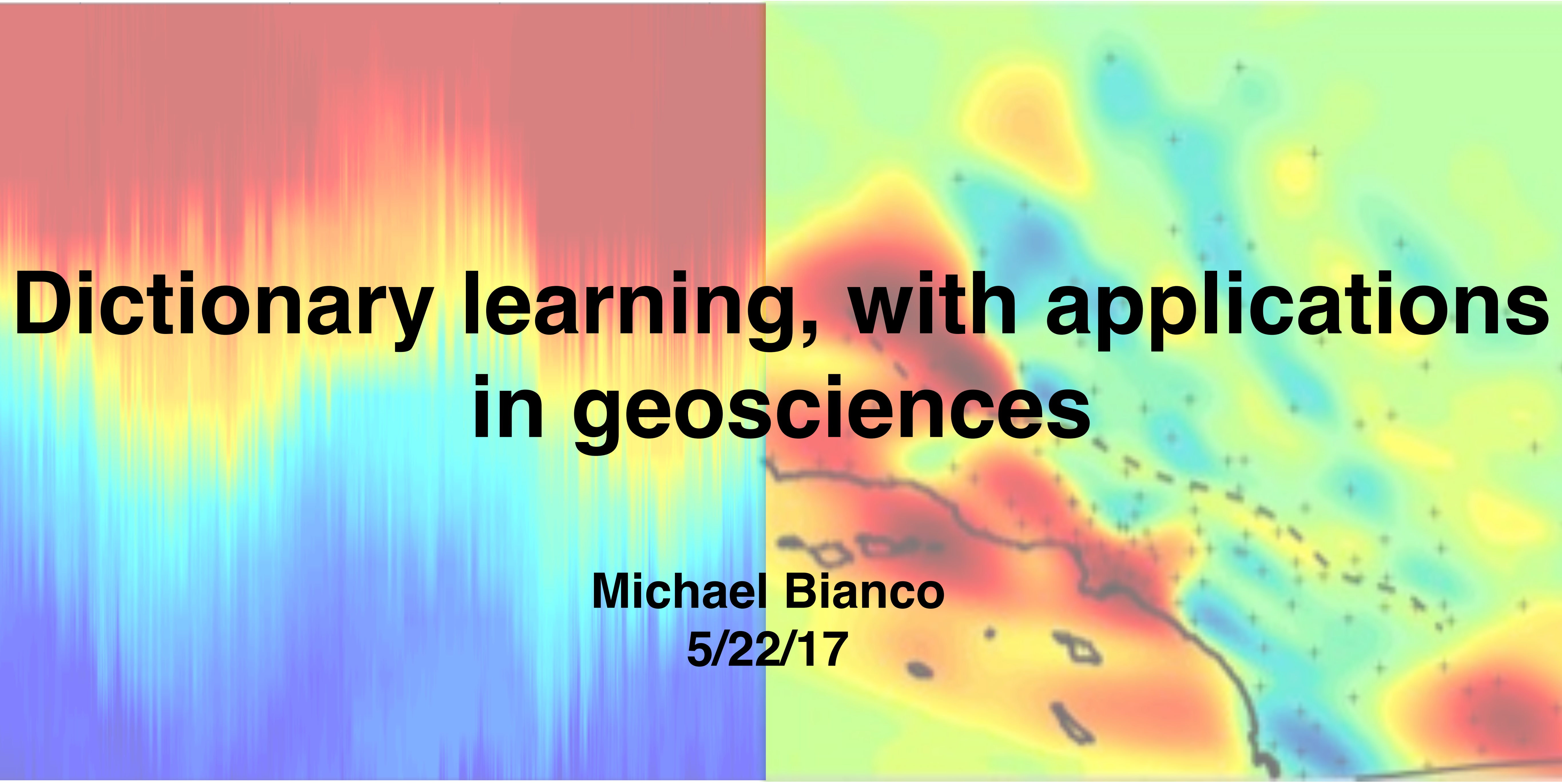


Dictionary learning, with applications in geosciences

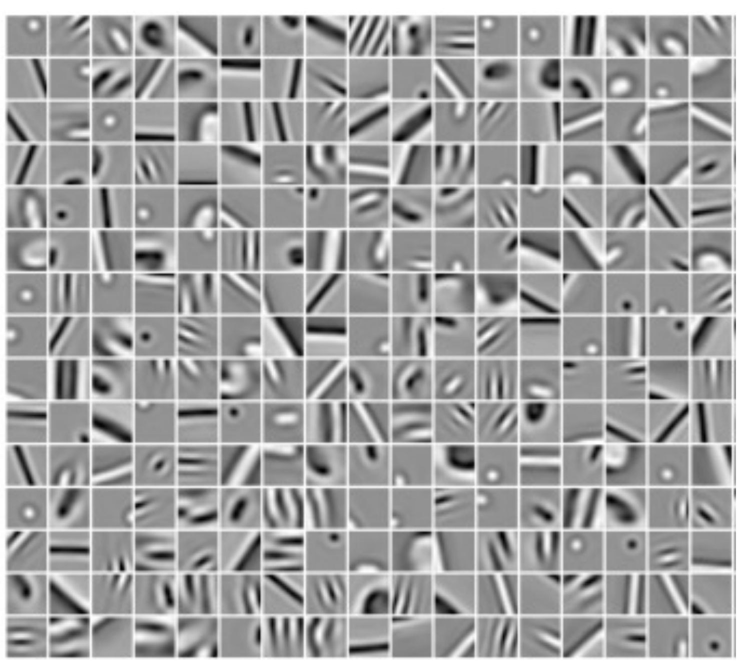
Michael Bianco

5/22/17

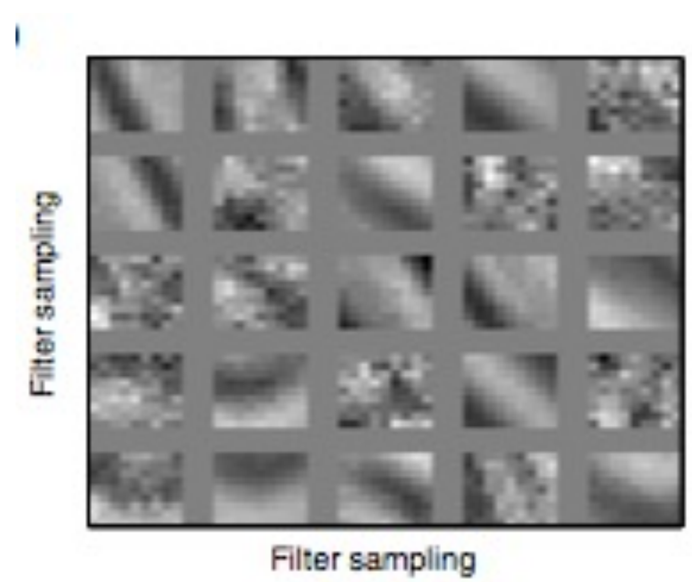
MARINE PHYSICAL LABORATORY

Dictionary learning

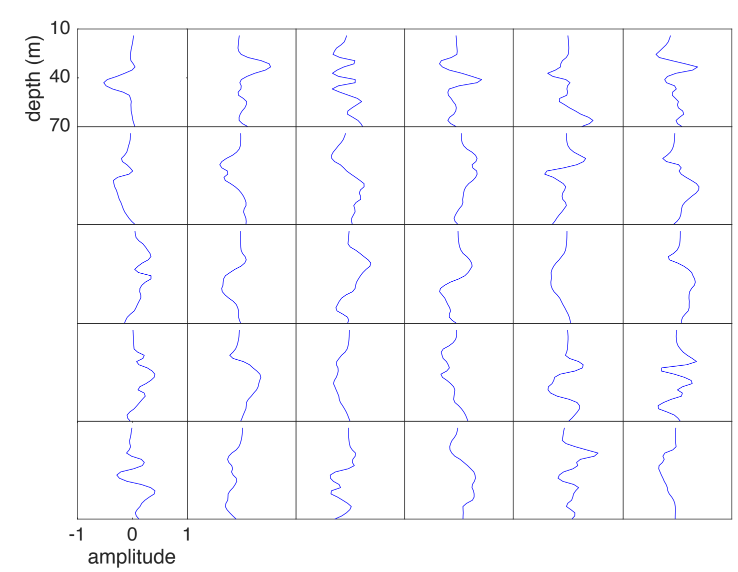
- Means of estimating sparse causes for given classes of signals, e.g. natural images, audio
- Originated in the neurosciences to estimate structure of V1 visual cortex cells from natural images
- Useful for regularization of general image denoising inverse problem, but only recent applications in the geosciences
  - Seismic survey image denoising
  - Dictionary learning of ocean sound speed profiles (SSPs)



Olshausen 2009



Beckouche 2014

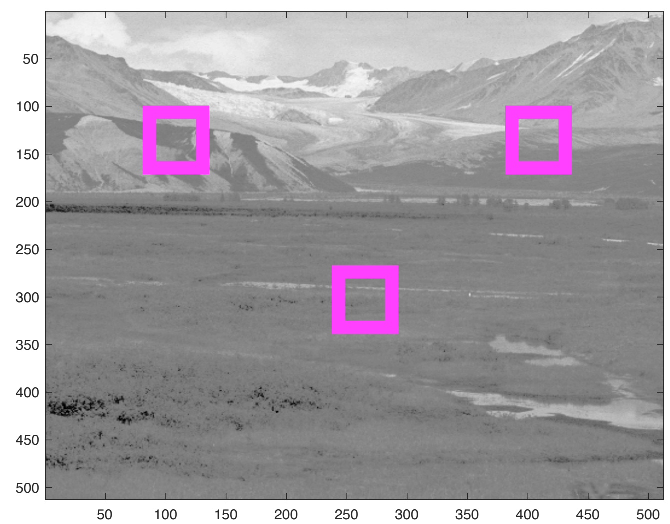


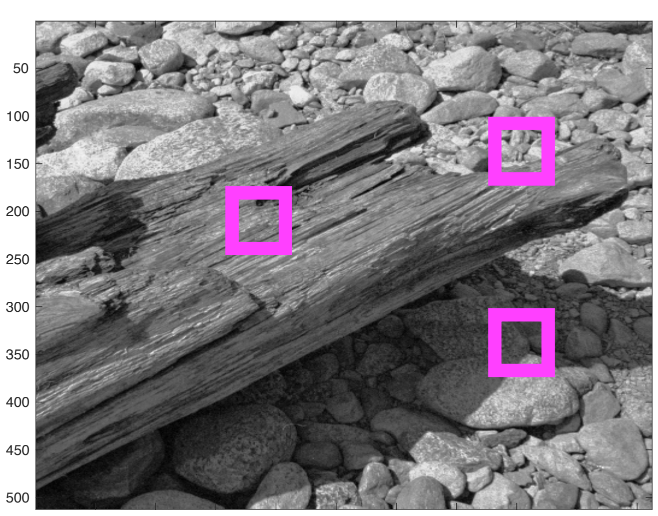
Bianco and Gerstoft 2017

Dictionary learning: Olshausen and Field 1997

- Seminal paper on learning dictionaries from a given class of signals
- Possible strategy of mammalian visual system for reducing redundancy in natural images

"Natural images"

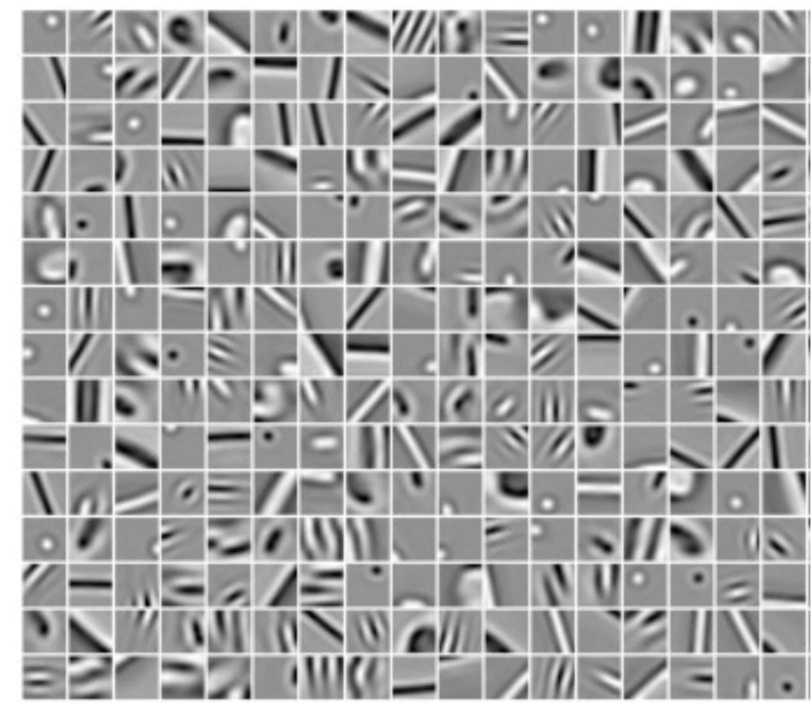




Observe random patches from corpus of natural images

Vectorize patches to obtain observations $\mathbf{Y}$

"Dictionary"



Estimate "dictionary" $\mathbf{\Phi}$ of basis functions which explain the structure observed in all the image patches

Overcomplete means # of functions > dimension of signal

Olshausen and Field 1997: image model with sparse prior

Assume that each image patch described by linear system

$$\mathbf{y}_k = \sum_n a_{nk}\phi_n = \mathbf{\Phi}\mathbf{a}_k \qquad \mathbf{y}_k = \mathbf{\Phi}\mathbf{a}_k + \mathbf{n}$$

Goal: estimate bases $\mathbf{\Phi}$ from observations $\mathbf{y}_k$
Probability of image patch arising from bases phi is

$$p(\mathbf{y}_k|\mathbf{\Phi}) = \int p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi})p(\mathbf{a}_k)d\mathbf{a}_k$$, with

