Inference and learning for sensory data
-
New priors, non-linear features, and the challenge of masking and occlusion

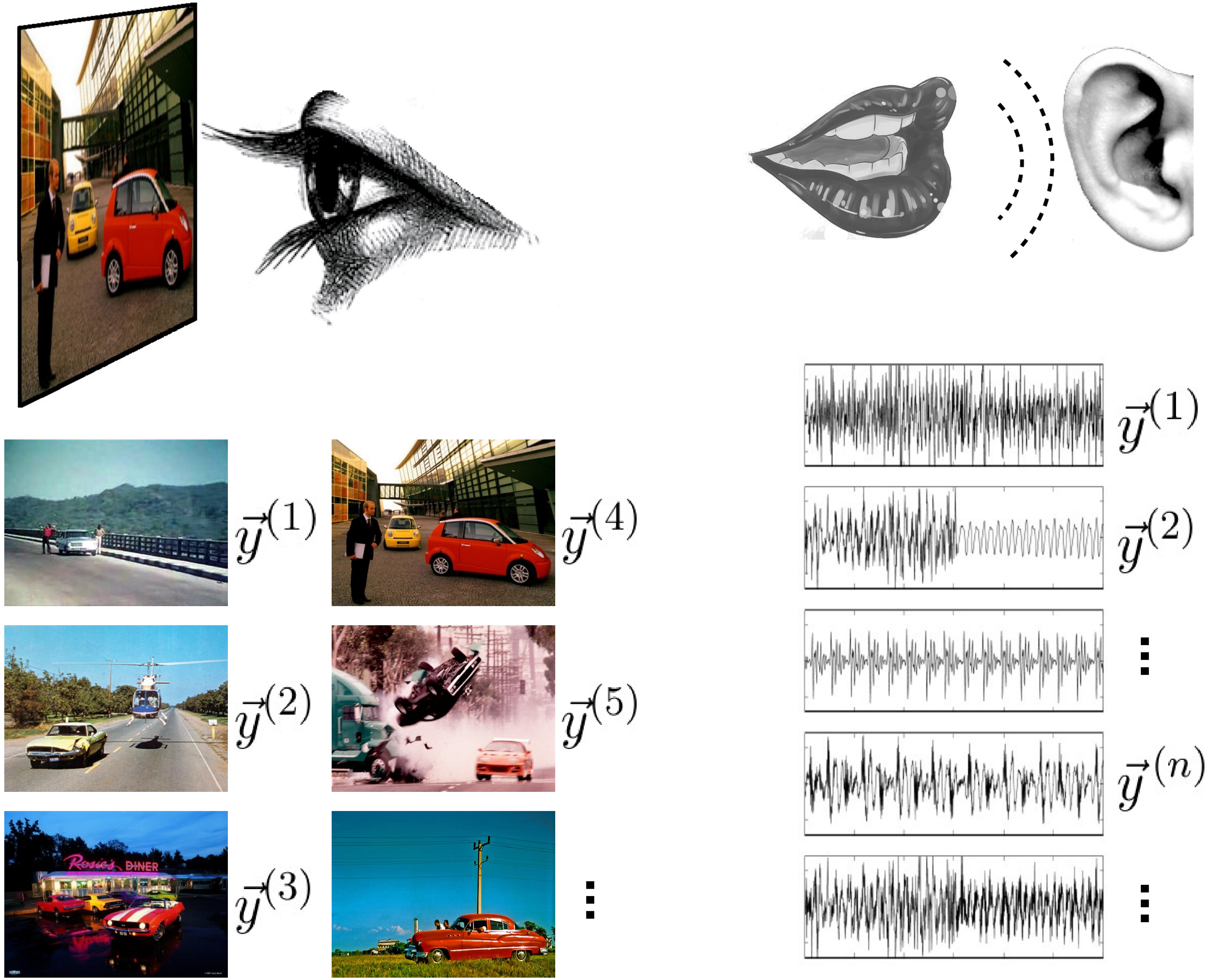
Jörg Lücke

Machine Learning

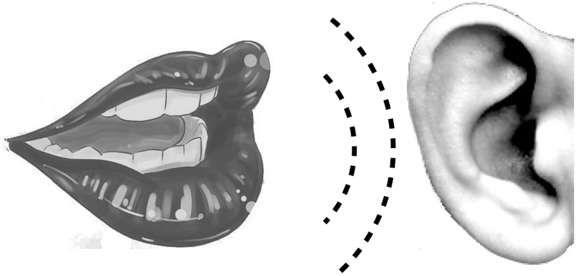
Cluster of Excellence Hearing4all, Dept for Medical Physics and Acoustics,
Carl-von-Ossietzky University Oldenburg, Germany

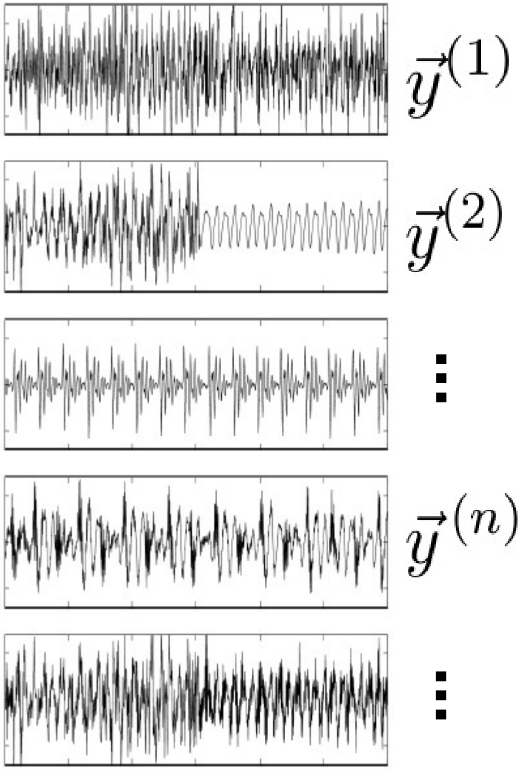
Oldenburg, March 6, 2015

Sensory Inference



Sensory Inference

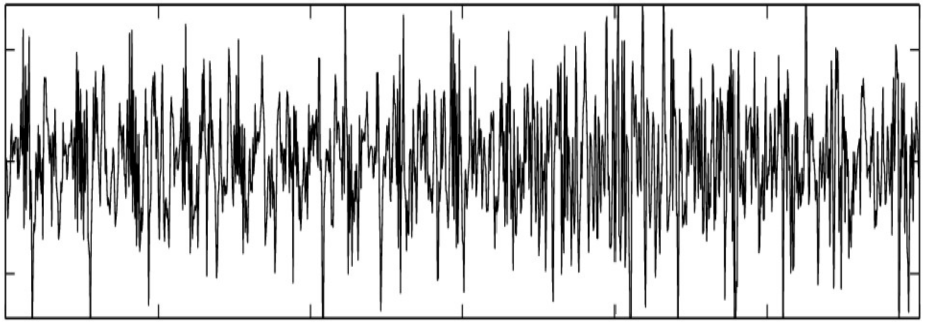


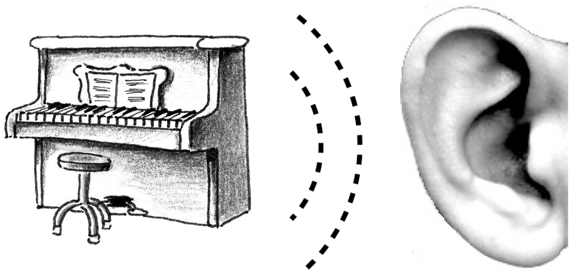


unconscious inference

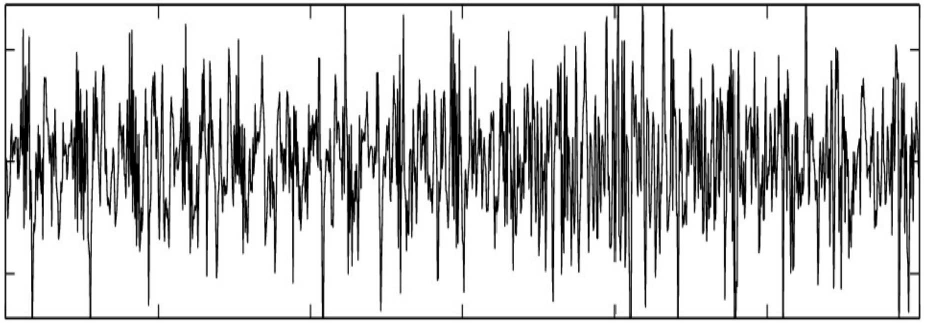
Helmholtz, *HB physiol. Optik*, 1867

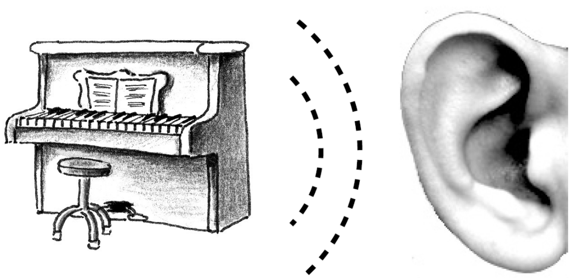
Sensory Inference: Example





Sensory Inference: Example



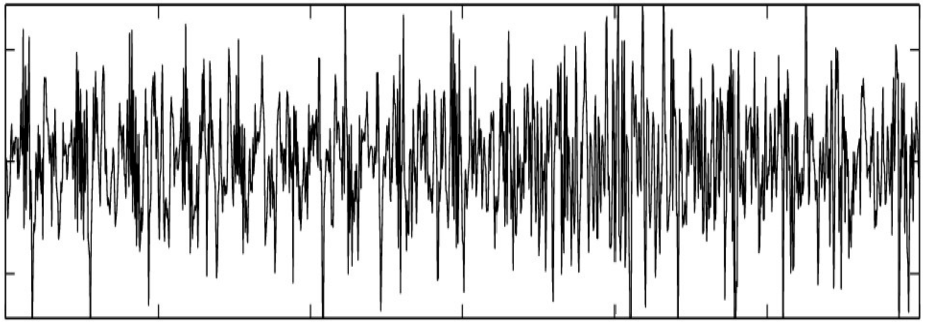


Inference Task:

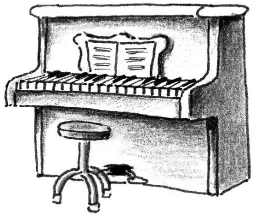
Which piano keys were pressed?

Build an artificial system that solves the task.

Sensory Inference: Example



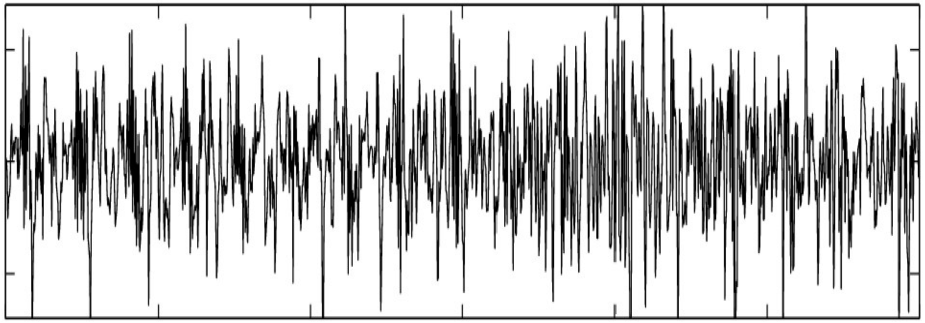
$(\vec{y}^{(n)})^T$



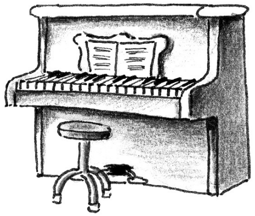
We re-express data point $\vec{y}^{(n)}$:

$$\vec{y}^{(n)} \approx \sum_{h=1}^{H} s_h^{(n)} \vec{W}_h$$

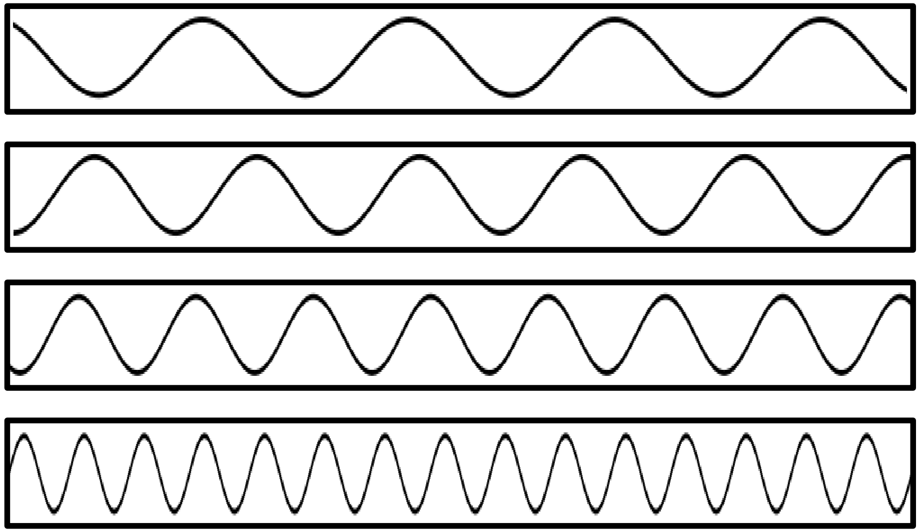
Sensory Inference: Example



$$(\vec{y}^{\,(n)})^T$$



For piano data, we would choose:



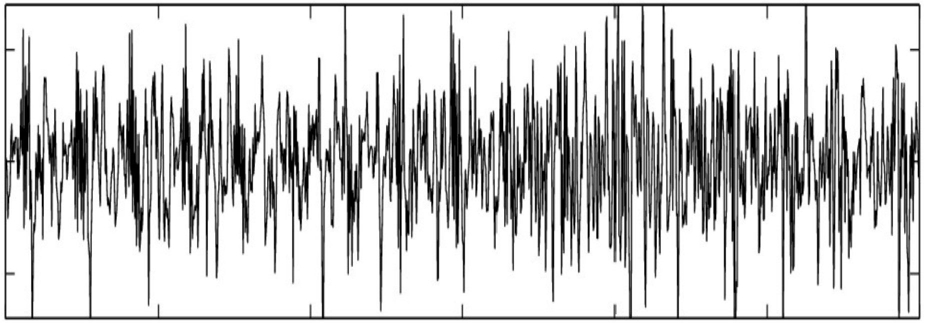
$$\vec{W}_1^T \qquad \vec{W}_2^T \qquad \vec{W}_3^T \qquad \vec{W}_4^T$$

etc.

We re-express data point $\vec{y}^{\,(n)}$:

$$\vec{y}^{\,(n)} \approx \sum_{h=1}^{H} s_h^{(n)}\, \vec{W}_h$$

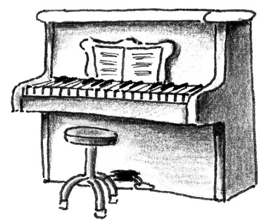
Sensory Inference: Example



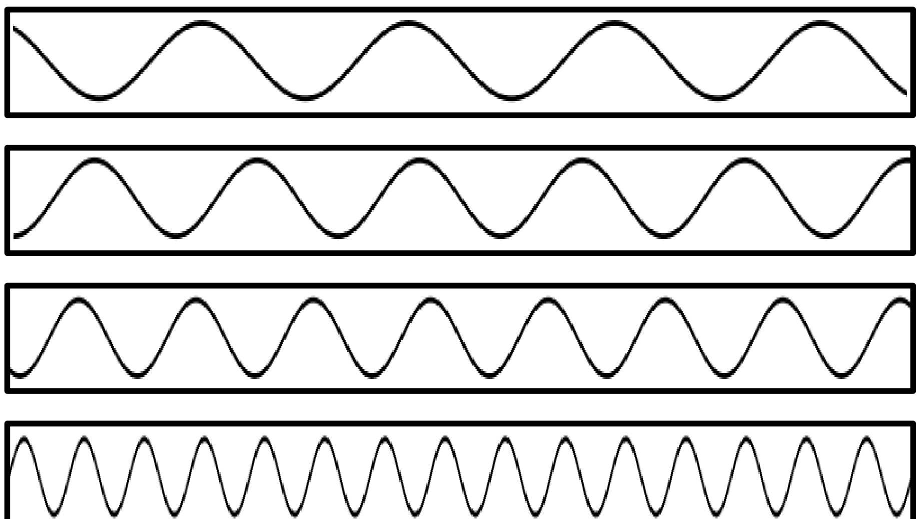
$$(\vec{y}^{\,(n)})^T$$

We re-express data point $\vec{y}^{\,(n)}$:

$$\vec{y}^{\,(n)} \approx \sum_{h=1}^{H} s_h^{(n)} \vec{W}_h$$



For piano data, we would choose:



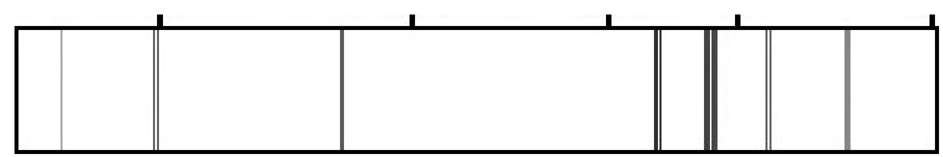
$\vec{W}_1^T$

$\vec{W}_2^T$

$\vec{W}_3^T$

$\vec{W}_4^T$

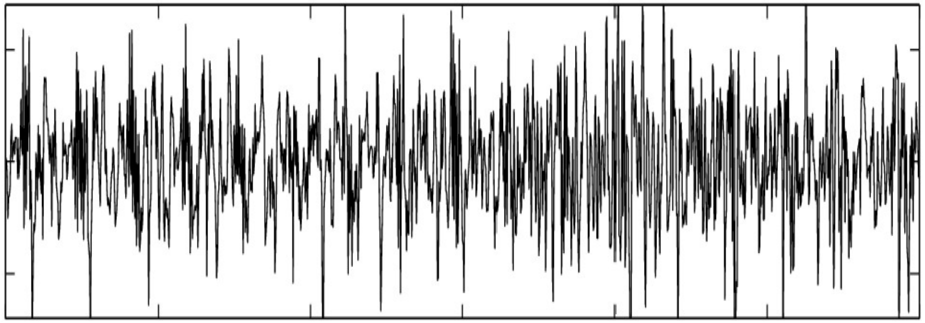
etc.



$$\vec{s}^{\,(n)}$$

$$p(\vec{s}\,|\,\vec{y}^{\,(n)})$$

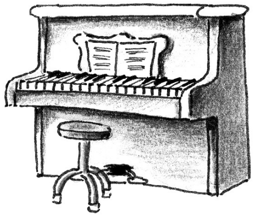
Sensory Inference: Example



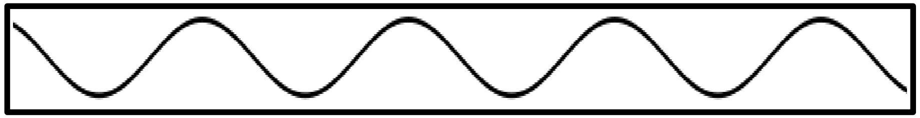
$$(\vec{y}^{(n)})^T$$

We re-express data point $\vec{y}^{(n)}$:

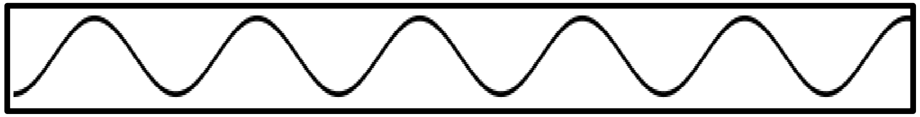
$$\vec{y}^{(n)} \approx \sum_{h=1}^{H} s_h^{(n)} \vec{W}_h$$



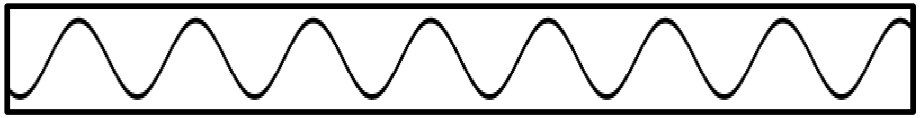
For piano data, we would choose:



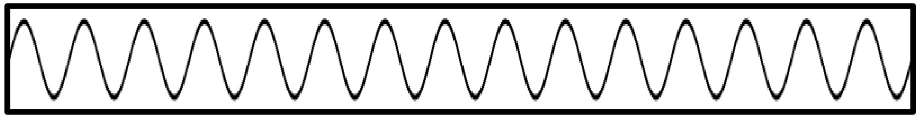
$\vec{W}_1^T$



$\vec{W}_2^T$



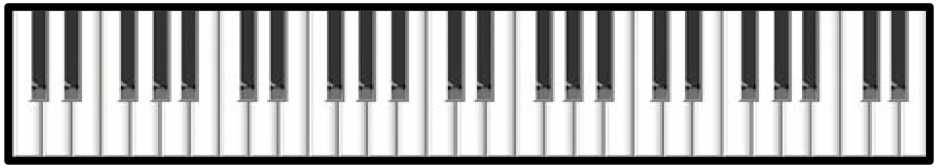
$\vec{W}_3^T$



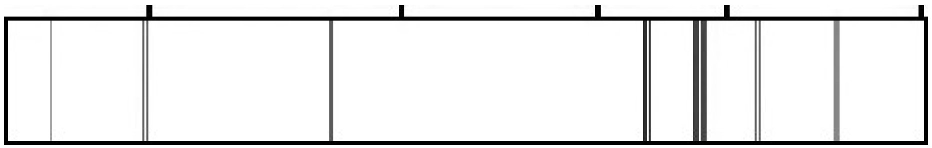
$\vec{W}_4^T$

etc.

Real data causes/components:



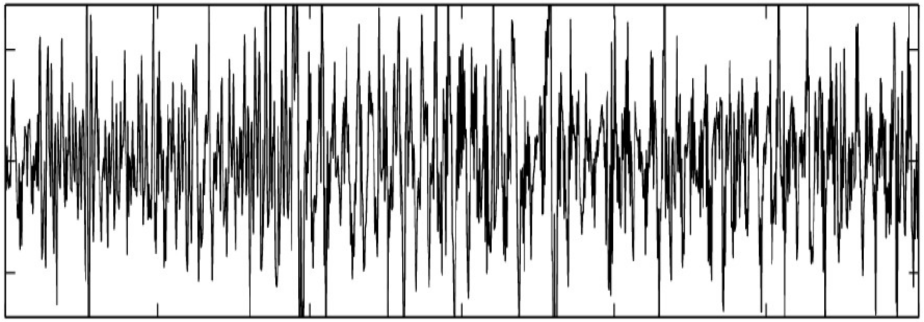
Estimates of causes/components:



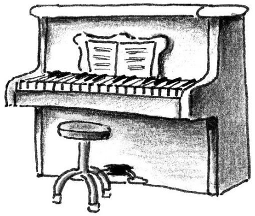
$$\vec{s}^{(n)}$$

$$p(\vec{s} \,|\, \vec{y}^{(n)})$$

Sensory Inference: Example



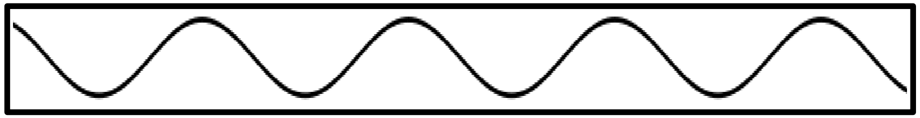
$$(\vec{y}^{\,(n)})^T$$



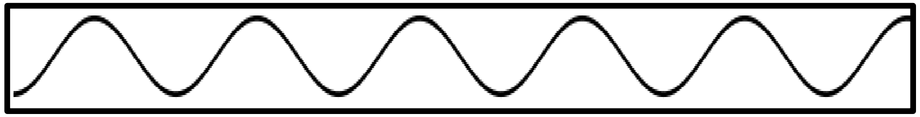
We re-express data point $\vec{y}^{\,(n)}$:

$$\vec{y}^{\,(n)} \approx \sum_{h=1}^{H} s_h^{(n)} \vec{W}_h$$

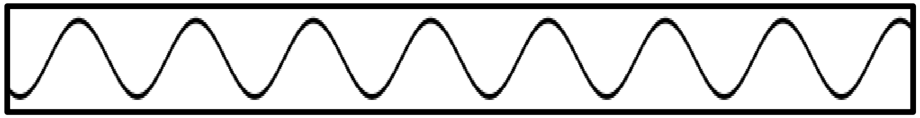
For piano data, we would choose:



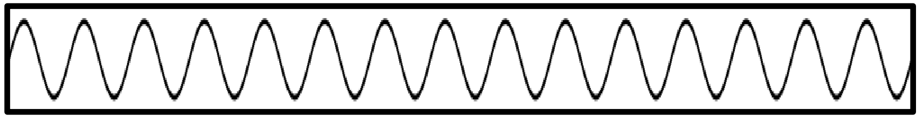
$\vec{W}_1^T$



$\vec{W}_2^T$



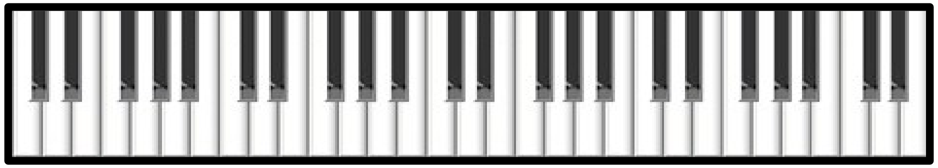
$\vec{W}_3^T$



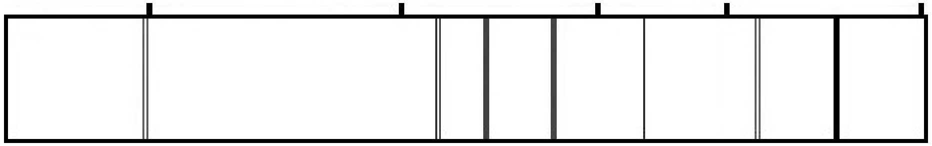
$\vec{W}_4^T$

etc.

Real data causes/components:



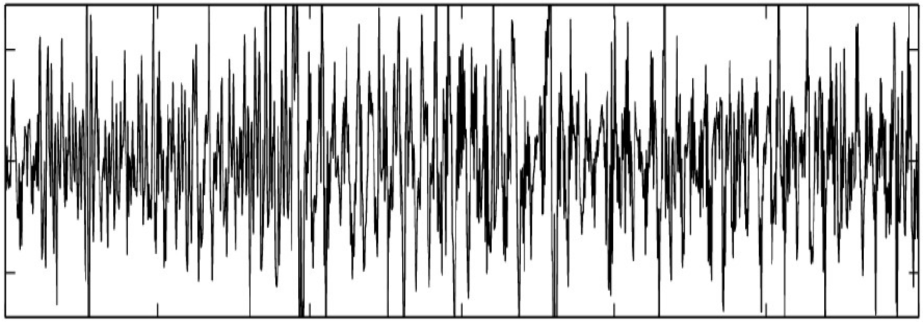
Estimates of causes/components:



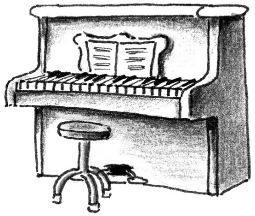
$$\vec{s}^{\,(n)}$$

$$p(\vec{s}\,|\,\vec{y}^{\,(n)})$$

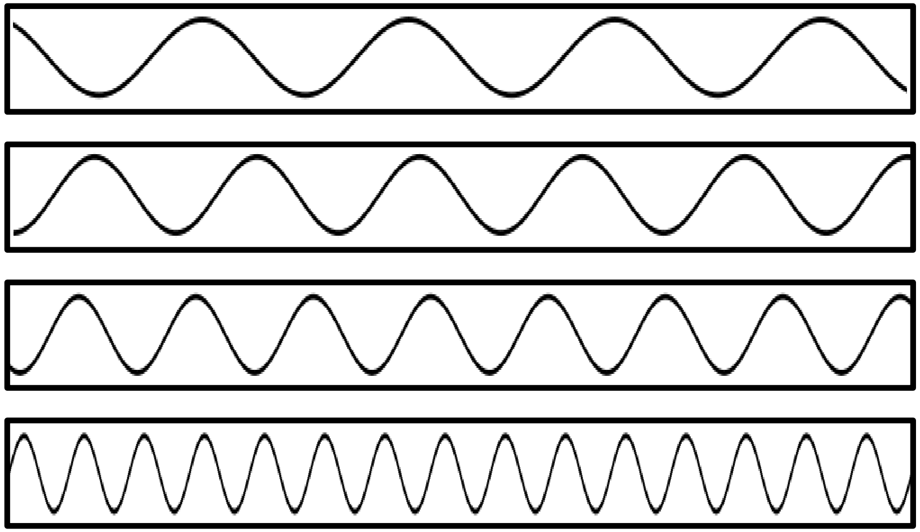
Sensory Inference: Example



$$(\vec{y}^{\,(n)})^T$$



For piano data, we would choose:



$\vec{W}_1^T$

$\vec{W}_2^T$

$\vec{W}_3^T$

$\vec{W}_4^T$

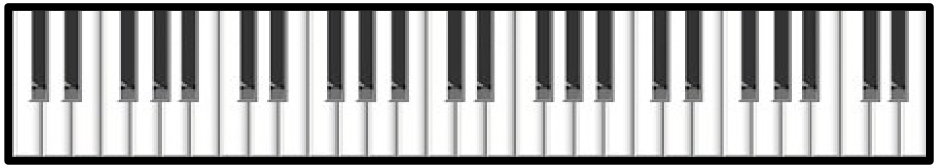
etc.

We re-express data point $\vec{y}^{\,(n)}$:

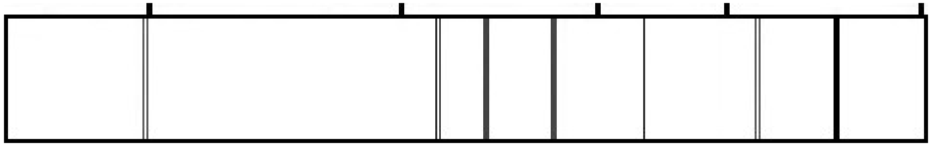
$$\vec{y}^{\,(n)} \approx \sum_{h=1}^{H} s_h^{(n)} \vec{W}_h$$

$$\vec{y}^{\,(n)} = \sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$$

Real data causes/components:



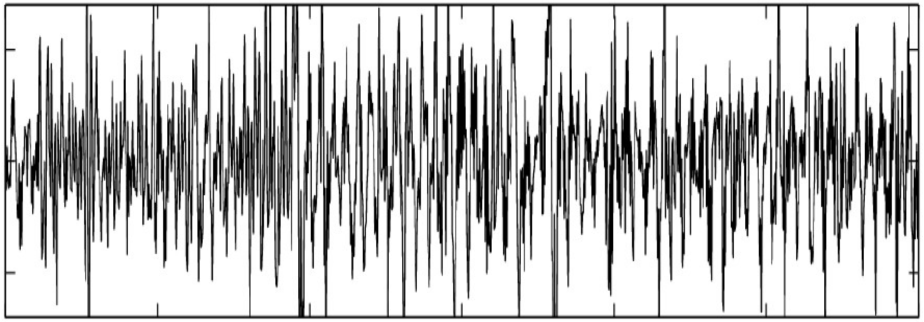
Estimates of causes/components:



$$\vec{s}^{\,(n)}$$

$$p(\vec{s} \,|\, \vec{y}^{\,(n)})$$

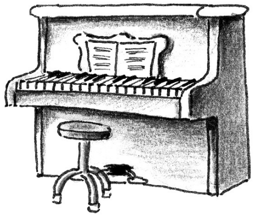
Sensory Inference: Example



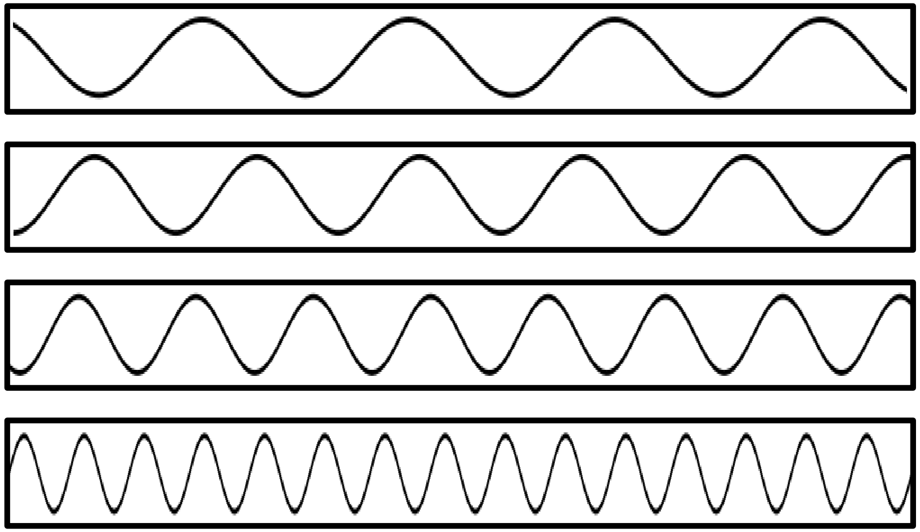
$$(\vec{y}^{\,(n)})^T$$

We re-express data point $\vec{y}^{\,(n)}$:

$$\vec{y}^{\,(n)} = \sum_h s_h^{(n)}\,\vec{W}_h + \vec{\eta}$$



For piano data, we would choose:



$\vec{W}_1^T$

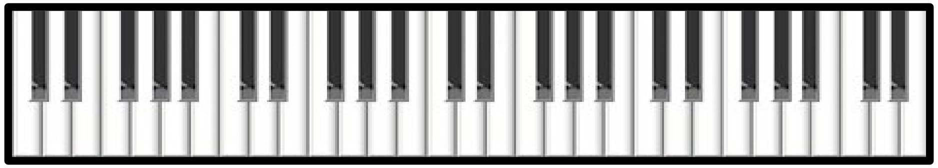
$\vec{W}_2^T$

$\vec{W}_3^T$

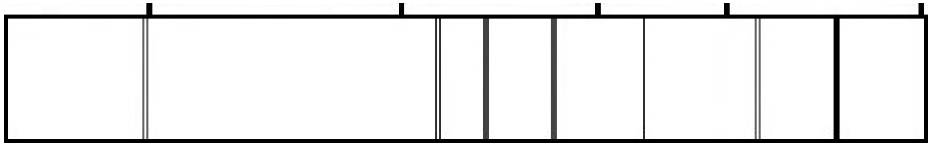
$\vec{W}_4^T$

etc.

