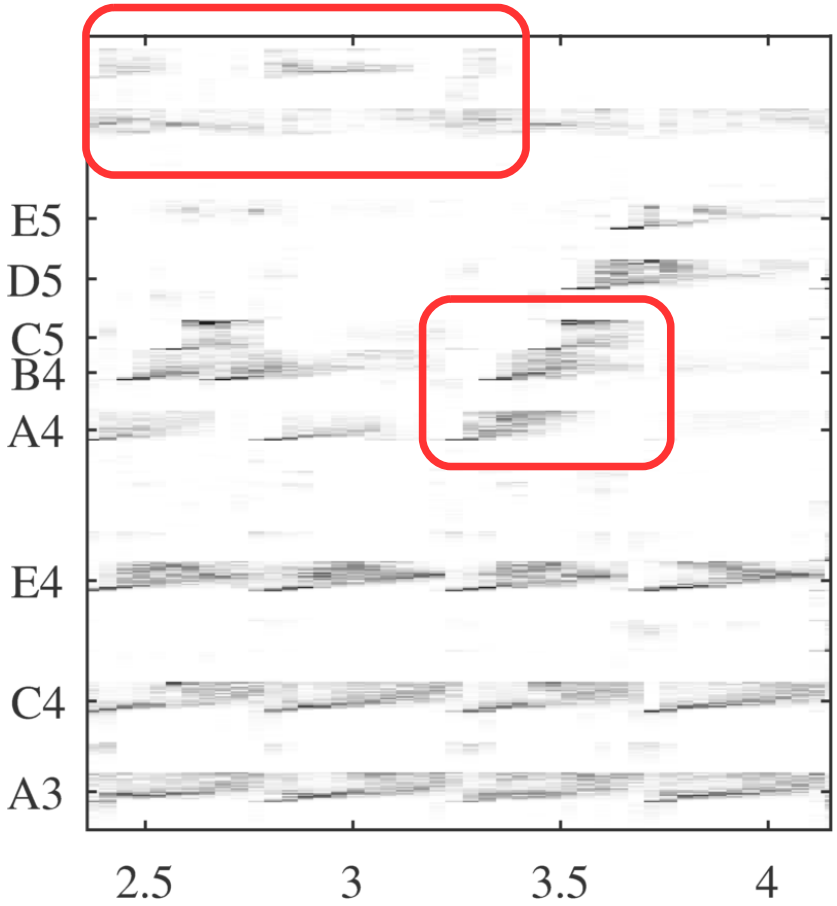
+

Non-Negativity

Parameter Estimation – Regularizers





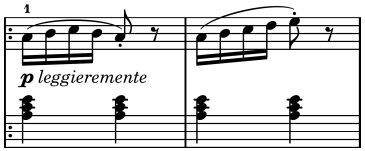


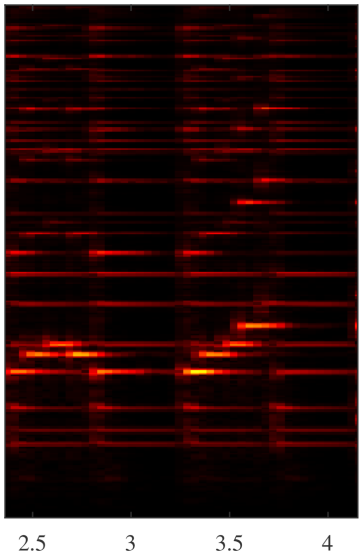
Generalized Kullback-Leibler

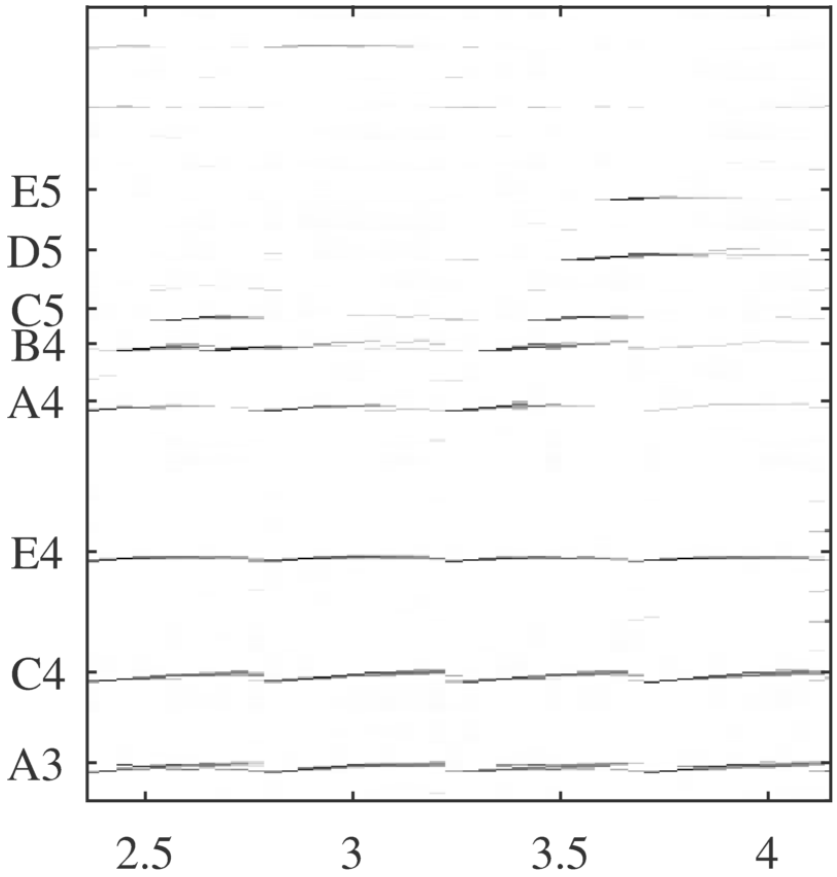
+

Non-Negativity

Parameter Estimation – Regularizers

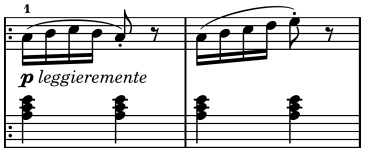


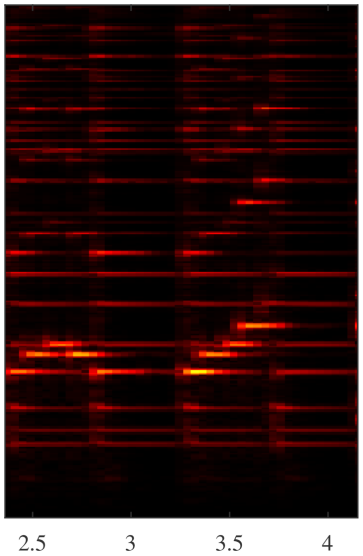


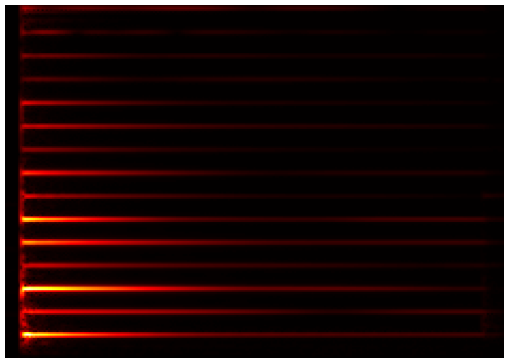


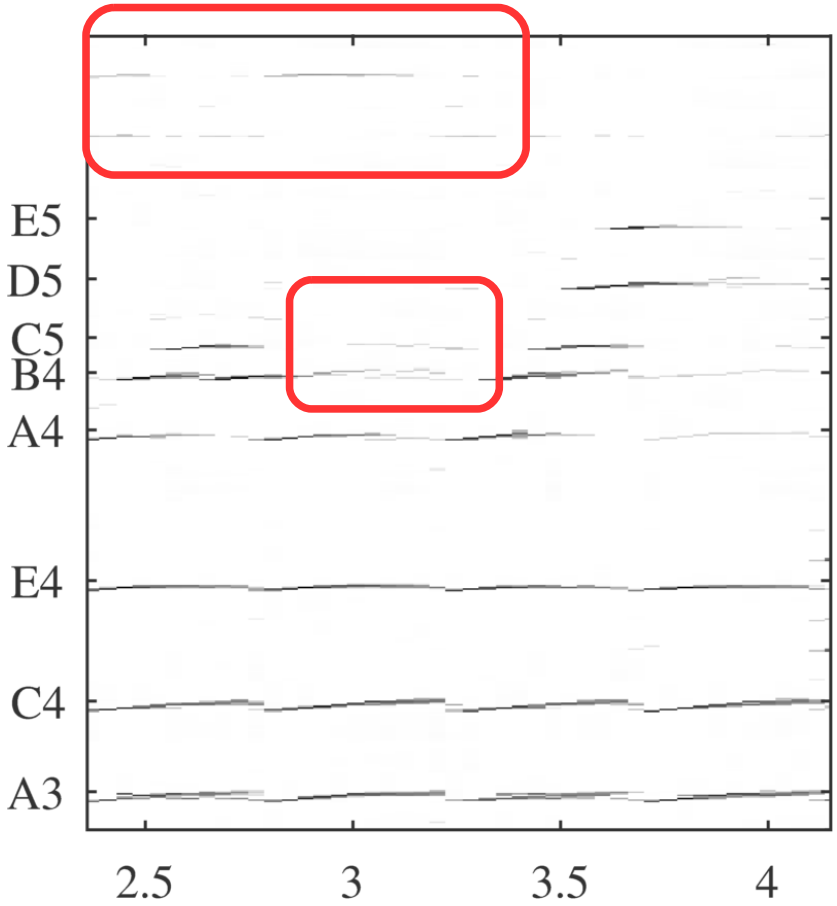
+ L1 Sparsity

Parameter Estimation – Regularizers



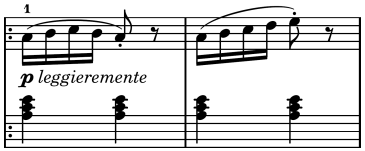


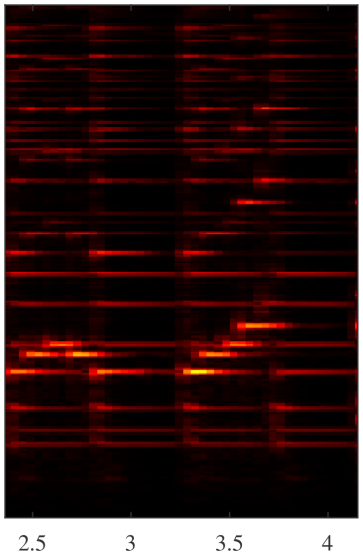


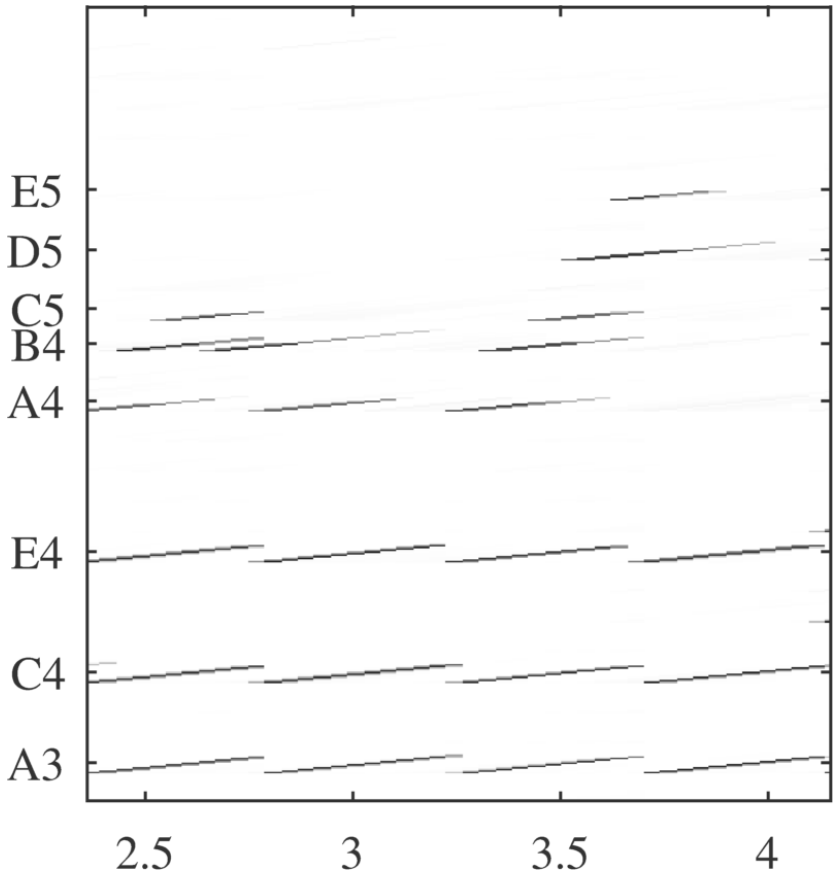


\+ L1 Sparsity

Parameter Estimation – Regularizers

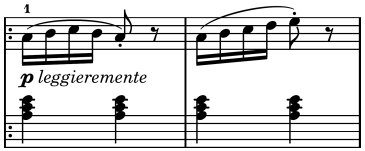


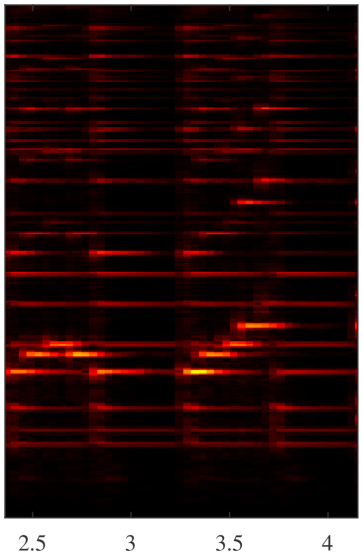


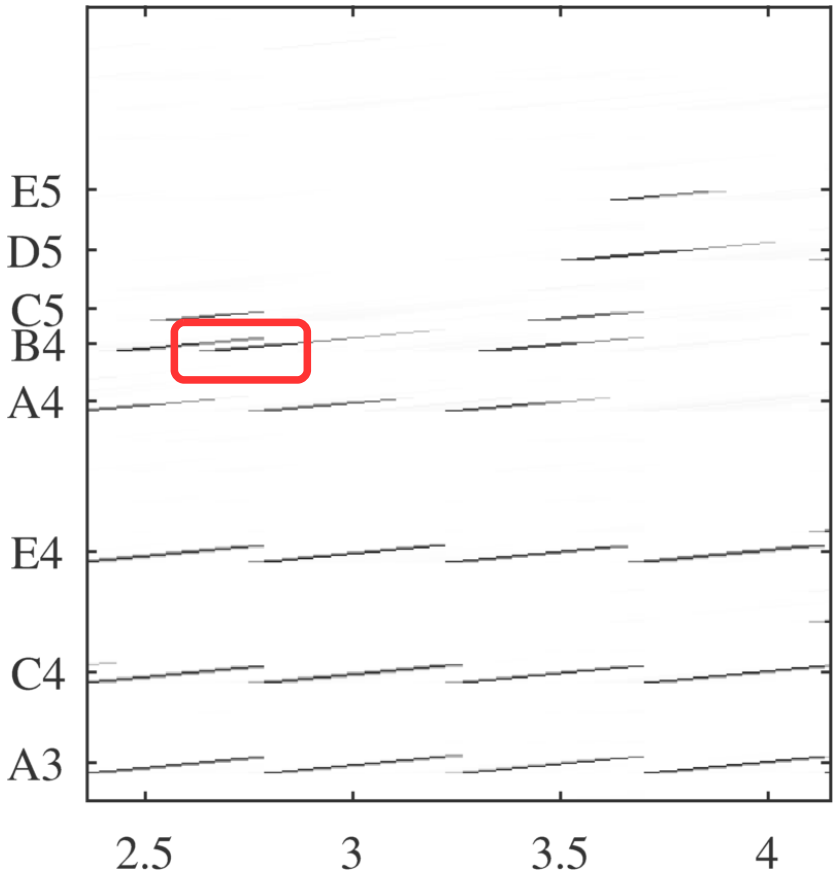


+ Total Directional Variation

Parameter Estimation – Regularizers

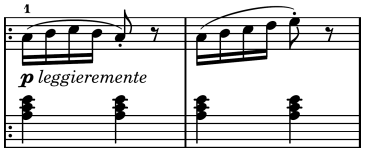