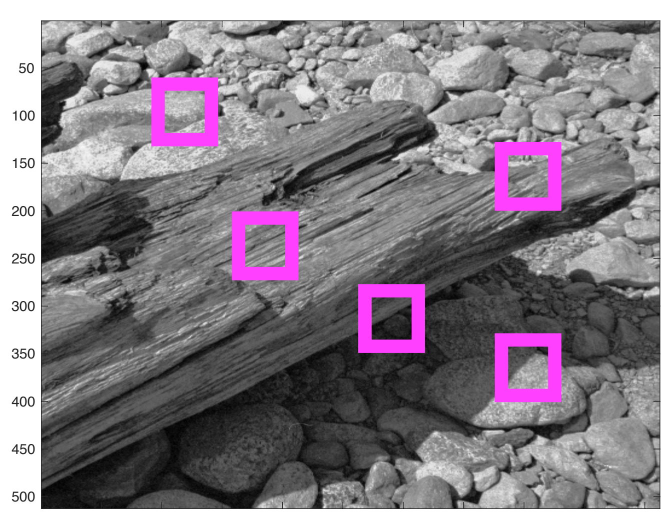
Likelihood

$$p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi}) = \frac{1}{Z_\sigma} e^{\frac{-\|\mathbf{y}_k - \mathbf{\Phi}\mathbf{a}_k\|_2^2}{2\sigma^2}}$$

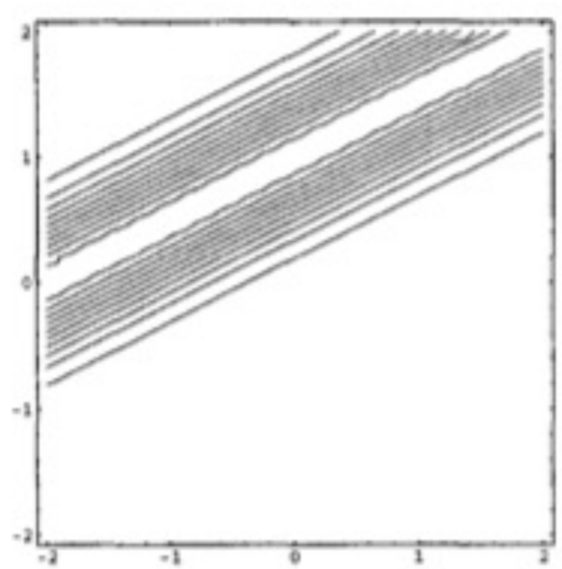
Independent, sparse prior

$$p(\mathbf{a}_k) = \prod p(a_{nk}) \quad p(a_{nk}) = \frac{1}{Z_\beta} e^{-\beta S(a_{nk})}$$

Image patches $\mathbf{y}_k$

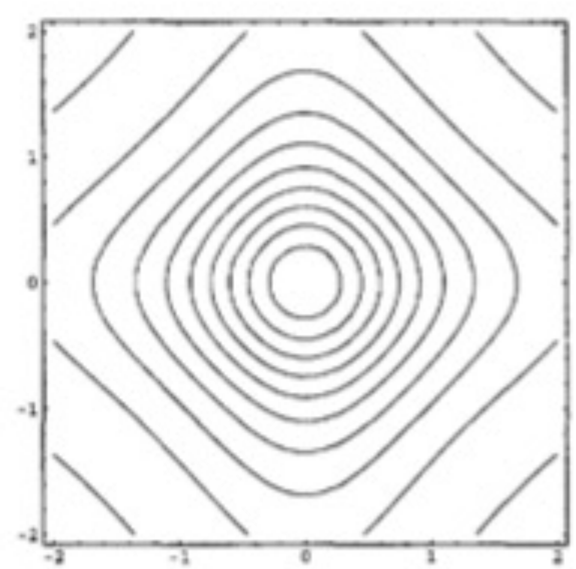


$p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi})$



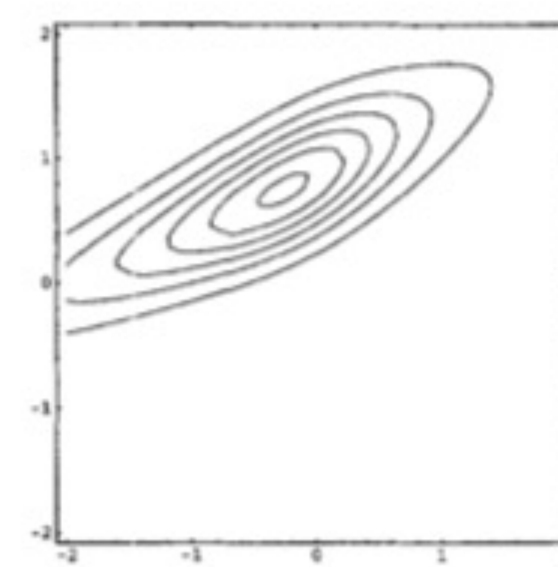
Likelihood

$p(\mathbf{a}_k)$



Prior

$p(\mathbf{y}_k, \mathbf{a}_k|\mathbf{\Phi}) = p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi})p(\mathbf{a}_k)$



Posterior

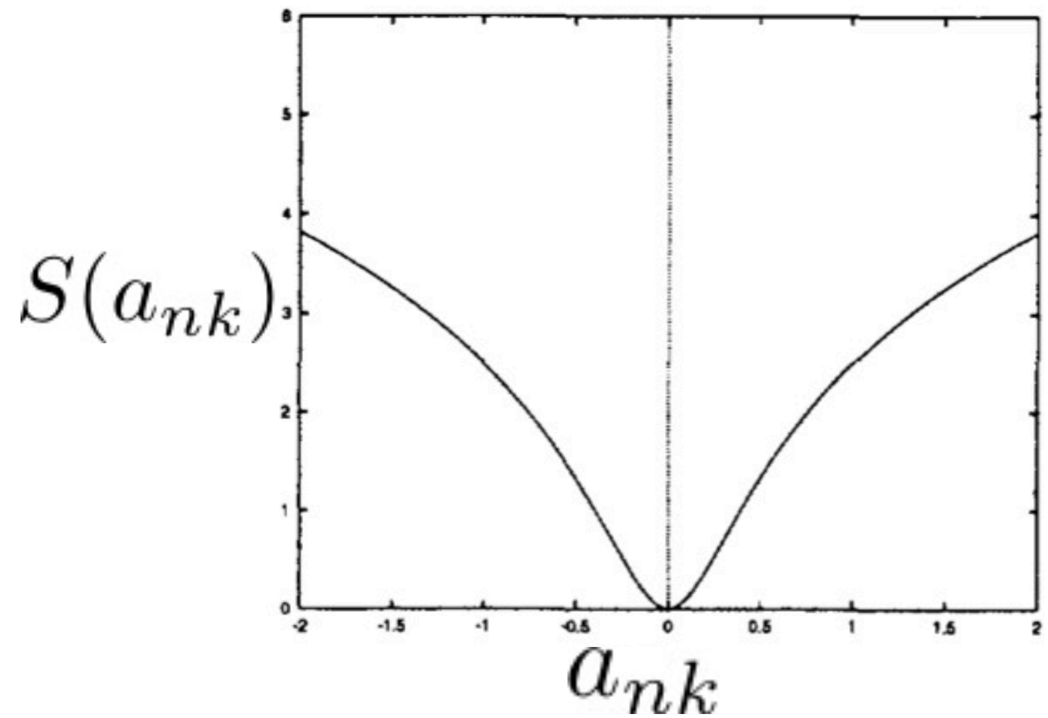
Olshausen and Field 1997- sparse prior induces sparse coefficients

Sparsity inducing prior

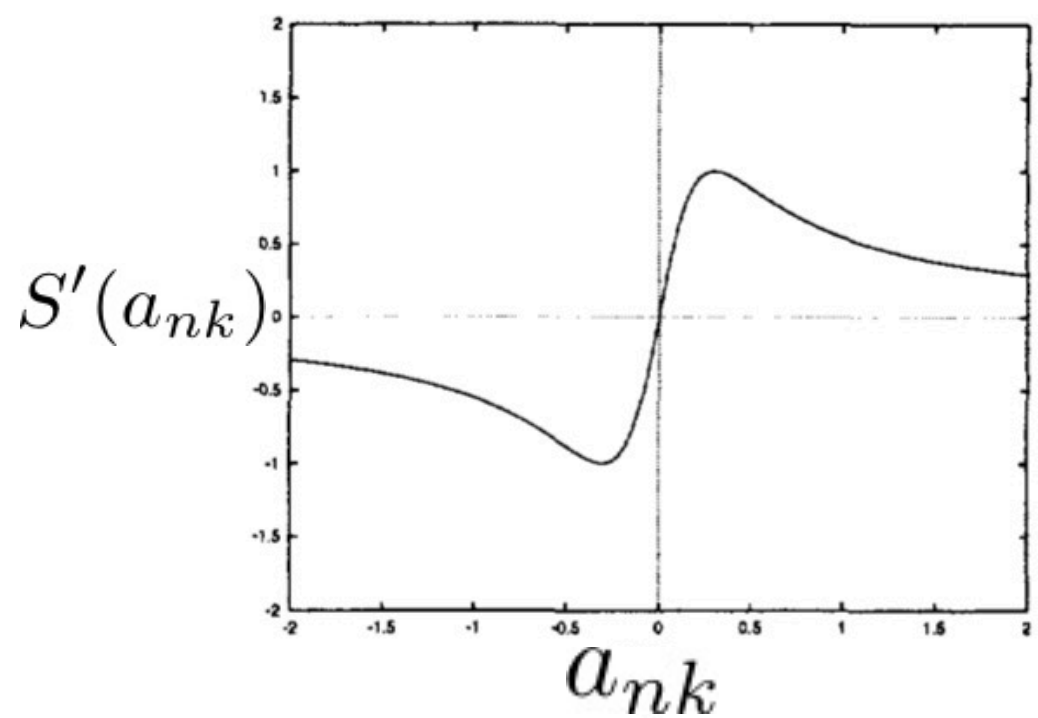
$$p(a_{nk}) = \frac{1}{Z_\beta} e^{-\beta S(a_{nk})}$$

$$S(a_{nk}) = \ln(1 + a_{nk})^2$$

"Cauchy distribution"



Derivative of prior induces sparsity in solution, as we'll see…

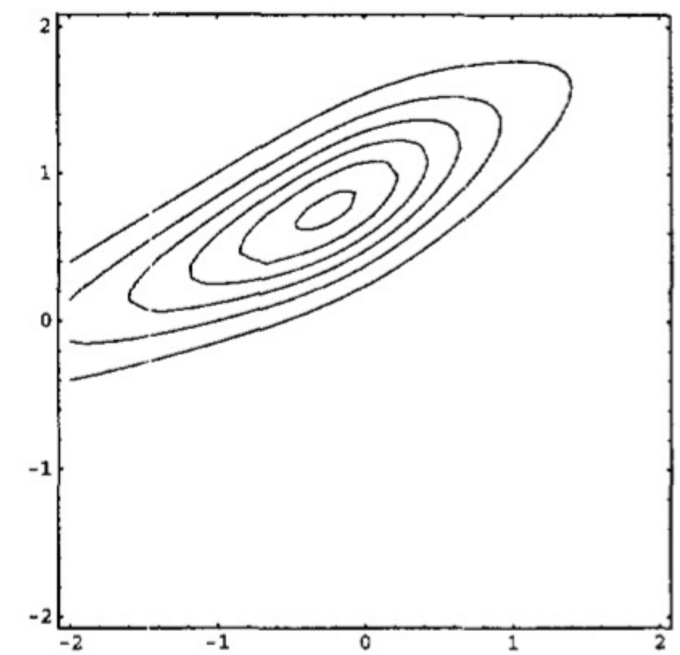


Olshausen and Field 1997 - derivation of Error function

Learn basis functions $\mathbf{\Phi}$ by minimizing Kullback-Leibler (KL) divergence between true images and those reproduced by model

$$KL = \int p^*(\mathbf{y}_k) \ln \frac{p^*(\mathbf{y}_k)}{p(\mathbf{y}_k|\mathbf{\Phi})} d\mathbf{y}_k$$

$$p(\mathbf{y}_k, \mathbf{a}_k|\mathbf{\Phi}) = p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi})p(\mathbf{a}_k)$$



Since $p^*(\mathbf{y}_k)$ is fixed, KL is minimized by maximizing log-likelihood (or minimizing negative log-likelihood) of image patches generated from model, hence

$$\{\widehat{\mathbf{\Phi}}, \widehat{\mathbf{a}}_k\} = \arg\min_{\mathbf{\Phi}} \left[\min_{\mathbf{a}_k} E(\mathbf{y}_k, \mathbf{a}_k|\mathbf{\Phi}) \right]$$

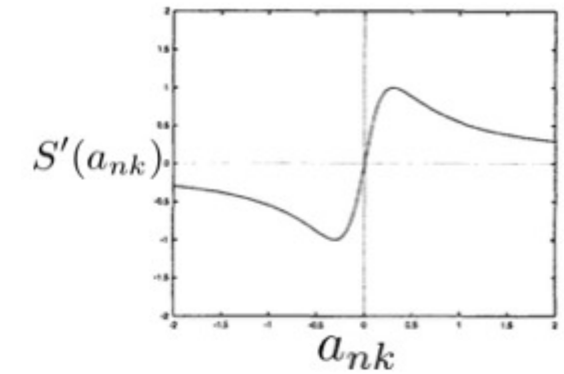
$$E(\mathbf{y}_k, \mathbf{a}_k|\mathbf{\Phi}) = -\ln p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi})p(\mathbf{a}_k)$$

Given: $p(\mathbf{y}_k|\mathbf{a}_k, \mathbf{\Phi}) = \frac{1}{Z_\sigma} e^{\frac{-\|\mathbf{y}_k - \mathbf{\Phi}\mathbf{a}_k\|_2^2}{2\sigma^2}}$ $\quad p(\mathbf{a}_k) = \prod p(a_{nk})$ $\quad p(a_{nk}) = \frac{1}{Z_\beta} e^{-\beta S(a_{nk})}$

$$E(\mathbf{y}_k, \mathbf{a}_k|\mathbf{\Phi}) = \|\mathbf{y}_k - \mathbf{\Phi}\mathbf{a}_k\|_2^2 + \lambda \sum_n S(a_{nk})$$