Real data causes/components:



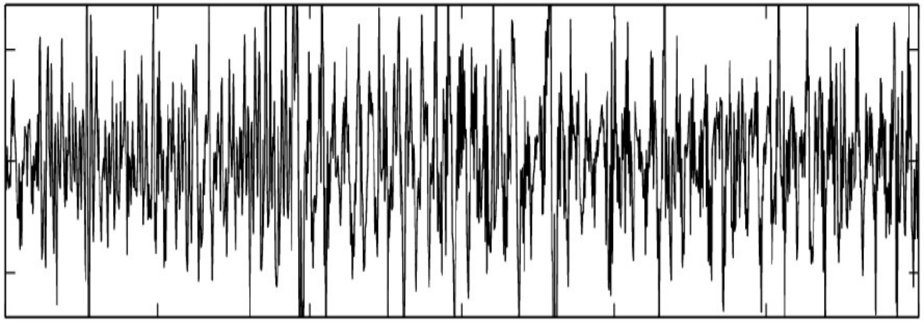
Estimates of causes/components:



$$\vec{s}^{\,(n)}$$

$$p(\vec{s}\,|\,\vec{y}^{\,(n)})$$

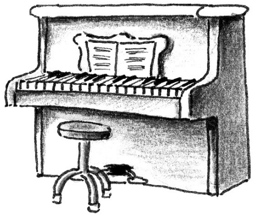
Sensory Inference: Example



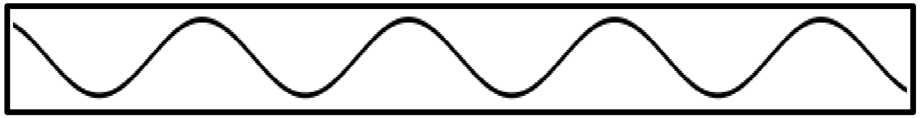
$$(\vec{y}^{(n)})^T$$

We re-express data point $\vec{y}^{(n)}$:

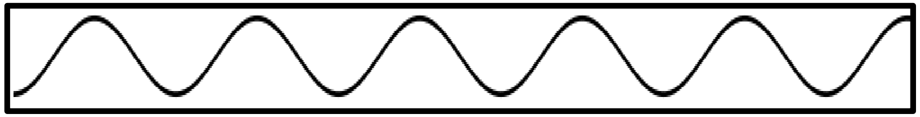
$$\vec{y}^{(n)} = \sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$$



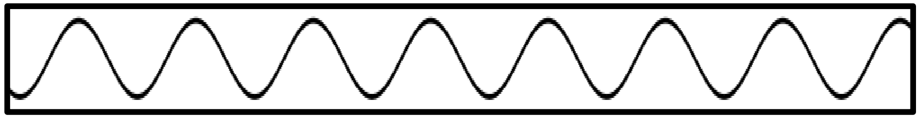
For piano data, we would choose:



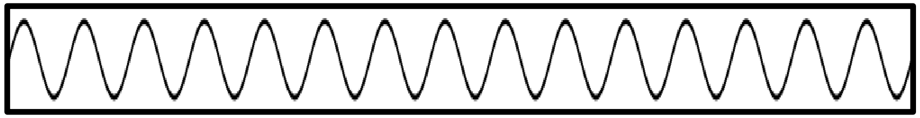
$\vec{W}_1^T$



$\vec{W}_2^T$



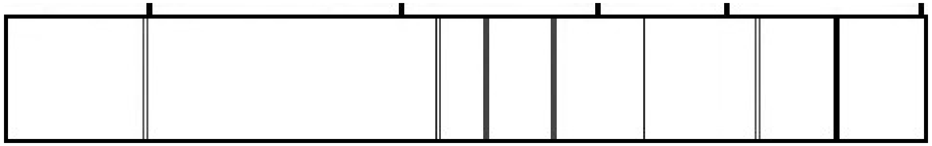
$\vec{W}_3^T$



$\vec{W}_4^T$

etc.

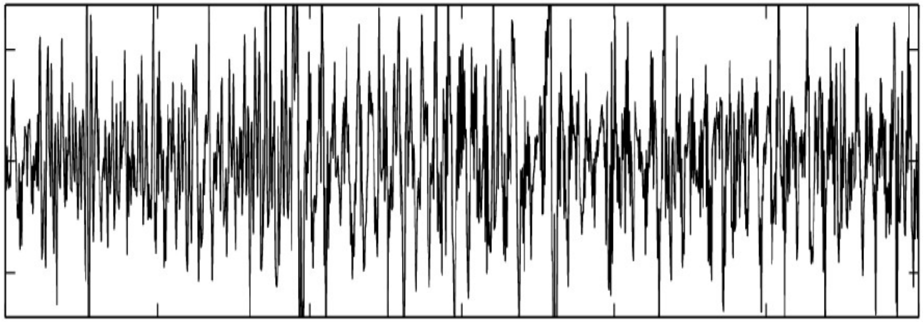
Estimates of causes/components:



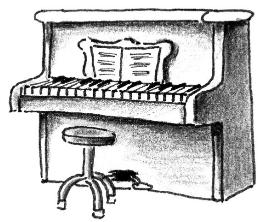
$$\vec{s}^{(n)}$$

$$p(\vec{s} \,|\, \vec{y}^{(n)})$$

Sensory Inference: Example



$$(\vec{y}^{\,(n)})^T$$



We re-express data point $\vec{y}^{\,(n)}$:

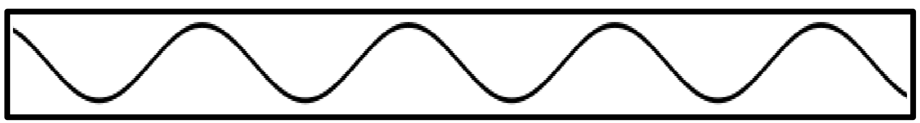
$$\vec{y}^{\,(n)} = \sum_h s_h^{(n)}\, \vec{W}_h + \vec{\eta}$$

Probabilistic **generative model** (e.g., SC):

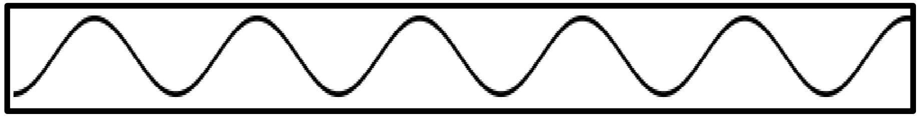
$$p(\vec{s}\,|\,\Theta) = \prod_h \frac{1}{\pi\,(1 + s_h^2)}$$

$$p(\vec{y}\,|\,\vec{s}, \Theta) = \mathcal{N}(\vec{y}; \sum_h s_h^{(n)}\, \vec{W}_h, \sigma^2\, \mathbb{1})$$

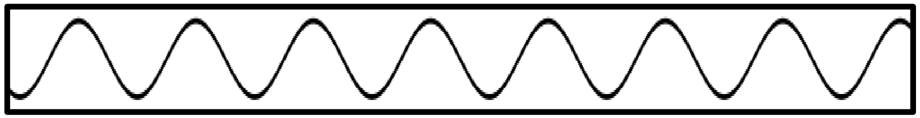
For piano data, we would choose:



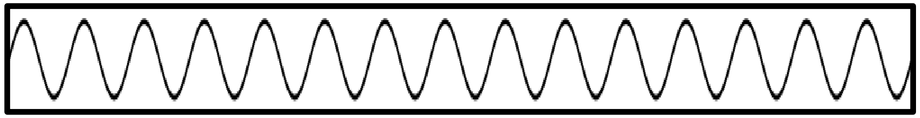
$\vec{W}_1^T$



$\vec{W}_2^T$



$\vec{W}_3^T$

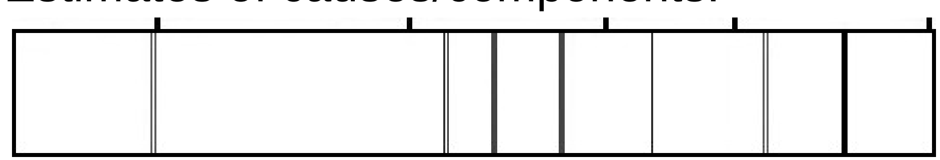


$\vec{W}_4^T$

⋮

dictionary

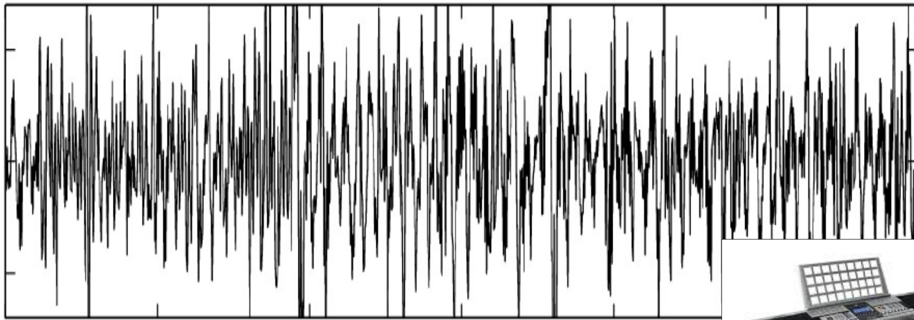
Estimates of causes/components:



$$\vec{s}^{\,(n)}$$

$$p(\vec{s}\,|\,\vec{y}^{\,(n)})$$

Sensory Inference: Example



$$(\vec{y}^{\,(n)})^T$$

We re-express data point $\vec{y}^{\,(n)}$:

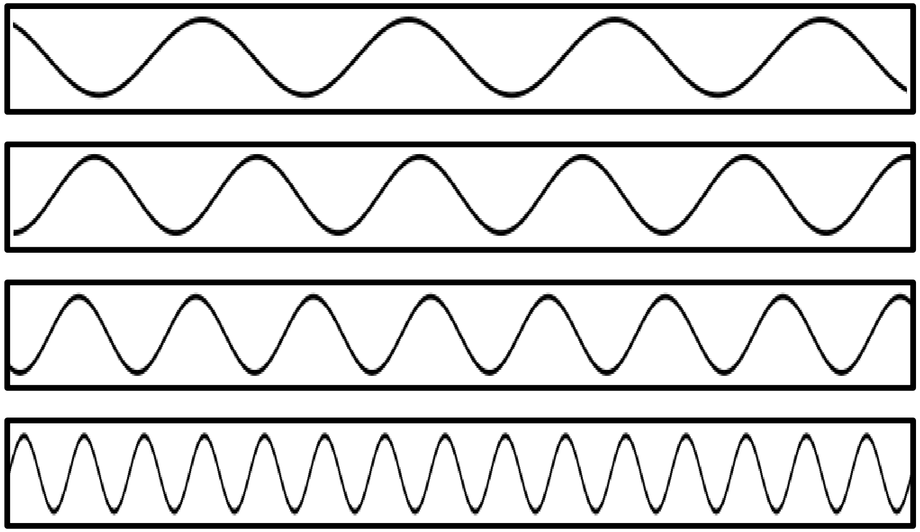
$$\vec{y}^{\,(n)} = \sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$$

Probabilistic **generative model** (e.g., SC):

$$p(\vec{s}\,|\,\Theta) = \prod_h \frac{1}{\pi\,(1+s_h^2)}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y}; \sum_h s_h^{(n)} \vec{W}_h, \sigma^2 \mathbb{1})$$

For piano data, we would choose:



$\vec{W}_1^T$

$\vec{W}_2^T$

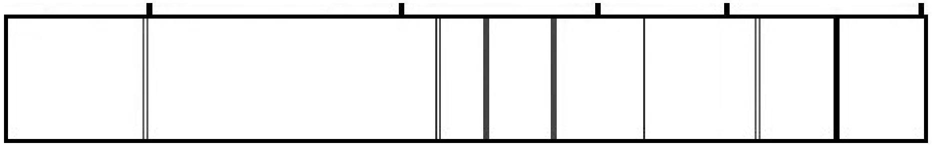
$\vec{W}_3^T$

$\vec{W}_4^T$

⋮

dictionary

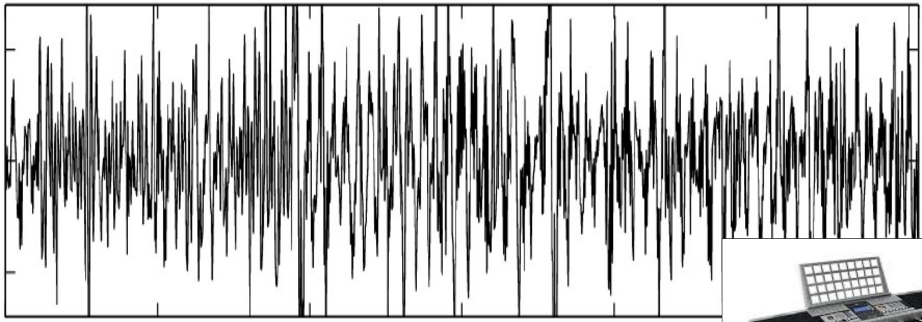
Estimates of causes/components:



$$\vec{s}^{\,(n)}$$

$$p(\vec{s}\,|\,\vec{y}^{\,(n)})$$

Sensory Inference: Example



$(\vec{y}^{\,(n)})^T$

We re-express data point $\vec{y}^{\,(n)}$:

$$\vec{y}^{\,(n)} = \sum_h s_h^{(n)}\, \vec{W}_h + \vec{\eta}$$

Probabilistic **generative model** (e.g., SC):

$$p(\vec{s}\,|\,\Theta) = \prod_h \frac{1}{\pi\,(1+s_h^2)}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y}; \sum_h s_h^{(n)}\, \vec{W}_h, \sigma^2\, \mathbb{1})$$

For a keyboard instrument:

?

$\vec{W}_1^T$

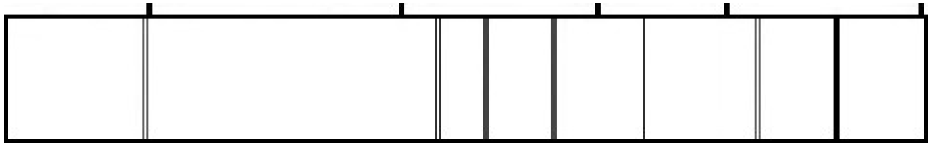
$\vec{W}_2^T$

$\vec{W}_3^T$

$\vec{W}_4^T$

⋮

Estimates of causes/components:

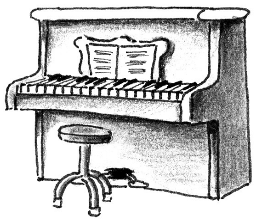
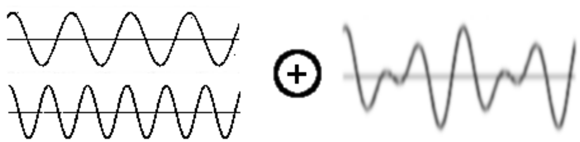
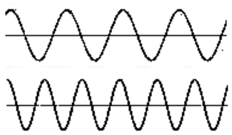
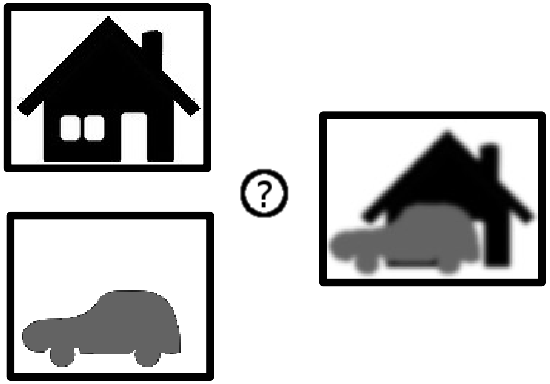
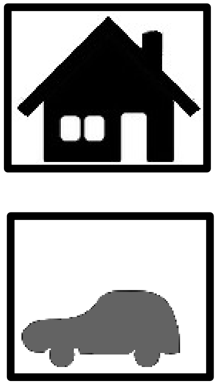


$\vec{s}^{\,(n)}$

$p(\vec{s}\,|\,\vec{y}^{\,(n)})$

Solution: learn dictionary from data

Non-linear components

generating system	combination of causes	model assumptions for $\vec{y}$	dictionary W
		$\sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$	
 street scene		$f(s_h^{(n)}, W) + \vec{\eta}$	

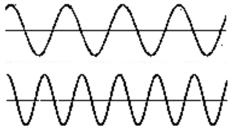
Non-linear components

Change model parameters W until:

real data $\approx$ model data

Measure for this similarity:

Data Likelihood

model assumptions for $\vec{y}$	dictionary W
$\sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$	
$f(s_h^{(n)}, W) + \vec{\eta}$	

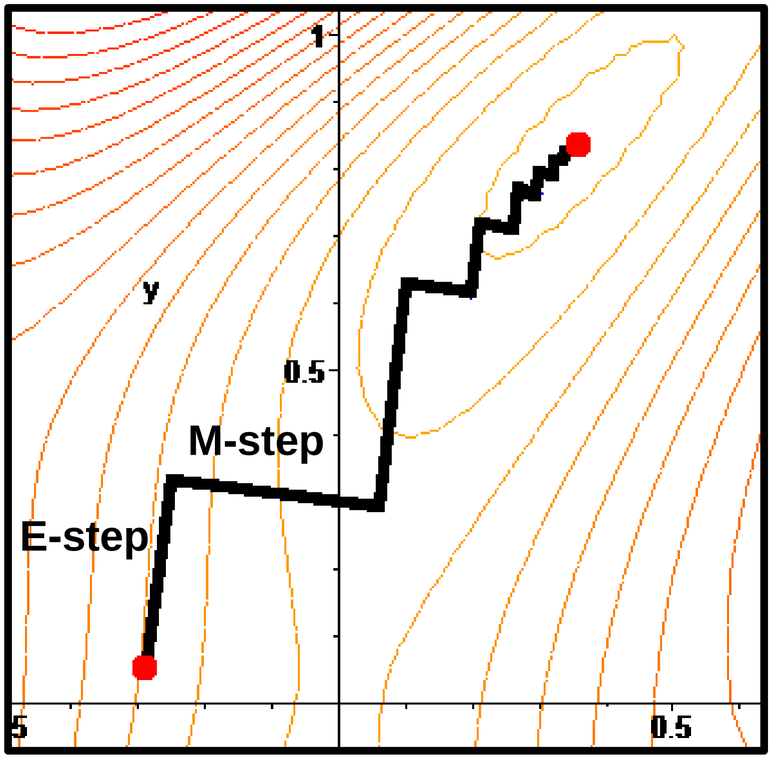
Non-linear components

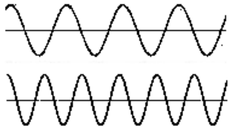
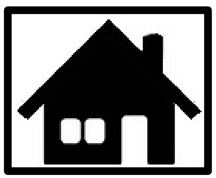
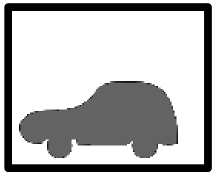
Change model parameters W until:

real data $\approx$ model data

Measure for this similarity:

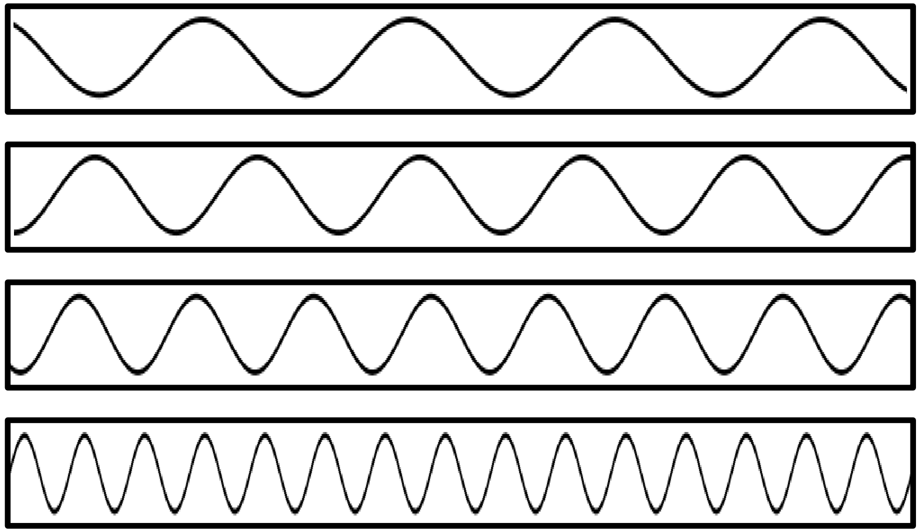
Data Likelihood



model assumptions for $\vec{y}$	dictionary W
$\sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$	
$f(s_h^{(n)}, W) + \vec{\eta}$ Dayan & Zemel, 1996; Lücke & Sahani, 2007, 2008; Lücke, 2009; Lücke et al. etc.	 

E.g., **E**xpectation **M**aximization (EM) framework.
Dempster, 1977; Neal & Hinton, 1998.

Dictionary Examples

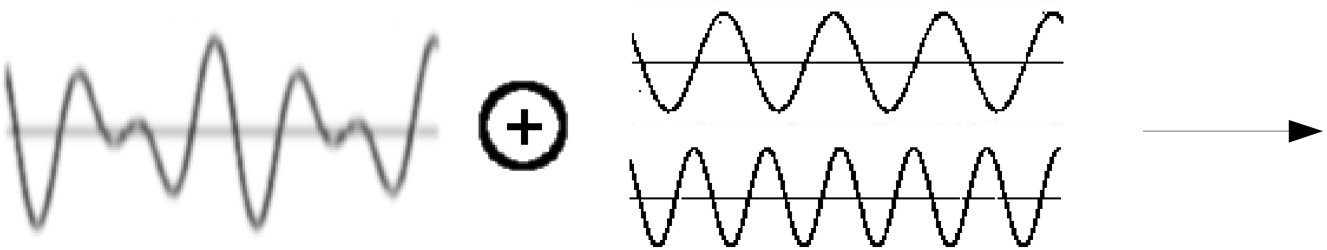


$\vec{W}_1^T$

$\vec{W}_2^T$

$\vec{W}_3^T$

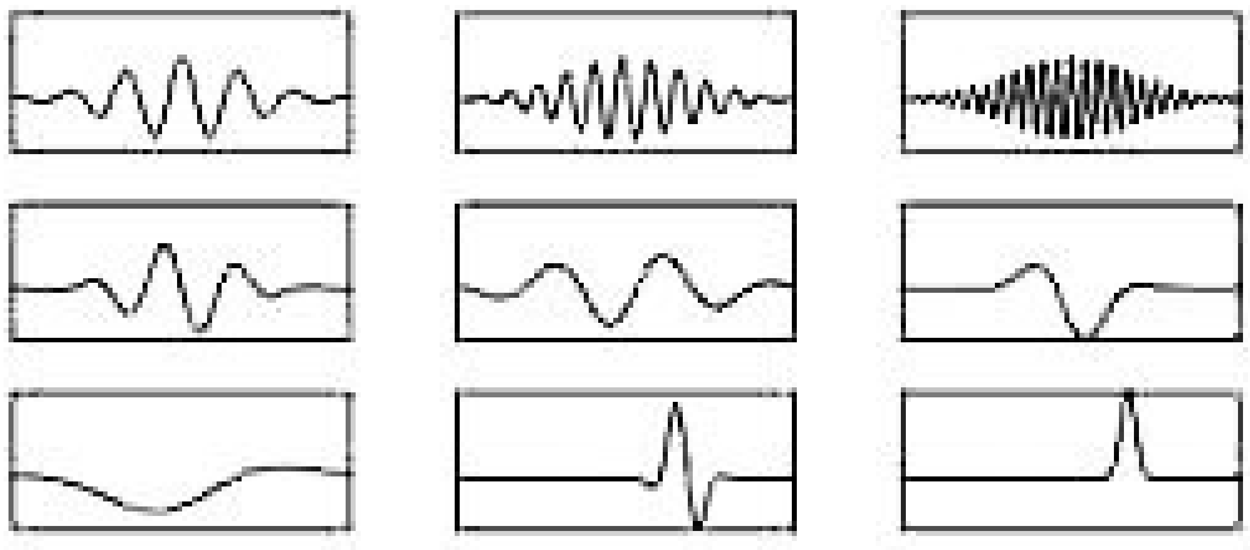
$\vec{W}_4^T$



$$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \vec{s}$$

$$(\vec{y}^{\,(n)})^T \qquad \vec{y}^{\,(n)} = \sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$$

Dictionary Examples

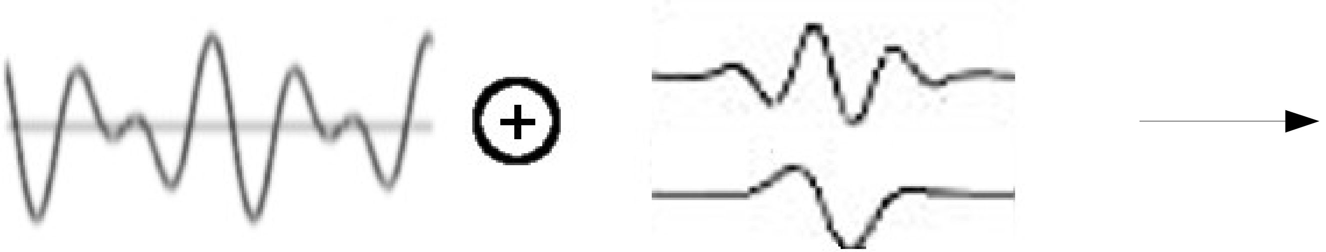


$\vec{W}_1^T$

$\vec{W}_2^T$

$\vec{W}_3^T$

$\vec{W}_4^T$



$$\vec{s} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

$(\vec{y}^{(n)})^T \qquad \vec{y}^{(n)} = \sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$

Dictionary Examples

$$\vec{y}^{(n)} = \sum_h s_h^{(n)} \vec{W}_h + \vec{\eta}$$

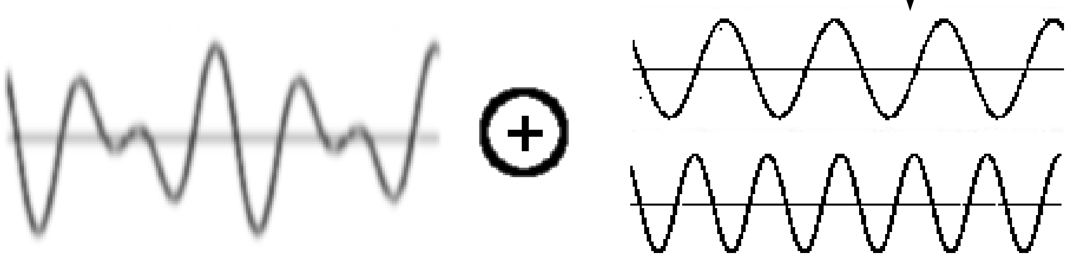
Olshausen & Field '96
… Sheikh et al., '14 ...

Gaussian prior

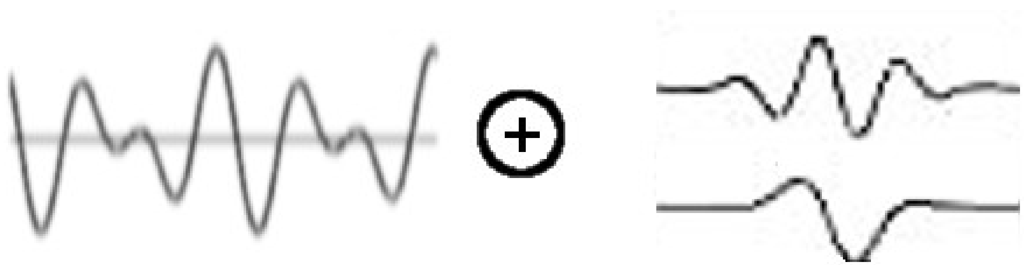
Sparse prior

PCA / Factor Analysis
(vgl. principal axis transform)

Sparse coding / (ICA)

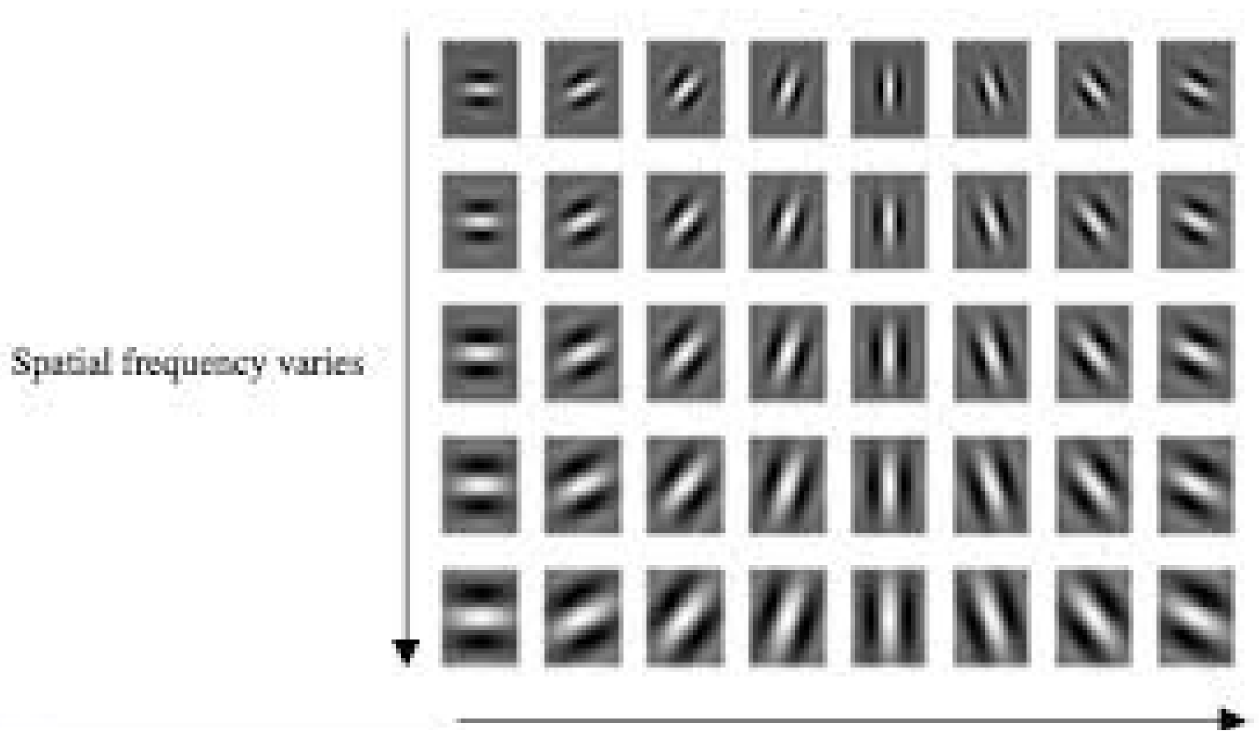


$$\vec{s} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$



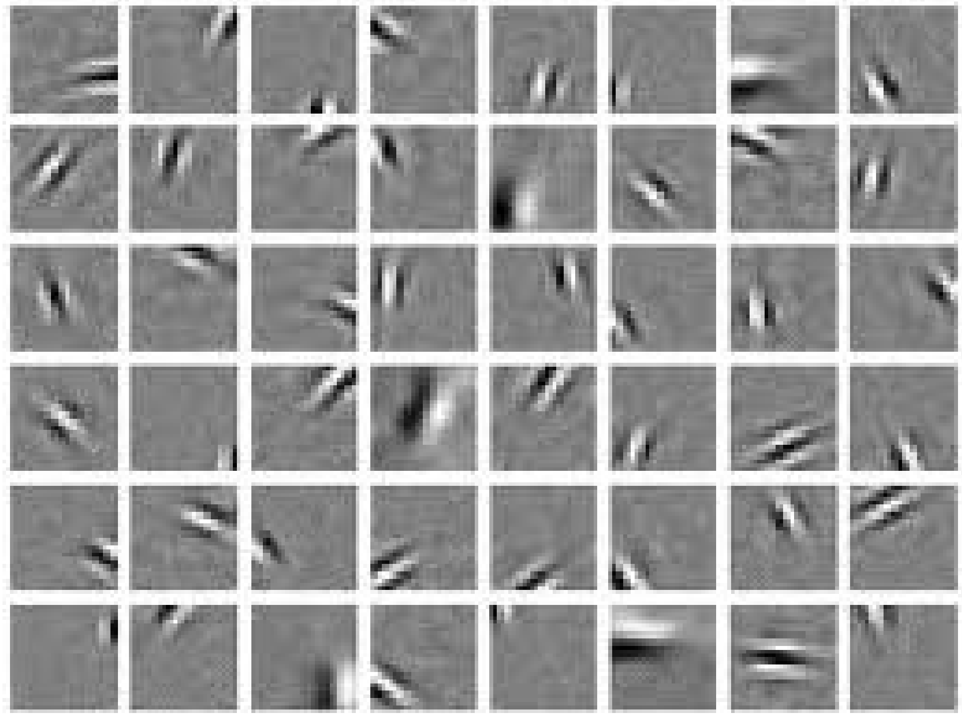
$$\vec{s} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

Dictionary Examples



$$\vec{W}_1^T$$
$$\vec{W}_2^T$$
$$\vec{W}_3^T$$
$$\vec{W}_4^T$$

Dictionary Examples

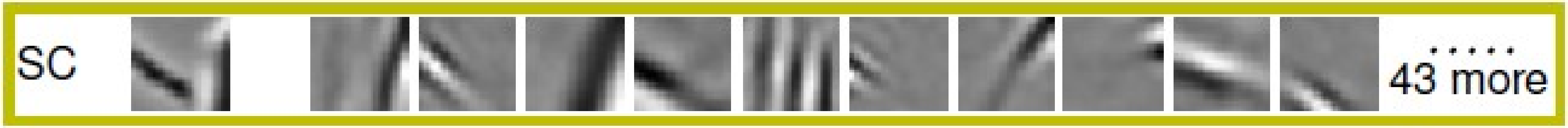


Olshausen & Field, *Nature* 1996

Lee, Battle, Raina, Ng, *NIPS* 2006;
Bornschein, Henniges, Lücke, *PLOS Comp Biology* 2013

Dictionary Examples

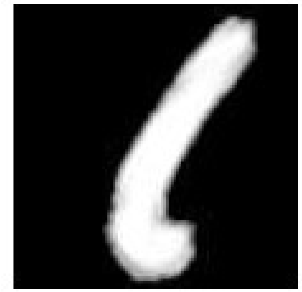
$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$



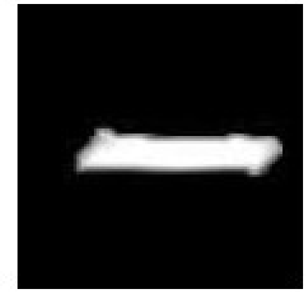
$$\vec{W}_1^T \; \vec{W}_2^T \; \dots$$

Dictionary Examples

$\vec{W}_1^T$



$\vec{W}_2^T$



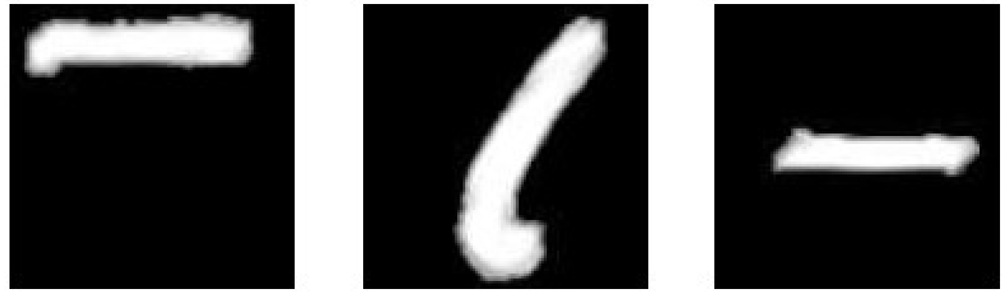
$\vec{W}_3^T$

linearity assumption not realistic

$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

$\vec{W}_1^T \; \vec{W}_2^T$ ...