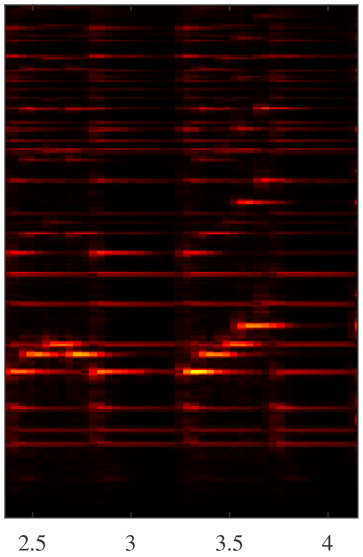


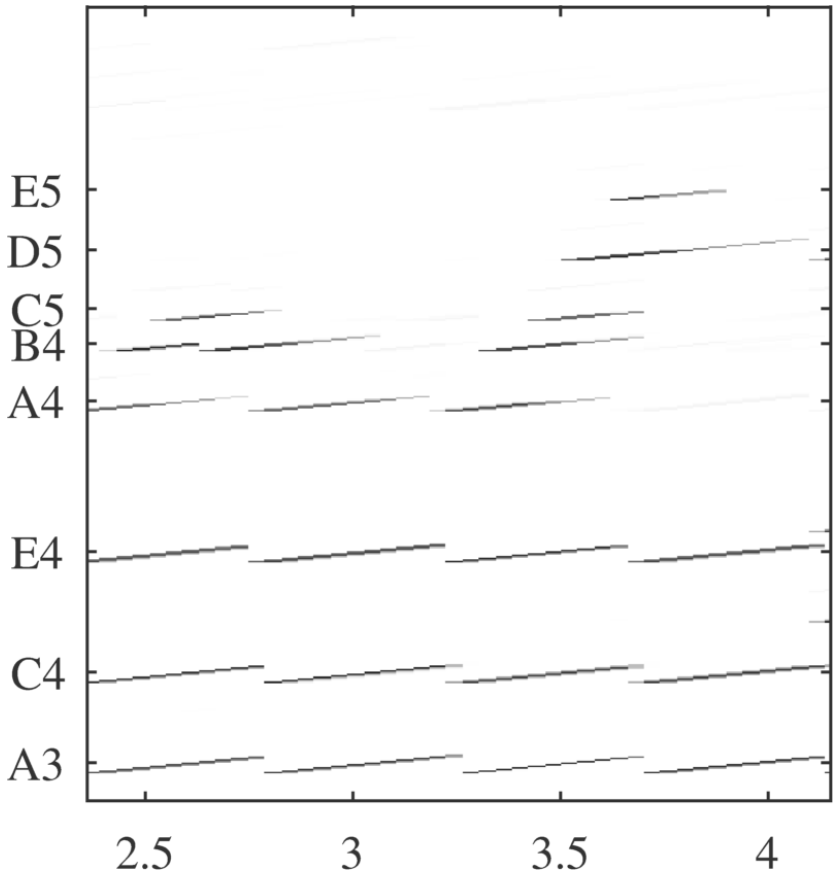


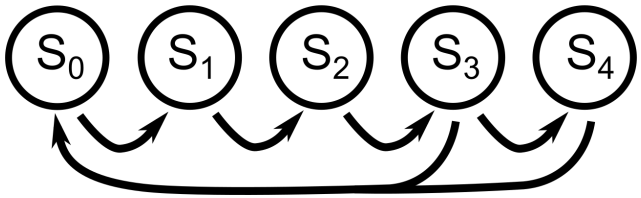
+ Total Directional Variation

Parameter Estimation – Regularizers



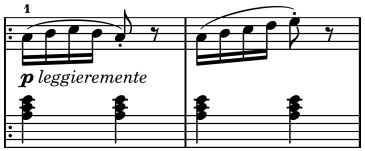


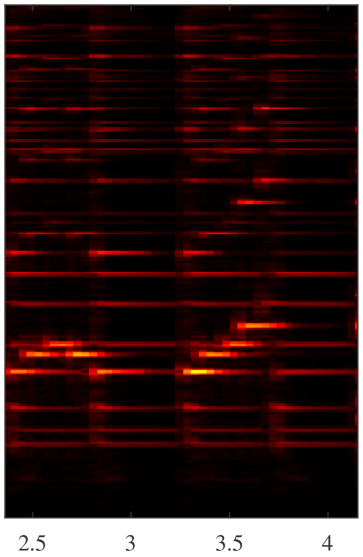


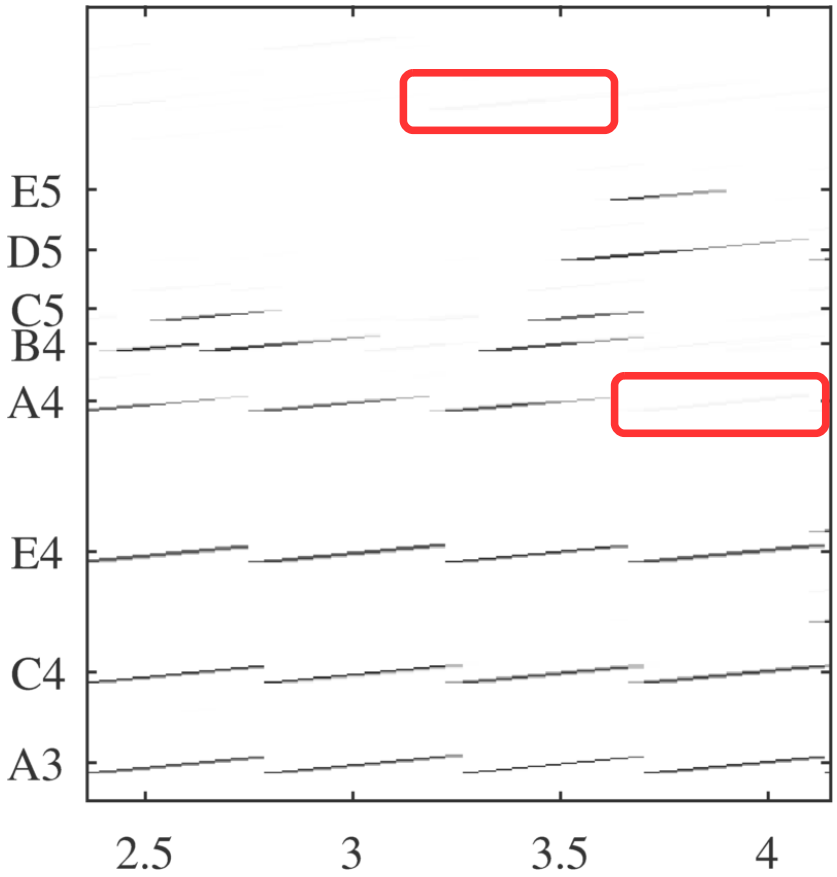


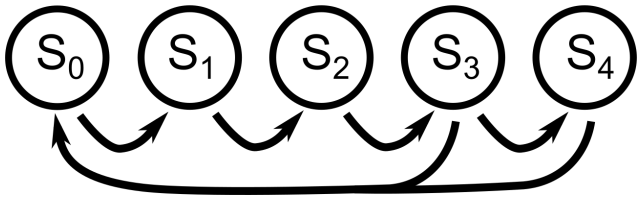
+ Markov-State Regularizer

Parameter Estimation – Regularizers



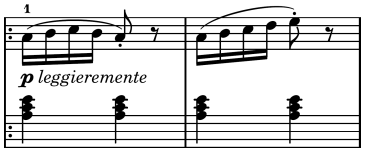


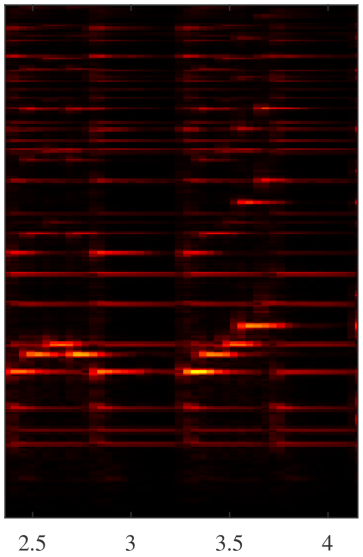


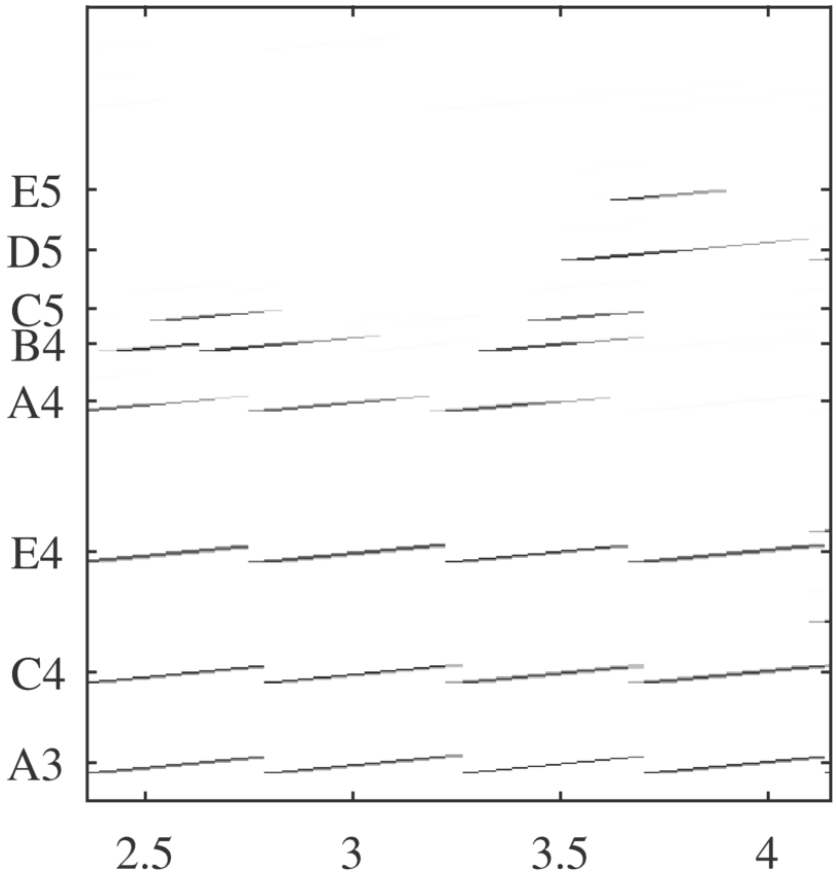


+ Markov-State Regularizer

Parameter Estimation – Regularizers

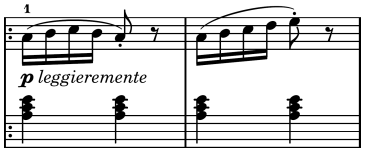


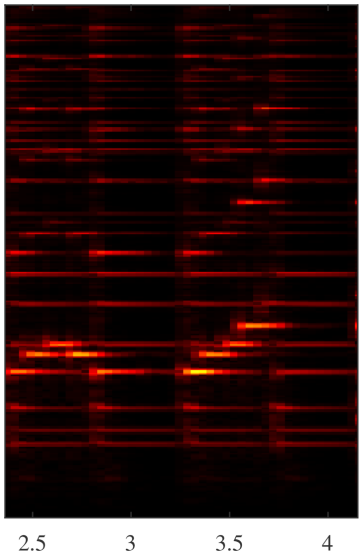


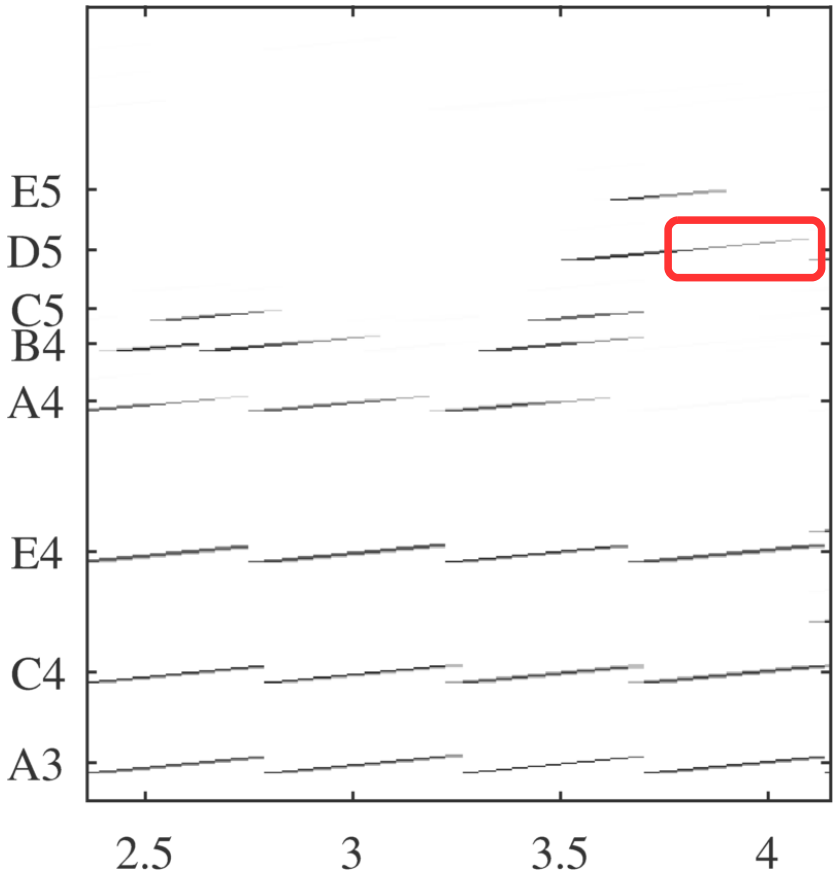


+ Threshold Regularizer

Parameter Estimation – Regularizers

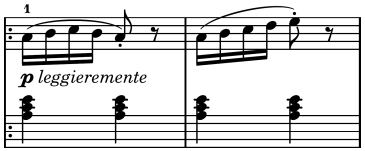


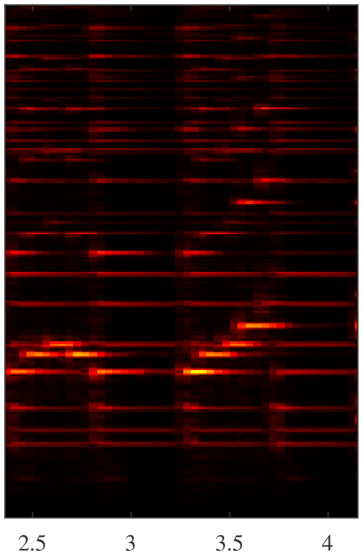


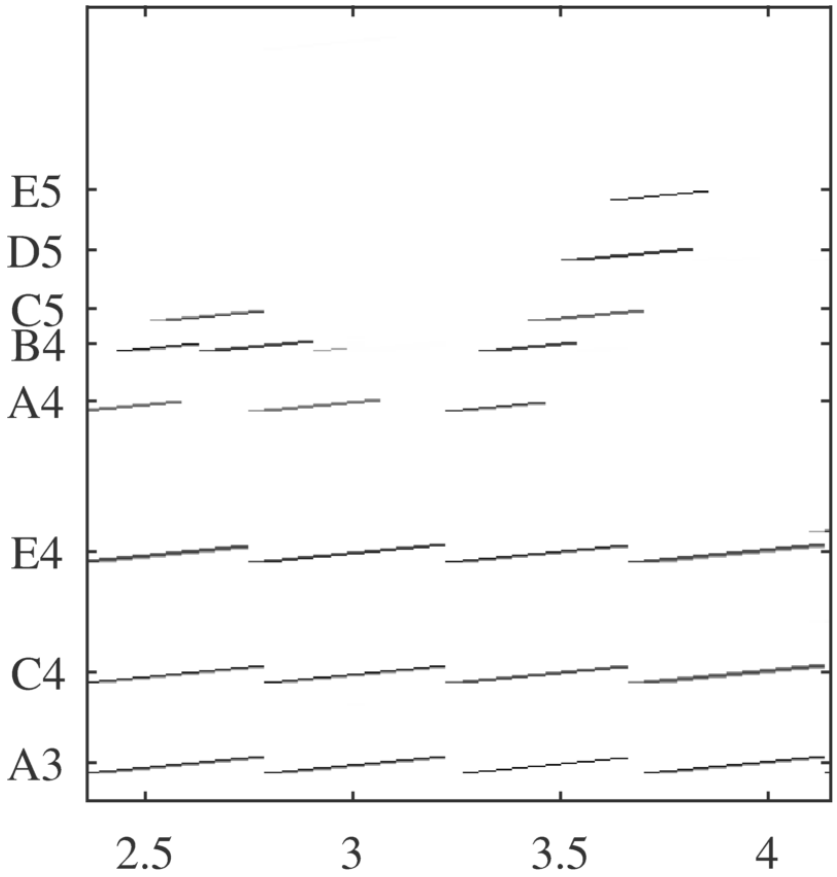