Olshausen and Field 1997 - gradients for network model

Rewriting Error function, take derivatives to find gradient



$$E(\mathbf{y}_k, \mathbf{a}|\mathbf{\Phi}) = \sum_m (y_{mk} - \sum_n \phi_{mn} a_{nk})^2 + \lambda \sum_n S(a_{nk})$$

$$\{\widehat{\mathbf{\Phi}}, \widehat{\mathbf{a}}_k\} = \arg\min_{\mathbf{\Phi}} \Big[\min_{\mathbf{a}_k} E(\mathbf{y}_k, \mathbf{a}|\mathbf{\Phi}) \Big]$$

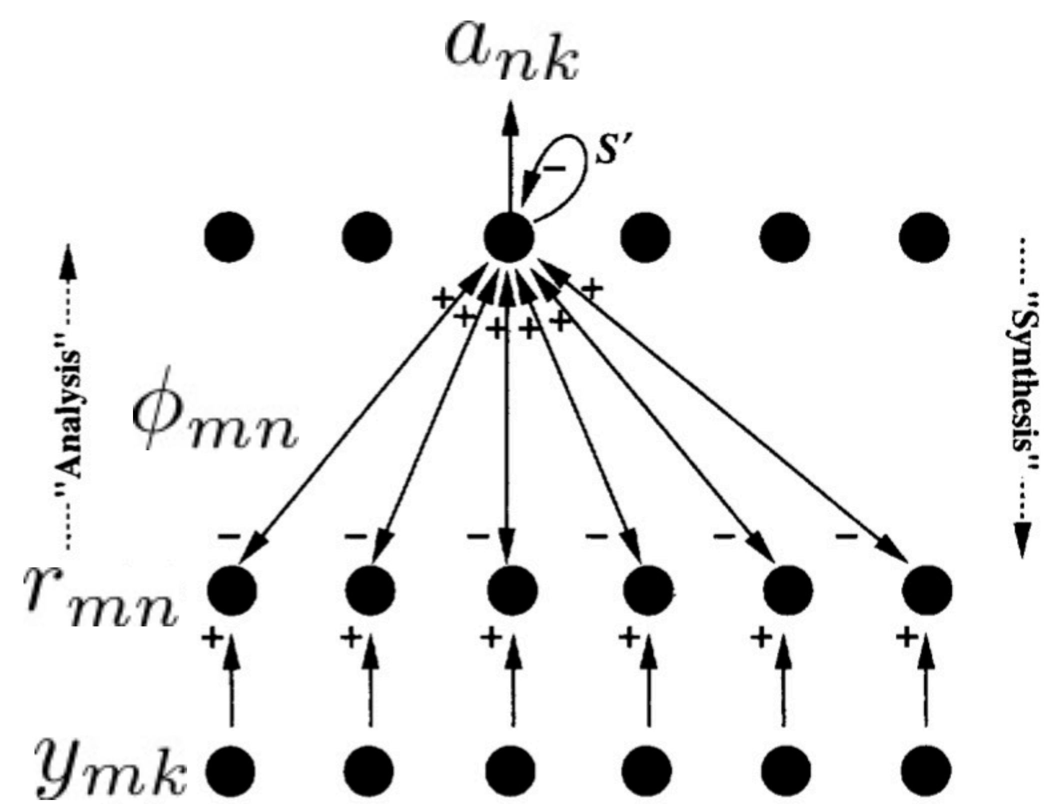
Update to a_{nk} with network (inner loop)

$$\dot{a}_{nk} = -\frac{dE}{da_{nk}} = \sum_m \phi_{mn} r_{mn} - \lambda S'(a_{nk})$$

with $\quad r_{mn} = y_{mk} - \sum_n \phi_{mn} a_{nk}$

Update to ϕ_{mn} with gradient descent (outer loop)

$$\Delta \phi_{mn} = \eta < a_{nk} r_{mn} >$$



"Hebbian" update

Olshausen and Field 1997 - gradients for network model

Can be rephrased as more recent canonical models, with Laplacian prior

$$\widehat{\mathbf{\Phi}} = \arg\min_{\mathbf{\Phi}} \sum_k \min_{\mathbf{a}_k} \{ \|\mathbf{\Phi}\mathbf{a}_k - \mathbf{y}_k\|_2^2 + \lambda \|\mathbf{a}_k\|_1 \}$$

Coefficients calculated using gradient descent, then dictionary updated by

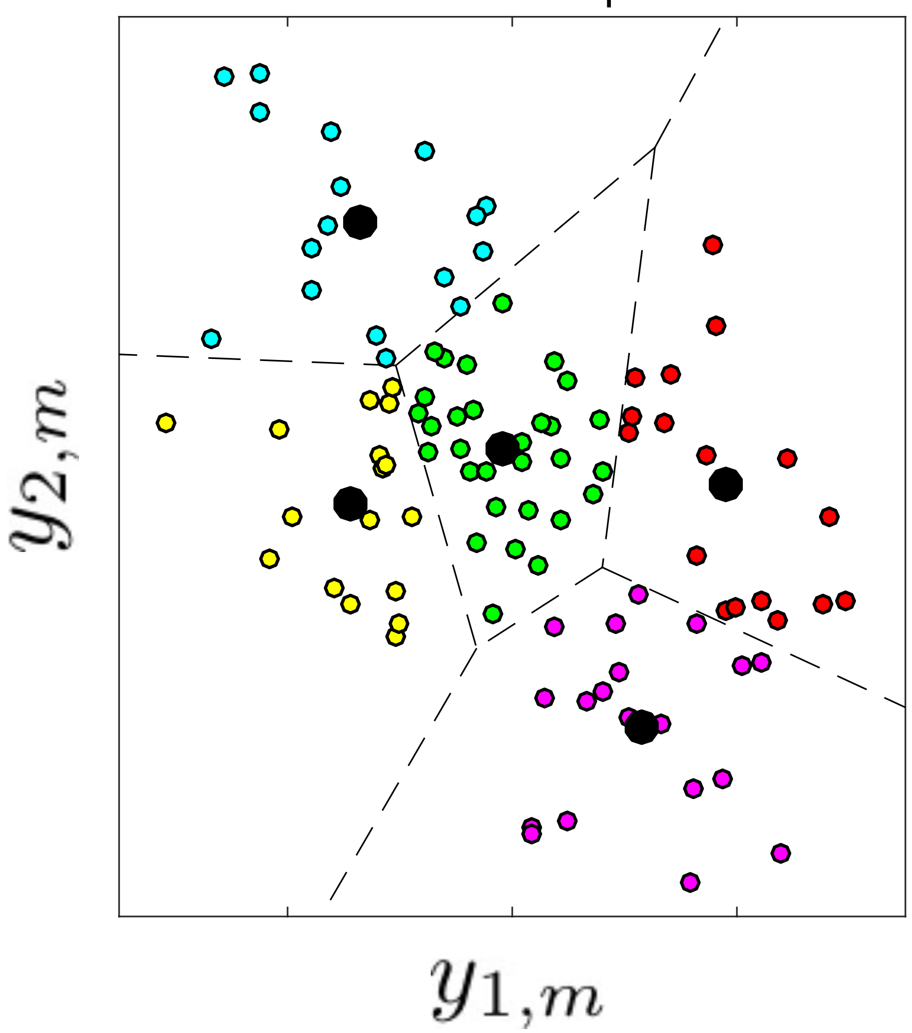
$$\mathbf{\Phi}^{(i+1)} = \mathbf{\Phi}^{(i)} - \eta \sum_k \left(\mathbf{\Phi}^{(i)}\mathbf{a}_k - \mathbf{y}_k\right)\mathbf{a}_k^T$$

....

This idea of iterative refinement is familiar: solving for coefficients, then updating basis functions

Iterative refinement: Vector Quantization and K-means

2D example



Vector quantization (VQ): means of compressing a set of data observations $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_M]$ using a nearest neighbor metric with codebook $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_N]$

$$R_n = \{i \mid \forall_{l \neq n}, \|\mathbf{y}_i - \mathbf{c}_n\|_2 < \|\mathbf{y}_i - \mathbf{c}_l\|_2\}$$

$$S_n(\mathbf{y}) = \begin{cases} 1 & \text{if } \mathbf{y} \in R_n \\ 0 & \text{otherwise,} \end{cases} \qquad \widehat{\mathbf{y}}_m = \sum_{i=1}^{N} S_i(\mathbf{y}_m)\mathbf{c}_i$$

K-means: finds optimal codebook for VQ

Given: training vectors $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_M] \in \mathbb{R}^{K \times M}$

Initialize: index $i = 0$, codebook $\mathbf{C}^0 = [\mathbf{c}_1^0, \ldots, \mathbf{c}_N^0] \in \mathbb{R}^{K \times N}$, MSE^0

I: Update codebook

1. Partition $\mathbf{Y}$ into N regions $(R_1, \ldots, R_N)$ by
   $R_n = \{i | \forall_{l \neq n}, \|\mathbf{y}_i - \mathbf{c}_n^i\|_2 < \|\mathbf{y}_i - \mathbf{c}_l^i\|_2\}$
2. Make code vectors centroids of $\mathbf{y}_j$ in partitions R_n
   $\mathbf{c}_n^{i+1} = \frac{1}{|R_n^i|} \sum_{j \in R_n^i} \mathbf{y}_j$

II. Check error

1. Calculate MSE^{i+1} from updated codebook $\mathbf{C}^{i+1}$
2. If $|\text{MSE}^{i+1} - \text{MSE}^i| < \eta$
   $i = i + 1$, return to I
   else

end

Relationship to canonical sparse processor

Sparse processor

$$\widehat{\mathbf{x}}_m = \arg\min_{\mathbf{x}_m} \|\mathbf{y}_m - \mathbf{Q}\mathbf{x}_m\|_2 \text{ subject to } \|\mathbf{x}_m\|_0 \leq T$$

VQ operators

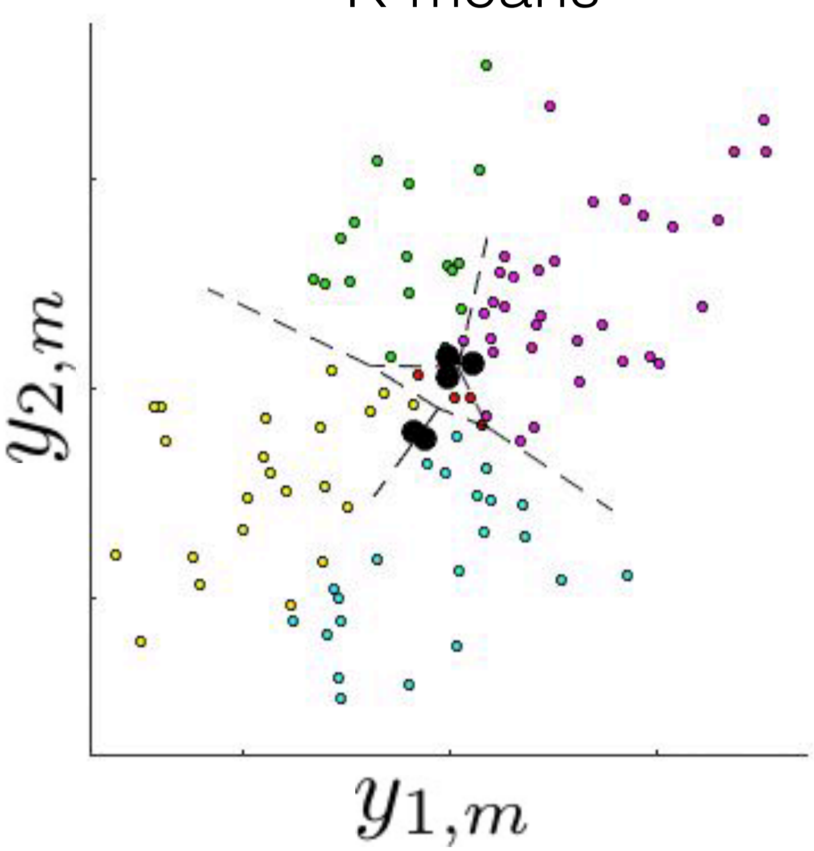
$$R_n = \{i \mid \forall_{l \neq n}, \|\mathbf{y}_i - \mathbf{c}_n\|_2 < \|\mathbf{y}_i - \mathbf{c}_l\|_2\}$$

$$\widehat{\mathbf{y}}_m = \sum_{i=1}^{N} S_i(\mathbf{y}_m)\mathbf{c}_i \qquad S_n(\mathbf{y}) = \begin{cases} 1 & \text{if } \mathbf{y} \in R_n \\ 0 & \text{otherwise,} \end{cases}$$