Dictionary Examples

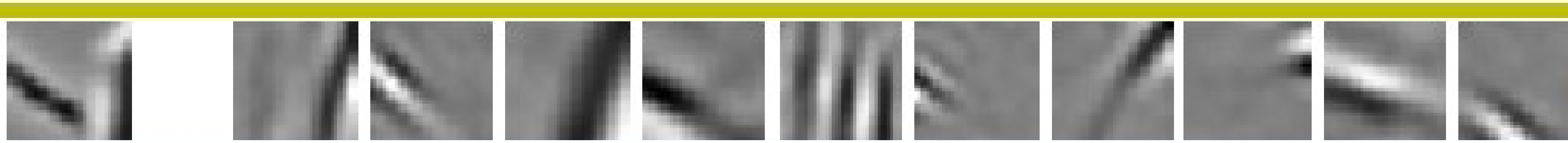


$\vec{W}_1^T$ $\vec{W}_2^T$ $\vec{W}_3^T$

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

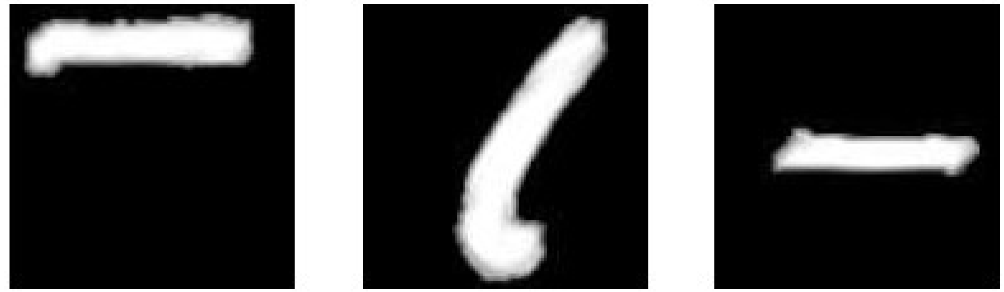
SC



...... 43 more

$\vec{W}_1^T$ $\vec{W}_2^T$ ...

Dictionary Examples



$$\vec{W}_1^T \quad \vec{W}_2^T \quad \vec{W}_3^T$$

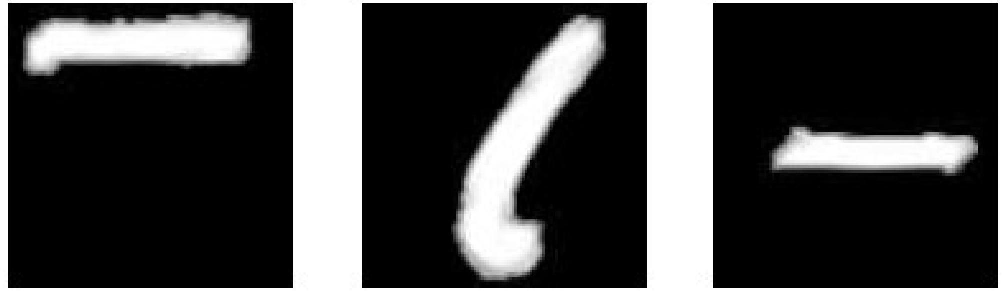
$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

PCA – standard approach, Pearson, 1901
FA – standard approach, e.g., Gorsuch, '83
ICA – has own conference, Comon, 1994
SC – Olshausen & Field, *Nature*, 1996
NMF – Lee & Seung, *Nature*, 1999
etc.

$$\vec{W}_1^T \, \vec{W}_2^T \; \ldots$$

Dictionary Examples



$\vec{W}_1^T \quad \vec{W}_2^T \quad \vec{W}_3^T$

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

crucial for
compressed sensing

$\vec{W}_1^T \, \vec{W}_2^T$ ...

Dictionary Examples

$\vec{W}_1^T$ $\vec{W}_2^T$ $\vec{W}_3^T$

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

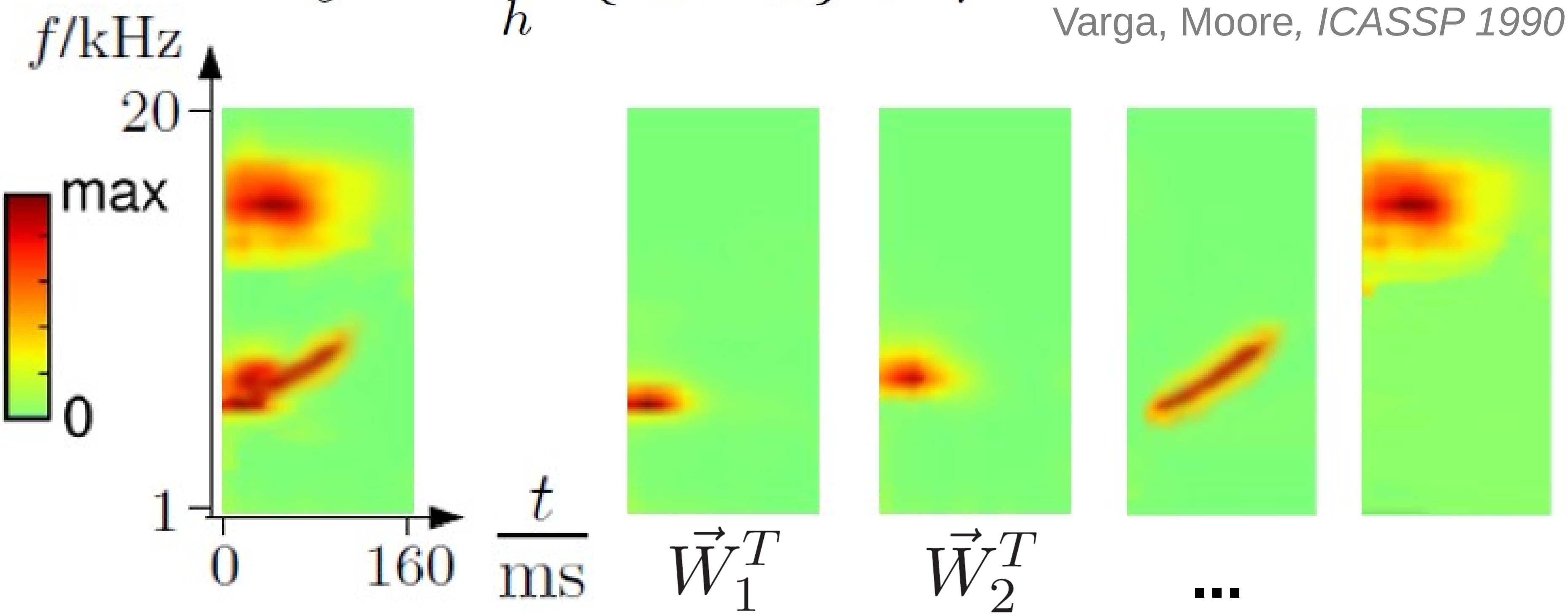
Dai et al., *NIPS* 2013
Bornschein et al., *PLOS CB* 2013
Shelton et al., *NIPS* 2012
Puertas, Bronschein, Lücke, *NIPS 2010*
Lücke, Sahani, *J Mach Learn Res* 2008
...
Roweis, *Eurospeech 2003*
Roweis, *NIPS 2002*
Varga & Moore, *ICASSP 1990*

$\vec{W}_1^T$ $\vec{W}_2^T$ ...

Cochleagram Dictionaries

$$\vec{y} = \max_{h}\{s_h \vec{W}_h\} + \vec{\eta}$$

Roweis, *Eurospeech 2003*
Roweis, *NIPS 2002*
Varga, Moore, *ICASSP 1990*



Computational Challenges

Example: Binary Prior

Binary Sparse Coding (BSC):

$$p(\vec{s}\,|\,\Theta) \;=\; \prod_h \pi^{s_h}\,(1-\pi\;)^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) \;=\; \mathcal{N}(\vec{y};\textstyle\sum_h s_h\,\vec{W}_h,\sigma^2\,\mathbb{1})$$

Henniges et al., 2010

Maximal Causes Analysis (MCA):

$$p(\vec{s}\,|\,\Theta) \;=\; \prod_h \pi^{s_h}\,(1-\pi\;)^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) \;=\; \mathcal{N}(\vec{y};\max_h\{s_h\,\vec{W}_h\},\sigma^2\,\mathbb{1})$$

Example: Binary Prior

Binary Sparse Coding (BSC):

$$p(\vec{s}\,|\,\Theta) \;=\; \prod_h \pi^{s_h}\,(1-\pi\;)^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) \;=\; \mathcal{N}(\vec{y};\textstyle\sum_h s_h\,\vec{W}_h,\sigma^2\,\mathbb{1})$$

Henniges et al., 2010

Optimization using Expectation Maximization (EM)

$$W^{\mathrm{new}} = \left(\sum_n \boldsymbol{y}^{(n)}\,\langle \boldsymbol{s}\,\rangle_p^T\right)\left(\sum_n \left\langle \boldsymbol{s}\,\boldsymbol{s}^T\right\rangle_p\right)^{-1}$$

$$\sigma^{\mathrm{new}} = \sqrt{\frac{1}{N\,D}\,\sum_n \left\langle \left|\left|\boldsymbol{y}^{(n)} - W\,\boldsymbol{s}\right|\right|^2\right\rangle_p}$$

$$\pi^{\mathrm{new}} = \frac{1}{ND}\sum_n \langle|\boldsymbol{s}|\rangle_{p^n}$$

$$\text{with } |\boldsymbol{s}| = \sum_{h=1}^{H} s_h$$

Example: Binary Prior

Binary Sparse Coding (BSC):

$$p(\vec{s}\,|\,\Theta) = \prod_h \pi^{s_h}\,(1-\pi\ )^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y};\textstyle\sum_h s_h\,\vec{W}_h,\sigma^2\,\mathbb{1})$$

Henniges et al., 2010

Expectation values scale exponentially.

$$\langle g(\boldsymbol{s})\rangle_p = \frac{\sum_{\boldsymbol{s}} p(\boldsymbol{s},\boldsymbol{y}^{(n)}\,|\,\Theta^{\mathrm{old}})\ g(\boldsymbol{s})}{\sum_{\tilde{\boldsymbol{s}}} p(\tilde{\boldsymbol{s}},\boldsymbol{y}^{(n)}\,|\,\Theta^{\mathrm{old}})}$$

Optimization using Expectation Maximization (EM)

$$W^{\mathrm{new}} = \Big(\sum_n \boldsymbol{y}^{(n)}\,\langle\boldsymbol{s}\rangle_p^T\Big)\Big(\sum_n \langle\boldsymbol{s}\,\boldsymbol{s}^T\rangle_p\Big)^{-1}$$

$$\sigma^{\mathrm{new}} = \sqrt{\frac{1}{N\,D}\sum_n \Big\langle \big|\big|\boldsymbol{y}^{(n)} - W\,\boldsymbol{s}\big|\big|^2\Big\rangle_p}$$

$$\pi^{\mathrm{new}} = \frac{1}{ND}\sum_n \langle|\boldsymbol{s}|\rangle_{p^n}$$

$$\text{with } |\boldsymbol{s}| = \sum_{h=1}^{H} s_h$$

Example: Binary Prior

Binary Sparse Coding (BSC):

$$p(\vec{s}\,|\,\Theta) = \prod_h \pi^{s_h}\,(1-\pi\ )^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y};\textstyle\sum_h s_h\,\vec{W}_h,\sigma^2\,\mathbb{1})$$

Henniges et al., 2010

Expectation values scale exponentially.

$$\langle g(\boldsymbol{s})\rangle_p = \frac{\sum_{\boldsymbol{s}} p(\boldsymbol{s},\boldsymbol{y}^{(n)}\,\|\,\Theta^{\mathrm{old}})\ g(\boldsymbol{s})}{\sum_{\tilde{\boldsymbol{s}}} p(\,\tilde{\boldsymbol{s}}\,,\boldsymbol{y}^{(n)}\,|\,\Theta^{\mathrm{old}})}$$

$$\langle g(\boldsymbol{s})\rangle_p = \frac{\sum_a p(\vec{s}_a,\vec{y}^{(n)}\,|\,\Theta')\ g(\vec{s}\,) + \sum_{\substack{a,b\\ a<b}} p(\vec{s}_{ab},\vec{y}^{(n)}\,|\,\Theta')\ g(\vec{s}\,) + \ldots}{p(\vec{0},\vec{y}^{(n)}\,|\,\Theta') + \sum_a p(\vec{s}_a,\vec{y}^{(n)}\,|\,\Theta') + \sum_{\substack{a,b\\ a<b}} p(\vec{s}_{ab},\vec{y}^{(n)}\,|\,\Theta') + \ldots}$$

Idea: Truncate the sums

where $\vec{s}_a := (0,\ldots,0,1,0,\ldots,0)$ with only $s_a = 1$

$\vec{s}_{ab} := (0,\ldots,0,1,0,\ldots,0,1,0,\ldots,0)$ with only $s_a = 1,\, s_b = 1,\, a \neq b$,

and $\vec{s}_{abc}$ etc. are defined analogously.

Example: Binary Prior

Binary Sparse Coding (BSC):

$$p(\vec{s}\,|\,\Theta) = \prod_h \pi^{s_h}\,(1-\pi\;)^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y};\textstyle\sum_h s_h\,\vec{W}_h,\sigma^2\,\mathbb{1})$$

Expectation values scale exponentially.

$$\big\langle g(\boldsymbol{s})\big\rangle_{\boldsymbol{q_n}} = \frac{\sum_{\boldsymbol{s}\in\mathcal{K}_n} p(\boldsymbol{s},\boldsymbol{y}^{(n)}\,\|\,\Theta^{\mathrm{old}})\;g(\boldsymbol{s})}{\sum_{\tilde{\boldsymbol{s}}\in\mathcal{K}_n} p(\,\tilde{\boldsymbol{s}}\,,\boldsymbol{y}^{(n)}\,|\,\Theta^{\mathrm{old}})}$$

$$\langle g(\boldsymbol{s})\rangle_p = \frac{\sum_a p(\vec{s}_a,\vec{y}^{(n)}\,|\,\Theta')\;g(\vec{s}\,) + \sum_{\substack{a,b\\a<b}} p(\vec{s}_{ab},\vec{y}^{(n)}\,|\,\Theta')\;g(\vec{s}\,) + \ldots}{p(\vec{0},\vec{y}^{(n)}\,|\,\Theta') + \sum_a p(\vec{s}_a,\vec{y}^{(n)}\,|\,\Theta') + \sum_{\substack{a,b\\a<b}} p(\vec{s}_{ab},\vec{y}^{(n)}\,|\,\Theta') + \ldots}$$

Such a truncation of sums is equivalent to a variational approximation:

$$\tilde{q}^{(n)}(\vec{s}\,;\Theta^{\mathrm{old}}) = \frac{p(\vec{s},\vec{y}^{(n)}\,|\,\Theta^{\mathrm{old}})}{\sum_{\vec{s}'\in\mathcal{K}_n} p(\vec{s}\,',\vec{y}^{(n)}\,|\,\Theta^{\mathrm{old}})}\,\delta(\vec{s}\in\mathcal{K}_n)$$

variational distribution (not factored)

Example: Binary Prior

Binary Sparse Coding (BSC):

$$p(\vec{s}\,|\,\Theta) = \prod_h \pi^{s_h}\,(1-\pi\ )^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y}; \textstyle\sum_h s_h \vec{W}_h, \sigma^2\,\mathbb{1})$$

Henniges et al., 2010

Expectation values scale exponentially.

$$\langle g(\boldsymbol{s})\rangle_p = \frac{\sum_{\boldsymbol{s}} p(\boldsymbol{s}, \boldsymbol{y}^{(n)}\,|\,\Theta^{\text{old}})\ g(\boldsymbol{s})}{\sum_{\tilde{\boldsymbol{s}}} p(\tilde{\boldsymbol{s}}, \boldsymbol{y}^{(n)}\,|\,\Theta^{\text{old}})}$$

Idea: Truncate the sums

$$\langle g(\boldsymbol{s})\rangle_p = \frac{\sum_a p(\vec{s}_a, \vec{y}^{(n)}\,|\,\Theta')\ g(\vec{s}) + \sum_{\substack{a,b\\a<b}} p(\vec{s}_{ab}, \vec{y}^{(n)}\,|\,\Theta')\ g(\vec{s}) + \ldots}{p(\vec{0}, \vec{y}^{(n)}\,|\,\Theta') + \sum_a p(\vec{s}_a, \vec{y}^{(n)}\,|\,\Theta') + \sum_{\substack{a,b\\a<b}} p(\vec{s}_{ab}, \vec{y}^{(n)}\,|\,\Theta') + \ldots}$$

Expectation Truncation:

$$q_n(\vec{s};\Theta) = \frac{1}{A}\, p(\vec{s}\,|\,\vec{y}^{(n)},\Theta)\ \delta(\vec{s} \in \mathcal{K}_n)$$

Relation to Other Approximations

exact: $$q_n(\vec{s};\Theta) \;=\; p(\vec{s}\,|\,\vec{y}^{\,(n)},\Theta)$$

MAP: $$q_n(\vec{s};\Theta) \;=\; \delta(\vec{s}-\vec{s}^{\,\mathrm{max}})$$

Laplace: $$q_n(\vec{s};\,\Theta) = \mathcal{N}(\vec{s};\,\vec{s}^{\,\mathrm{max}},\Sigma)$$

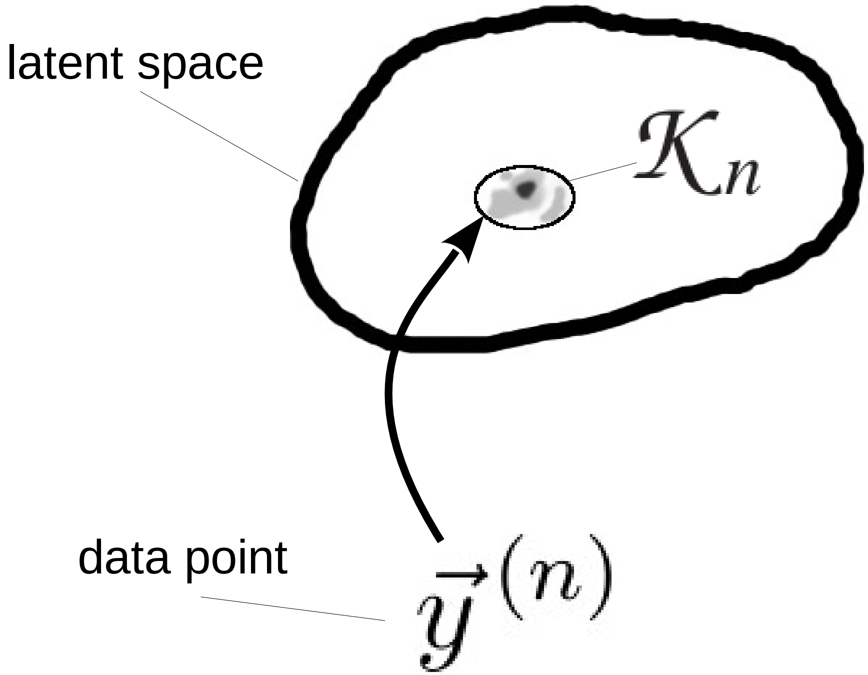
mean-field: $$q_n(\vec{s};\,\Theta) = \prod_h q^{(n)}_{h,\vec{\lambda}_n}(s_h;\Theta)$$

truncated: $$\boxed{q_n(\vec{s};\Theta) \;=\; \frac{1}{A}\,p(\vec{s}\,|\,\vec{y}^{\,(n)},\Theta)\,\delta(\vec{s}\in\mathcal{K}_n)}$$