+ Threshold Regularizer

Parameter Estimation – Regularizers







+ Binary Markov-State Regularizer
+ Strict Coupling Regularizer

Parameter Estimation

$$
\begin{aligned}
f(A) :=& D_{KL}(V, PA) \\
&+ \chi_{\mathbb{R}_{\geq 0}^{K \times L \times N}}(A) \\
&+ \lambda_1 \|A\|_1 \\
&+ \lambda_2 \|\Delta_D[A]\|_1 \\
&+ \chi_{\mathcal{M}}(A) \\
&+ \chi_{\mathcal{T}}(A) \\
&+ \lambda_3 \|A - B \odot G\| \\
&+ \chi_{\widetilde{\mathcal{M}}}(B) \\
&+ \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Parameter Estimation

$$
\begin{aligned}
f(A) :=& D_{KL}(V, PA) \\
&+ \chi_{\mathbb{R}_{\geq 0}^{K \times L \times N}}(A) \\
&+ \lambda_1 \|A\|_1 \\
&+ \lambda_2 \|\Delta_D[A]\|_1 \\
&+ \chi_{\mathcal{M}}(A) \\
&+ \chi_{\mathcal{T}}(A) \\
&+ \lambda_3 \|A - B \odot G\| \\
&+ \chi_{\widetilde{\mathcal{M}}}(B) \\
&+ \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Highly Non-Differentiable

Parameter Estimation

$$
\begin{aligned}
f(A) := & D_{KL}(V, PA) \\