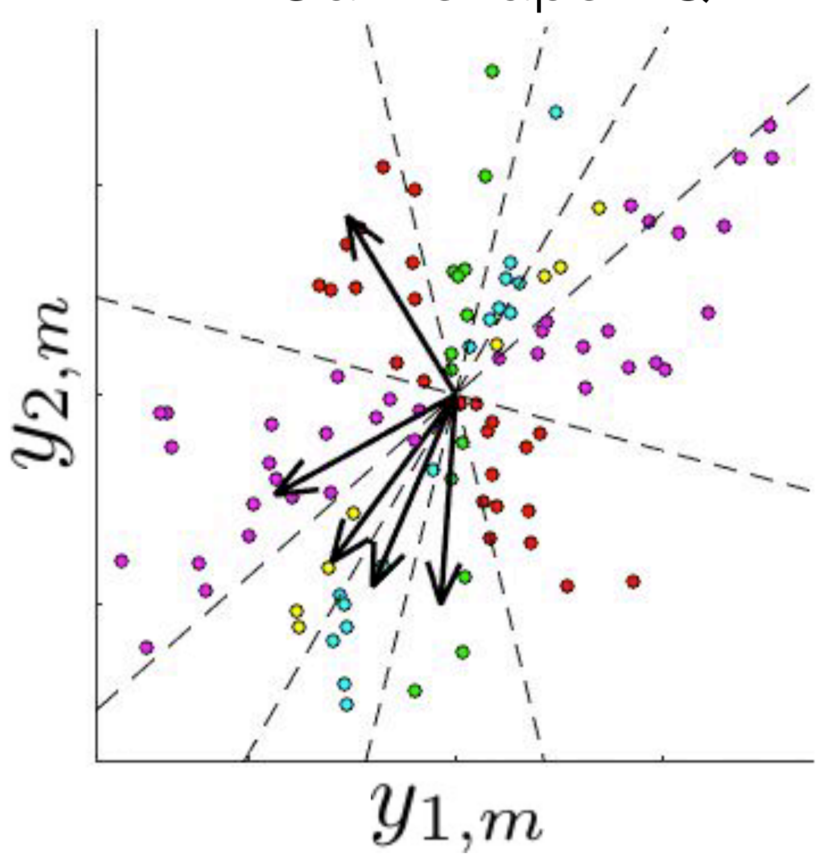
Dictionary learning objective

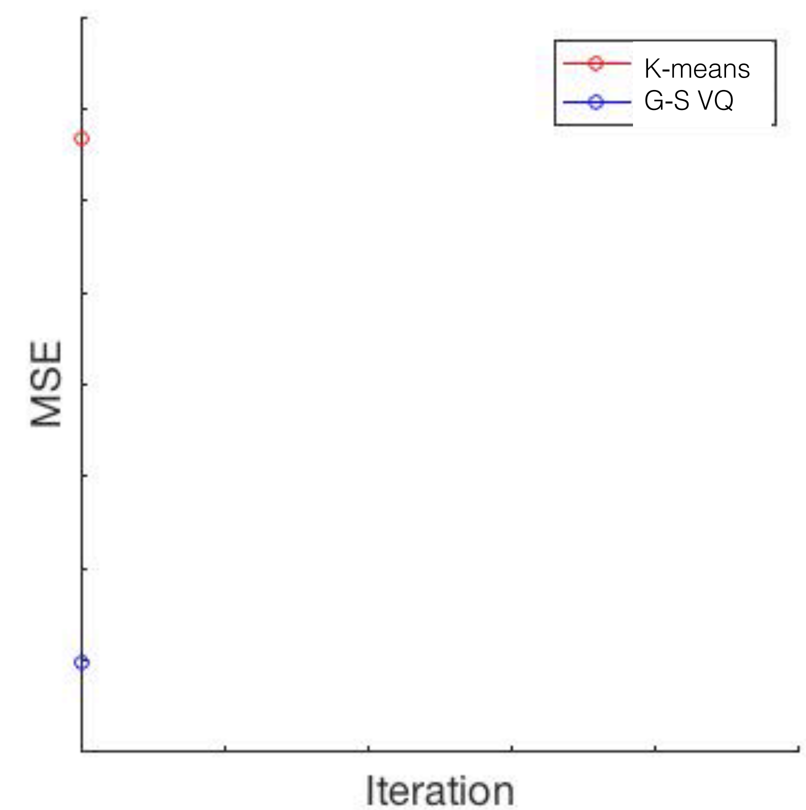
$$\min_{\mathbf{Q}}\{\min_{\mathbf{X}}\|\mathbf{Y} - \mathbf{Q}\mathbf{X}\|_F^2 \text{ subject to } \forall m,\ \|\mathbf{x}_m\|_0 \leq T\}$$

K-means



Gain-shape VQ





MOD algorithm: Extending K-means to dictionary learning problem

Method of Optimal Directions (MOD) [Engan 2000]

$$\min_{\mathbf{Q}}\{\min_{\mathbf{X}}\|\mathbf{Y}-\mathbf{Q}\mathbf{X}\|_F^2 \;\text{ subject to }\; \forall m,\; \|\mathbf{x}_m\|_0 \le T\}$$

MOD algorithm:

1. COEFFICIENTS: Solve for coefficients **X=[x_1…x_i]** for fixed **Q** using orthogonal matching pursuit (OMP)
2. DICTIONARY UPDATE: Solve for dictionary **Q=[q_1…q_i]**, by inverting the coefficient matrix **X**, and normalizing dictionary entries to have unit norm.

$$\widehat{\mathbf{Q}} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}$$

…. repeat until convergence

Simple and flexible but, a few drawbacks:

- computationally expensive to invert coefficient matrix **X**
- since keeping coefficients in **X** fixed during dictionary update, slow convergence

K-SVD algorithm

K-SVD [Aharon 2006]: Learn optimal dictionary for sparse representation of data

$$\min_{\mathbf{Q}}\{\min_{\mathbf{X}}\|\mathbf{Y}-\mathbf{QX}\|_F^2 \ \text{ subject to } \ \forall m,\ \|\mathbf{x}_m\|_0 \le T\}$$

K-SVD algorithm:

1. Solve for coefficients **X=[x_1…x_i]** for fixed **Q** using OMP
2. Solve (1) for dictionary **Q=[q_1…q_i]**, updating both **Q** and **X** from the SVD of representation error

$$\|\mathbf{Y}-\mathbf{QX}\|_F = \left\| \underbrace{\left(\mathbf{Y}-\sum_{j\neq k}\mathbf{q}_j\mathbf{x}_T^j\right)} - \mathbf{q}_k\mathbf{x}_T^k \right\|_F$$

$$= \|\mathbf{E}_k - \mathbf{q}_k\mathbf{x}_T^k\|$$

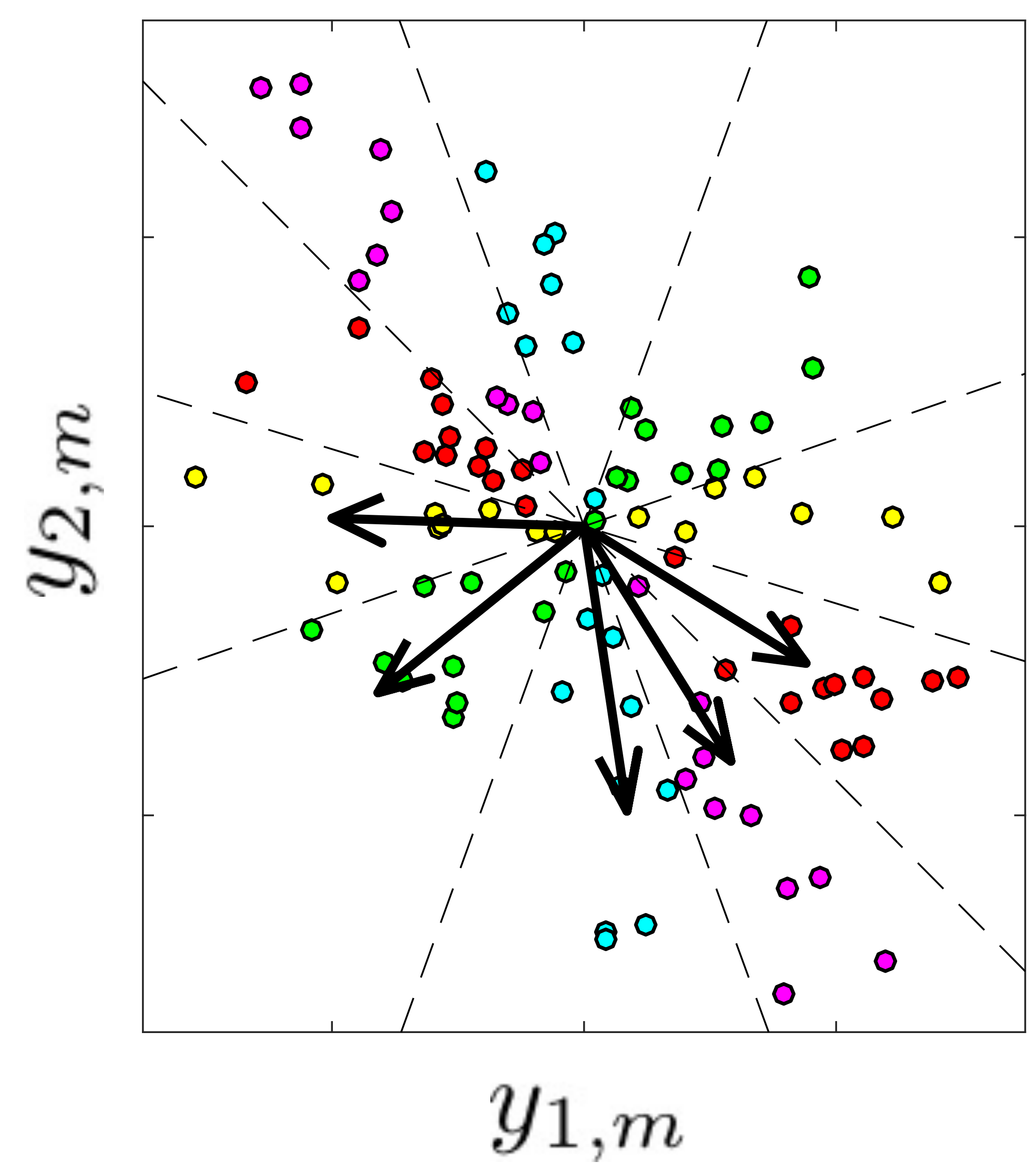
update **q_k, x_k** by SVD

$$\mathbf{E}_k^e = \mathbf{USV}^T$$

$$\mathbf{q}_k = \mathbf{U}(:,1), \mathbf{x}_T^k = \mathbf{V}(:,1)\mathbf{S}(1,1)$$