Expectation Truncation

Lücke, Eggert, *JMLR* 2010

$$p(\vec{s}\,|\,\vec{y}^{\,(n)},\Theta)$$



ET

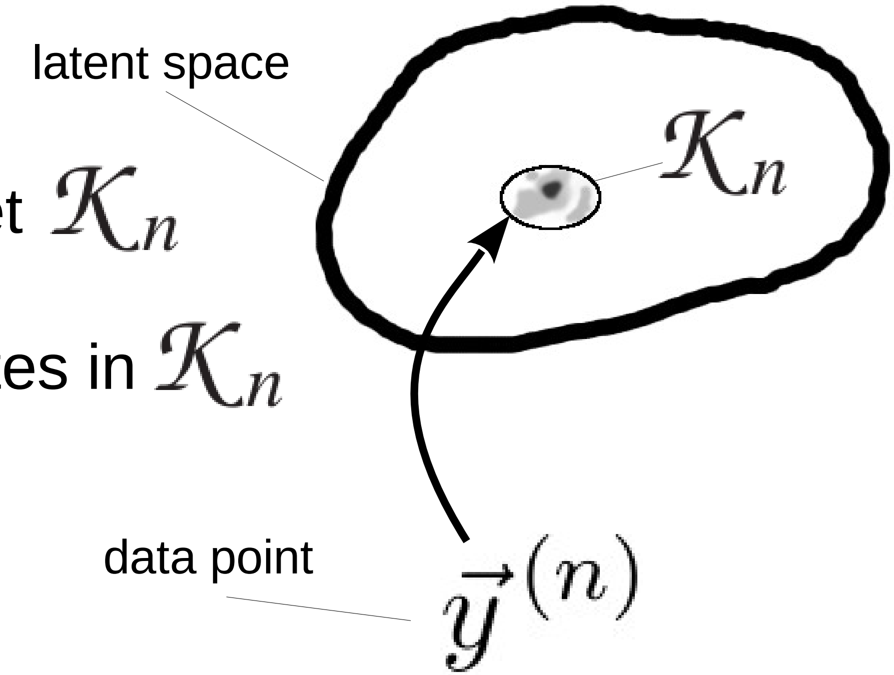
Expectation Truncation

Lücke, Eggert, *JMLR* 2010

$$p(\vec{s} \,|\, \vec{y}^{(n)}, \Theta)$$

preselect subset $\mathcal{K}_n$

evaluate all states in $\mathcal{K}_n$



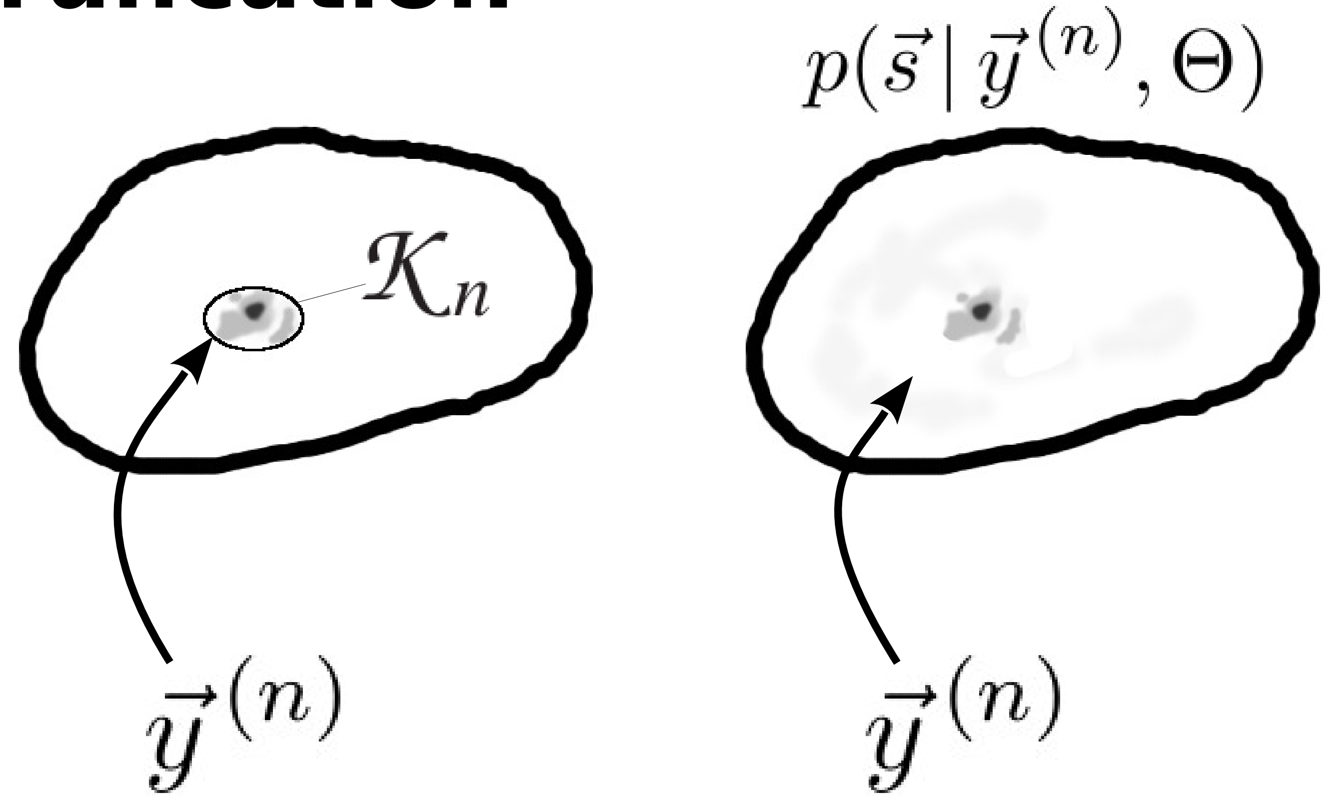
ET

Expectation Truncation

Lücke, Eggert, *JMLR* 2010

preselect subset $\mathcal{K}_n$

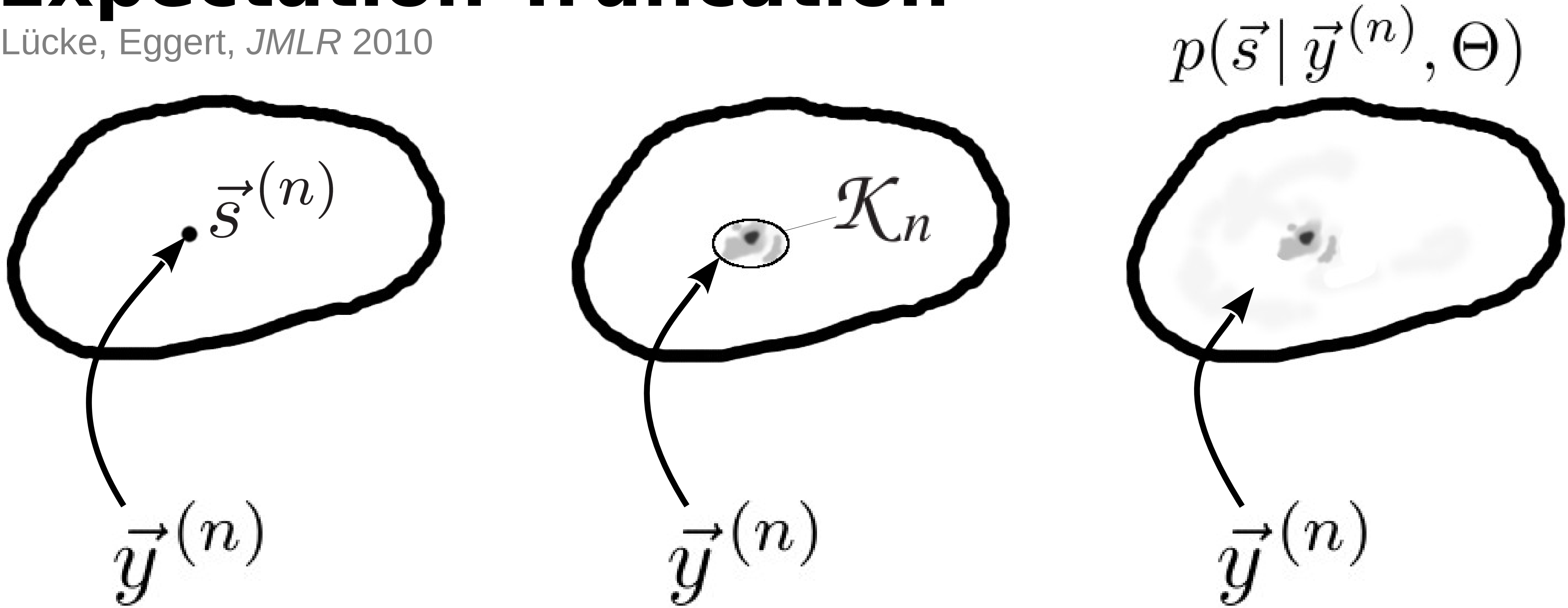
evaluate all states in $\mathcal{K}_n$



ET **optimal case**

Expectation Truncation

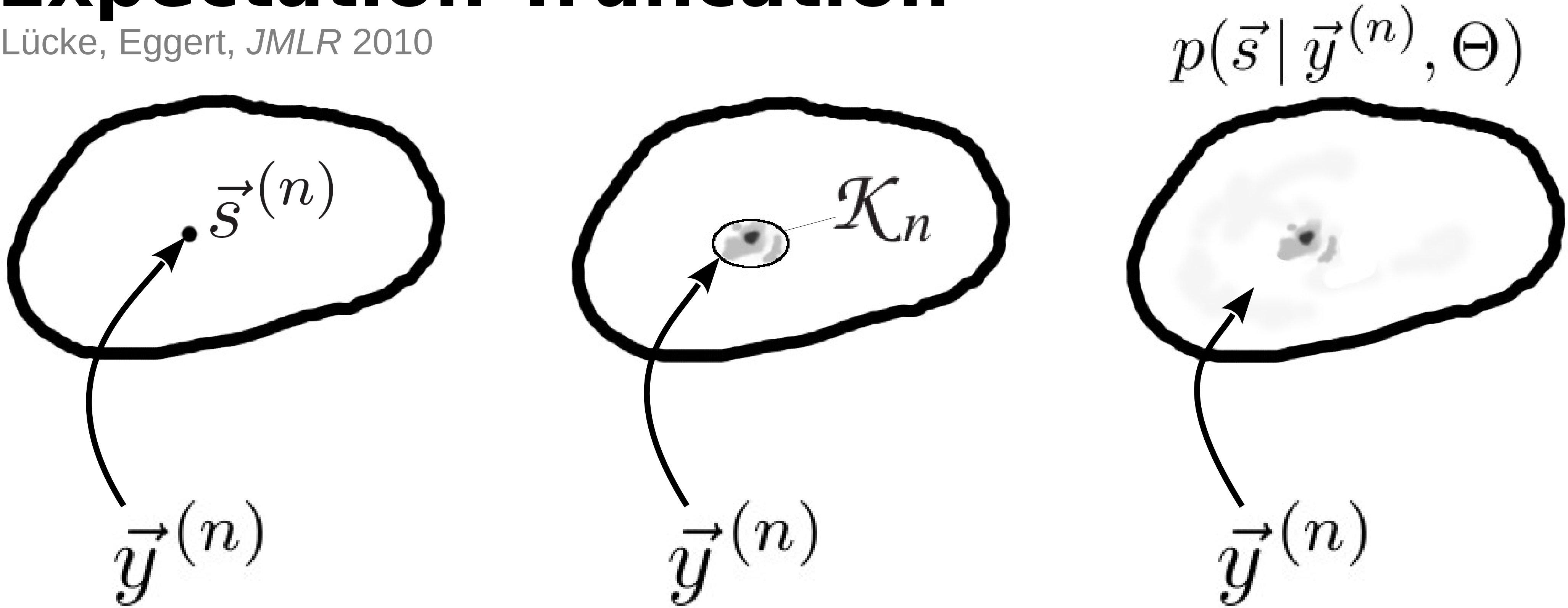
Lücke, Eggert, *JMLR* 2010



deterministic **ET** **optimal case**

Expectation Truncation

Lücke, Eggert, *JMLR* 2010

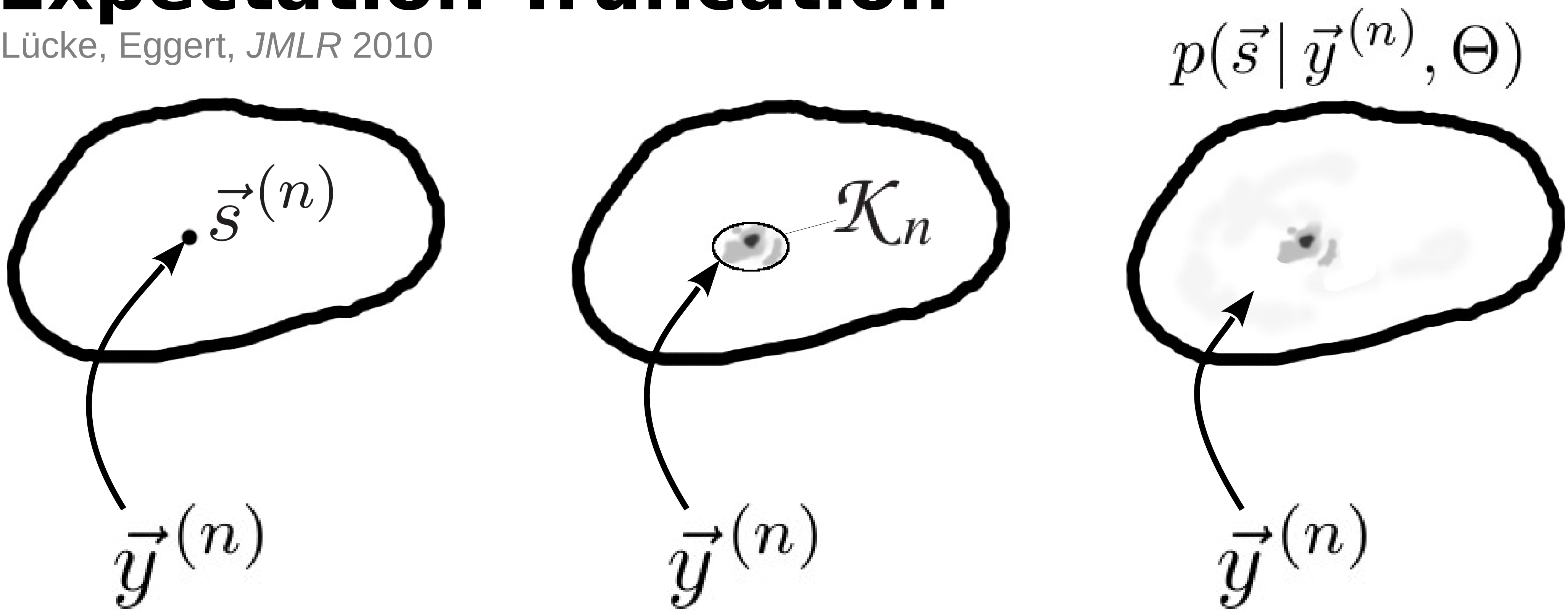


Exact: $$q_n(\vec{s};\Theta) \;=\; p(\vec{s}\,|\,\vec{y}^{(n)},\Theta)$$

MAP: $$q_n(\vec{s};\Theta) \;=\; \delta(\vec{s}-\vec{s}^{\,\mathrm{max}})$$

Expectation Truncation

Lücke, Eggert, *JMLR* 2010

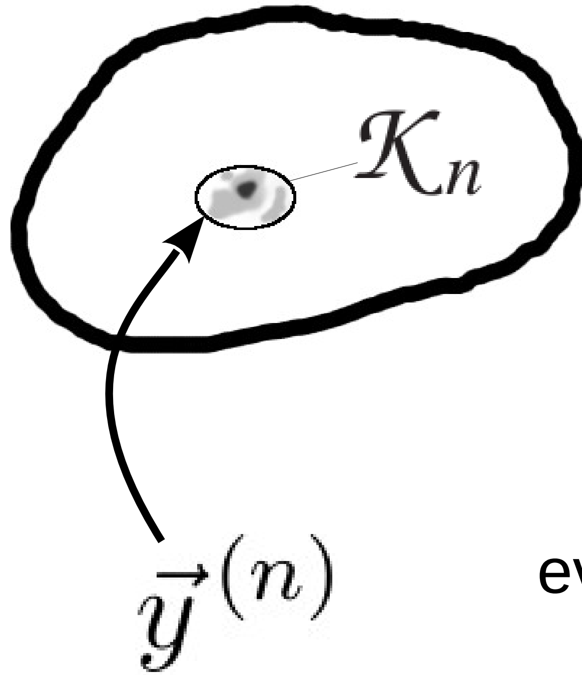


Exact: $$q_n(\vec{s};\Theta) = p(\vec{s}\,|\,\vec{y}^{(n)},\Theta)$$

ET: $$q_n(\vec{s};\Theta) = \frac{1}{A}\,p(\vec{s}\,|\,\vec{y}^{(n)},\Theta)\,\delta(\vec{s}\in\mathcal{K}_n)$$

MAP: $$q_n(\vec{s};\Theta) = \delta(\vec{s}-\vec{s}^{\,\mathrm{max}})$$

Expectation Truncation (ET)



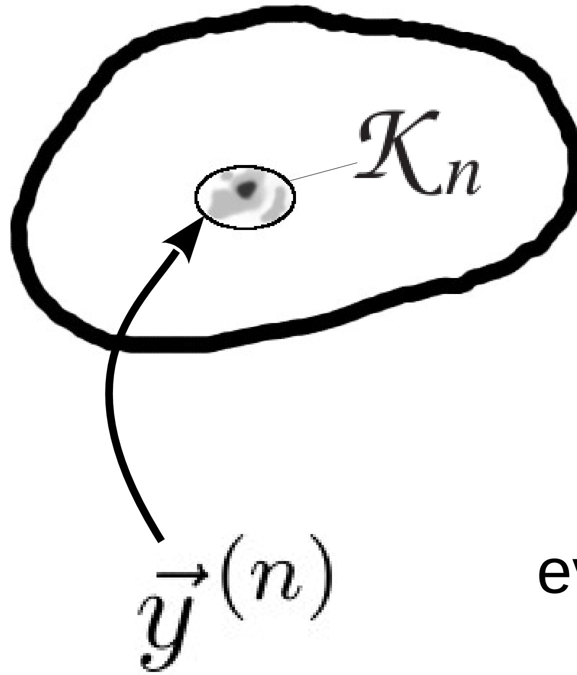
preselect subset $\mathcal{K}_n$

evaluate states in $\mathcal{K}_n$

discriminative approaches ⟷ generative approaches

Expectation Truncation (ET)

Lücke, Eggert, *JMLR* 2010
Henniges et al., *LVA* 2010
Puertas et al., *NIPS* 2010
Shelton et al., *NIPS* 2011
Exarchakis et al., *LVA* 2012
Dai, Lücke, *CVPR* 2012a
Dai, Lücke, *CVPR* 2012b
Shelton et al., *NIPS* 2012
Bornschein et al., *PLOS CB* 2013
Sheikh et al., *JMLR* 2014
Henniges et al., *JMLR* 2014



preselect subset $\mathcal{K}_n$

evaluate states in $\mathcal{K}_n$

discriminative approaches ⟷ generative approaches

Expectation Truncation (ET)

Lücke, Eggert, *JMLR* 2010

variational approximation	correlations	multiple modes	example papers
max a-posteriori (MAP)	no	no	Olshausen, Field, 1996; A. Ng et al.
Gaussian	yes	no	Opper et al.; Seeger, 2008
mean-field	no	yes	Titsias et al., 2011; Goodfellow … Bengio, 2012;

Expectation Truncation (ET)

Lücke, Eggert, *JMLR* 2010

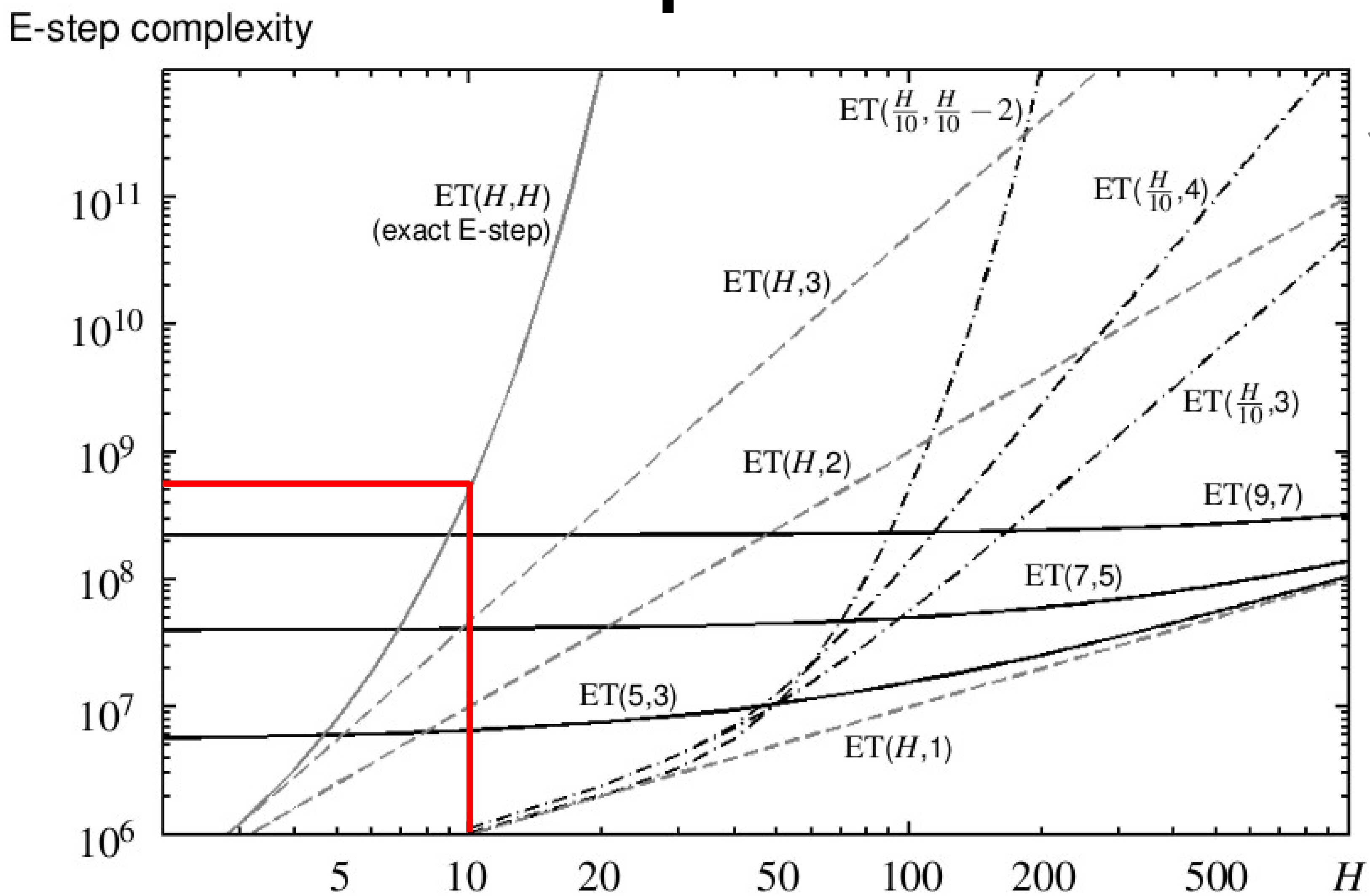
variational approximation	correlations	multiple modes	example papers
max a-posteriori (MAP)	no	no	Olshausen, Field, 1996; A. Ng et al.
Gaussian	yes	no	Opper et al.; Seeger, 2008
mean-field	no	yes	Titsias et al., 2011; Goodfellow … Bengio, 2012;
expectation truncation	yes	yes	Sheikh, Shelton, Lücke, 2012; Puertas … '10; Dai&Lücke, '12;

Expectation Truncation (ET)

Lücke, Eggert, *JMLR* 2010

variational approximation	correlations	multiple modes	broad posterior distributions
max a-posteriori (MAP)	no	no	no
Gaussian	yes	no	yes
mean-field	no	yes	yes
expectation truncation	yes	yes	no

Expectation Truncation

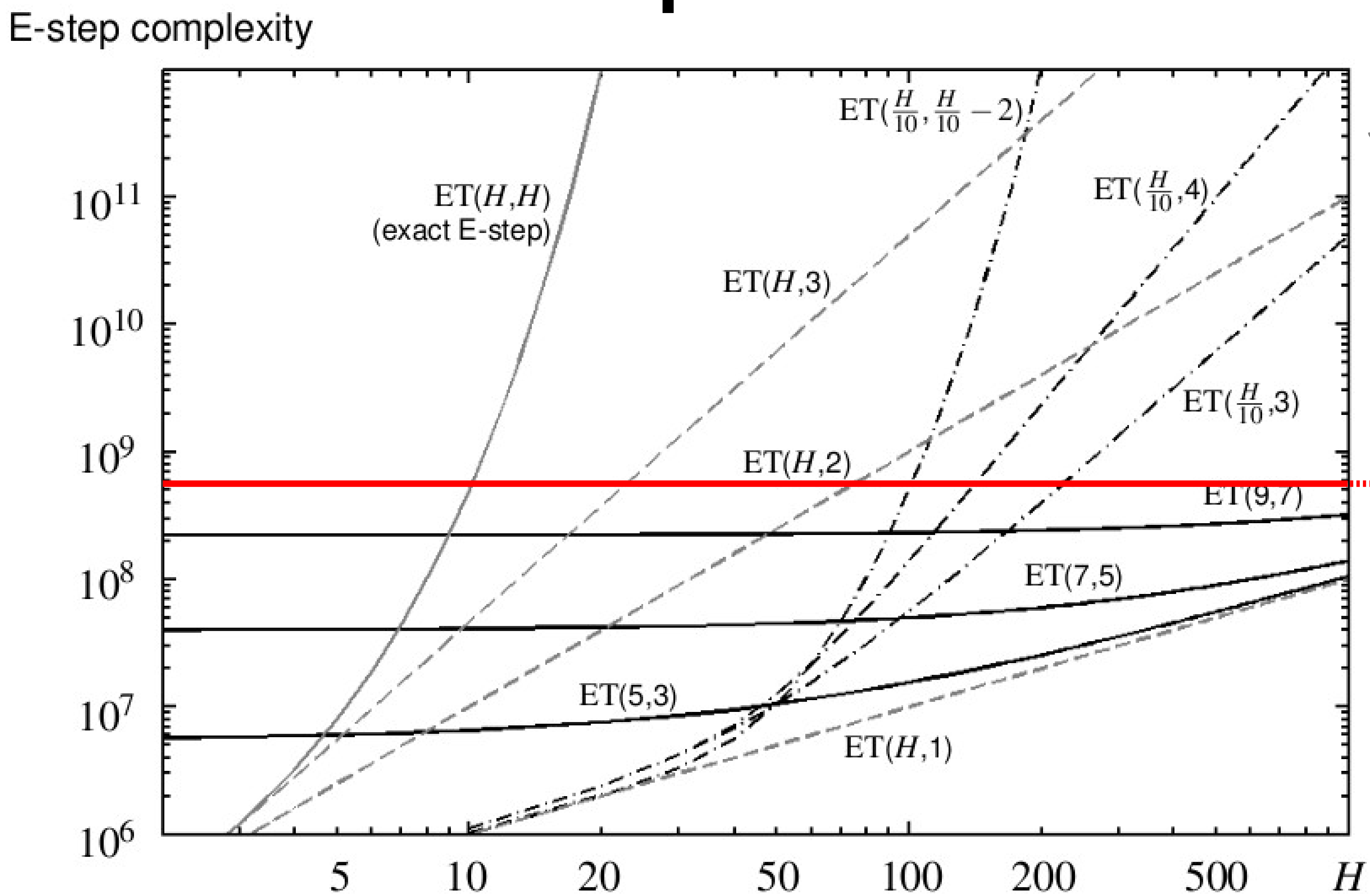


Lücke, Eggert, *JMLR* 2010;

exact EM

$$\mathcal{O}(e^H)$$

Expectation Truncation



Lücke, Eggert, *JMLR* 2010;

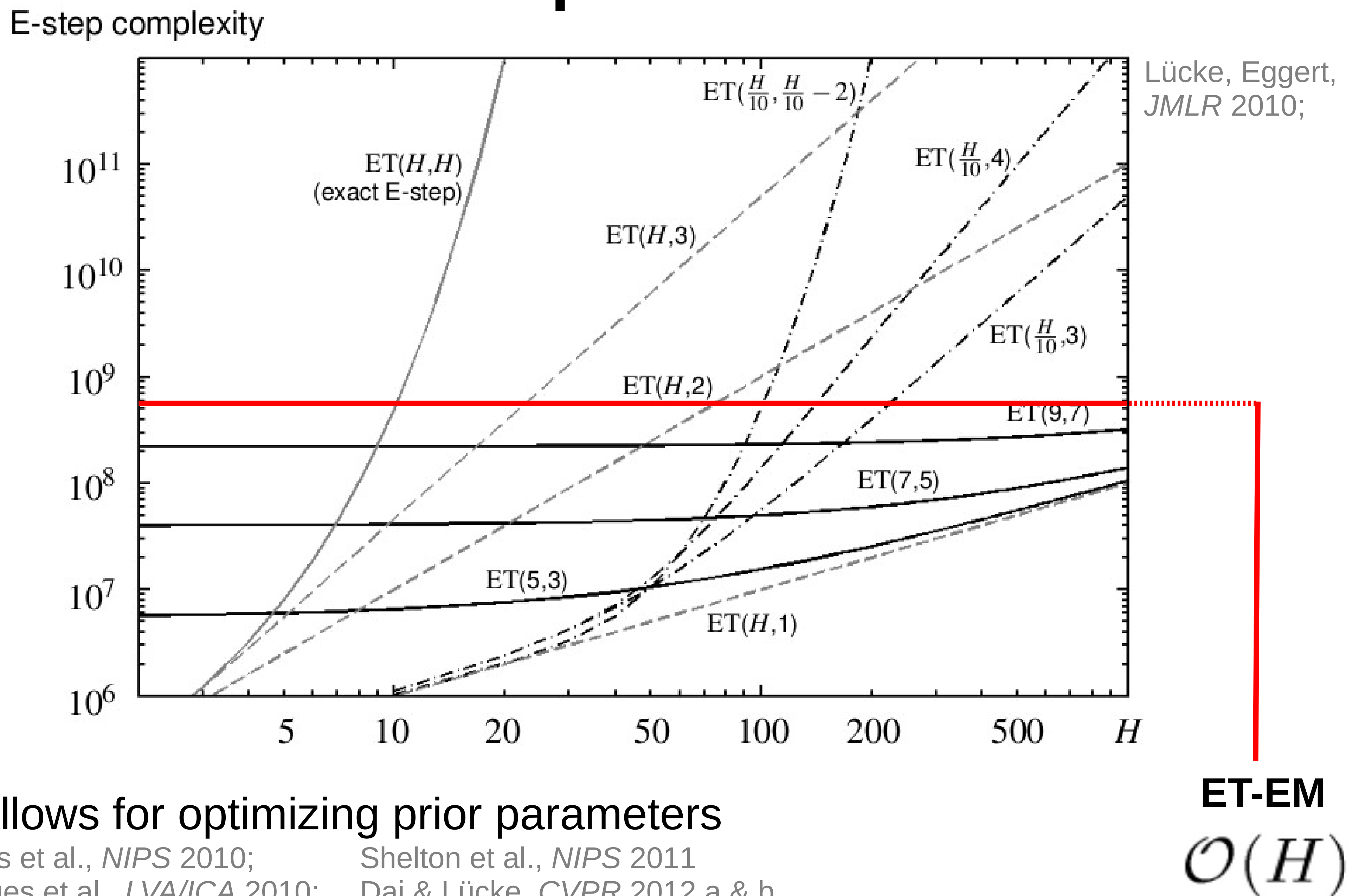
ET parametrizes accuracy

ET-EM

$\mathcal{O}(H)$

Expectation Truncation

Expectation Truncation

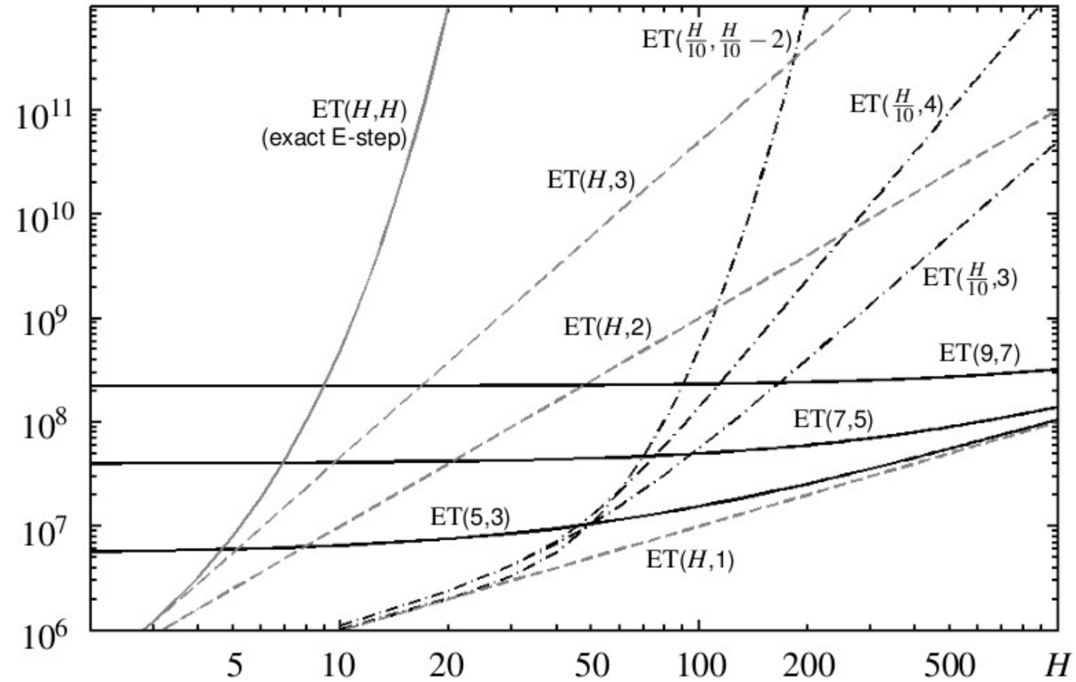


Lücke, Eggert, *JMLR* 2010;

ET allows for optimizing prior parameters

Puertas et al., *NIPS* 2010;
Henniges et al., *LVA/ICA* 2010;
Lücke, Eggert, JMLR 2010;
Shelton et al., *NIPS* 2011
Dai & Lücke, *CVPR* 2012 a & b
Shelton *et al., NIPS 2012*

E-step complexity



MCA generative model:

$$p(\vec{s}\,|\,\Theta) = \prod_h \pi_h^{s_h}\,(1-\pi_h)^{1-s_h}$$

$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y};\max_h\{s_h^{(n)}\,\vec{W}_h\},\sigma^2\,\mathbb{1})$$

- multiple modes
- correlations

} { non-linear models
advanced linear

Applicable to large-scale since 2010.

up to 4000 cores or GOLD (16 GPUs)

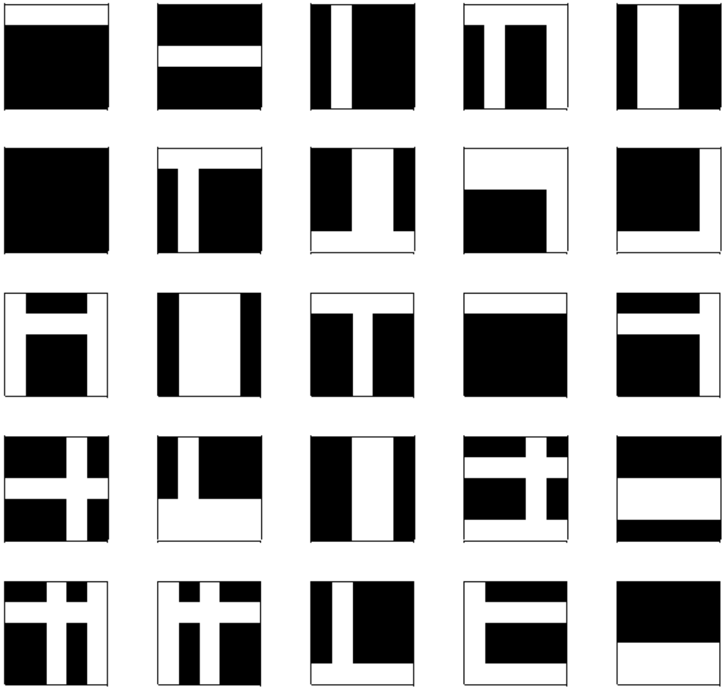
- Problem: local likelihood optima
→ simulated annealing

- Problem: no closed-form M-steps
→ general sol. for non-linear models
Dai, Lücke, *TPAMI* 2014
Lücke et al., *NIPS* 2009
Lücke, Sahani, *JMLR* 2008

- Problem: E-step comput. intractable
Dai et al.,, *NIPS* 2013
Shelton et al., *NIPS* 2012
Shelton et al., *NIPS* 2011
Puertas et al., *NIPS* 2010
Lücke, Eggert, *JMLR* 2010
Lücke et al., *NIPS* 2009
Lücke, Sahani, *JMLR* 2008

Bars Test

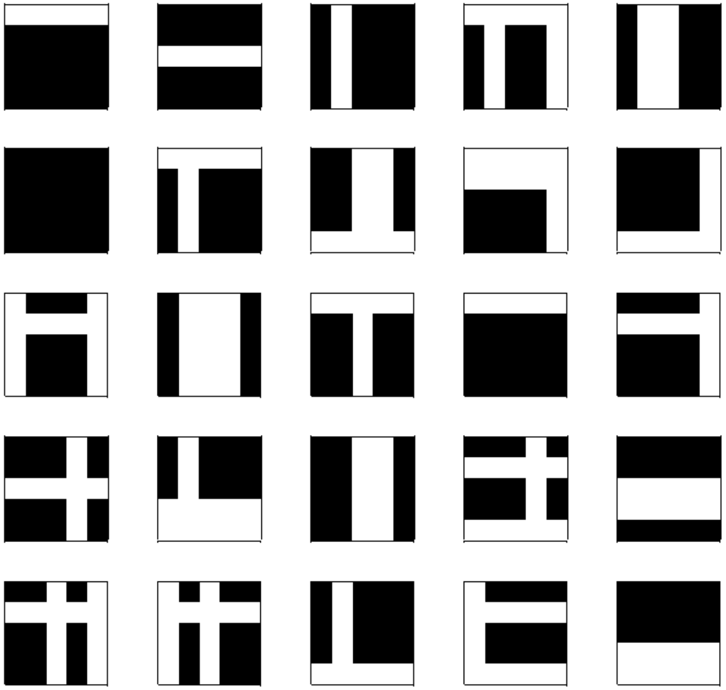
$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$



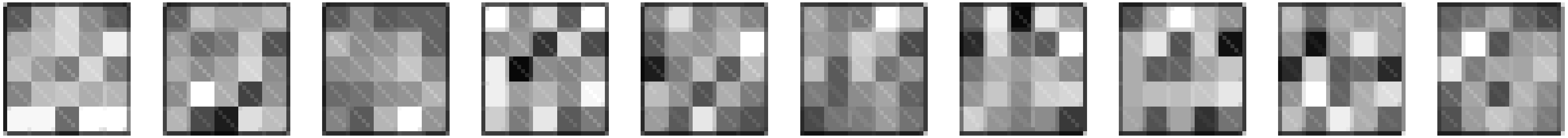
The Bars Test, Földiák, 1990

Bars Test

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$



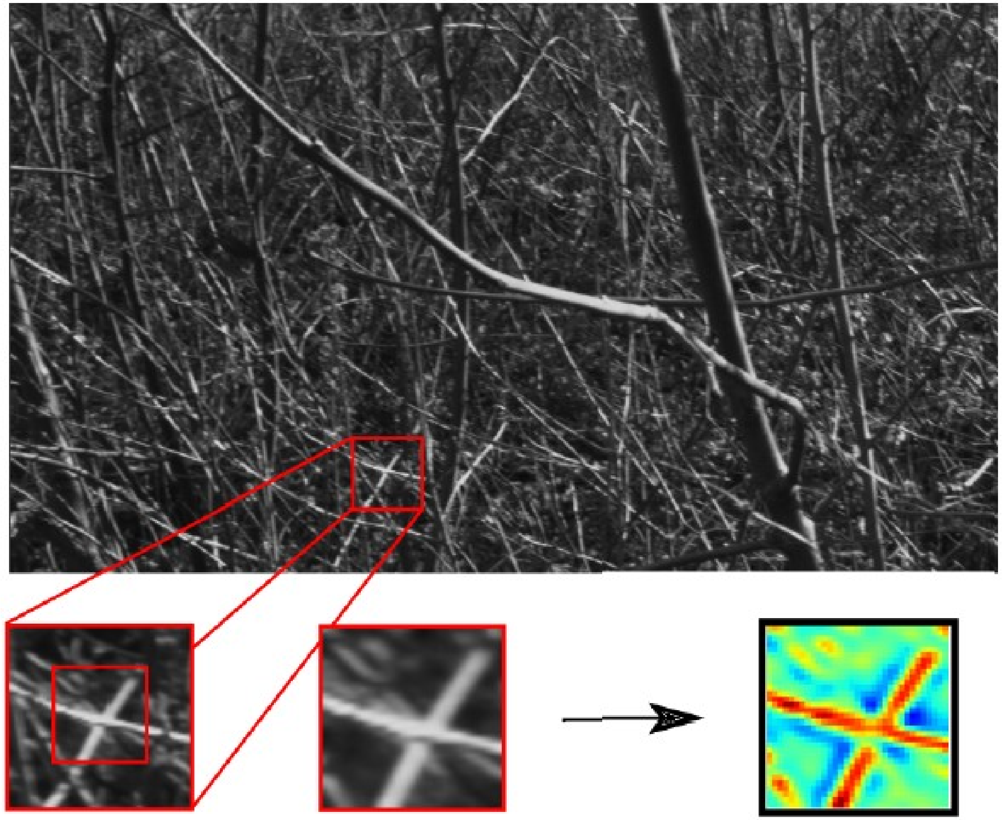
The Bars Test, Földiák, 1990



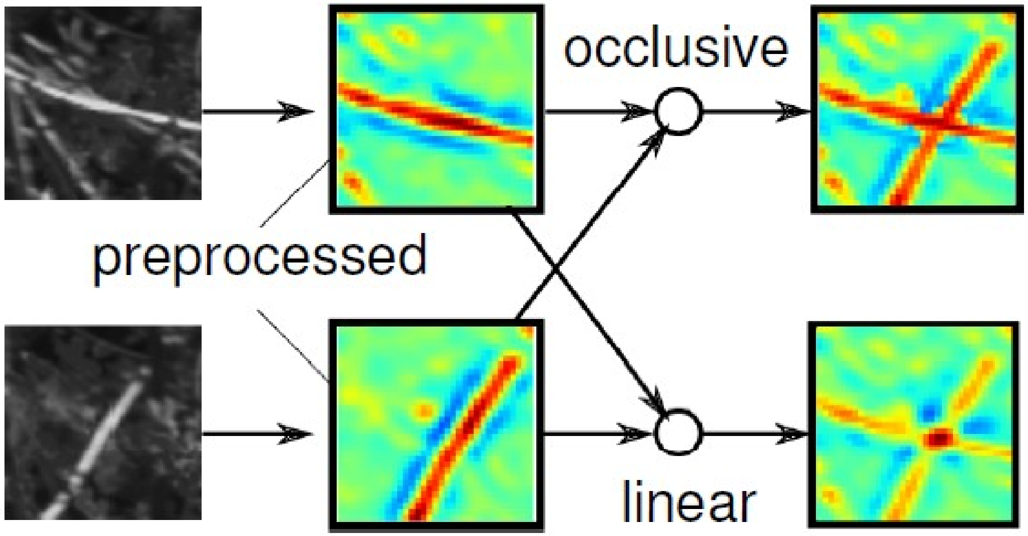
Lücke and Sahani, *JMLR* 2008
Lücke and Eggert, *JMLR* 2010