& + \chi_{\mathbb{R}_{>0}^{K \times L \times N}}(A) \\
& + \lambda_1 \|A\|_1 \\
& + \lambda_2 \|\Delta_D [A]\|_1 \\
& + \chi_{\mathcal{M}}(A) \\
& + \chi_{\mathcal{T}}(A) \\
& + \lambda_3 \|A - B \odot G\| \\
& + \chi_{\widetilde{\mathcal{M}}}(B) \\
& + \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Highly Non-Differentiable

Infinity as Value

Parameter Estimation

$$
\begin{aligned}
f(A) := & D_{KL}(V, PA) \\
& + \chi_{\mathbb{R}^{K\times L\times N}_{>0}}(A) \\
& + \lambda_1 \|A\|_1 \\
& + \lambda_2 \|\Delta_D[A]\|_1 \\
& + \chi_{\mathcal{M}}(A) \\
& + \chi_{\mathcal{T}}(A) \\
& + \lambda_3 \|A - B \odot G\| \\
& + \chi_{\widetilde{\mathcal{M}}}(B) \\
& + \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Highly Non-Differentiable

Infinity as Value

Highly Non-Convex

Parameter Estimation

$$
\begin{aligned}
f(A) := & D_{KL}(V, PA) \\
& + \chi_{\mathbb{R}^{K \times L \times N}_{>0}}(A) \\
& + \lambda_1 \|A\|_1 \\