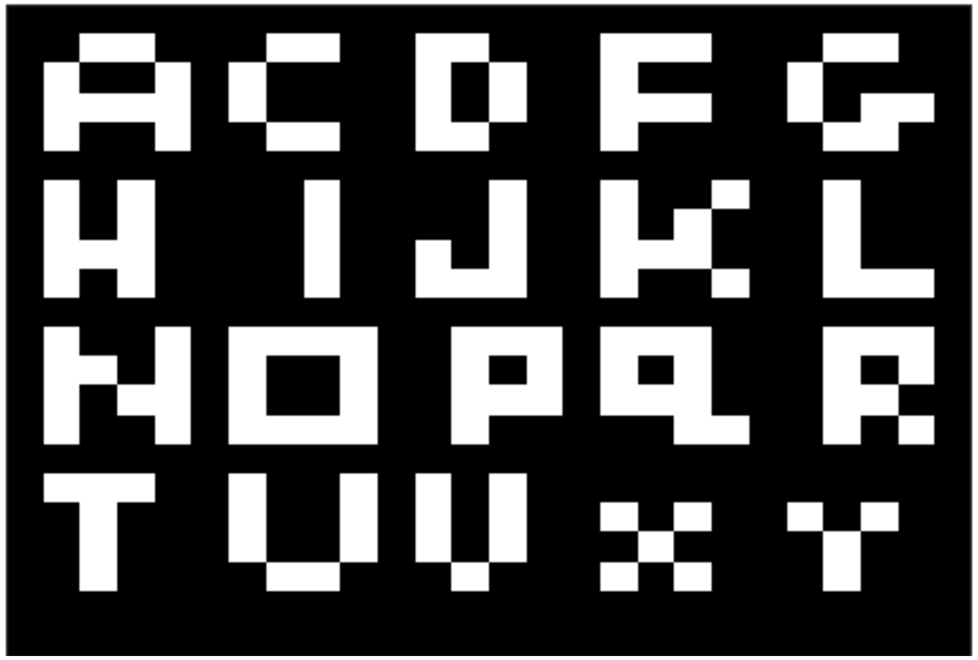
…. repeat until convergence

2D example

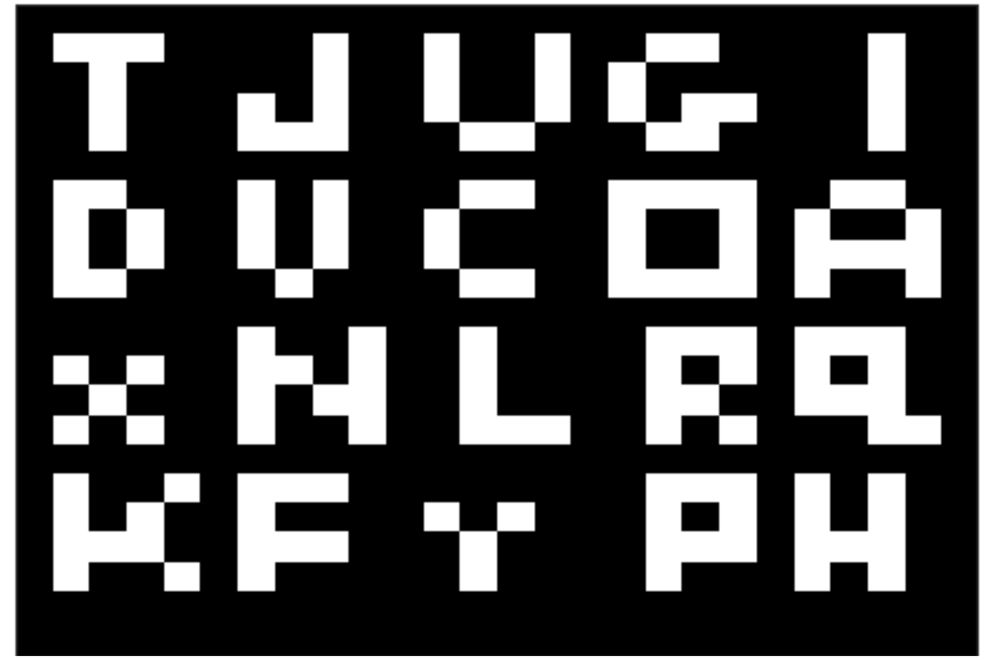


Example: Denoising alphabet with K-SVD algorithm

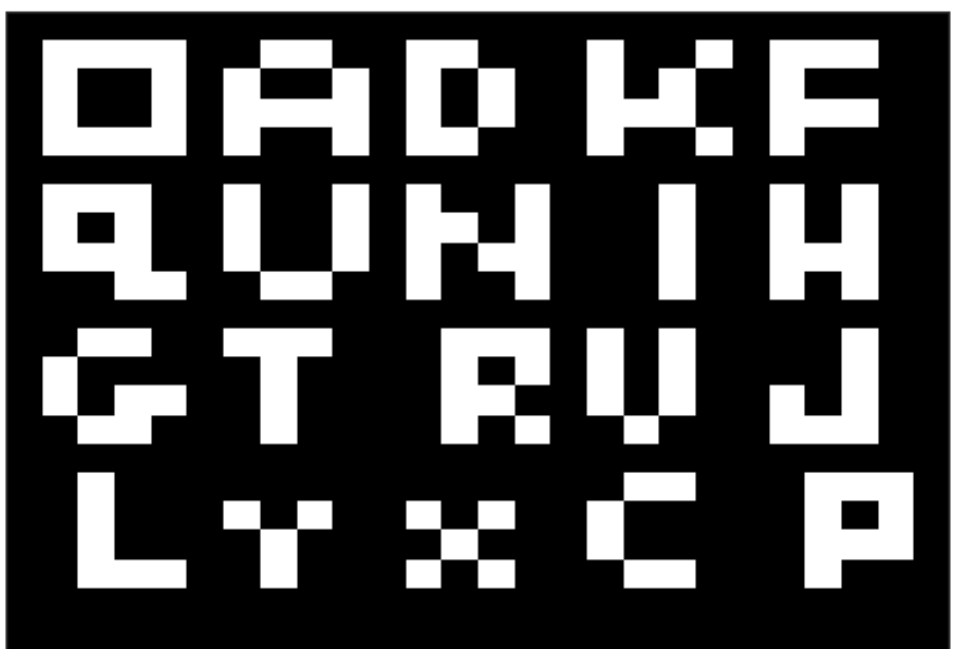
True alphabet



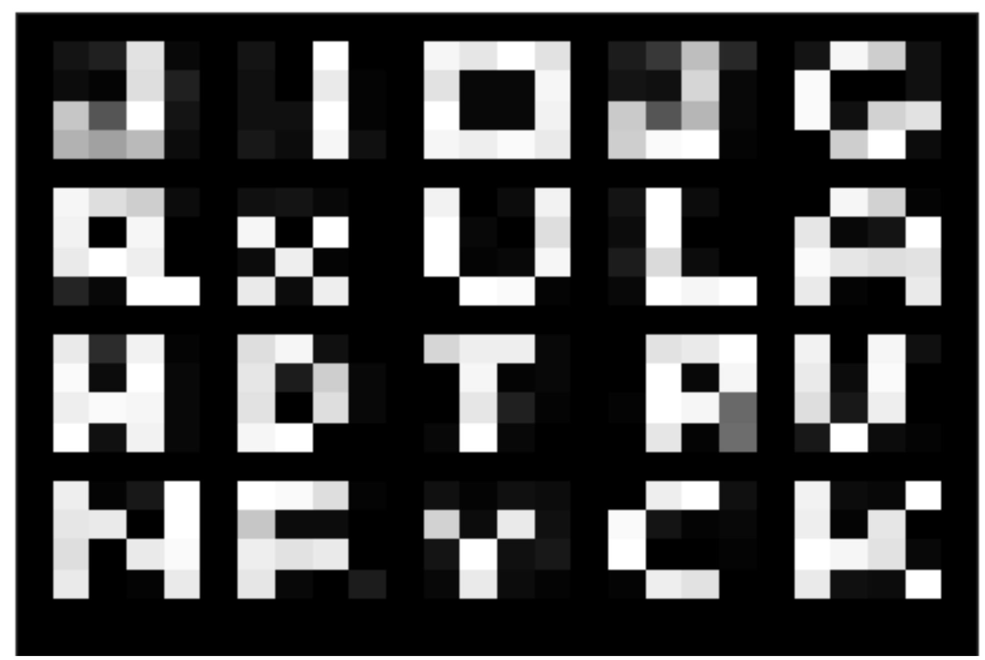
Recovered alphabet (no noise)



Recovered alphabet (noise std = .01)

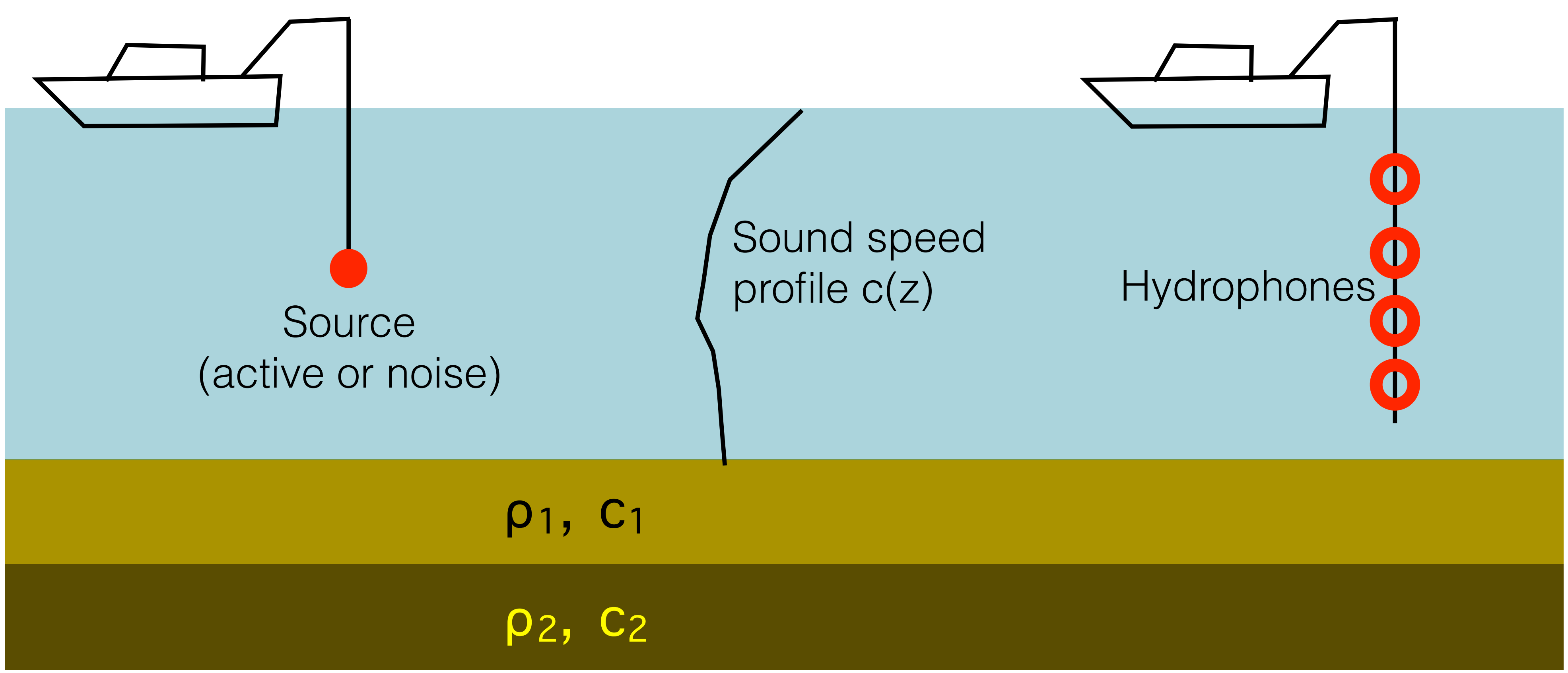


Recovered alphabet (noise std = .5)

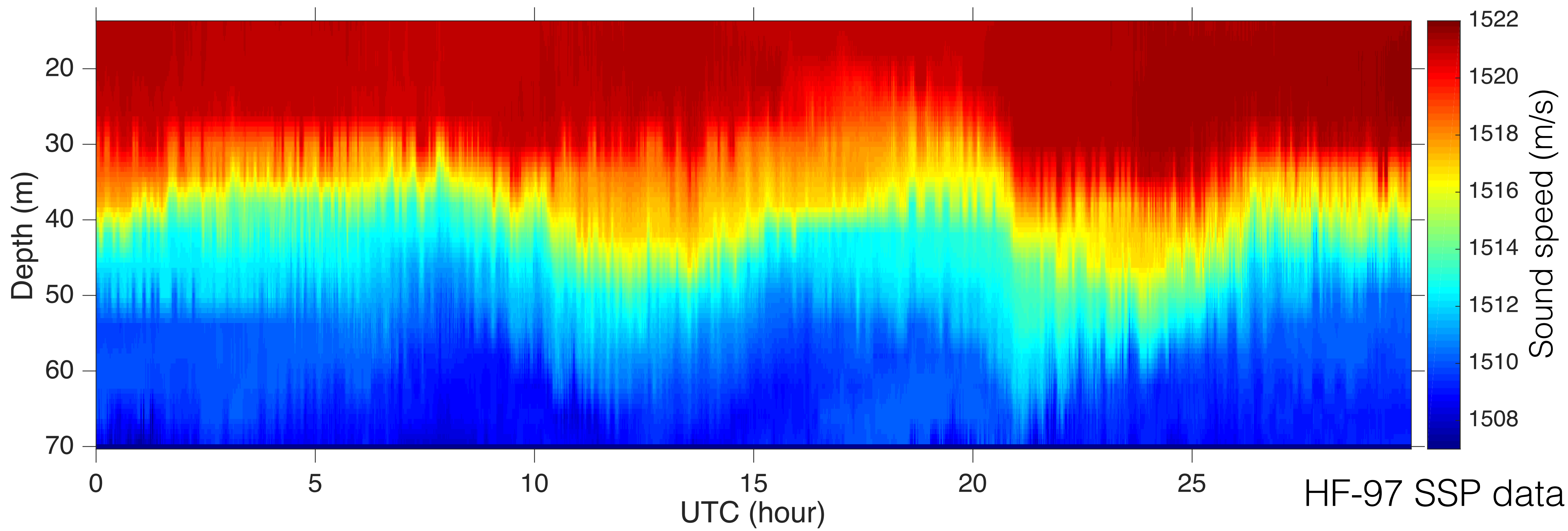


Dictionary learning of SSPs: motivation

- Acoustic observations from ocean contain information about ocean environment
- The inversion of environment parameters is limited by physics and signal processing assumptions



Sound speed profiles

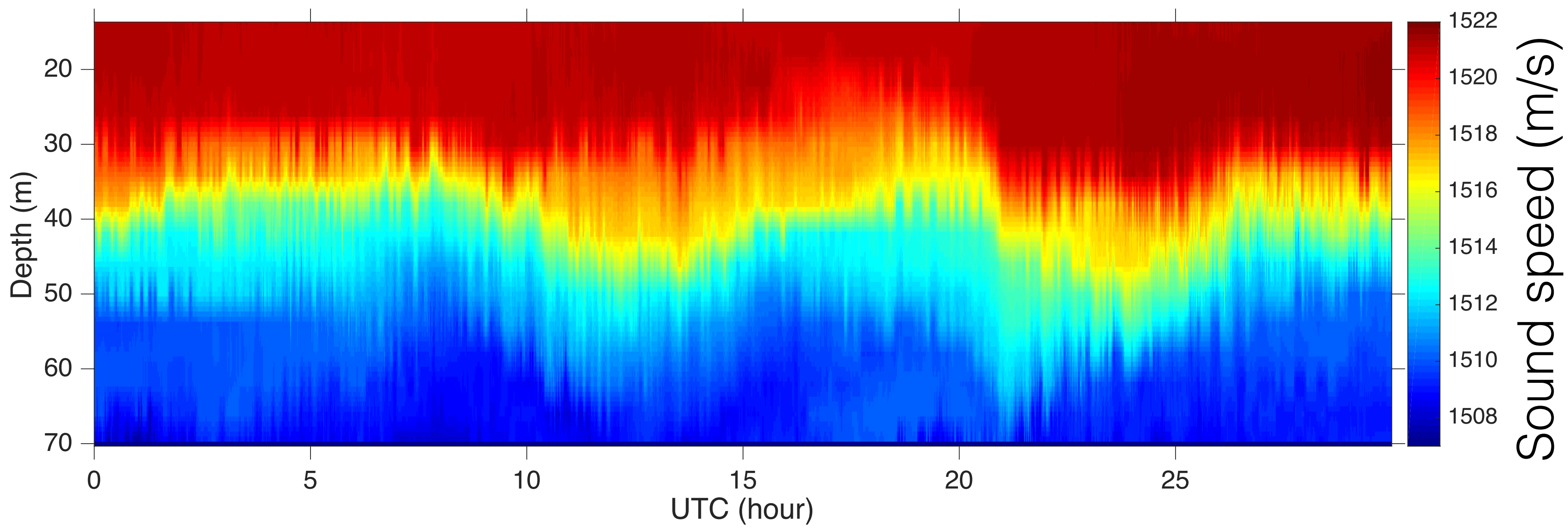


HF-97 SSP data

- Sound speed profiles (SSPs) in the ocean are often highly variable with fine scale fluctuations
- Acoustic inversion of SSPs is ill-posed and traditionally regularized using EOFs
- Dictionaries obtained via unsupervised learning may provide better representation of SSP dynamics

Dictionary learning of sound speed profiles

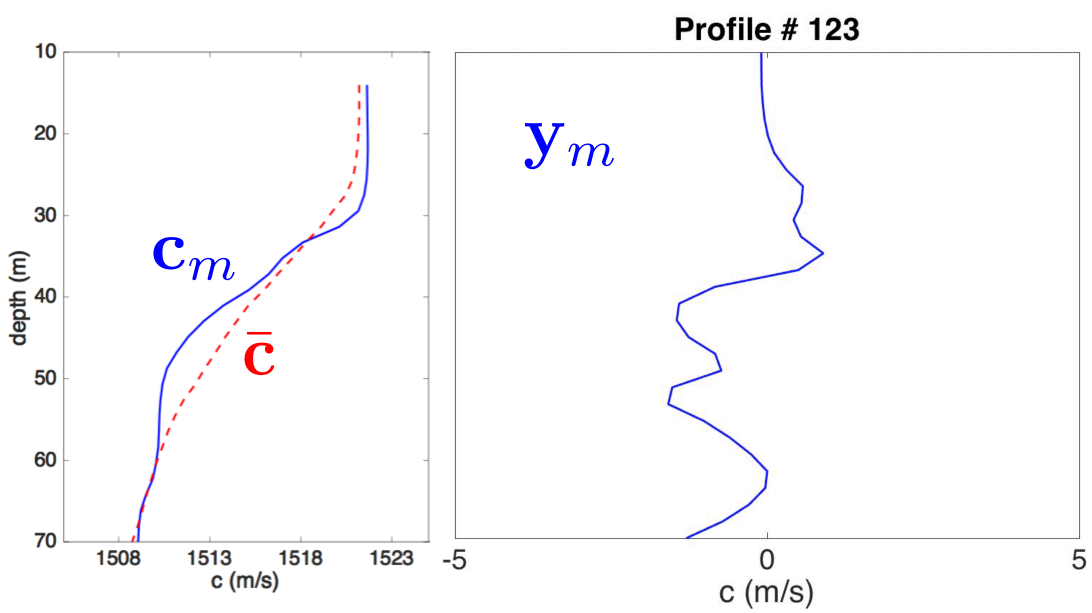
Bianco and Gerstoft JASA 2017 (published)