Encoding of Visual Scenes

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$



LGN



$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

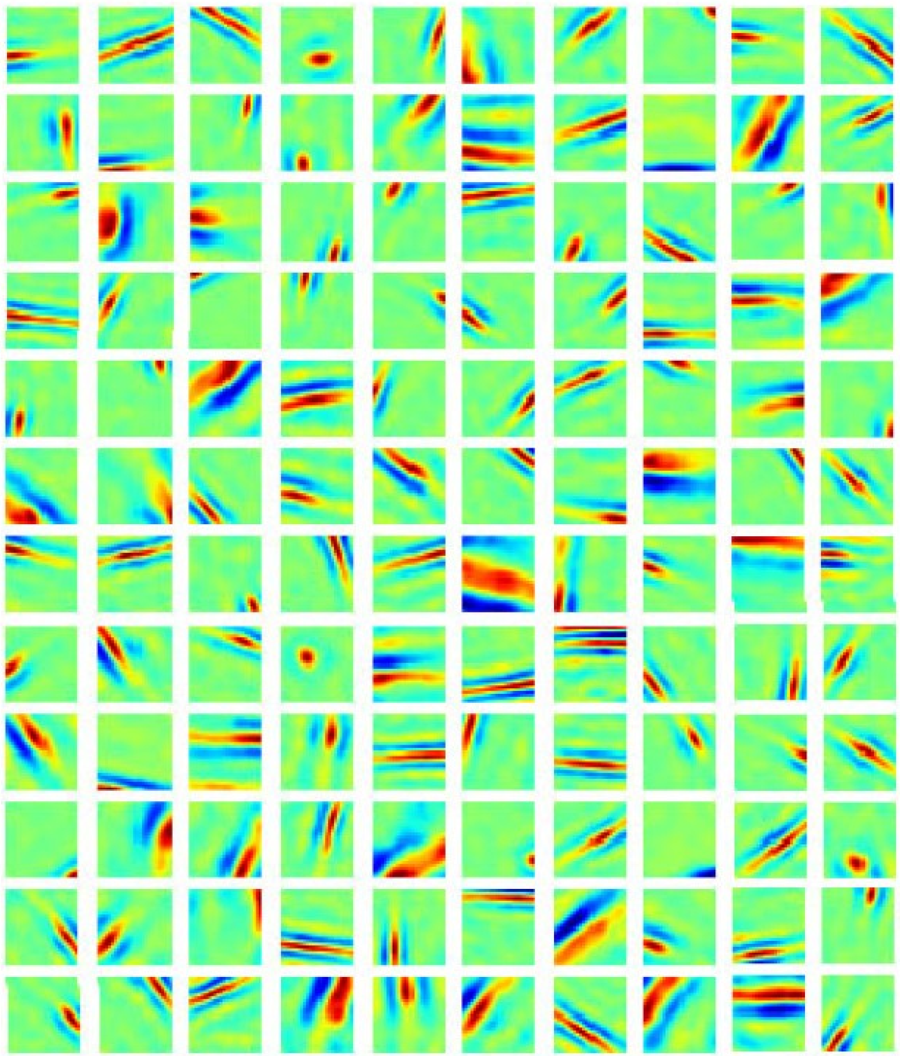
Sparse Coding

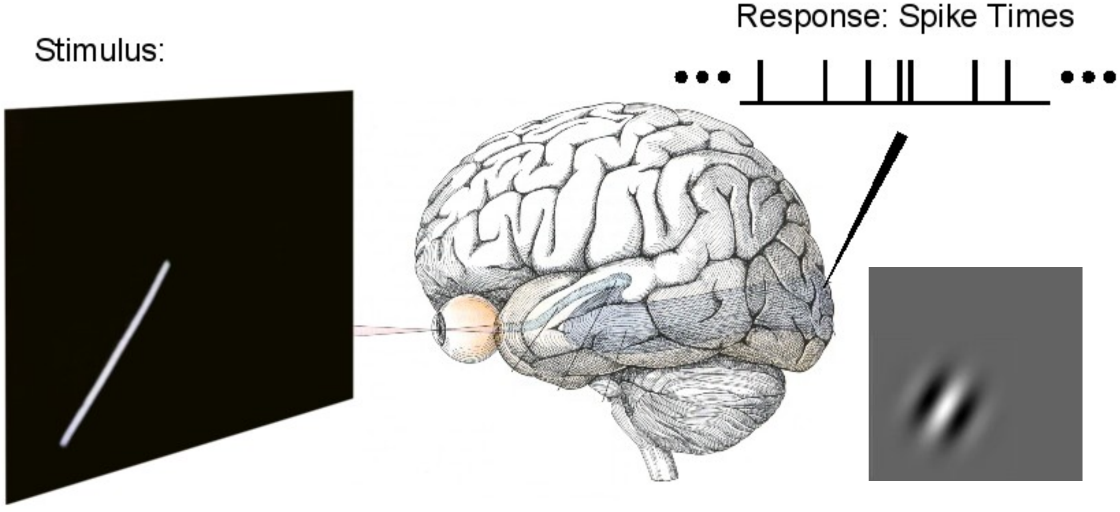
$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

Non-linear Sparse Coding

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

Puertas et al. 2010; Bornschein et. al., 2013



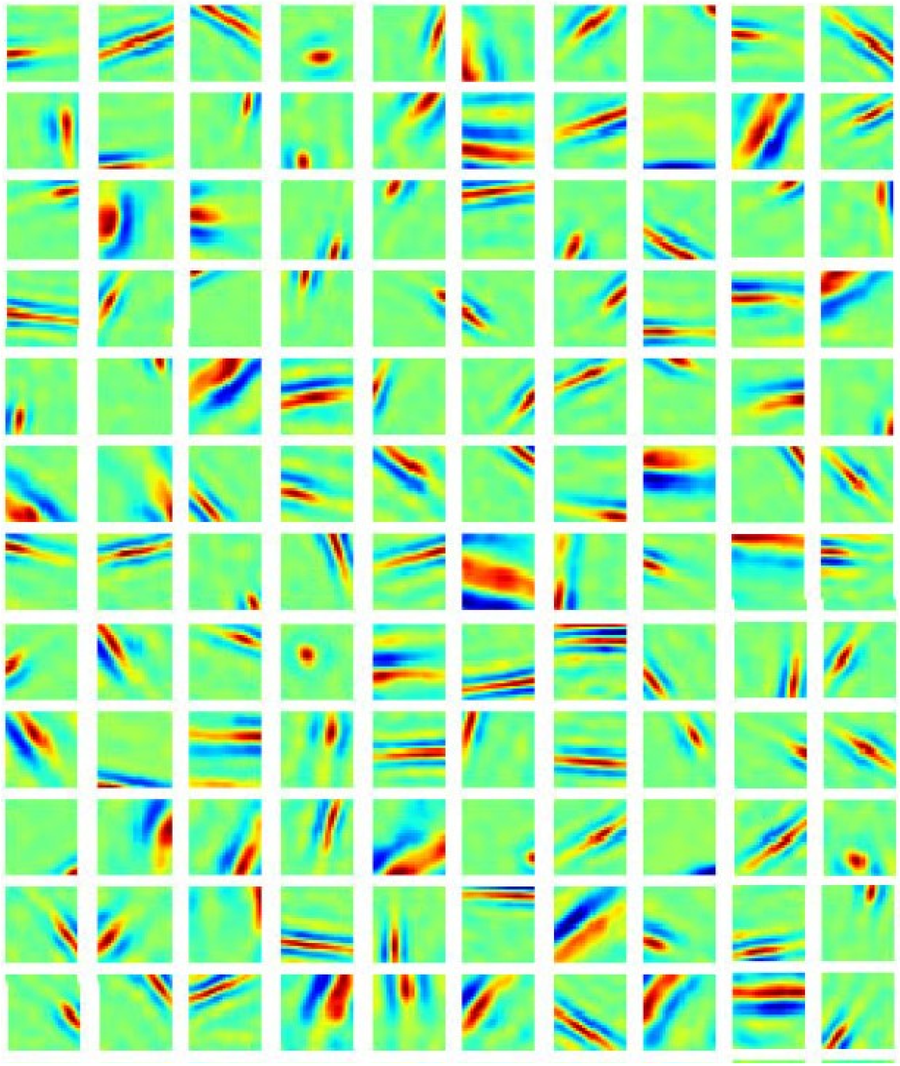


Olshausen & Field
Nature 1996

Nobel Prize in Physiology 1981
Hubel & Wiesel

Sparse Coding

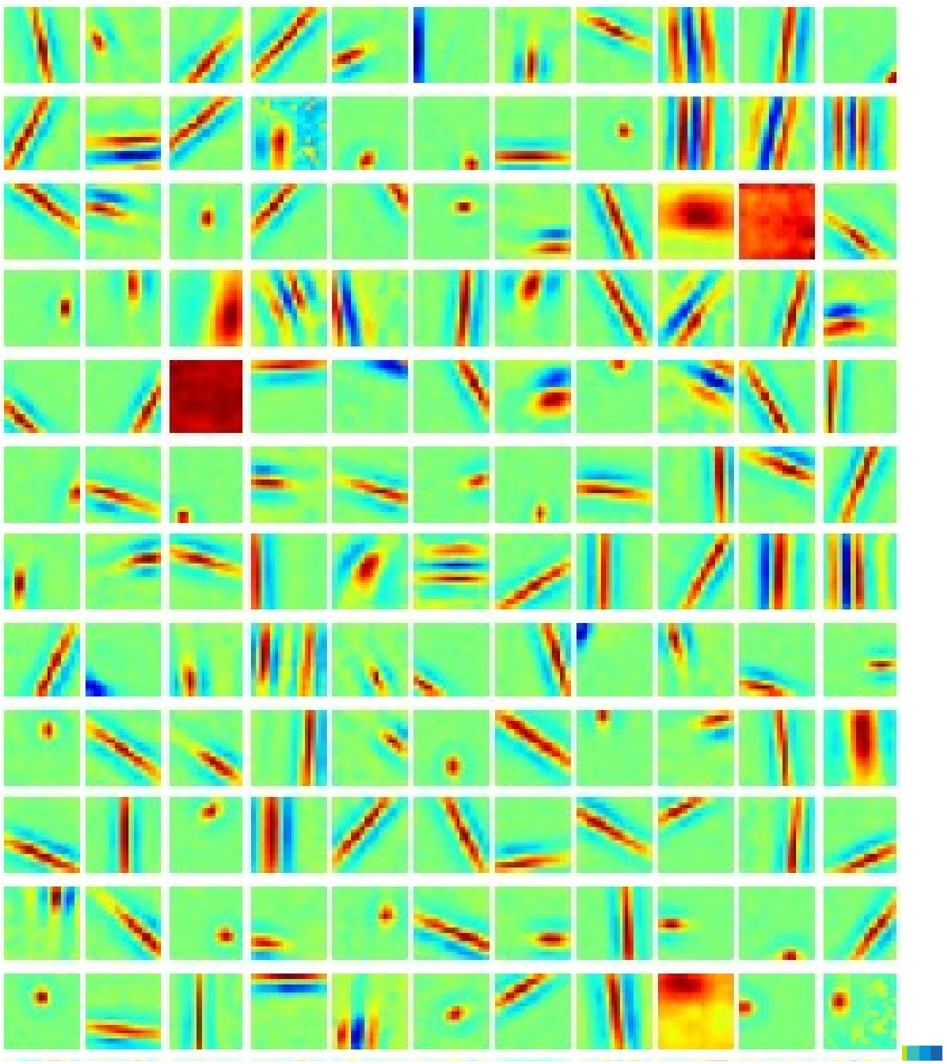
$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$



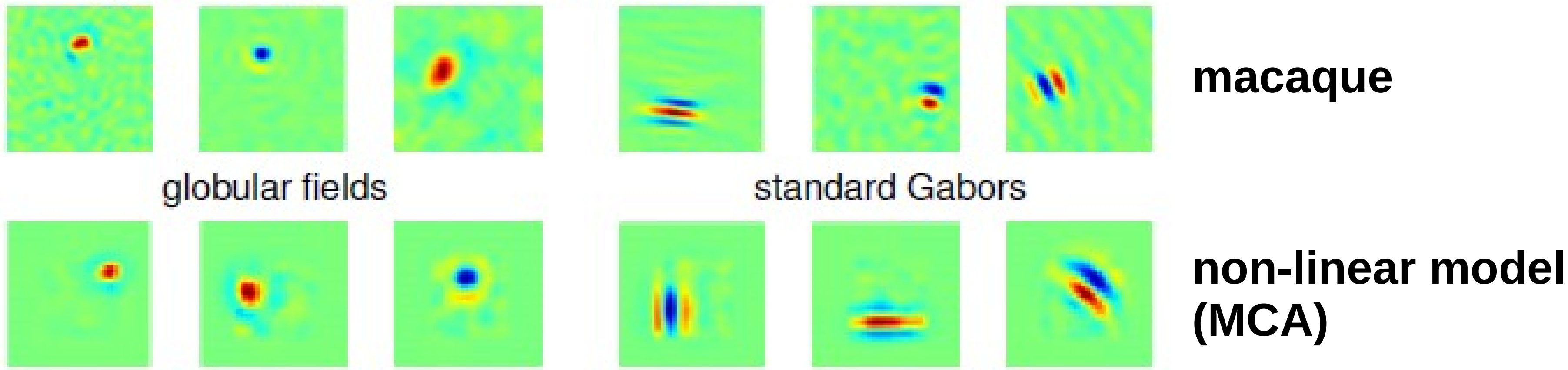
Non-linear Sparse Coding

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

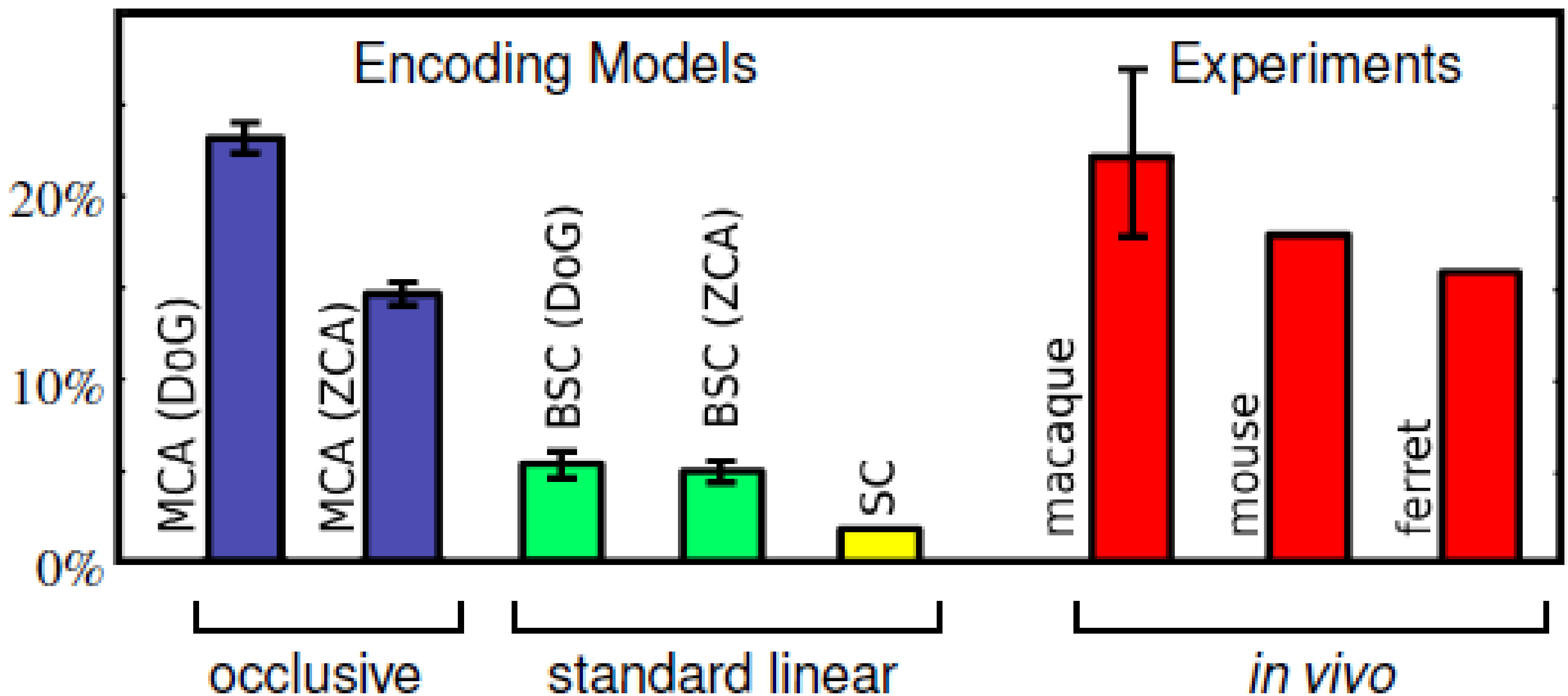
Puertas et al. 2010; Bornschein et. al., 2013



Two types of simple cell RFs are measured



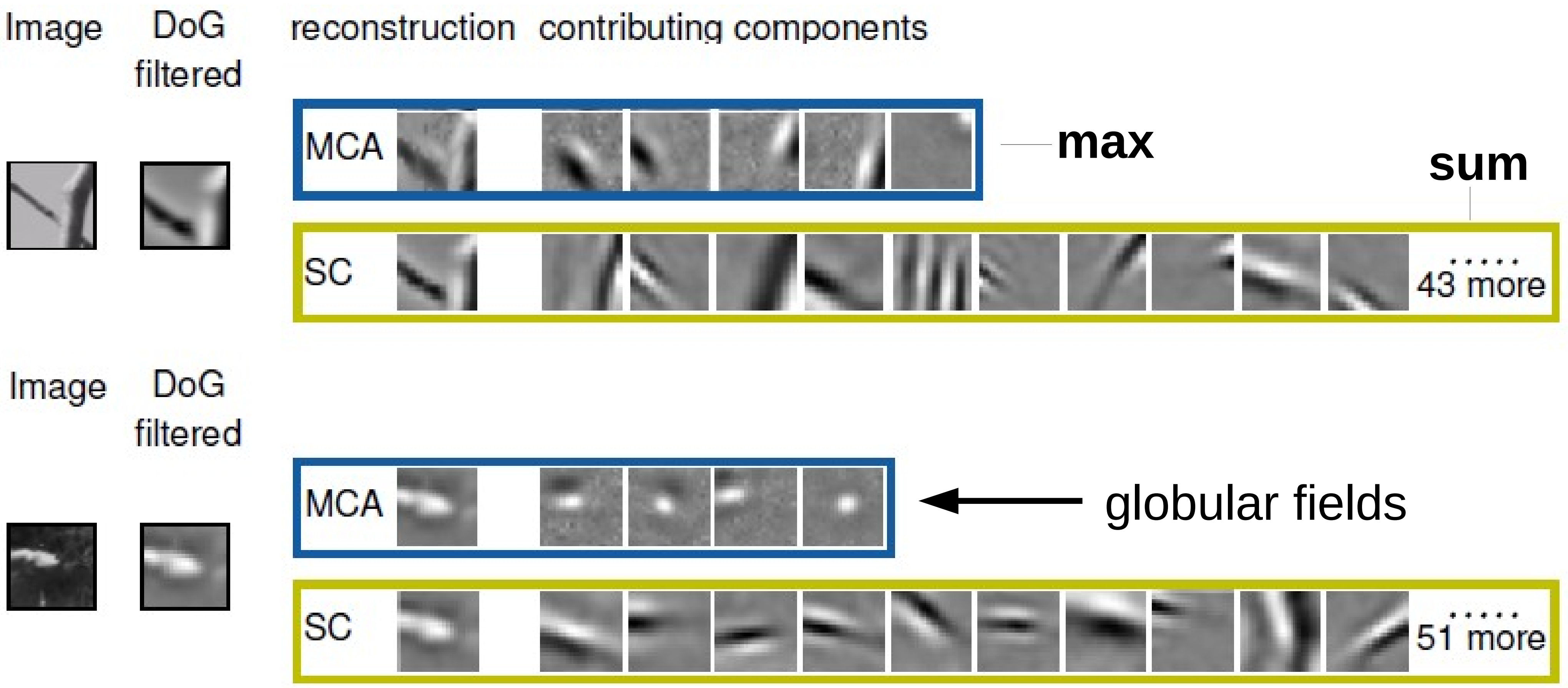
Predicted and Measured Percentages of Globular RFs



Bornschein, Henniges, Lücke, *PLOS Comp Biology* 2013

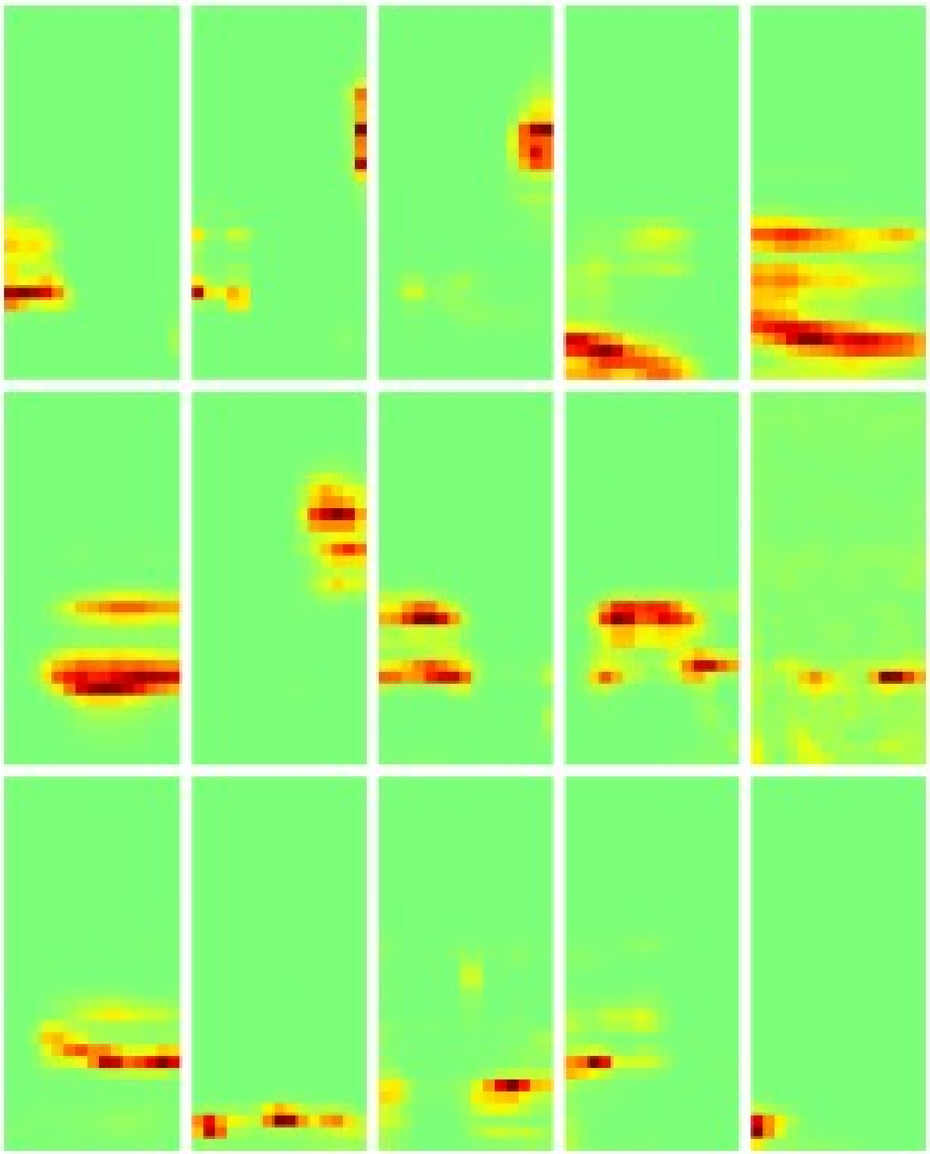
Encoding of Visual Scenes

("Viterbi" for sparse coding)



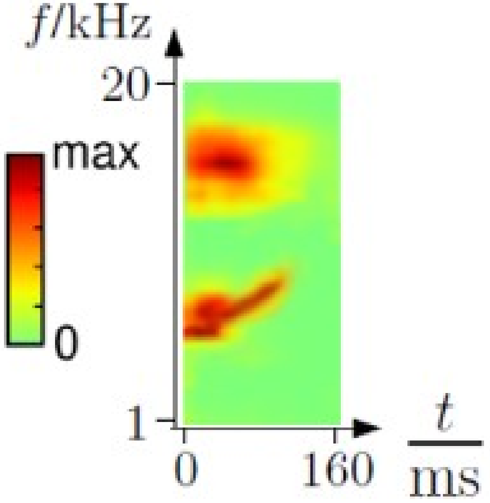
Bornschein, Henniges, Lücke, *PLOS Comp Biology* 2013

Encoding of Acoustic Scenes

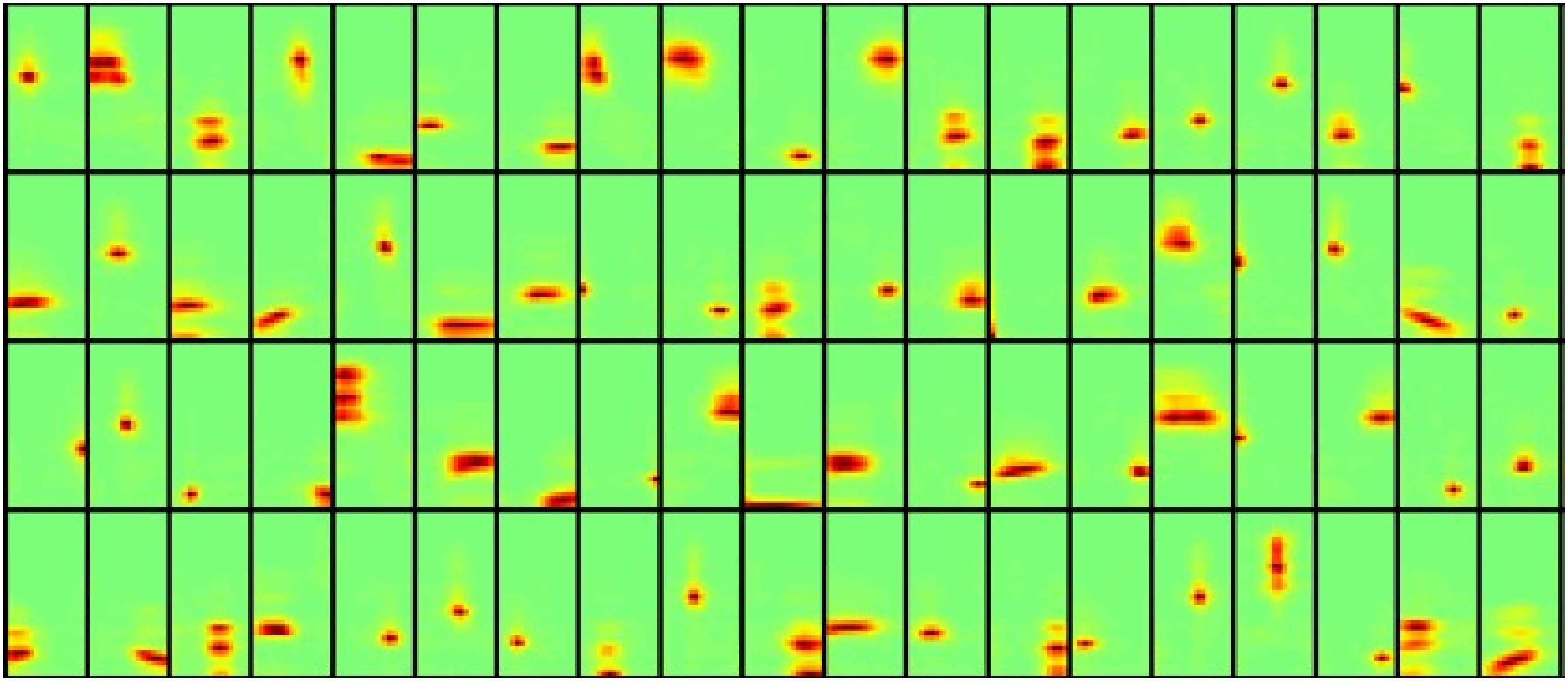


Apply non-linear model:

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$



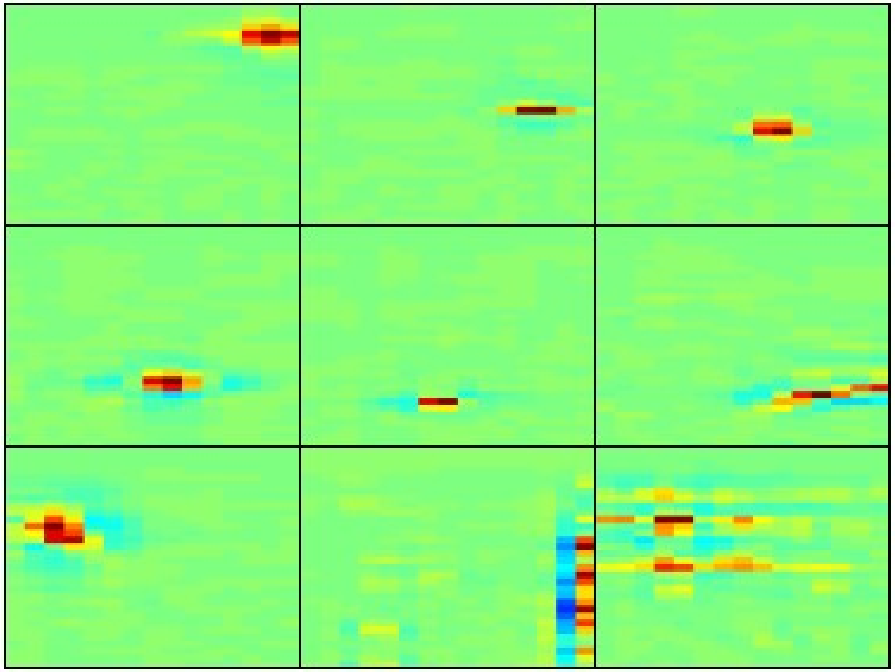
Encoding of Acoustic Scenes



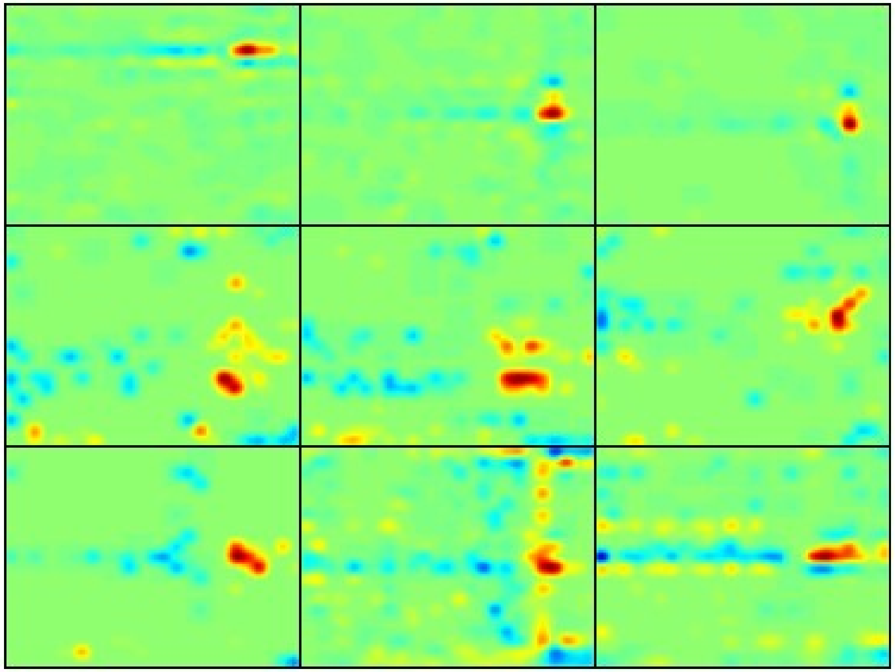
generative fields (selection of H=1000 fields)

Currently ongoing work (Univ. Oldenburg / UC Berkeley / Univ. Cambridge / TU Berlin)

Encoding of Acoustic Scenes



STRFs learned by the model



Ferret A1 recordings

Both estimated using (regularized) reverse correlation.