& + \lambda_2 \|\Delta_D[A]\|_1 \\
& + \chi_{\mathcal{M}}(A) \\
& + \chi_{\mathcal{T}}(A) \\
& + \lambda_3 \|A - B \odot G\| \\
& + \chi_{\widetilde{\mathcal{M}}}(B) \\
& + \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Highly Non-Differentiable

Infinity as Value

Highly Non-Convex

"How to minimize such a function?"

Proximal Methods

Parikh and Boyd [12]:

"Much like Newton's method is a standard tool for solving unconstrained smooth optimization problems of modest size, proximal algorithms can be viewed as an analogous tool for non-smooth, constrained, large-scale, or distributed versions of these problems."

Proximal Methods

Parikh and Boyd [12]:

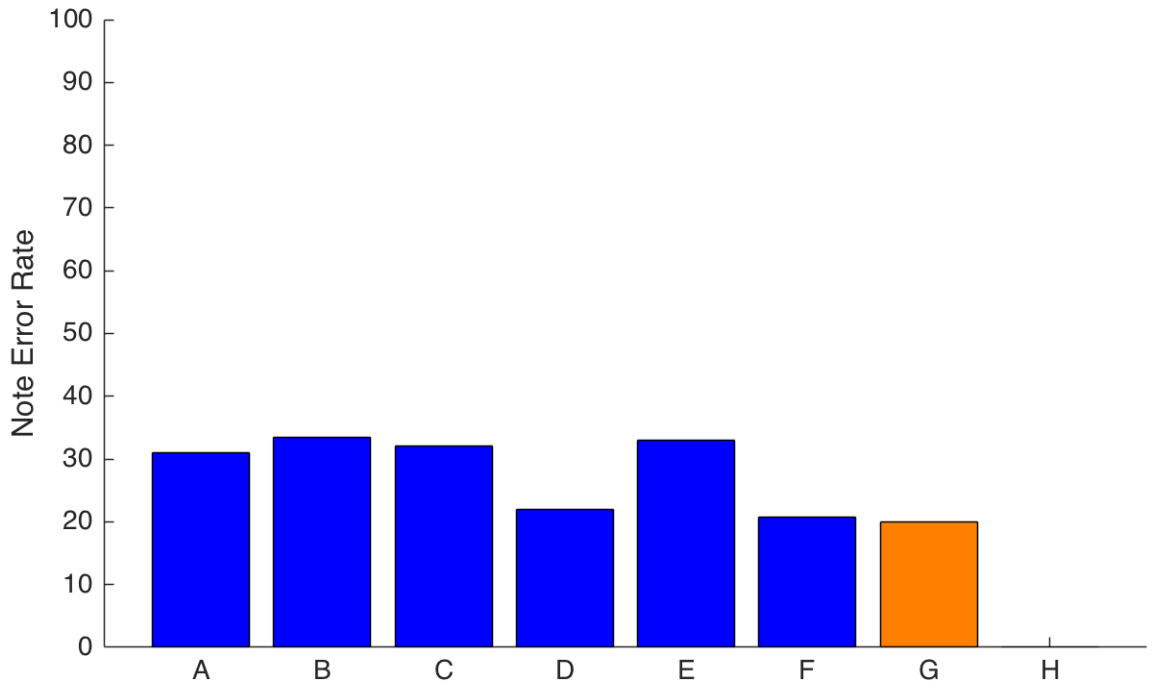
"Much like Newton's method is a standard tool for solving unconstrained smooth optimization problems of modest size, proximal algorithms can be viewed as an analogous tool for non-smooth, constrained, large-scale, or distributed versions of these problems."