HF-97 Experiment

- 30 hours of SSP data
- Used 1000 profiles for dictionary learning
- *K* = 30 point SSP's (interpolated from 15 measurements)

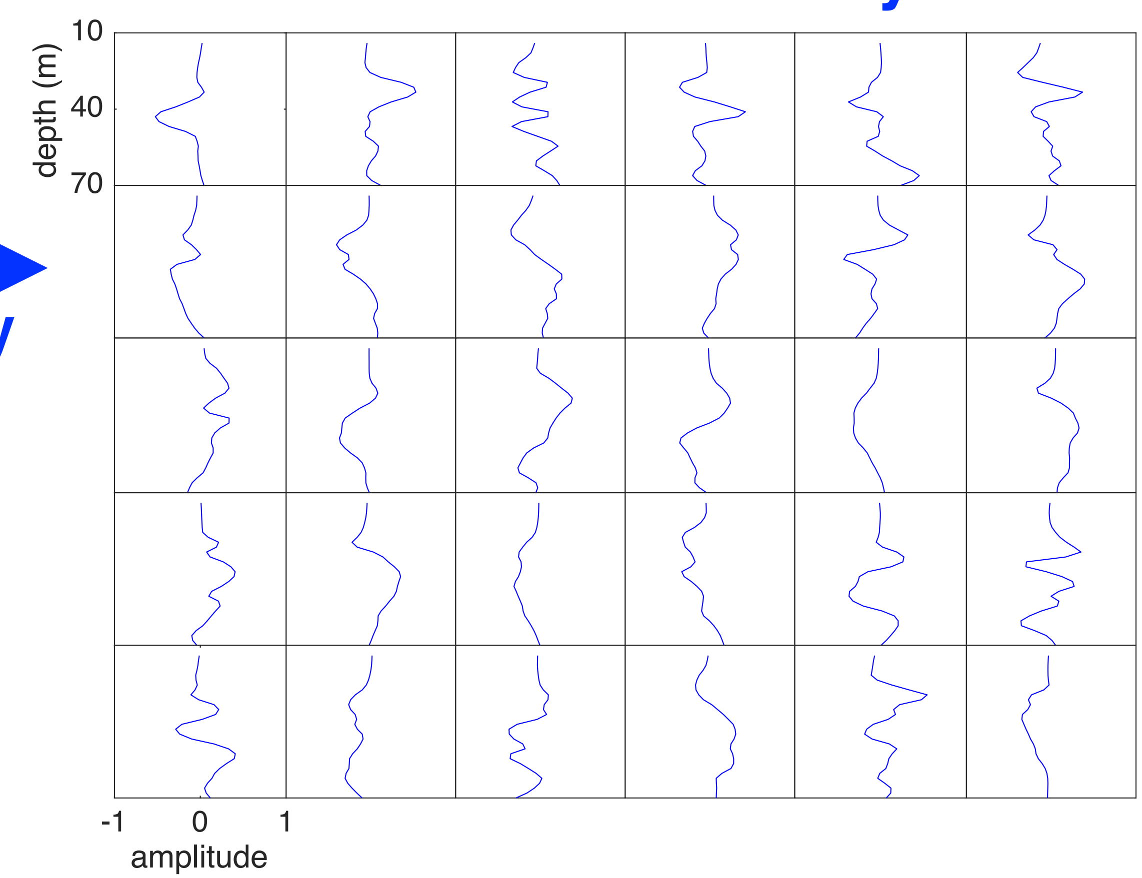
SSP Variability



$$\mathbf{y}_m = \mathbf{c}_m - \bar{\mathbf{c}}$$

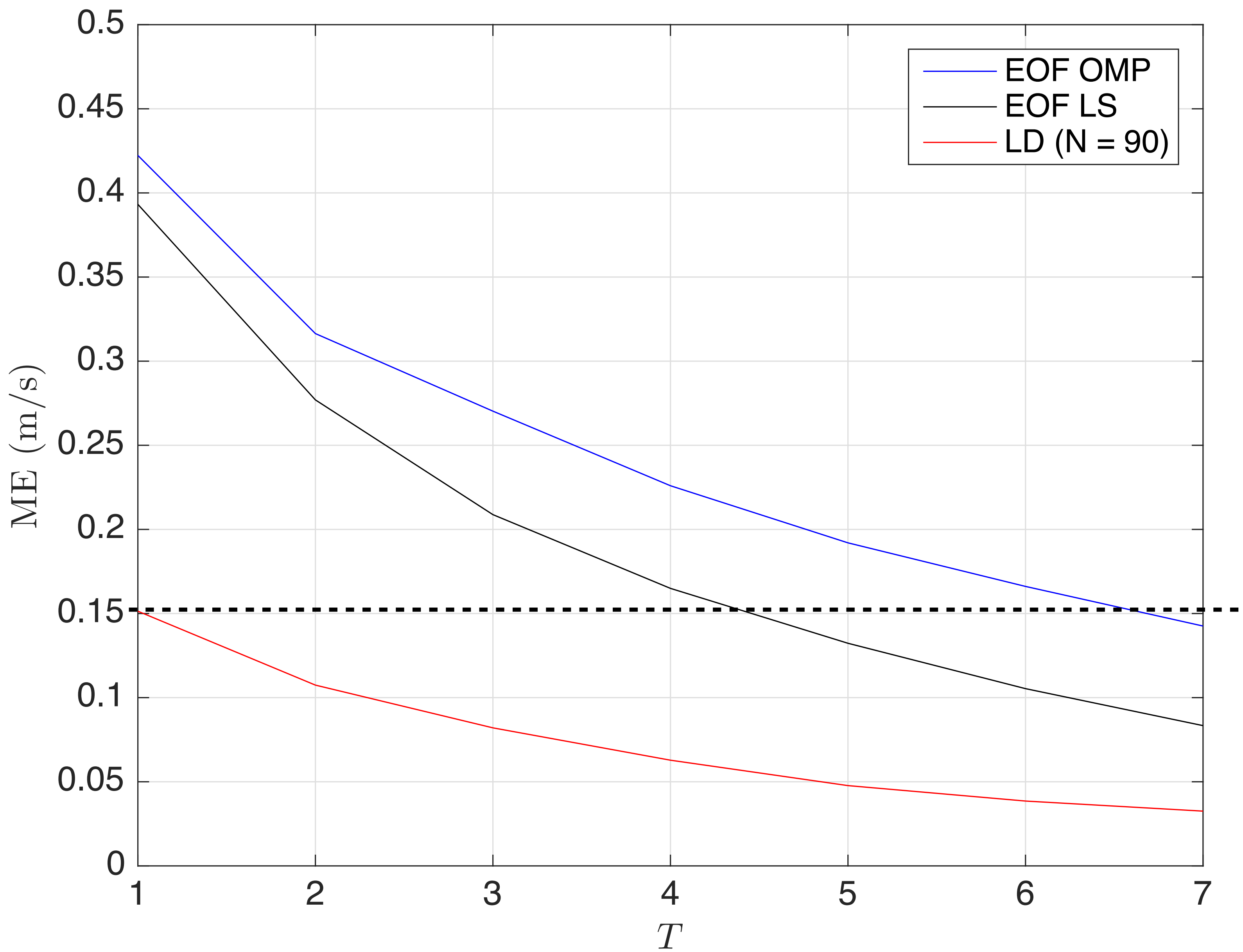
Dictionary Learning

'Learned Dictionary'



SSP reconstruction error using Dictionary Learning

Based on 1000 profiles from HF-97



LS: Least squares
OMP: Sparse processor

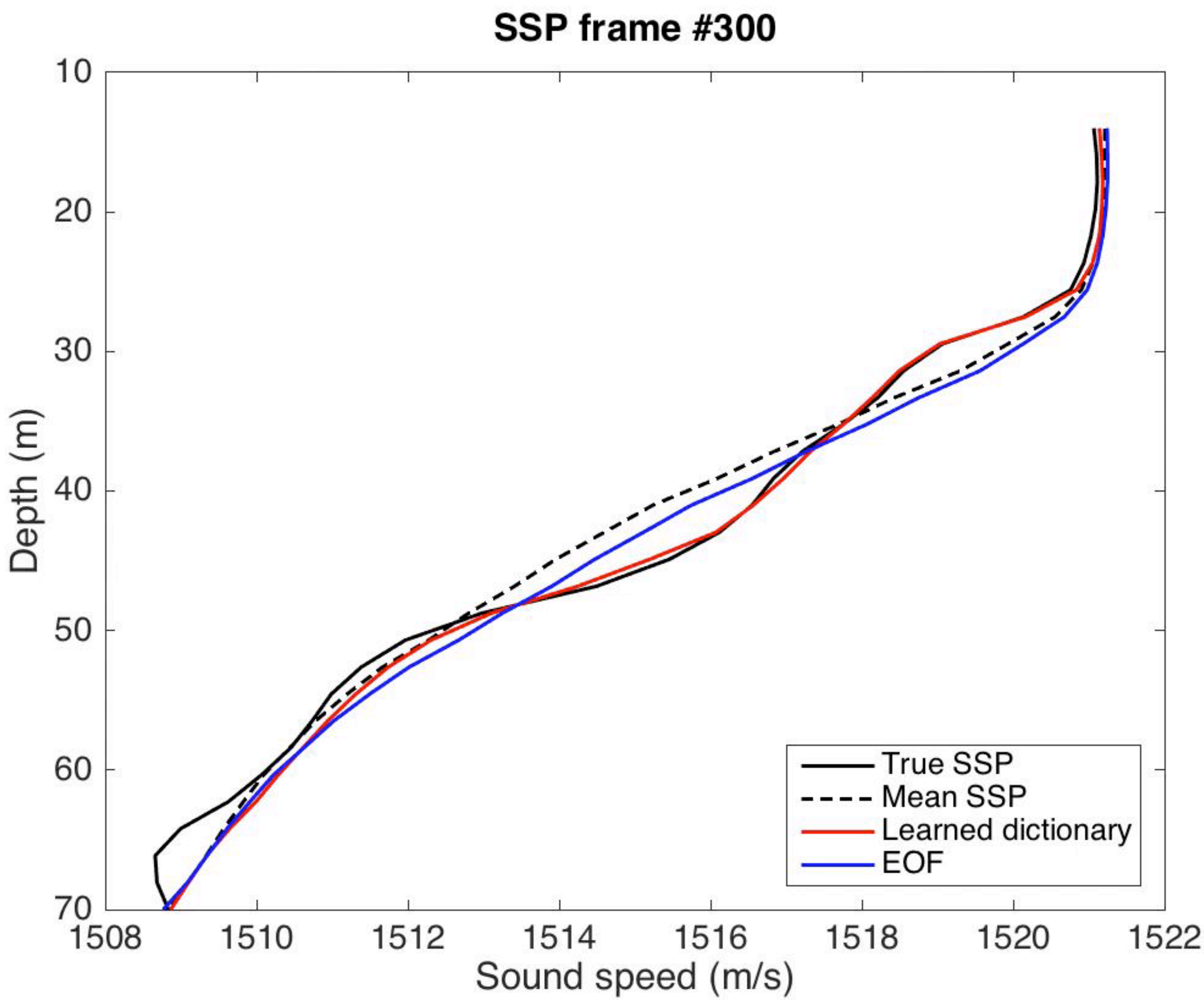
Mean Error (ME):

$$\mathrm{ME} = \frac{1}{KM}\|\mathbf{Y} - \widehat{\mathbf{Y}}\|_1$$

- One entry from Learned Dictionary fits SSP data better than 6 EOFs
- Learned dictionary (LD) reconstruction error less than 50% of EOF error

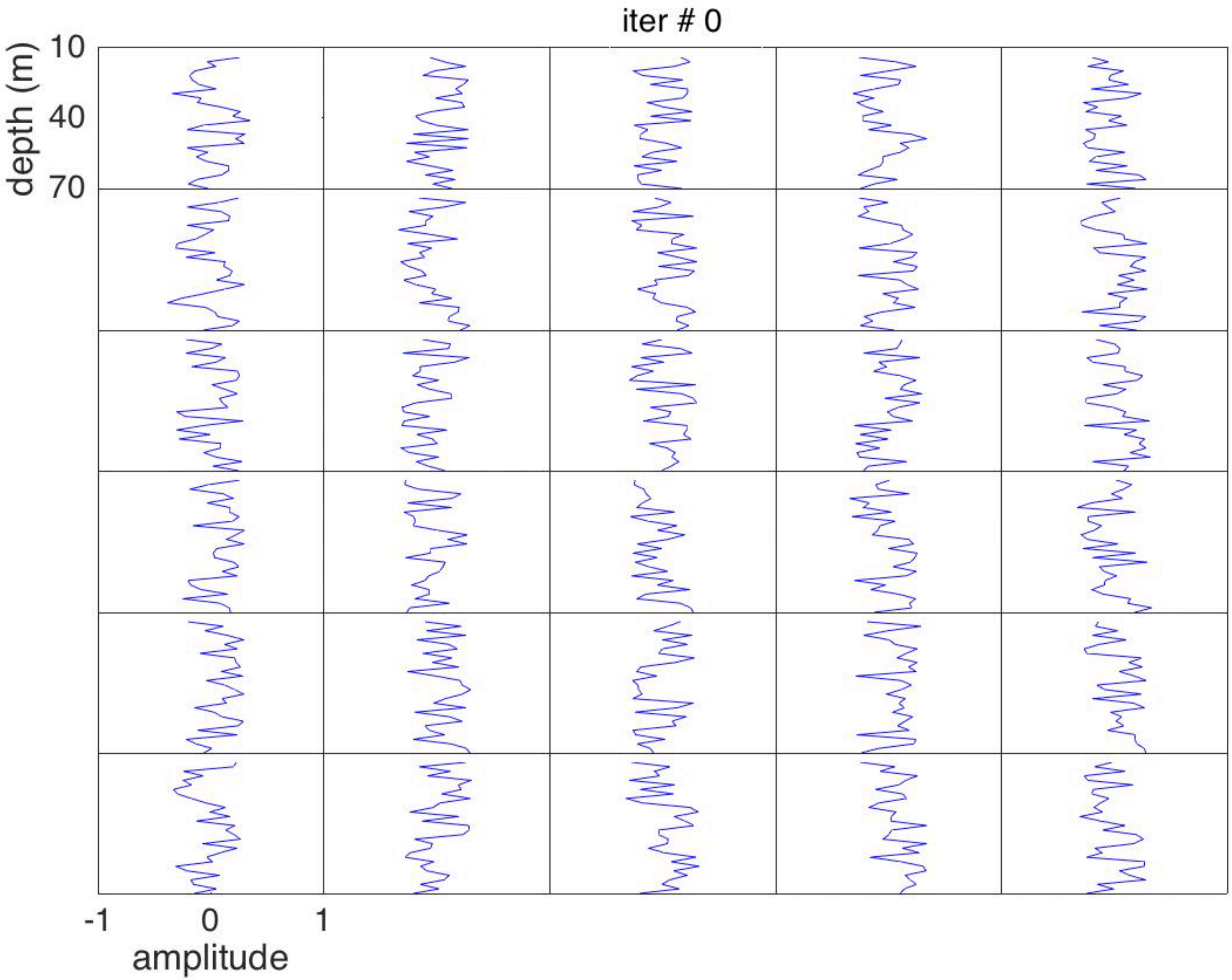
SSP reconstruction using Dictionary Learning