Currently ongoing work (Univ. Oldenburg / UC Berkeley / Univ. Cambridge / TU Berlin)

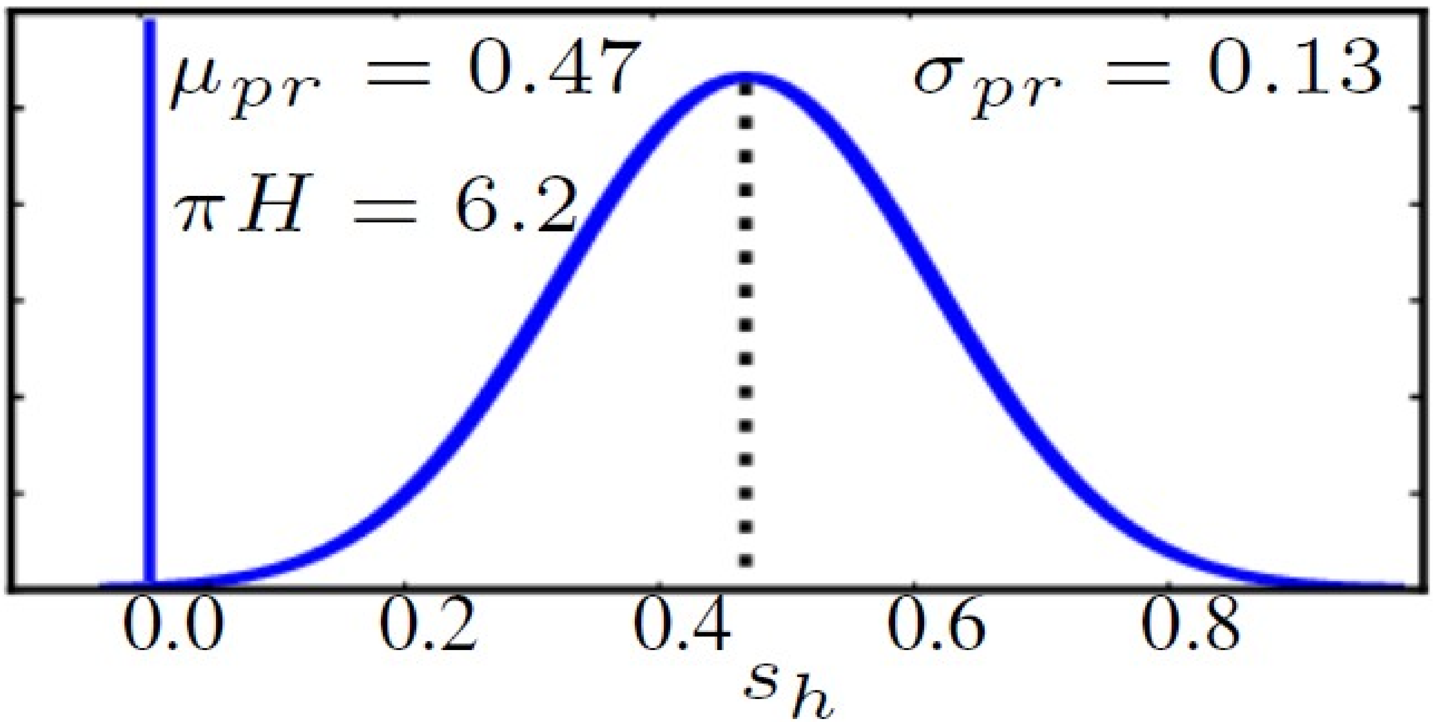
Example: Spike-and-Slab Sparse Coding (GSC)

$$p(\vec{s}\,|\Theta) \;=\; \mathcal{B}(\vec{s};\vec{\pi}) = \prod_{h=1}^{H} \pi_h^{s_h}\,(1-\pi_h)^{1-s_h}$$

$$p(\vec{z}\,|\Theta) \;=\; \mathcal{N}(\vec{z};\,\vec{\mu},\Psi)$$

$$p(\vec{y}\,|\,\vec{s},\vec{z},\Theta) = \mathcal{N}(\vec{y};\,W(\vec{s}\odot\vec{z}),\Sigma)$$

Titsias, Lazaro-Gredilla, *NIPS* '11
Sheikh, Lücke, *LVA* '12
Goodfellow et al., *ICML* '12
Shelton et al., *NIPS* '12
… and more



Example: Gaussian Sparse Coding (GSC)

$$p(\vec{s}\,|\Theta) \;=\; \mathcal{B}(\vec{s};\vec{\pi}) = \prod_{h=1}^{H} \pi_h^{s_h}\,(1-\pi_h)^{1-s_h}$$

$$p(\vec{z}\,|\Theta) \;=\; \mathcal{N}(\vec{z};\,\vec{\mu},\Psi) \qquad p(\vec{y}\,|\,\vec{s},\vec{z},\Theta) = \mathcal{N}(\vec{y};\,W(\vec{s}\odot\vec{z}),\Sigma)$$

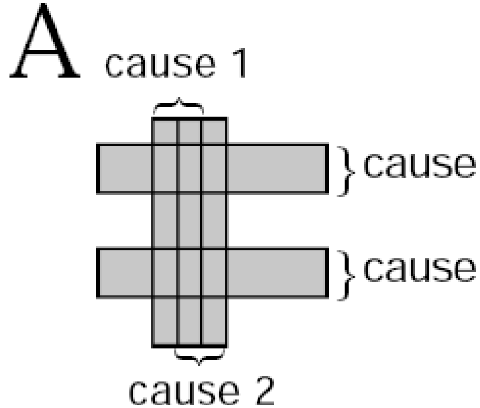
Noise	PSNR (dB)						
	Noisy img	MTMKL$^{exp.}$	K-SVD$^{mis.}$	*K-SVD^{match}	Beta pr.	GSC (H=64)	GSC (H=256)
σ=15	24.59	**34.29**	30.67	34.22	34.19	32.68 (H'=10,γ=8)	33.78 (H'=18,γ=3)
σ=25	20.22	31.88	31.52	32.08	31.89	31.10 (H'=10,γ=8)	**32.01** (H'=18,γ=3)
σ=50	14.59	28.08	19.60	27.07	27.85	28.02 (H'=10,γ=8)	**28.35** (H'=10,γ=8)

Lücke, Sheikh, *LVA* 2012;
Sheikh, Shelton, Lücke, *JMLR* 2014.

GSC is state-of-the-art in denoising.

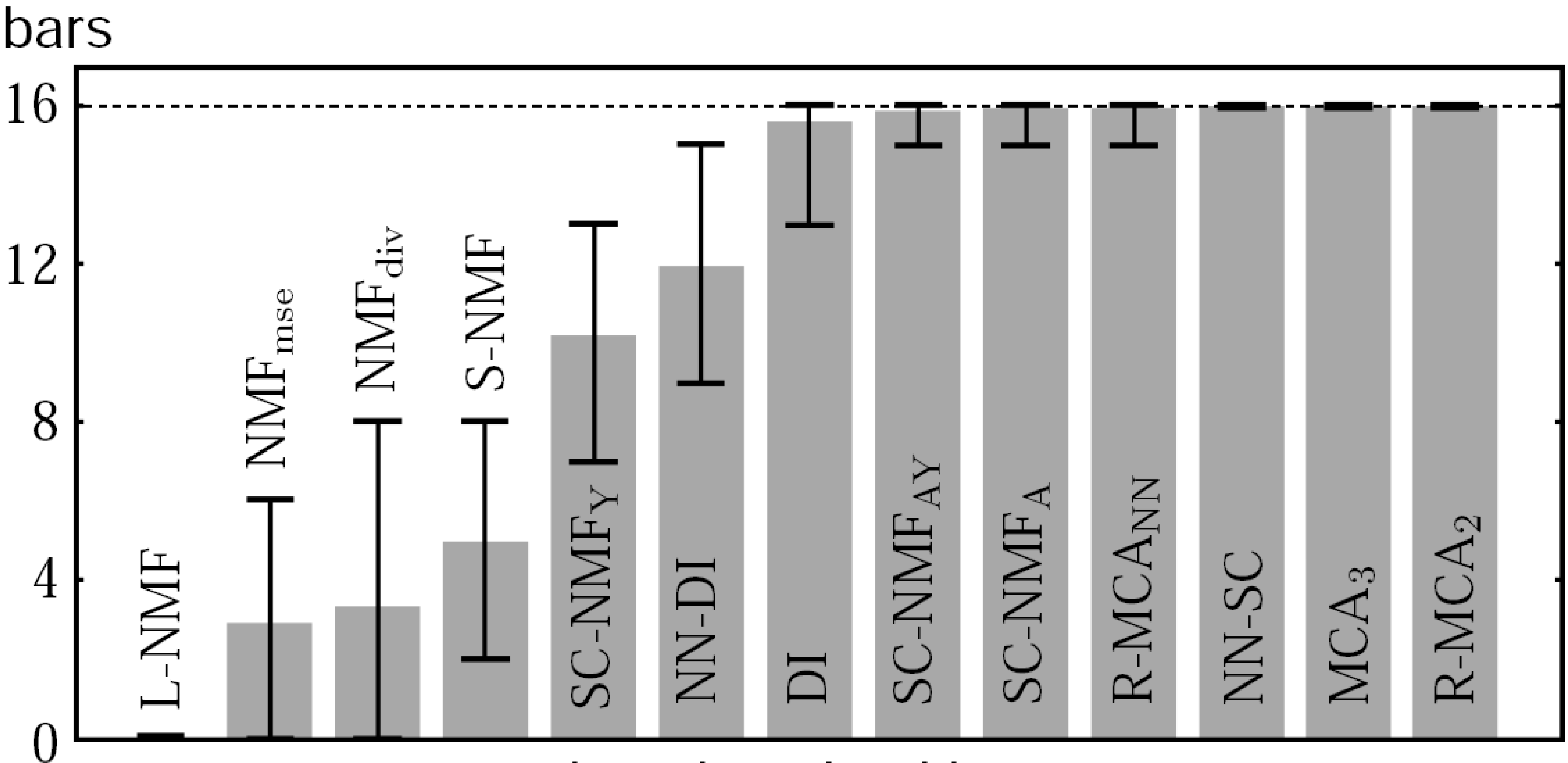
… but denoising is just one tasks.

Maximal Causes



Maximal Causes Analysis (MCA) is state-of-the-art

Source:
Hoyer, *J Mach Learning Res*, 2004
Spratling, *J Mach Learning Res*, 2006
Lücke/Sahani, *J Mach Learning Res*, 2008



Lücke and Sahani, *JMLR*, 2008

Selection of Statistical Models

noisy-OR
J. Bornschein

occlusion
M. Henniges

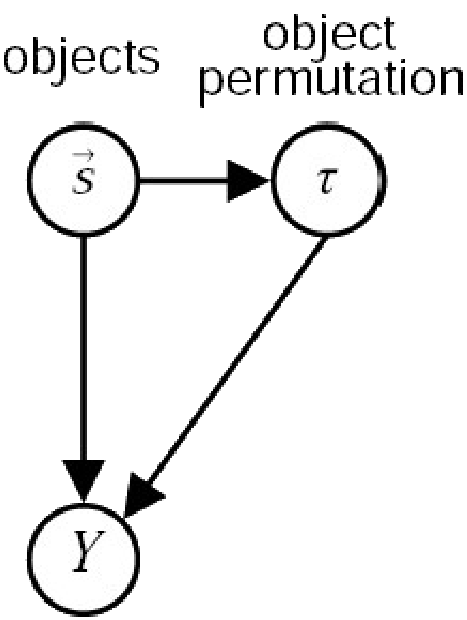
exclusion
Z. Dai

mixtures
C. Keck,
S. Sheikh,
C. Savin

linear SC
A.-S. Sheikh,
M. Henniges
J. Shelton

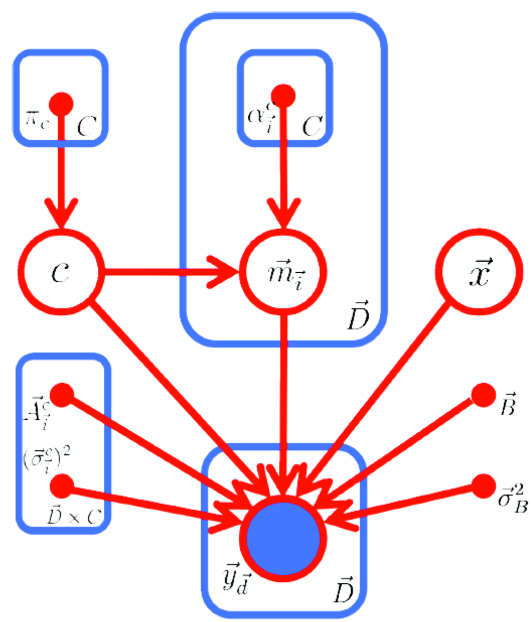
Mask Feature

$$\vec{T}_d(S;\Theta) = W_{h_o d}\,\vec{T}_{h_o}$$
$$h_o = \mathrm{argmax}_h\{\tau(h)W_{hd}\}$$



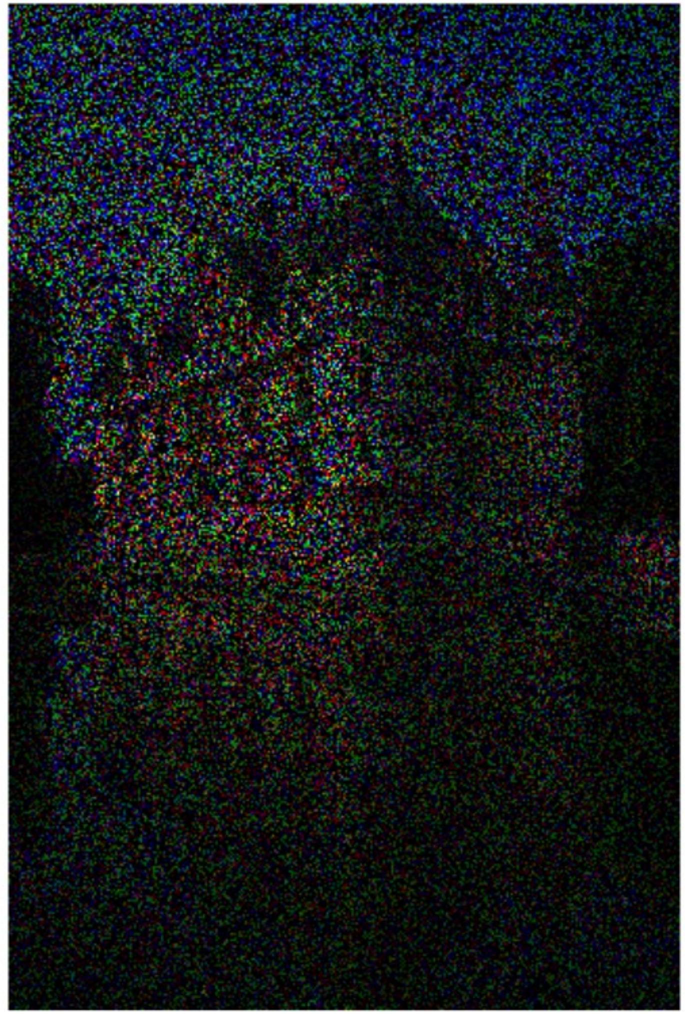
Graphical Model

Lücke et al., *NIPS* 2009



Example App: Inpainting

Original image



80% lost pixels

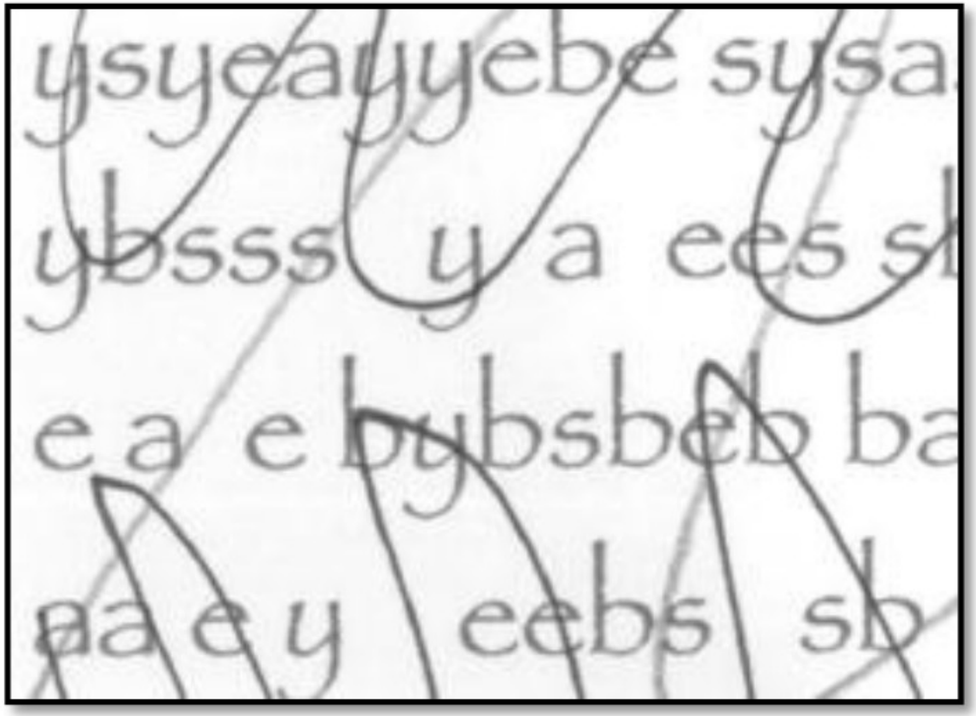
reconstruction

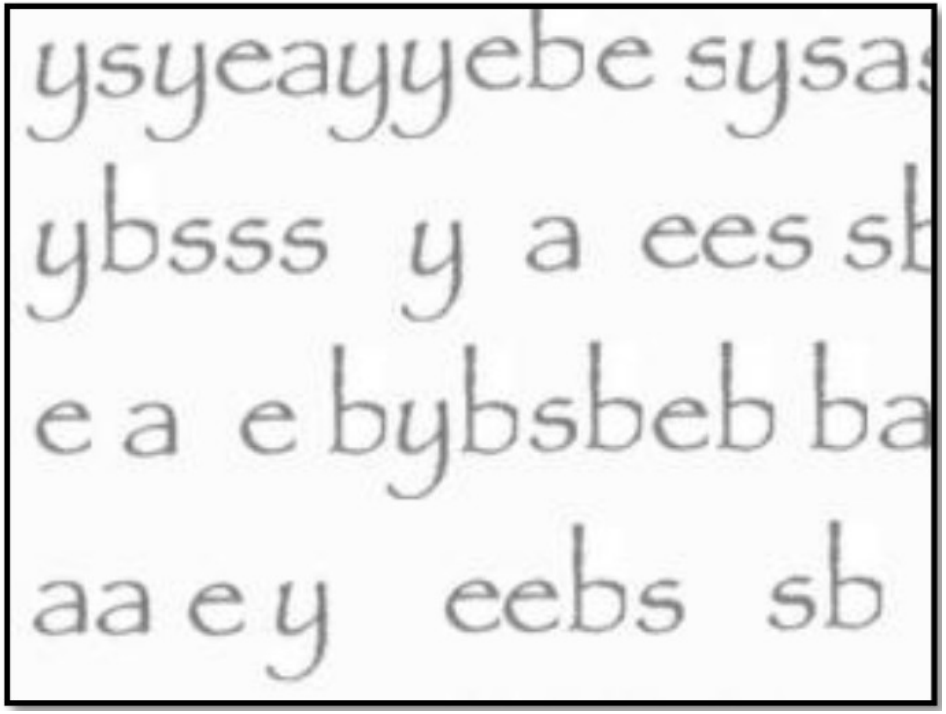
spike-and-slab prior

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

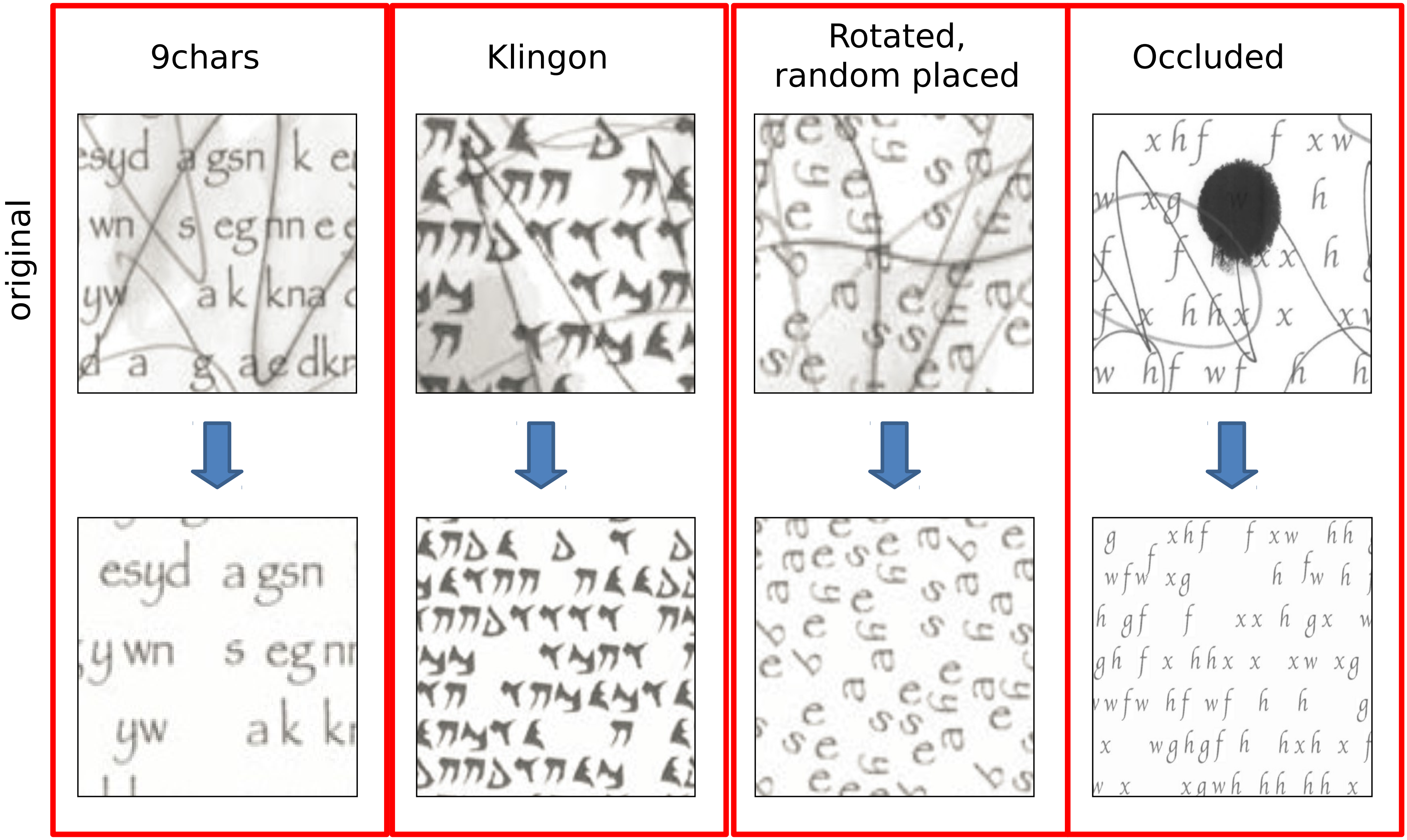
Shelton et al.,
NIPS 2012

Example: Structured Noise Removal

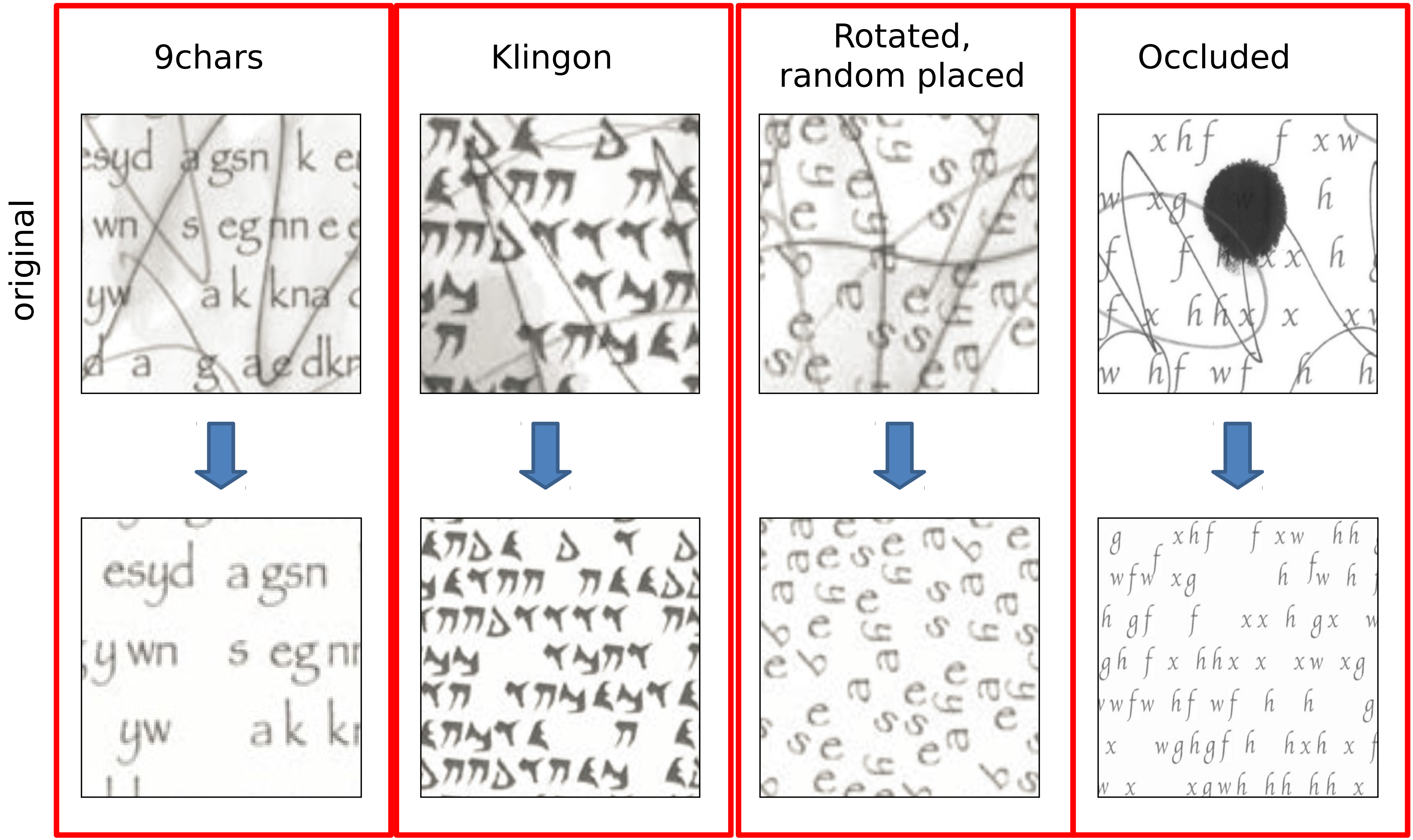




Example: Structured Noise Removal



Example: Structured Noise Removal



Dai & Lücke, *CVPR* 2012, oral presentation, Google award.
Dai & Lücke, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2014.

Thanks to:

Cluster of Excellence Hearing4all
Deutsche Forschungsgemeinschaft

Non-linear Probabilistic Models for Representational Recognition and Unsupervised Learning in Vision
Deutsche Forschungsgemeinschaft

SPONSORED BY THE
Federal Ministry of Education and Research

Bernstein Focus Neurotechnology Frankfurt
Bundesministerium für Bildung und Forschung

LOROC
Honda Research Institute Europe

Thanks to:

Christian Keck
Jörg Bornschein
Marc Henniges
Dennis Forster
Abdul Saboor Sheikh
Jacquelyn Shelton
Jörg Lücke
Zhenwen Dai
Georgios Exarchakis

Project Researchers
Univ. Oldenburg, TU Berlin, Univ. Frankfurt

Thanks to:

Richard Turner
Cristina Savin (now IST Austria)
Cambridge University, UK

Roland Memisevic
Jörg Bornschein
University of Montreal, CA

Maneesh Sahani
Gatsby Unit, UCL, UK

Bruno Olshausen
Nicol Harper
Berkeley University, US

Jörg Lücke
University of Oldenburg

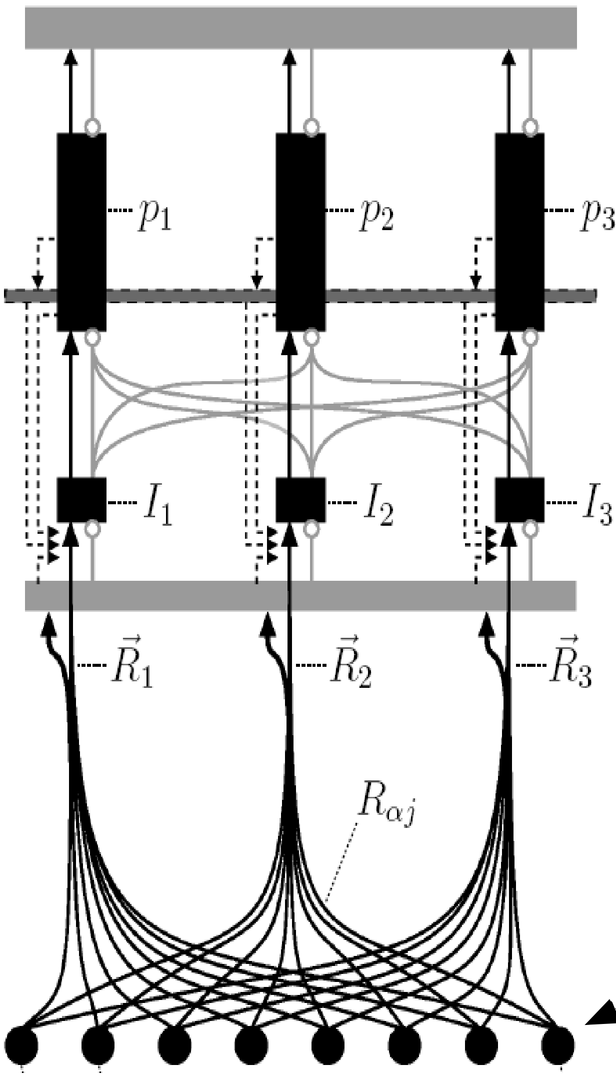
Julian Eggert
Honda-RI, Europe

Pietro Berkes
Enthought, Ltd., UK
Brandeis Univ., USA

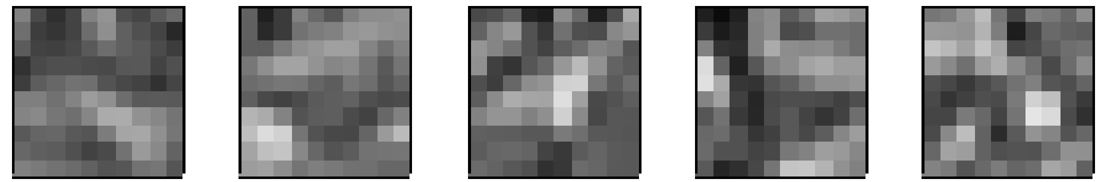
Marc-Thilo Figge
Universität Jena

Thank you.