$\Rightarrow$ Proximal Methods among the most important classes of methods in optimization

$\Rightarrow$ Not enough time now but could give separate tutorial

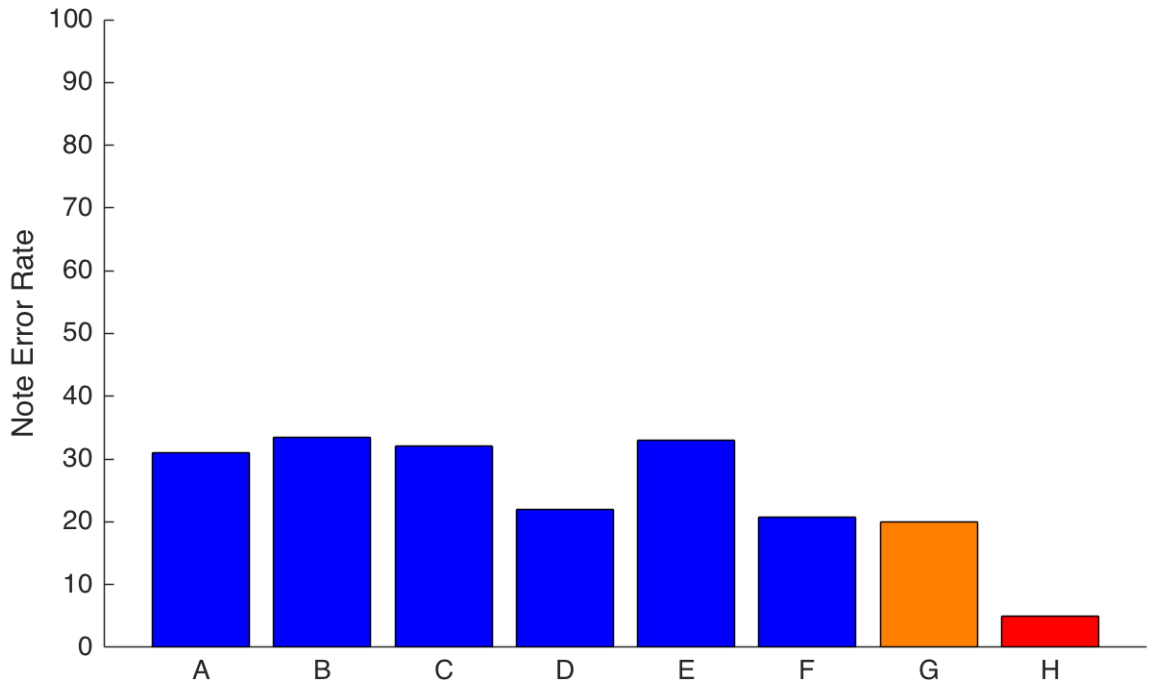
(there is a short intro hidden after the Thank You slide)

Music Transcription – State of the Art



A: Vincent et al [4] (Harmonic Decomposition)
B: Benetos et al [5] (Shift Invariant NMF)
C: Boeck et al [7] (LSTM)
D: O'Hanlon/Plumbley [8] (Group Sparsity)
E: Sigtia et al [6] (Conv-Net / LSTM Hybrid)
F: Boeck et al [9] (Conv-Net / F-Anova)
G: Ewert et al [10]
H:

Music Transcription – State of the Art



A: Vincent et al [4] (Harmonic Decomposition)
B: Benetos et al [5] (Shift Invariant NMF)
C: Boeck et al [7] (LSTM)
D: O'Hanlon/Plumbley [8] (Group Sparsity)
E: Sigtia et al [6] (Conv-Net / LSTM Hybrid)
F: Boeck et al [9] (Conv-Net / F-Anova)
G: Ewert et al [10]
H: Ewert et al [11]

(*) Evaluation methodologies and thus numbers not always comparable (take with a grain of salt)

Regularization in Deep Networks

Understanding the numerics and behaviour of optimization processes essential for successful machine learning.

Regularization in Deep Networks

Understanding the numerics and behaviour of optimization processes essential for successful machine learning.

$\Rightarrow$ Transfer concepts to deep networks (Daniel Stoller, Delia Fano Yela)

Conclusions

- *Understanding behaviour* of optimization processes is important for every application involving machine learning
- *Regularization + Relaxation*:
  - more than L_1/L_2
  - Can make all the difference
- *Proximal methods:* Powerful tool

References

[1] Marolt, Matija. "A connectionist approach to automatic transcription of polyphonic piano music." *IEEE Transactions on Multimedia* 6.3 (2004).

[2] Klapuri, Anssi. "Multiple Fundamental Frequency Estimation by Summing Harmonic Amplitudes.", *International Society for Music Information Retrieval Conference (ISMIR)*, 2006.

[3] FitzGerald, Derry, Matt Cranitch, and Eugene Coyle. "Extended nonnegative tensor factorisation models for musical sound source separation." *Computational Intelligence and Neuroscience* (2008).

[4] Vincent, Emmanuel, Nancy Bertin, and Roland Badeau. "Adaptive harmonic spectral decomposition for multiple pitch estimation." *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 18.3 (2010).

[5] Benetos, Emmanouil, and Simon Dixon. "A shift-invariant latent variable model for automatic music transcription." *Computer Music Journal* 36.4 (2012).

[6] Sigtia, Siddharth, Emmanouil Benetos, and Simon Dixon. "An end-to-end neural network for polyphonic piano music transcription." *IEEE/ACM Transactions on Audio, Speech and Language Processing* 24.5 (2016).

[7] Böck, Sebastian, and Markus Schedl. "Polyphonic piano note transcription with recurrent neural networks.", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

[8] O'Hanlon, Ken, Hidehisa Nagano, and Mark D. Plumbley. "Structured sparsity for automatic music transcription.", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

[9] Kelz, R., Dorfer, M., Korzeniowski, F., Böck, S., Arzt, A., & Widmer, G. (2016). On the Potential of Simple Framewise Approaches to Piano Transcription. *arXiv preprint arXiv:1612.05153*.

[10] Ewert, Sebastian, Mark D. Plumbley, and Mark Sandler. "A dynamic programming variant of non-negative matrix deconvolution for the transcription of struck string instruments.", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[11] Ewert, Sebastian, and Mark Sandler. "Piano transcription in the studio using an extensible alternating directions framework." *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.11 (2016).

References

[12] Parikh, Neal, and Stephen Boyd. "Proximal algorithms." *Foundations and Trends in Optimization* 1.3 (2014): 127-239.

[13] Eckstein, J. and D. P. Bertsekas, “On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators,” Mathematical Programming, vol. 55 (1992).