HF-97: One coefficient from Learned Dictionary vs. One EOF coefficient



Learning dictionary from HF-97 SSP variation

Q random initialized, converges within 15 iterations



LD solution space much smaller than EOFs

Inversion for SSP:

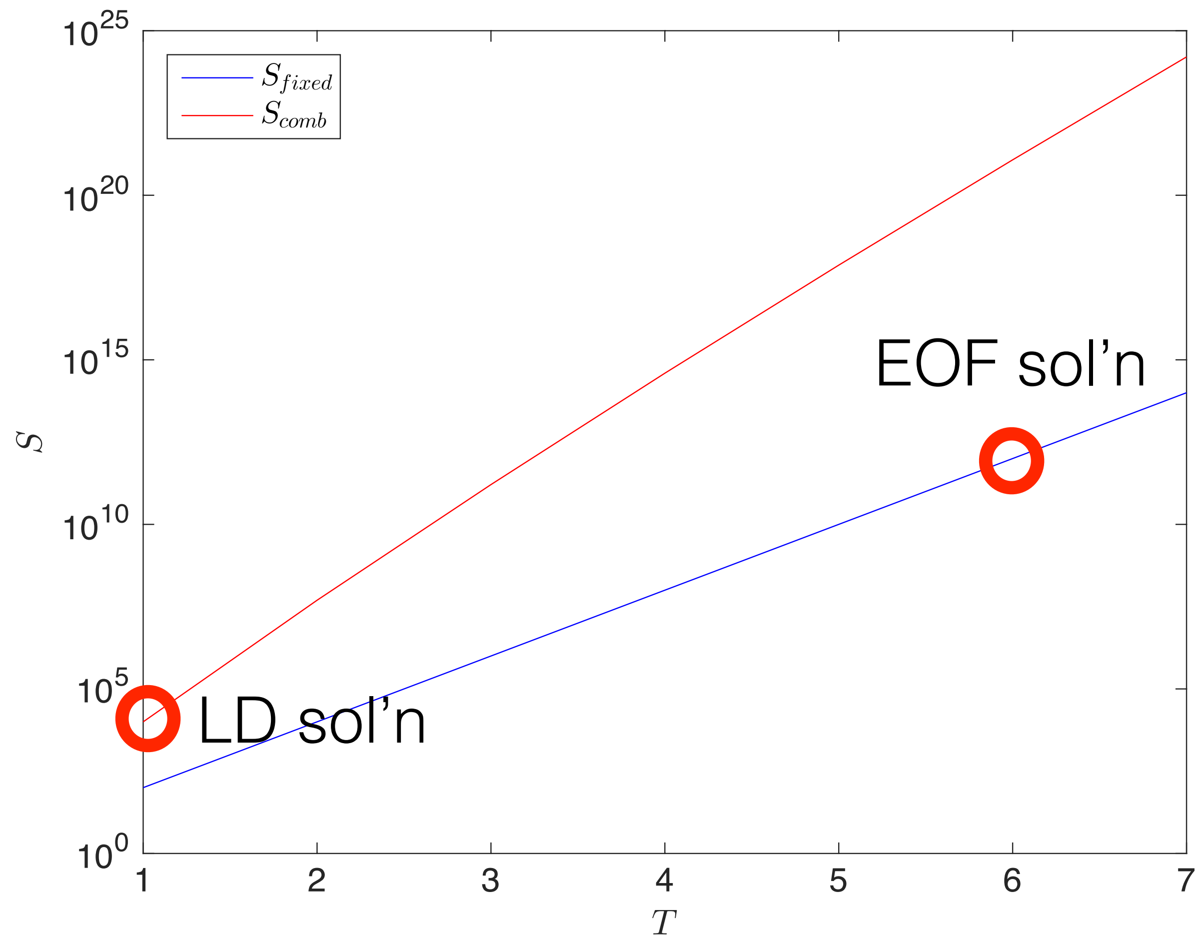
Assuming a potentially non-linear mapping:

- EOF solution: *T* leading order coefficients (fixed indices)

$$S_{\text{fixed}} = H^T$$

- LD solution: *T*-non-zero coefficients (combinatorial indices)

$$S_{\text{comb}} = H^T \frac{N!}{T!(N-T)!}$$

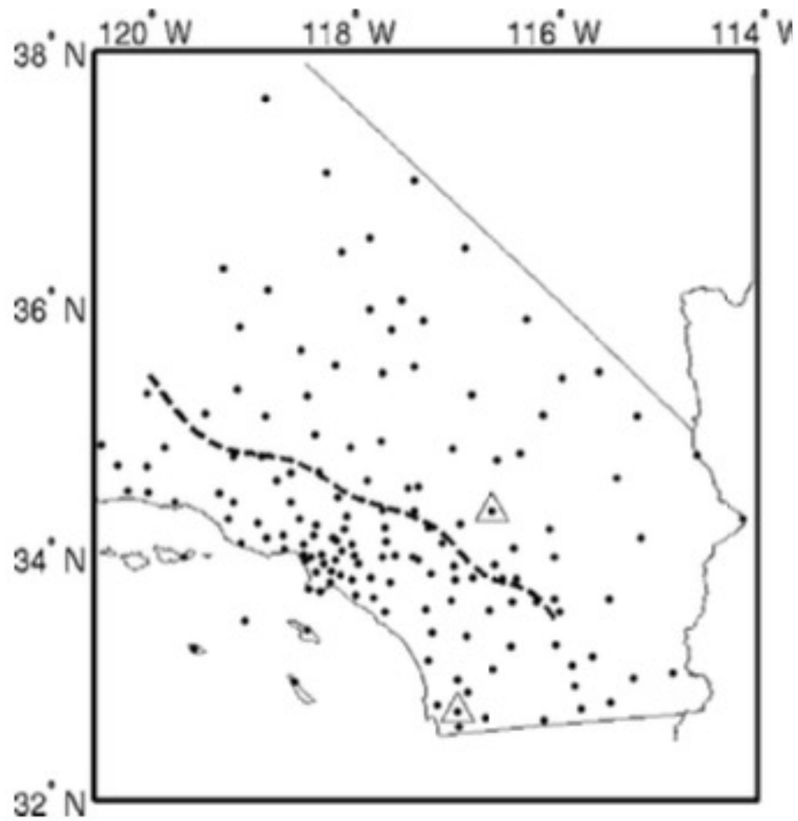


- Since 6 EOFs or 1 LD entry required, if coefficients discretized in *H*=100 coefficients number of possible solutions are

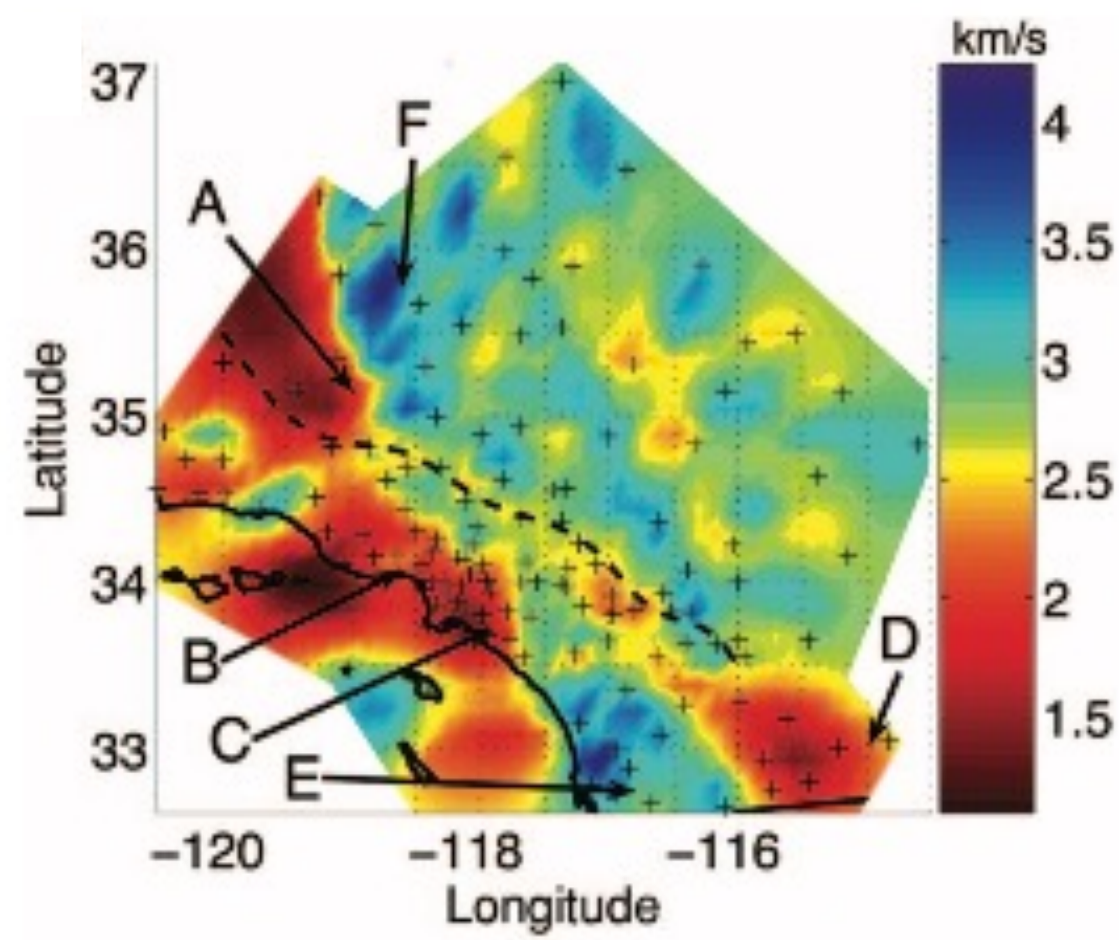
EOFs: $S_{fixed} = 10^{12}$ solutions LD: $S_{comb} = 10^{4}$ solutions

Adaptive patch based seismic tomography: motivation

- The earth contains both smooth and discontinuous variations in wave speed (e.g. Moho, faults) at multiple scales
- Most existing inversion methods regularize inversion of seismic data by assuming exclusively smooth, discontinuous, or block constant wave speeds for inversion, which may be unrealistic
- Propose adaptive approach based on image denoising algorithms
- Want to avoid Markov-chain Monte Carlo (MCMC) formulations of seismic inversion



Southern California Seismic Network



Travel time tomography

From the basic relation, get travel time: $t = \dfrac{d}{c}$

$c =$ wave speed

Travel time for ray from station *i* to *j* (r_{ij})

$$t_{ij} = \int \frac{dr_{ij,k}}{c_k} = \int s_k dr_{ij,k}, \quad s_k = \frac{1}{c_k}$$

"slowness"

For discrete blocks

$$t_{ij} = \sum_{k \in r_{ij}} s_k \delta r_{ij,k}$$

Can write formulation in matrix notation

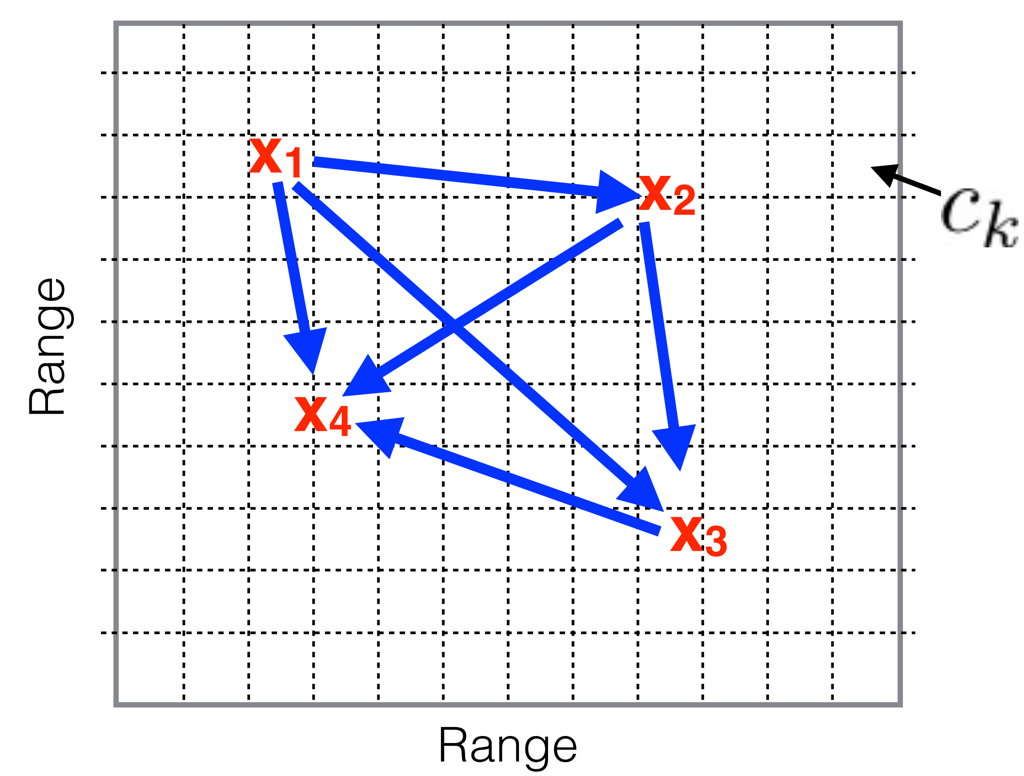
$$\begin{bmatrix} t_{12} \\ \vdots \\ t_{ij} \end{bmatrix} = \begin{bmatrix} \delta r_{12,1} & \cdots & \delta r_{12,k} \\ \vdots & \ddots & \vdots \\ \delta r_{12,k} & \cdots & \delta r_{ij,k} \end{bmatrix} \begin{bmatrix} s_1 \\ \vdots \\ s_k \end{bmatrix} \longrightarrow \boxed{\mathbf{t} = \mathbf{A}\mathbf{s}}$$