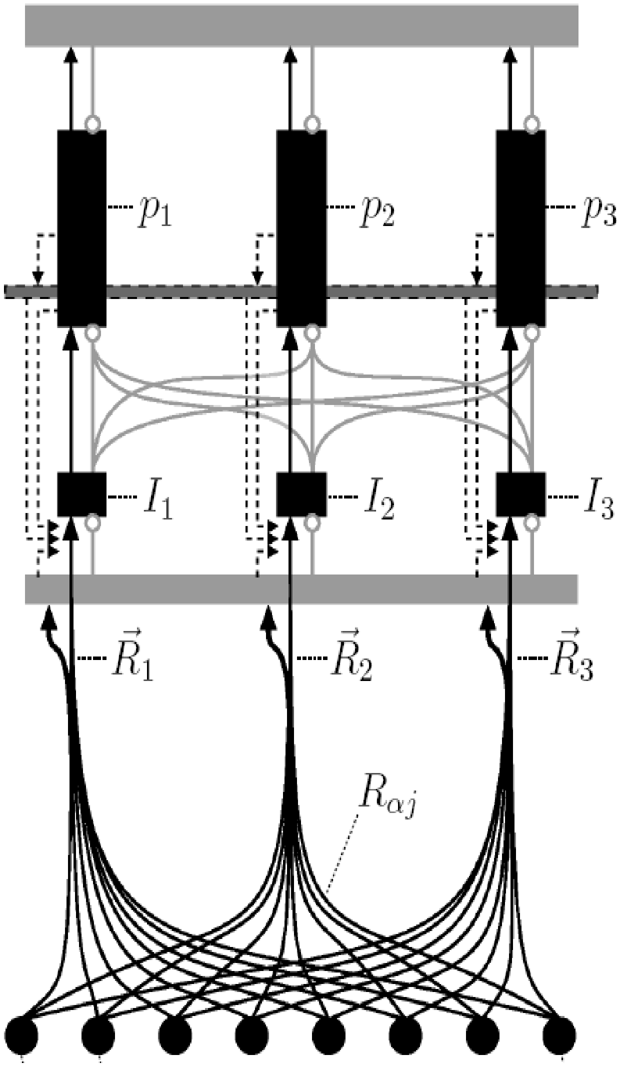
Cortical Circuits

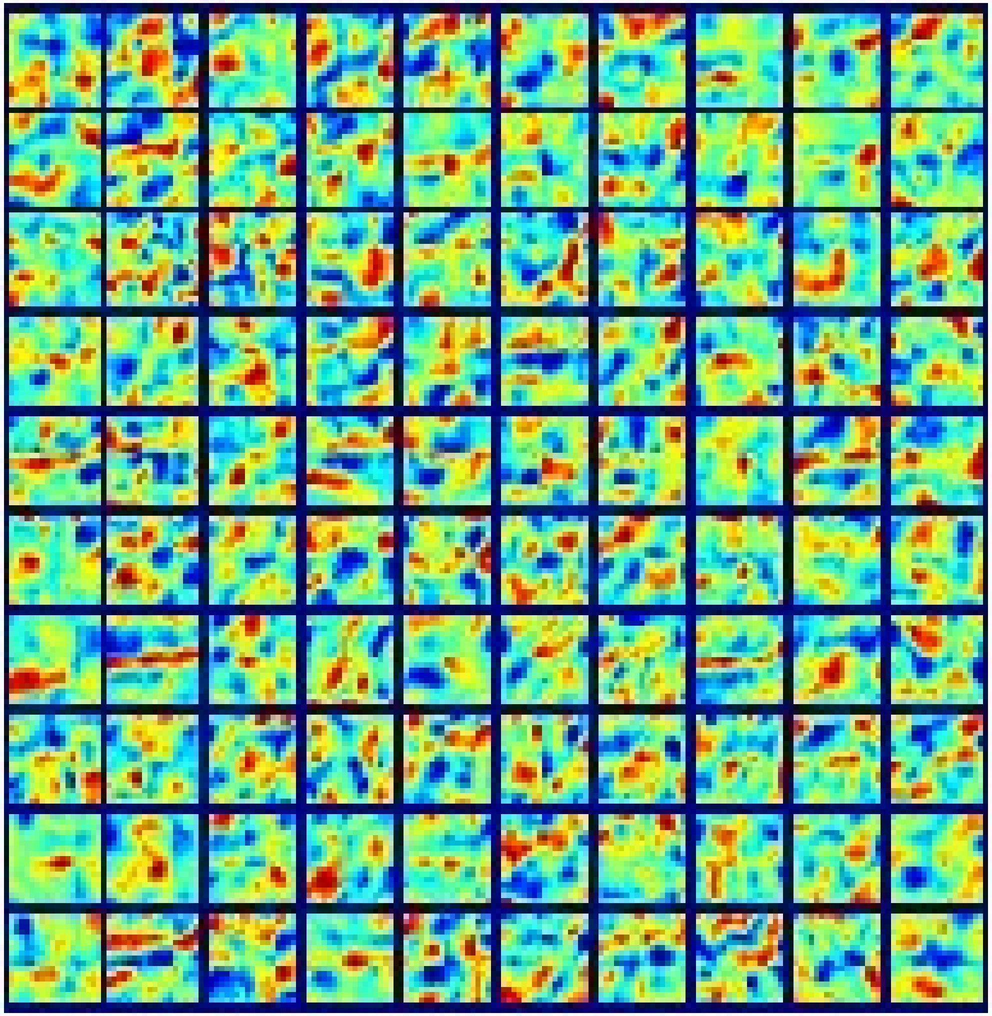


Use Natural Image Patches as Input



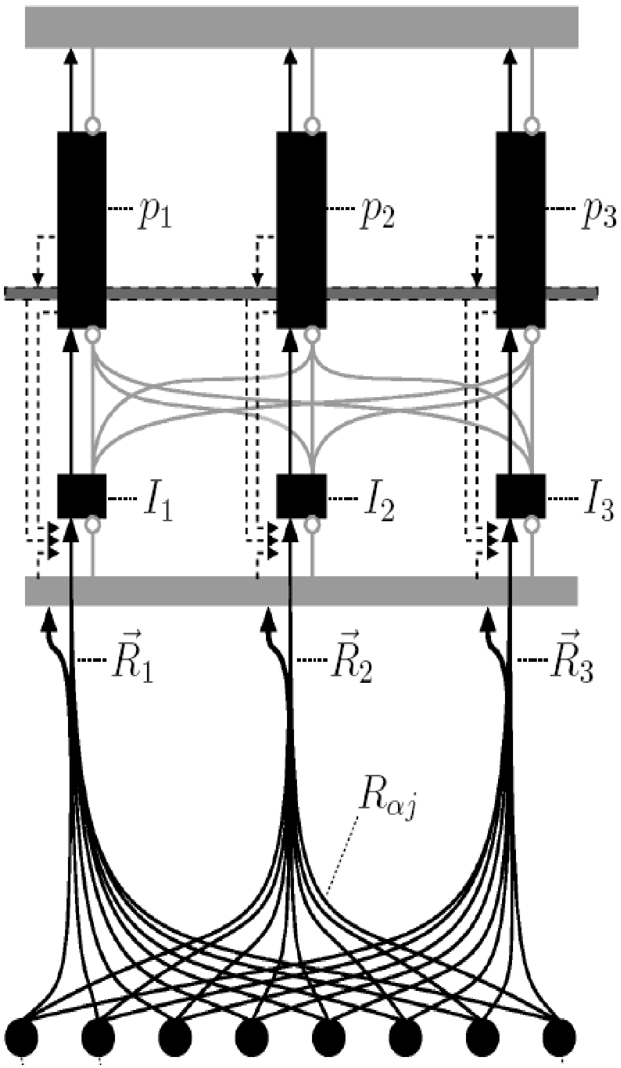
Cortical Circuits

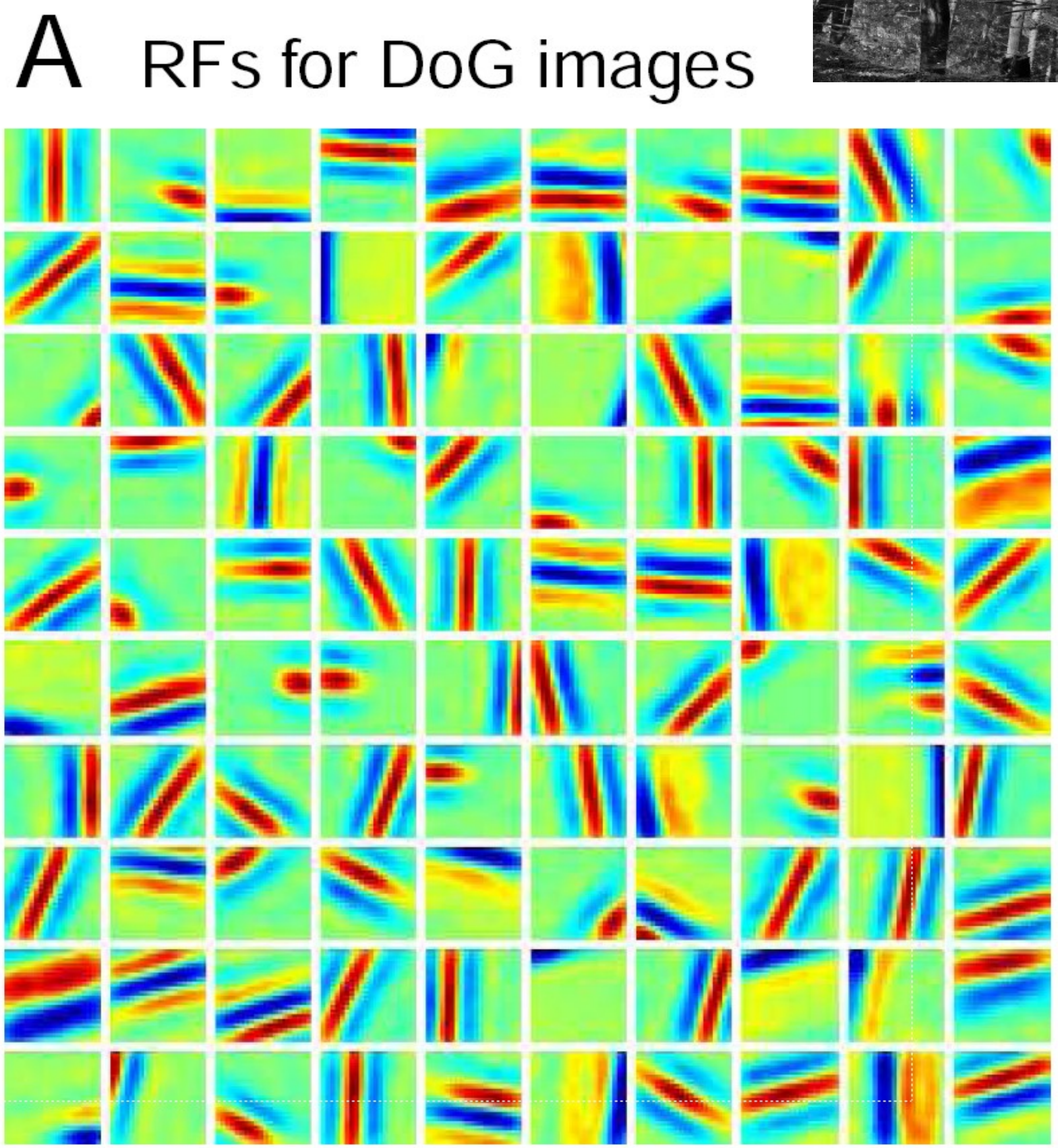




Numerical simulations of stochastic and non-linear differential equations.

Cortical Circuits



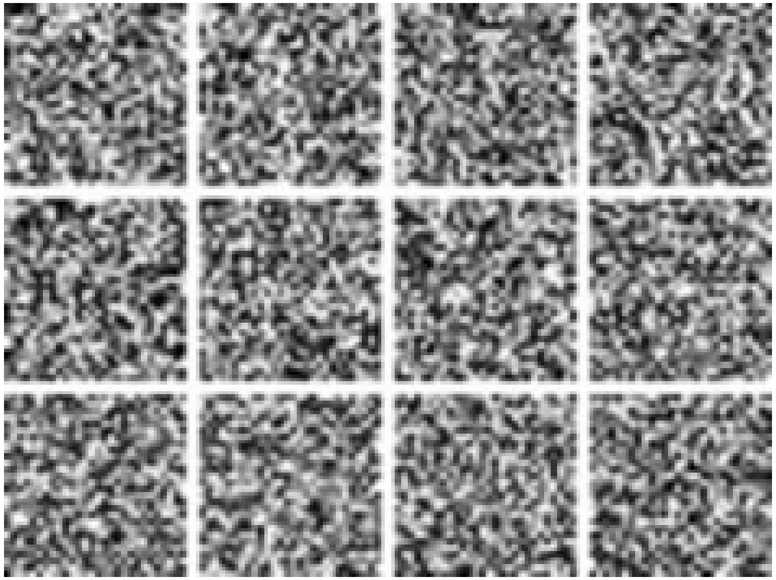


Lücke, *Neural Computation*, 2009.

Application NMF

$$W \leftarrow W \odot \frac{\sum_n \vec{y}^{(n)} < \vec{s}^T >_{q_n}}{\sum_n W < \vec{s}\vec{s}^T >_{q_n}}$$

$$\left\langle g(\vec{s},\Theta^{\mathrm{old}}) \right\rangle_{q_n} \approx \frac{\sum_{\vec{s}\in\mathcal{K}_n} p(\vec{s},\vec{y}^{(n)}\,|\,\Theta^{\mathrm{old}})\; g(\vec{s},\Theta^{\mathrm{old}})}{\sum_{\vec{s}'\in\mathcal{K}_n} p(\vec{s}',\vec{y}^{(n)}\,|\,\Theta^{\mathrm{old}})}$$

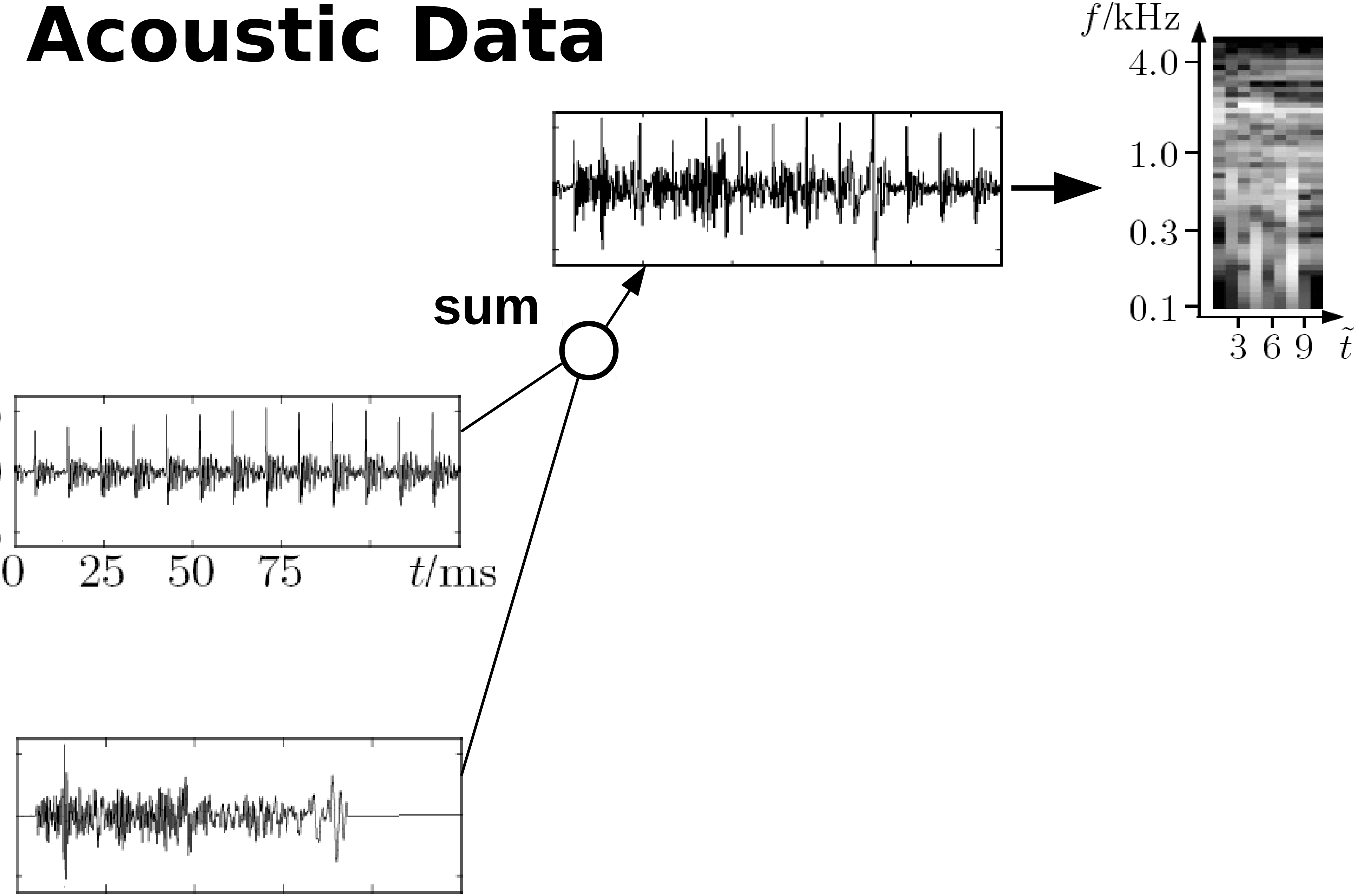


Application to MNIST data.

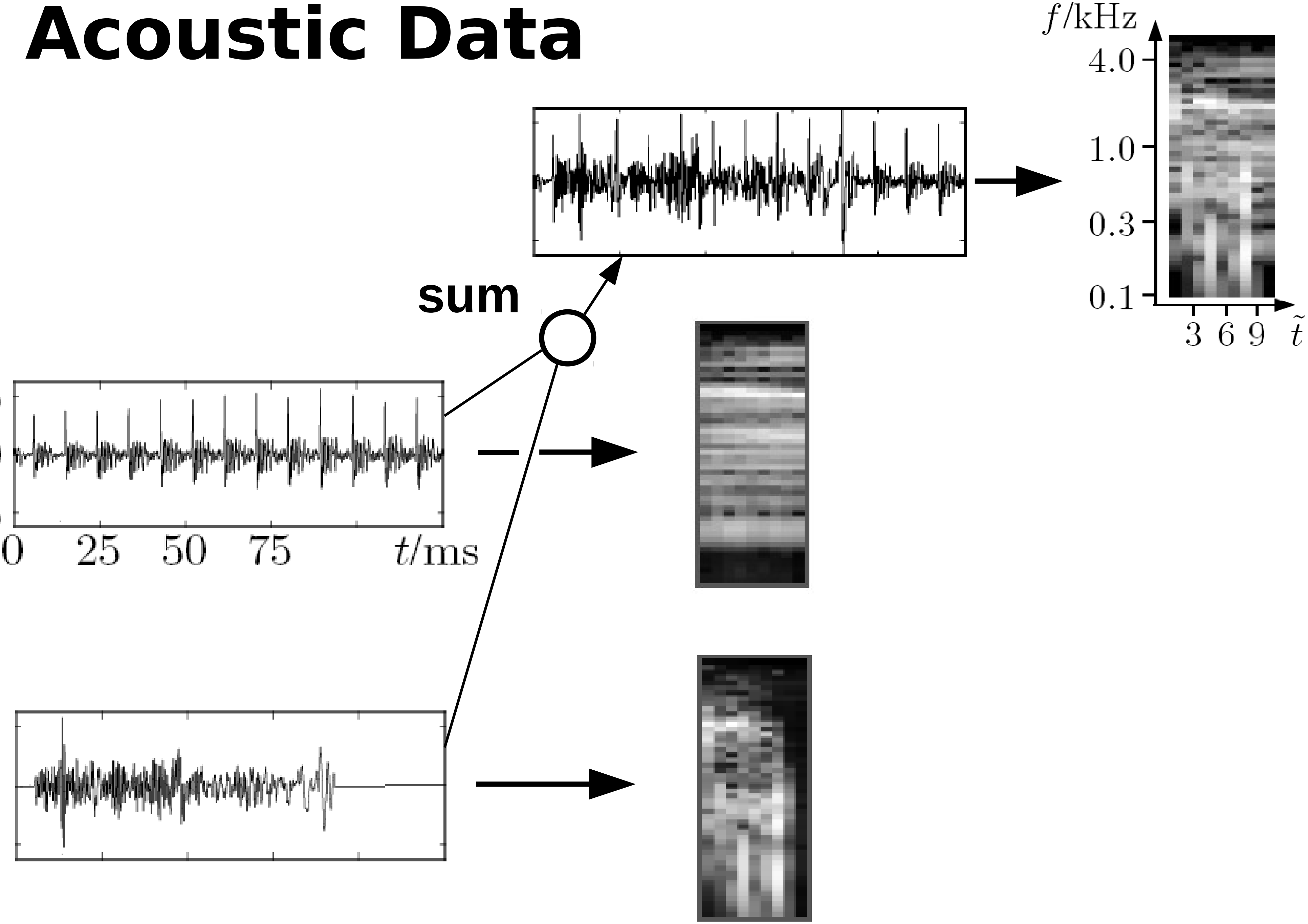
The model can be constraint by allowing only for positive $\vec{s}$ and positive W.

We obtain a generative version of NMF (with binary latents).

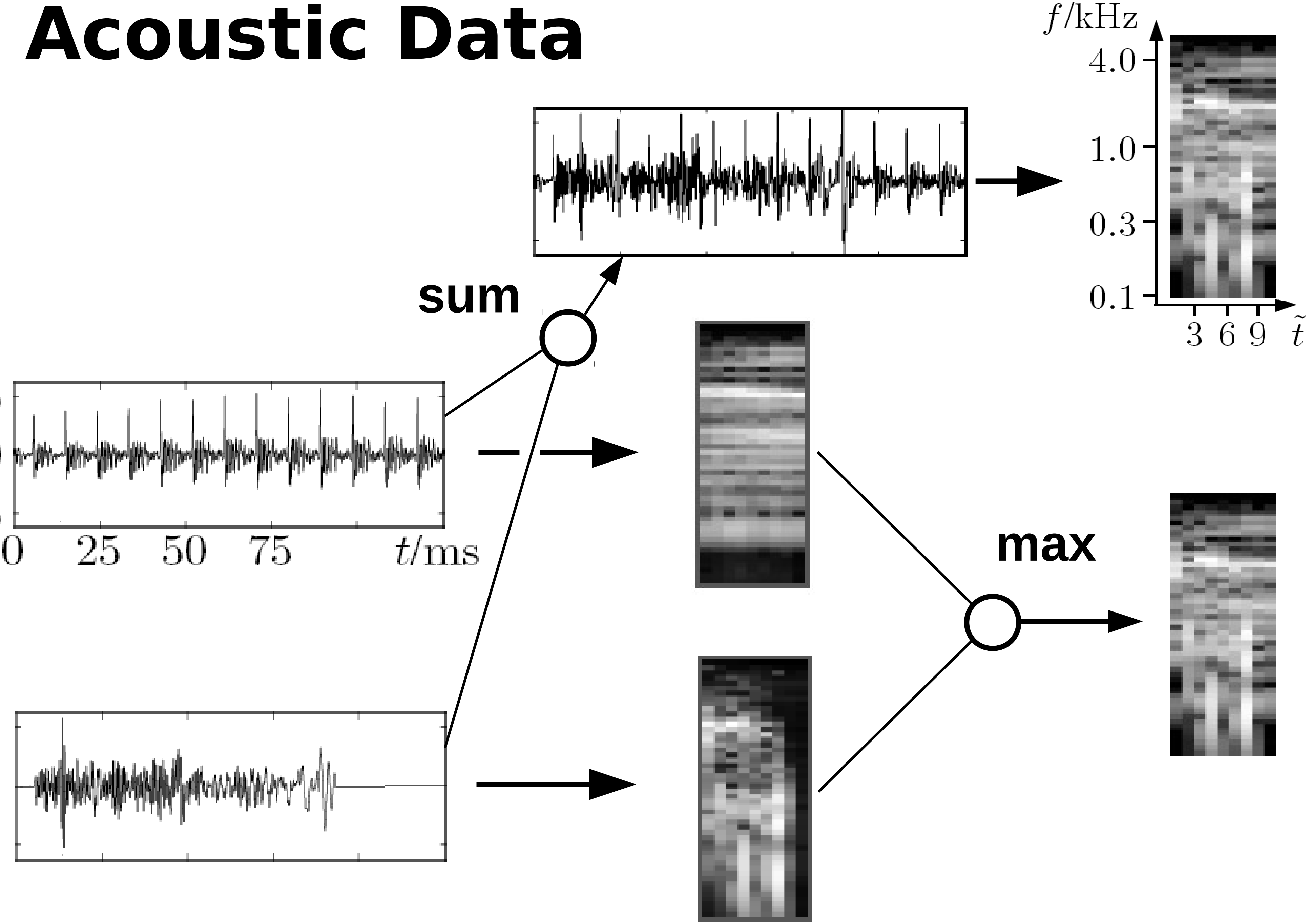
Acoustic Data



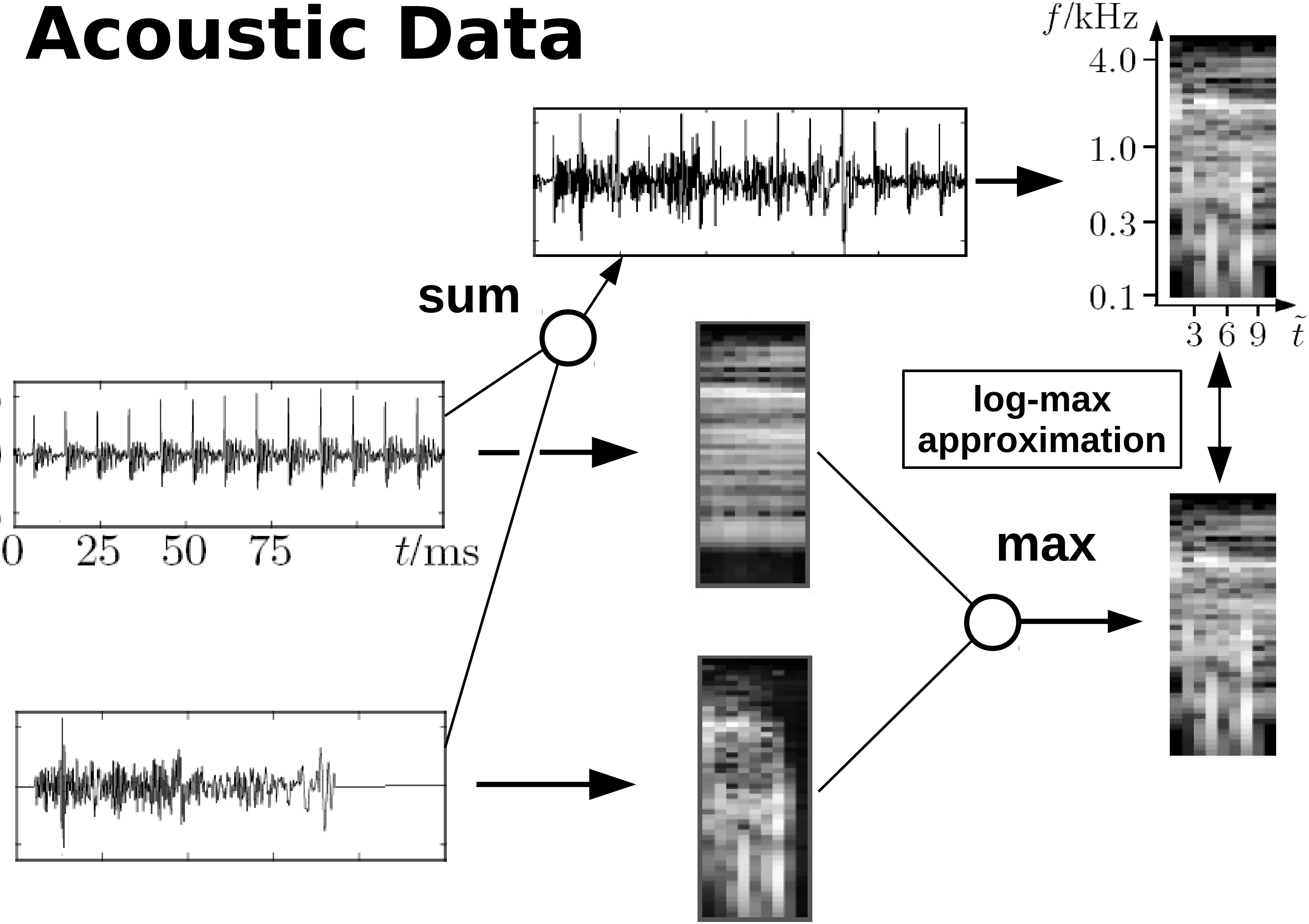
Acoustic Data



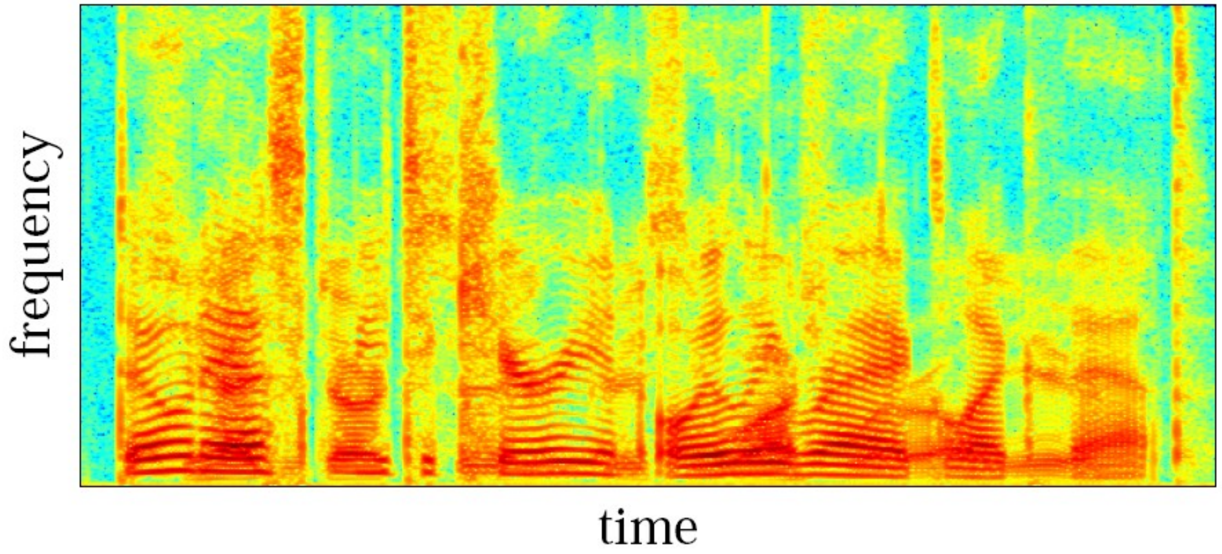
Acoustic Data



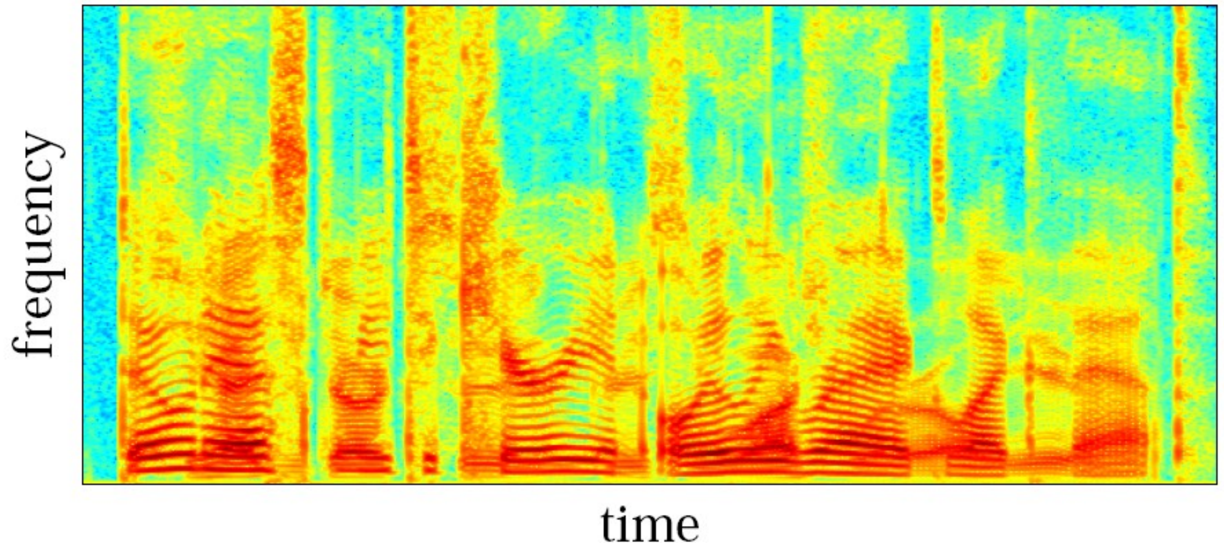
Acoustic Data



Acoustic Data



Log-spectrogram
of a mixture of
two sound sources.



Max of the two individual
log-spectrograms.