[14] Keskar, Nitish Shirish, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. "On large-batch training for deep learning: Generalization gap and sharp minima." *arXiv preprint arXiv:1609.04836* (2016).

[15] Goodfellow, Ian J., Oriol Vinyals, and Andrew M. Saxe. "Qualitatively characterizing neural network optimization problems." *arXiv preprint arXiv:1412.6544* (2014).

[16] Graves, Alex, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks.”, ACM *International Conference on Machine Learning*, 2006.

[17] Hershey, John R., Jonathan Le Roux, and Felix Weninger. "Deep unfolding: Model-based inspiration of novel deep architectures." *arXiv preprint arXiv:1409.2574* (2014).

[18] Ewert, Sebastian, and Mark B. Sandler. "Structured Dropout for Weak Label and Multi-Instance Learning and Its Application to Score-Informed Source Separation.”, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

Thanks!*

(*) Up next: *Introduction to Proximal Methods and ADMM (another 6-7 slides)*

Difficult to minimize

$$
\begin{aligned}
f(A) :=& D_{KL}(V, PA) \\
&+ \chi_{\mathbb{R}_{\geq 0}^{K \times L \times N}}(A) \\
&+ \lambda_1 \|A\|_1 \\
&+ \lambda_2 \|\Delta_D[A]\|_1 \\
&+ \chi_{\mathcal{M}}(A) \\
&+ \chi_{\mathcal{T}}(A) \\
&+ \lambda_3 \|A - B \odot G\| \\
&+ \chi_{\widetilde{\mathcal{M}}}(B) \\
&+ \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Difficult to minimize

$$
\begin{aligned}
f(A) :=& D_{KL}(V, PA) \\
&+ \chi_{\mathbb{R}_{\geq 0}^{K \times L \times N}}(A) \\
&+ \lambda_1 \|A\|_1 \\
&+ \lambda_2 \|\Delta_D[A]\|_1 \\
&+ \chi_{\mathcal{M}}(A) \\
&+ \chi_{\mathcal{T}}(A) \\
&+ \lambda_3 \|A - B \odot G\| \\
&+ \chi_{\widetilde{\mathcal{M}}}(B) \\
&+ \chi_{\widetilde{\mathcal{T}}}(G)
\end{aligned}
$$

Idea:
"This would be much easier if we could minimize these individually!"

ADMM – Mini-Introduction

Alternating Directions Method of Multipliers (ADMM) solves problems of the following type:

$$\begin{aligned} \underset{x,z}{\text{argmin}} \quad & f(x) + g(z) \\ \text{subject to} \quad & Bx + Cz = c \end{aligned}$$

ADMM – Mini-Introduction

Alternating Directions Method of Multipliers (ADMM) solves problems of the following type:

$$\begin{aligned} \underset{x,z}{\text{argmin}} \quad & f(x) + g(z) \\ \text{subject to} \quad & Bx + Cz = c \end{aligned}$$

equivalent to...

$$\underset{\beta}{\text{argmax}} \inf_{x,z} L_\rho(x, z, \beta),$$

$$L_\rho(x, z, \beta) := f(x) + g(z) + \langle \beta, Bx + Cz - c \rangle + (\rho/2)\left\| Bx + Cz - c \right\|_2^2$$

ADMM – Mini-Introduction

Alternating Directions Method of Multipliers (ADMM) solves problems of the following type:

$$\underset{x,z}{\operatorname{argmin}} \quad f(x) + g(z)$$

$$\text{subject to} \quad Bx + Cz = c$$

equivalent to...

$$\underset{\beta}{\operatorname{argmax}} \inf_{x,z} L_\rho(x, z, \beta),$$

$$L_\rho(x, z, \beta) := f(x) + g(z) + \langle \beta, Bx + Cz - c \rangle + (\rho/2)\left\| Bx + Cz - c \right\|_2^2$$

ADMM – Mini-Introduction

Alternating Directions Method of Multipliers (ADMM) solves problems of the following type:

$$\underset{x,z}{\operatorname{argmin}} \; f(x) + g(z)$$

$$\text{subject to} \quad Bx + Cz = c$$

equivalent to...

$$\underset{\beta}{\operatorname{argmax}} \inf_{x,z} L_\rho(x, z, \beta),$$

$$L_\rho(x, z, \beta) := f(x) + g(z) + \langle \beta, Bx + Cz - c \rangle + (\rho/2) \left\| Bx + Cz - c \right\|_2^2$$

ADMM – Mini-Introduction

Simple Algorithm:

$$(x^{k+1}, z^{k+1}) := \operatorname*{argmin}_{x,z} L_\rho(x, z, \beta^k)$$

$$\beta^{k+1} := \beta^k + \rho(Bx^{k+1} + Cz^{k+1} - c)$$

ADMM – Mini-Introduction

Simple Algorithm:

$$(x^{k+1}, z^{k+1}) := \operatorname*{argmin}_{x,z} L_\rho(x, z, \beta^k)$$

$$\beta^{k+1} := \beta^k + \rho(Bx^{k+1} + Cz^{k+1} - c)$$

Problem: Difficult

ADMM – Mini-Introduction

Simple Algorithm:

$$(x^{k+1}, z^{k+1}) := \underset{x,z}{\operatorname{argmin}}\, L_\rho(x, z, \beta^k)$$

$$\beta^{k+1} := \beta^k + \rho(Bx^{k+1} + Cz^{k+1} - c)$$

Problem: Difficult

ADMM:

$$x^{k+1} := \underset{x}{\operatorname{argmin}}\, L_\rho(x, z^k, \beta^k)$$

$$z^{k+1} := \underset{z}{\operatorname{argmin}}\, L_\rho(x^{k+1}, z, \beta^k)$$

$$\beta^{k+1} := \beta^k + \rho(Bx^{k+1} + Cz^{k+1} - c)$$

Breakthrough (Bertsekas [13]): This converges under VERY mild conditions

ADMM – Mini-Introduction

How is that useful? Our case:

$$\underset{A}{\operatorname{argmin}} \sum_{i=1}^{I} f_i(C_i A)$$

ADMM – Mini-Introduction

How is that useful? Our case:

$$\underset{A}{\operatorname{argmin}} \sum_{i=1}^{I} f_i(C_i A)$$

equivalent to...

$$\begin{aligned} \underset{x_1,\ldots,x_I,A}{\operatorname{argmin}} \quad & \sum_{i=1}^{I} f_i(x_i) \\ \text{subject to} \quad & \forall i \in \{1,\ldots,I\} : x_i = C_i A \end{aligned}$$

ADMM – Mini-Introduction

How is that useful? Our case:

$$\underset{A}{\operatorname{argmin}} \sum_{i=1}^{I} f_i(C_i A)$$

equivalent to...

$$\begin{aligned} \underset{x_1,\ldots,x_I,A}{\operatorname{argmin}} \quad & \sum_{i=1}^{I} f_i(x_i) \\ \text{subject to} \quad & \forall i \in \{1,\ldots,I\} : x_i = C_i A \end{aligned}$$

Almost ready for ADMM!

ADMM – Mini-Introduction

Getting ready to apply ADMM:

$$\begin{aligned} \underset{x_1,\ldots,x_I,A}{\text{argmin}} \quad & \sum_{i=1}^{I} f_i(x_i) \\ \text{subject to} \quad & \forall i \in \{1,\ldots,I\} : x_i = C_i A \end{aligned}$$

equivalent to...