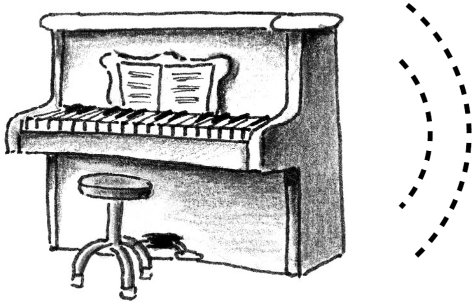
Source: Roweis, 2004

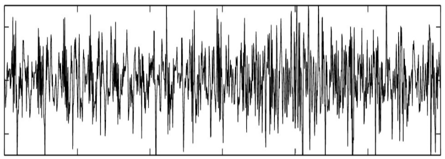
Log-max approximation (Moore, 1983)

Acoustic Data

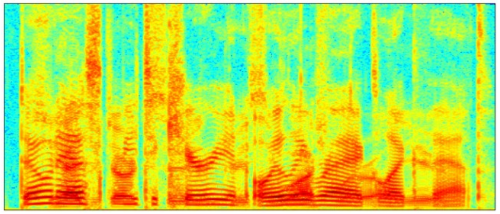
$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

$$\vec{y} = f\left(s_{1:H}\,\vec{W}_{1:H}\right) + \vec{\eta}$$





$\vec{y}$

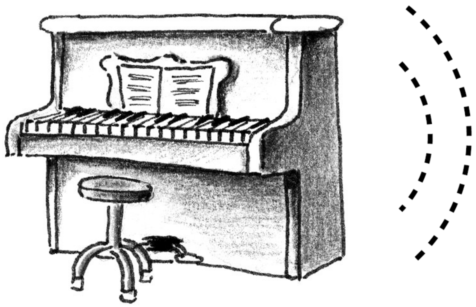


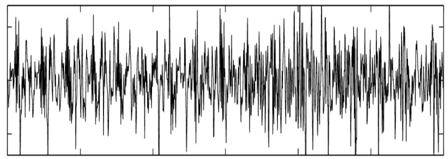
$\vec{y}$

Acoustic Data

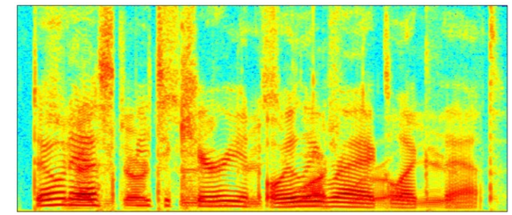
$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$





$\vec{y}$



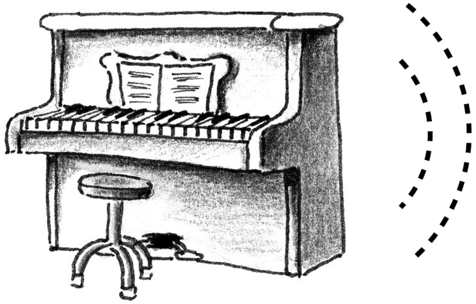
$\vec{y}$

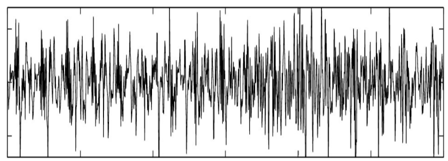
Bornschein et al., *PLOS CB* 2013
Shelton et al., *NIPS* 2012
Puertas, Bronschein, Lücke, *NIPS 2010*
Lücke, Sahani, *J Mach Learn Res 2008*
...
Roweis, *Eurospeech 2003*
Roweis*, NIPS 2002*
Varga, Moore*, ICASSP 1990*

Acoustic Data

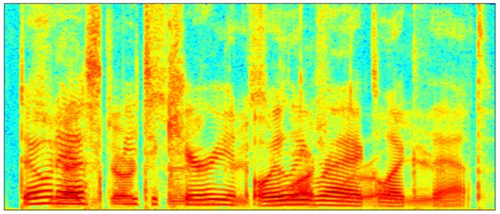
$$\vec{y} = \sum_h s_h \vec{W}_h + \vec{\eta}$$

$$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$$

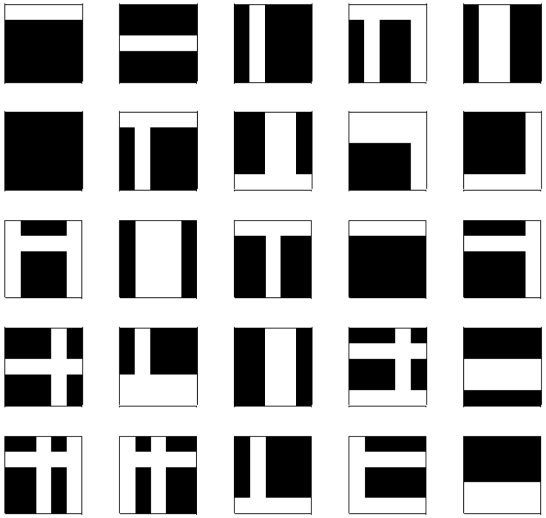




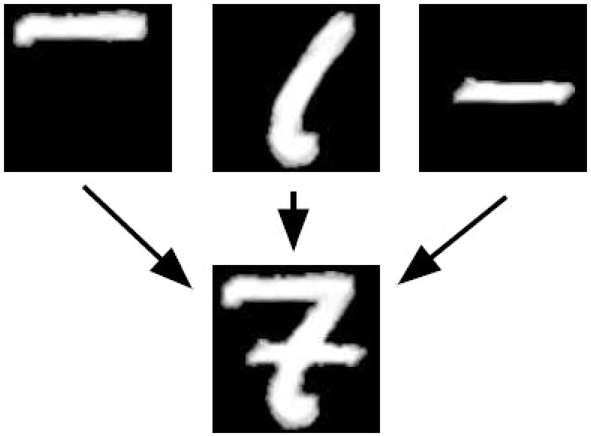
$\vec{y}$



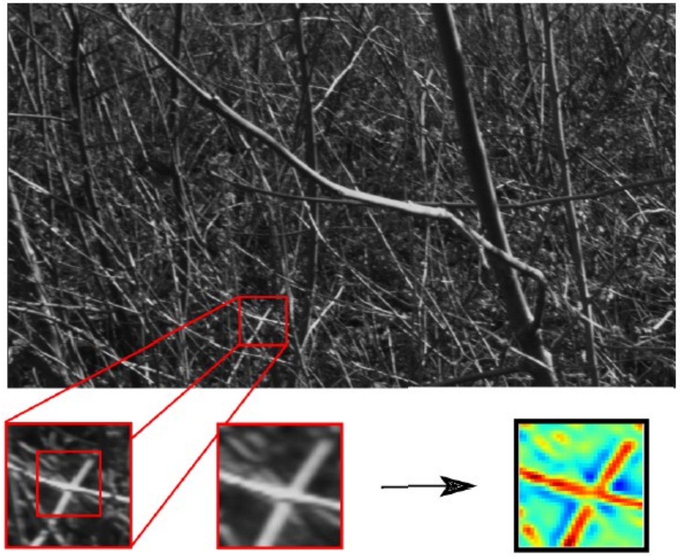
$\vec{y}$



The Bars Test.
Földiák, 1990



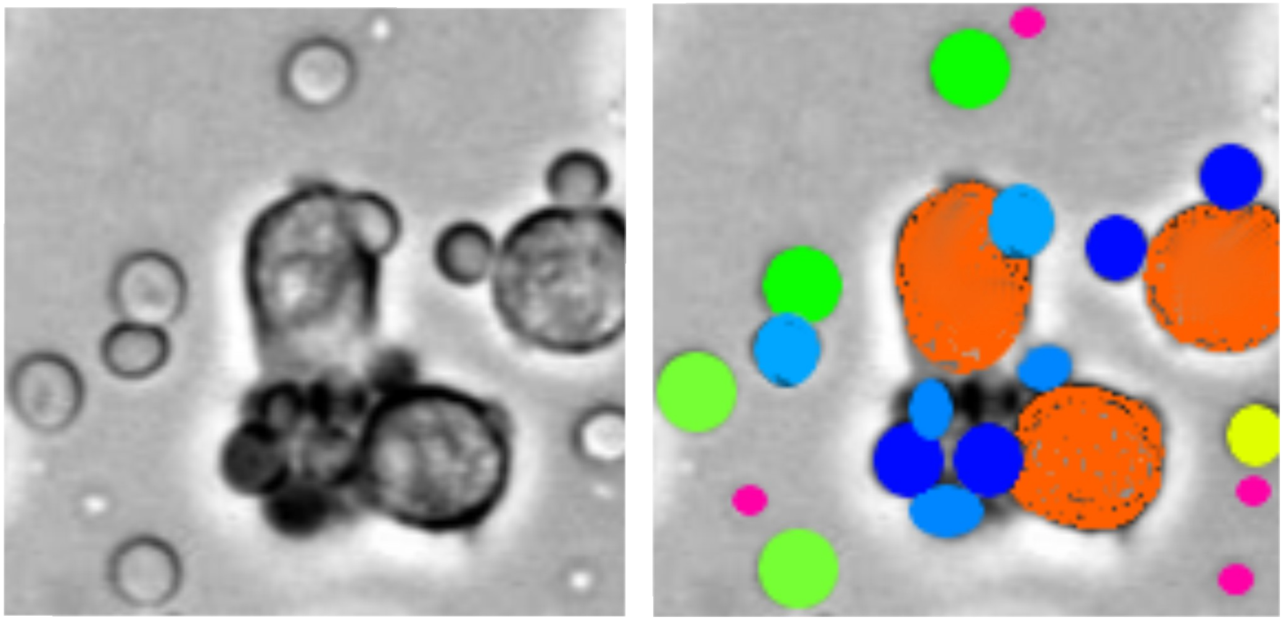
hand-written digits
(e.g., MNIST)



Natural image patches

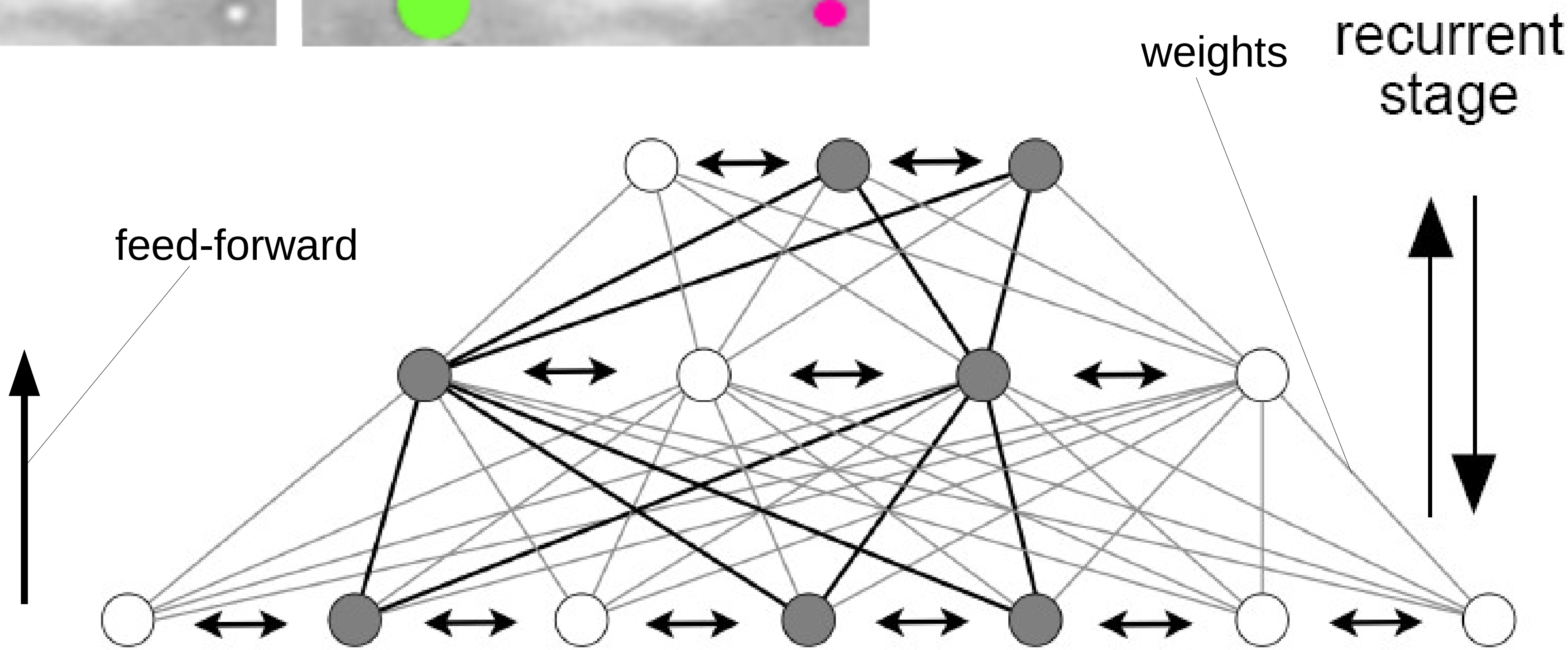
10s hidden **10s-100s hidden** **100s-1000s hidden**

Other projects

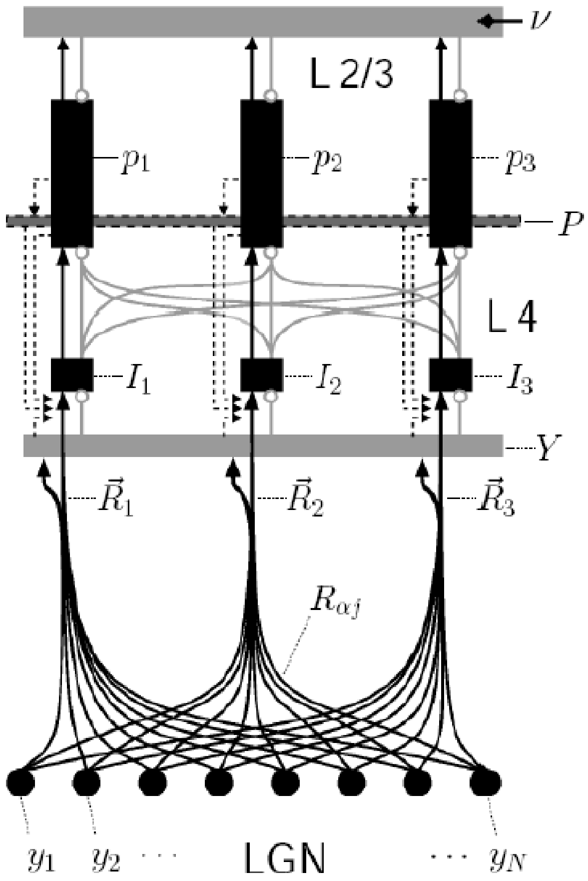


Microscopy
Image Analysis

Deep Learning Architectures
for Pattern Recognition
Keck, Savin, Lücke
PLOS Comp Bio, 2012

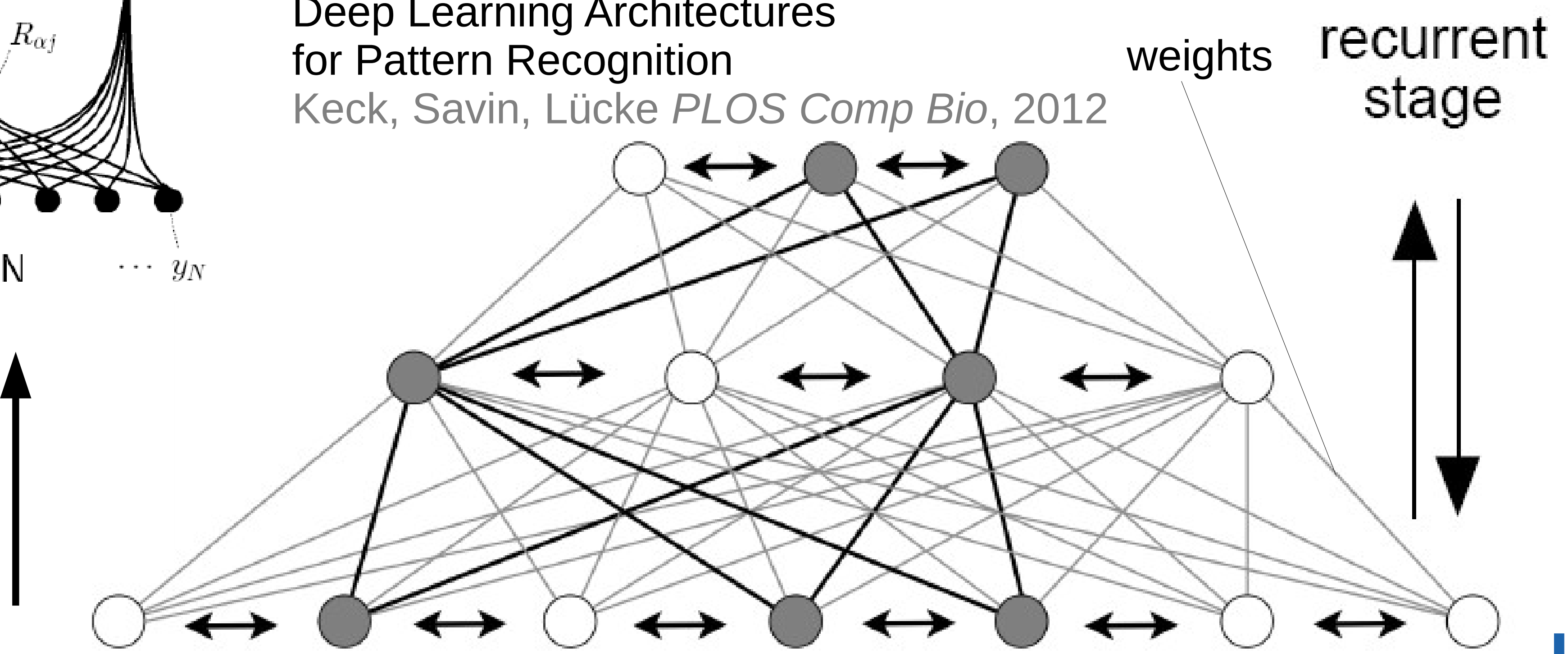


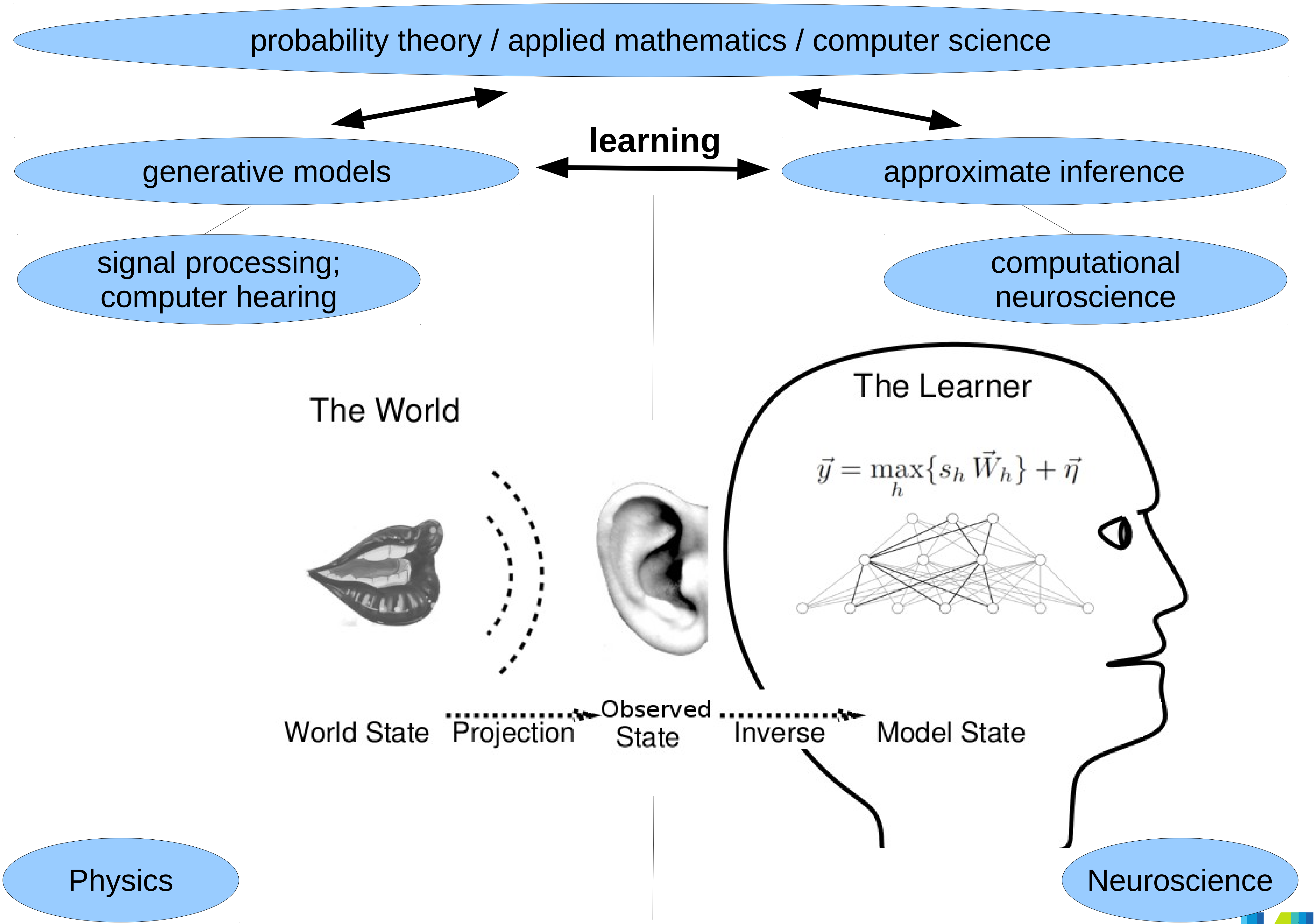
Other projects



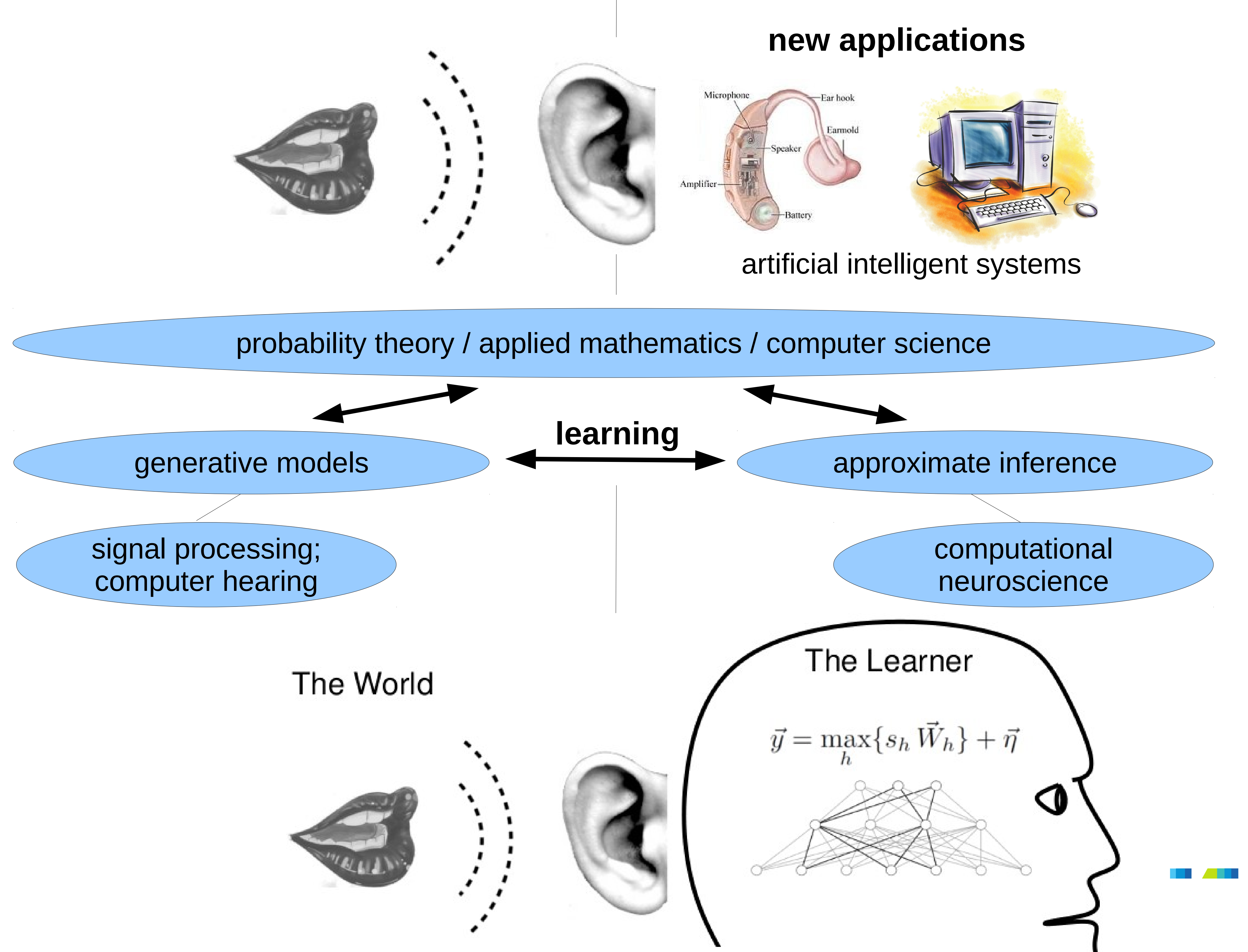
Deep Learning Architectures
for Pattern Recognition
Lücke, *Neural Comp* 2009
Lücke, *ICANN* 2005-2007
Lücke, *Neural Networks* 2004
Lücke, Malsburg, *Neural Comp* 2004
...

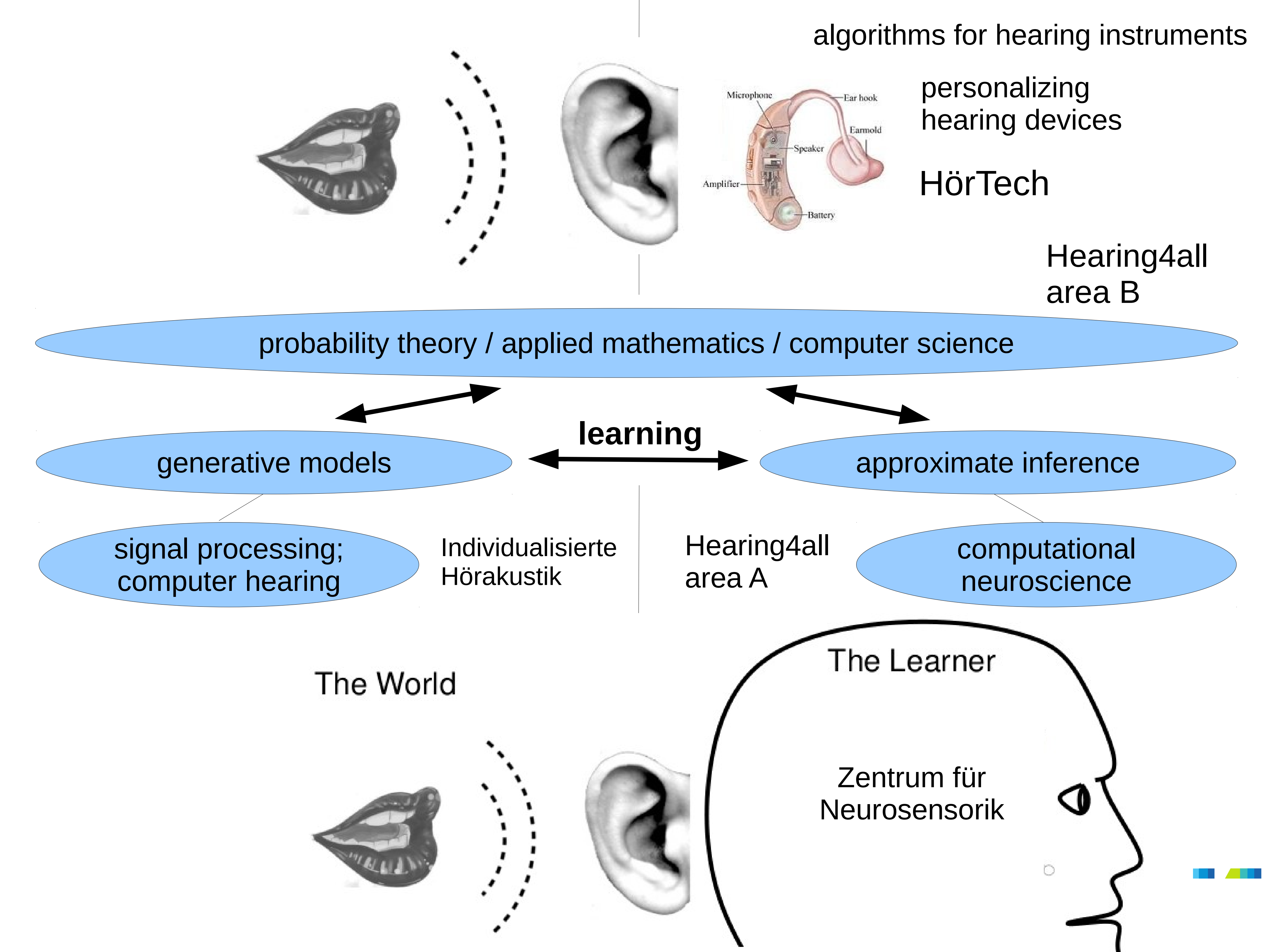
Deep Learning Architectures
for Pattern Recognition
Keck, Savin, Lücke *PLOS Comp Bio*, 2012





probability theory / applied mathematics / computer science
learning
generative models
approximate inference
signal processing;
computer hearing
computational
neuroscience
The World
The Learner
$\vec{y} = \max_h \{ s_h \vec{W}_h \} + \vec{\eta}$
World State
Projection
Observed
State
Inverse
Model State
Physics
Neuroscience


new applications
Microphone
Ear hook
Earmold
Speaker
Amplifier
Battery
artificial intelligent systems
probability theory / applied mathematics / computer science
learning
generative models
approximate inference
signal processing;
computer hearing
computational
neuroscience
The World
The Learner
$\vec{y} = \max_{h}\{s_h \vec{W}_h\} + \vec{\eta}$

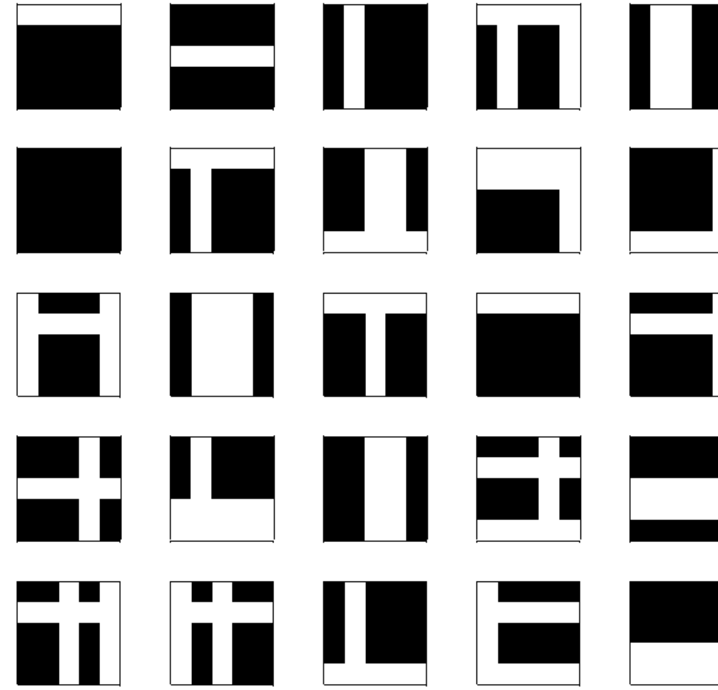

algorithms for hearing instruments
Microphone
Ear hook
Earmold
Speaker
Amplifier
Battery
personalizing hearing devices
HörTech
Hearing4all area B
probability theory / applied mathematics / computer science
learning
generative models
approximate inference
signal processing; computer hearing
Individualisierte Hörakustik
Hearing4all area A
computational neuroscience
The World
The Learner
Zentrum für Neurosensorik

Linear Causes

Linear generative models:

$$p(\vec{s}\,|\,\Theta) = \prod_h \cdots$$

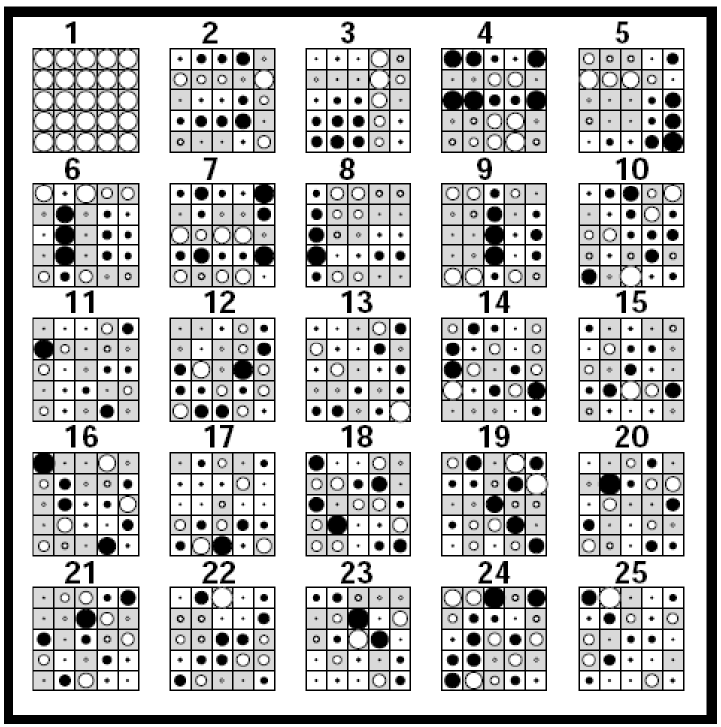
$$p(\vec{y}\,|\,\vec{s},\Theta) = \mathcal{N}(\vec{y}; \textstyle\sum_h s_h^{(n)}\,\vec{W}_h, \sigma^2\,\mathbb{1})$$



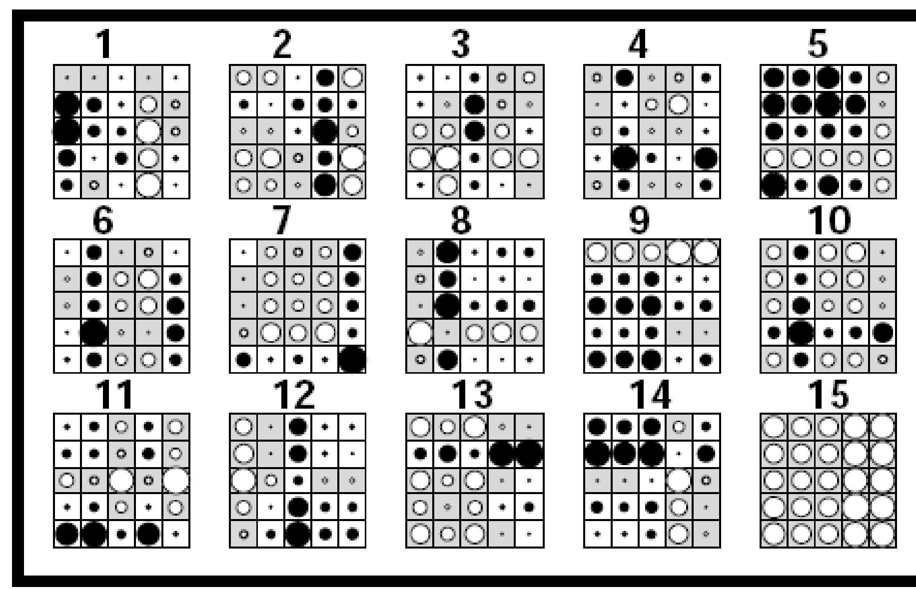
The Bars Test, Földiák, 1990

Obtain basis functions:

PCA



ICA 15



Source: Hochreiter & Schmidhuber, *Neural Comp*, 1999