ADMM – Mini-Introduction

Getting ready to apply ADMM:

$$\begin{aligned} \underset{x_1,\ldots,x_I,A}{\text{argmin}} \quad & \sum_{i=1}^{I} f_i(x_i) \\ \text{subject to} \quad & \forall i \in \{1,\ldots,I\} : x_i = C_i A \end{aligned}$$

equivalent to...

$$\begin{aligned} \underset{X,\overline{A}}{\text{argmin}} \quad & f(X) + g(\overline{A}) \\ \text{subject to} \quad & X - C\overline{A} = 0 \end{aligned}$$

$$\begin{aligned} X &:= (x_1,\ldots,x_I) \\ f(X) &:= \sum_{i=1}^{I} f_i(x_i) \\ \mathcal{A} &:= \{\overline{A} := (A_1,\ldots,A_I) | A_1 = \ldots = A_I\} \\ g &:= \chi_{\mathcal{A}} \end{aligned}$$

ADMM – Mini-Introduction

Getting ready to apply ADMM:

Our Objective Function

$$\underset{x_1,\ldots,x_I,A}{\operatorname{argmin}} \quad \sum_{i=1}^{I} f_i(x_i)$$

$$\text{subject to} \quad \forall i \in \{1,\ldots,I\} : x_i = C_i A$$

equivalent to...

$$\underset{X,\overline{A}}{\operatorname{argmin}} \quad f(X) + g(\overline{A})$$

$$\text{subject to} \quad X - C\overline{A} = 0$$

$$X := (x_1,\ldots,x_I)$$

$$f(X) := \sum_{i=1}^{I} f_i(x_i)$$

$$\mathcal{A} := \{\overline{A} := (A_1,\ldots,A_I) | A_1 = \ldots = A_I\}$$

$$g := \chi_{\mathcal{A}}$$

ADMM – Mini-Introduction

Getting ready to apply ADMM:

Our Objective Function

Linear Constraints

$$\underset{x_1,\ldots,x_I,A}{\operatorname{argmin}} \quad \sum_{i=1}^{I} f_i(x_i)$$

$$\text{subject to} \quad \forall i \in \{1,\ldots,I\} : x_i = C_i A$$

equivalent to...

$$\underset{X,\overline{A}}{\operatorname{argmin}} \quad f(X) + g(\overline{A})$$

$$\text{subject to} \quad X - C\overline{A} = 0$$

$$X := (x_1,\ldots,x_I)$$

$$f(X) := \sum_{i=1}^{I} f_i(x_i)$$

$$\mathcal{A} := \{\overline{A} := (A_1,\ldots,A_I) | A_1 = \ldots = A_I\}$$

$$g := \chi_{\mathcal{A}}$$

ADMM – Mini-Introduction

Getting ready to apply ADMM:

Our Objective Function

Linear Constraints

Copies of A are identical

$$\underset{x_1,\ldots,x_I,A}{\operatorname{argmin}} \sum_{i=1}^{I} f_i(x_i)$$

$$\text{subject to} \quad \forall i \in \{1,\ldots,I\} : x_i = C_i A$$

equivalent to...

$$\underset{X,\overline{A}}{\operatorname{argmin}} \; f(X) + g(\overline{A})$$

$$\text{subject to} \quad X - C\overline{A} = 0$$

$$X := (x_1,\ldots,x_I)$$

$$f(X) := \sum_{i=1}^{I} f_i(x_i)$$

$$\mathcal{A} := \{\overline{A} := (A_1,\ldots,A_I) | A_1 = \ldots = A_I\}$$

$$g := \chi_{\mathcal{A}}$$

ADMM – Mini-Introduction

Getting ready to apply ADMM:

$$\underset{x_1,\ldots,x_I,A}{\text{argmin}} \quad \sum_{i=1}^{I} f_i(x_i)$$

$$\text{subject to} \quad \forall i \in \{1,\ldots,I\} : x_i = C_i A$$

Our Objective Function

Linear Constraints

Copies of A are identical

equivalent to...

$$\underset{X,\overline{A}}{\text{argmin}} \quad f(X) + g(\overline{A})$$

$$\text{subject to} \quad X - C\overline{A} = 0$$

$$X := (x_1,\ldots,x_I)$$

$$f(X) := \sum_{i=1}^{I} f_i(x_i)$$

$$\mathcal{A} := \{\overline{A} := (A_1,\ldots,A_I) | A_1 = \ldots = A_I\}$$

$$g := \chi_{\mathcal{A}}$$

How is that useful?

ADMM – Mini-Introduction

Applying ADMM:

ADMM – Mini-Introduction

Applying ADMM:

We need to minimize the Lagrangian w.r.t. X:

$$L_\rho(X, \overline{A}, \beta) := \sum_{i=1}^{I} f_i(x_i) + \langle \beta_i, x_i - C_i A_i \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i \right\|_2^2 + \chi_{\mathcal{A}}(A_1, \ldots, A_I)$$

ADMM – Mini-Introduction

Applying ADMM:

We need to minimize the Lagrangian w.r.t. X:

$$L_\rho(X, \overline{A}, \beta) := \sum_{i=1}^{I} f_i(x_i) + \langle \beta_i, x_i - C_i A_i \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i \right\|_2^2 + \chi_{\mathcal{A}}(A_1, \ldots, A_I)$$

Each x_i only occurs in three terms.

ADMM – Mini-Introduction

Applying ADMM:

We need to minimize the Lagrangian w.r.t. X:

$$L_\rho(X, \overline{A}, \beta) := \sum_{i=1}^{I} f_i(x_i) + \langle \beta_i, x_i - C_i A_i \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i \right\|_2^2 + \chi_{\mathcal{A}}(A_1, \ldots, A_I)$$

Each x_i only occurs in three terms.
We can optimize EACH term in the objective individually!

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

ADMM – Mini-Introduction

Applying ADMM:

We need to minimize the Lagrangian w.r.t. X:

$$L_\rho(X, \overline{A}, \beta) := \sum_{i=1}^{I} f_i(x_i) + \langle \beta_i, x_i - C_i A_i \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i \right\|_2^2 + \chi_{\mathcal{A}}(A_1, \ldots, A_I)$$

Each x_i only occurs in three terms.
We can optimize EACH term in the objective individually!

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

This is much easier!

ADMM – Mini-Introduction

Applying ADMM:

We need to minimize the Lagrangian w.r.t. X:

$$L_\rho(X, \overline{A}, \beta) := \sum_{i=1}^{I} f_i(x_i) + \langle \beta_i, x_i - C_i A_i \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i \right\|_2^2 + \chi_{\mathcal{A}}(A_1, \ldots, A_I)$$

Each x_i only occurs in three terms.
We can optimize EACH term in the objective individually!

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

This is much easier!

Map-Step in Map-Reduce!

ADMM – Mini-Introduction

We will stop here.

ADMM – Mini-Introduction

We will stop here. However, next steps would be:

ADMM – Mini-Introduction

We will stop here. However, next steps would be:

1.) Solve for X (separately for each x_i):

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

ADMM – Mini-Introduction

We will stop here. However, next steps would be:

1.) Solve for X (separately for each x_i):

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

2.) Solve Augmented Lagrangian for the second variable $\overline{A}$.

ADMM – Mini-Introduction

We will stop here. However, next steps would be:

1.) Solve for X (separately for each x_i):

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

2.) Solve Augmented Lagrangian for the second variable $\overline{A}$.

3.) Perform gradient ascent on beta.

ADMM – Mini-Introduction

We will stop here. However, next steps would be:

1.) Solve for X (separately for each x_i):

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \left\| x_i - C_i A_i^k \right\|_2^2$$

2.) Solve Augmented Lagrangian for the second variable $\overline{A}$.

3.) Perform gradient ascent on beta.

=> Iteration of these steps converges to solution (Bertsekas 1990)

ADMM – Mini-Introduction

We will stop here. However, next steps would be:

1.) Solve for X (separately for each x_i):

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}} f_i(x_i) + \langle \beta_i^k, x_i - C_i A_i^k \rangle + \frac{\rho}{2} \| x_i - C_i A_i^k \|_2^2$$

2.) Solve Augmented Lagrangian for the second variable $\overline{A}$.

3.) Perform gradient ascent on beta.

=> Iteration of these steps converges to solution (Bertsekas 1990)

Thanks!