$\mathbf{t} \in \mathbb{R}^M, \quad M = n_{\text{rays}}$

$\mathbf{A} \in \mathbb{R}^{M \times K}$

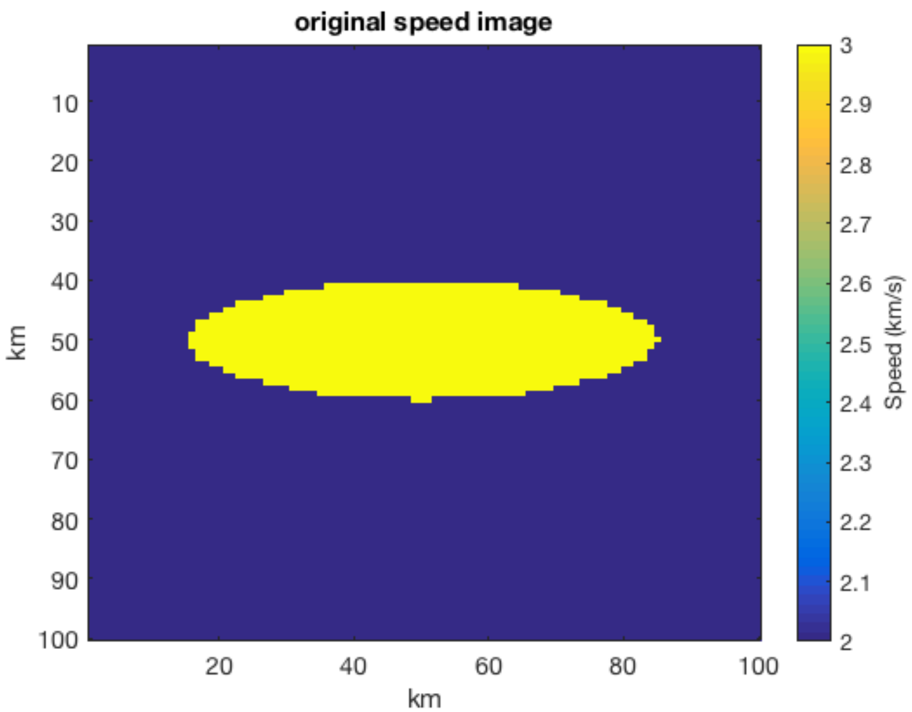
$\mathbf{s} \in \mathbb{R}^K$

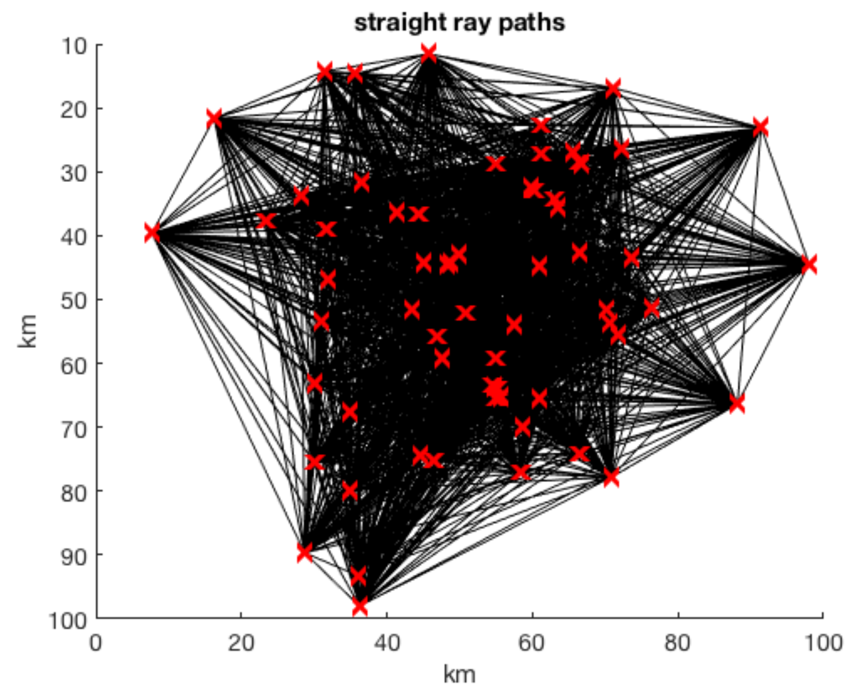
2D map of wave speed



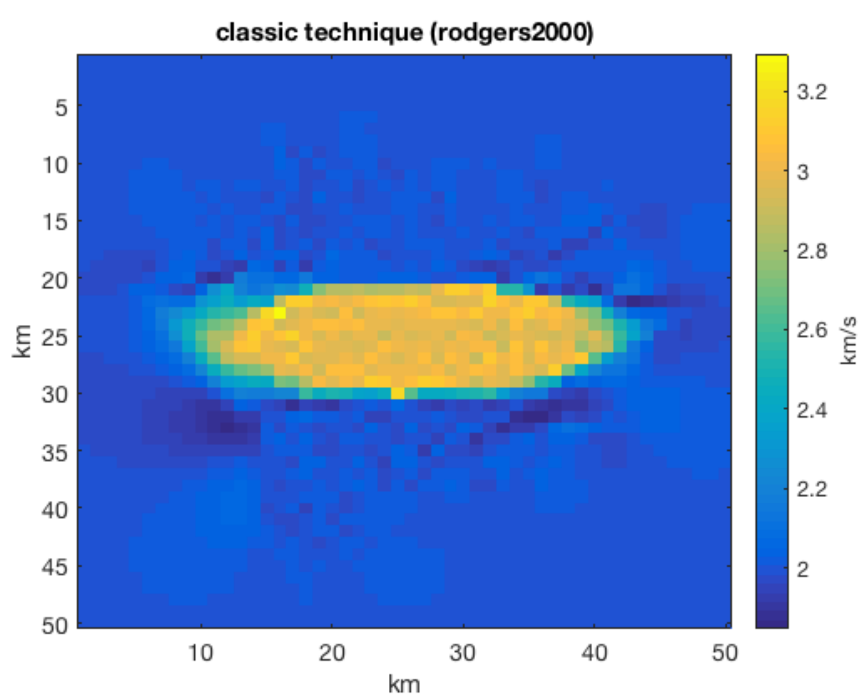
$$n_{\text{rays}} = \frac{n_{sta}(n_{sta}-1)}{2}$$

Example Inversion (unrealistic but illustrative)

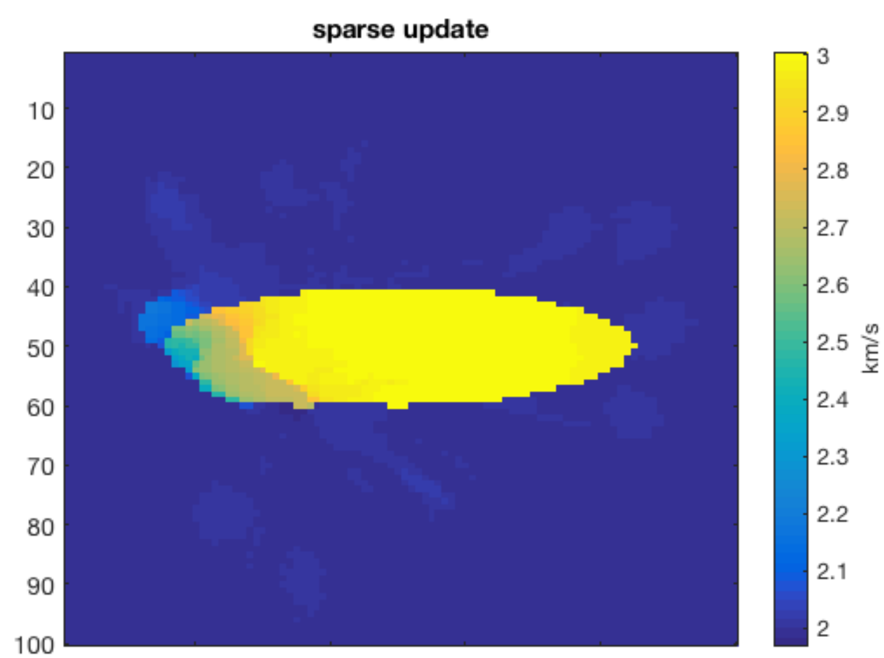




Semi-gaussian distributed stations:
64 stations
2016 ray paths

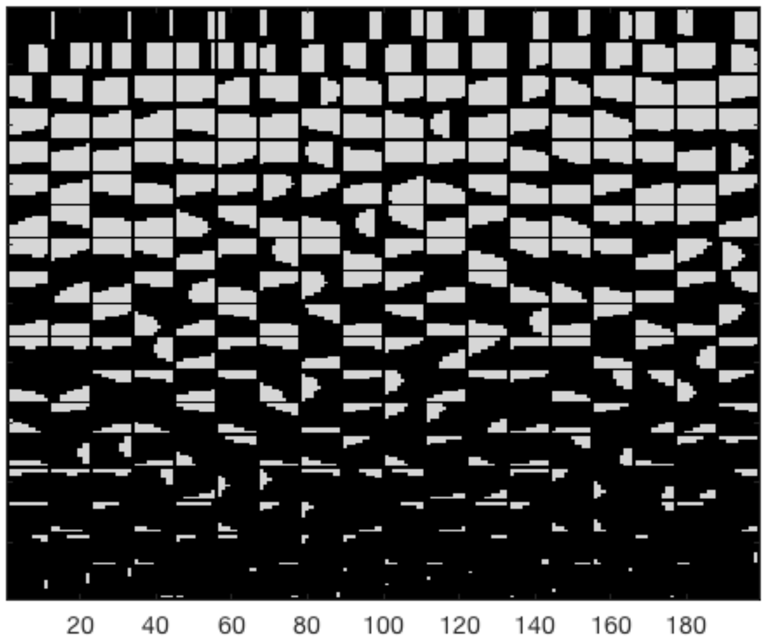


Classic



Patch sparsity

Dictionary (local priors)



322 elements

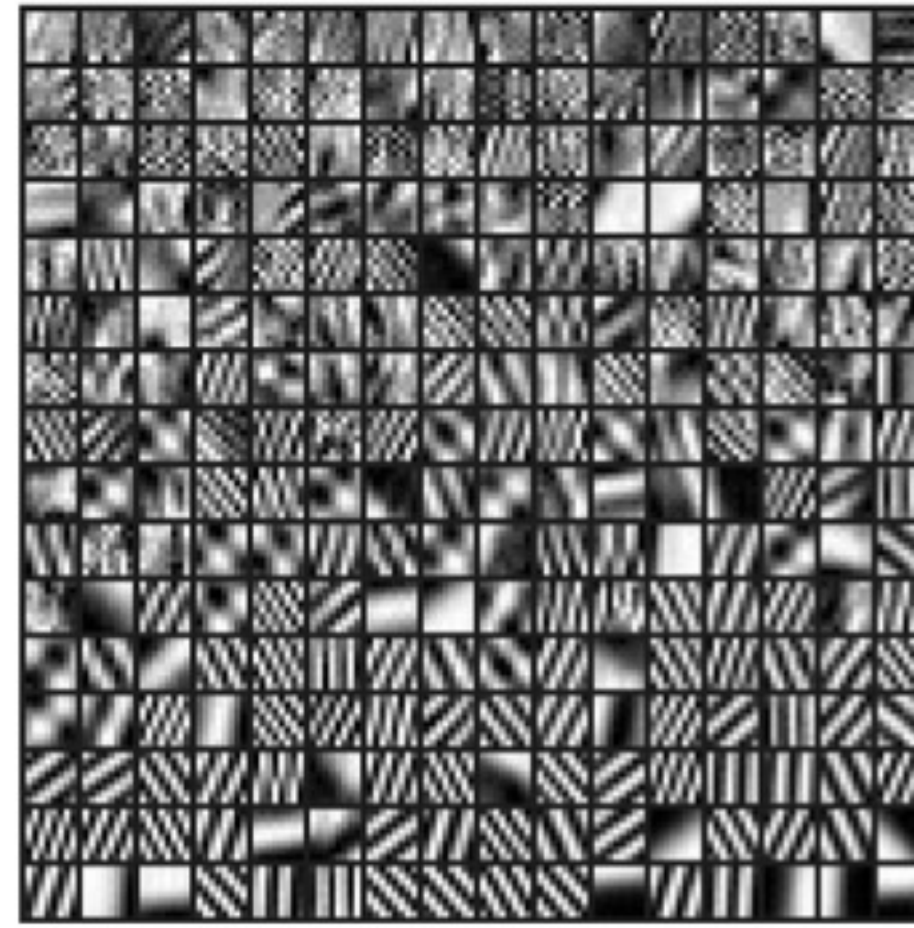
Patch based image denoising

Original Image

Noisy Image (22.1307 dB, σ=20)

Denoised Image Using
Adaptive Dictionary (30.8295 dB)

Learned dictionary (256 atoms, 8x8 pixels)

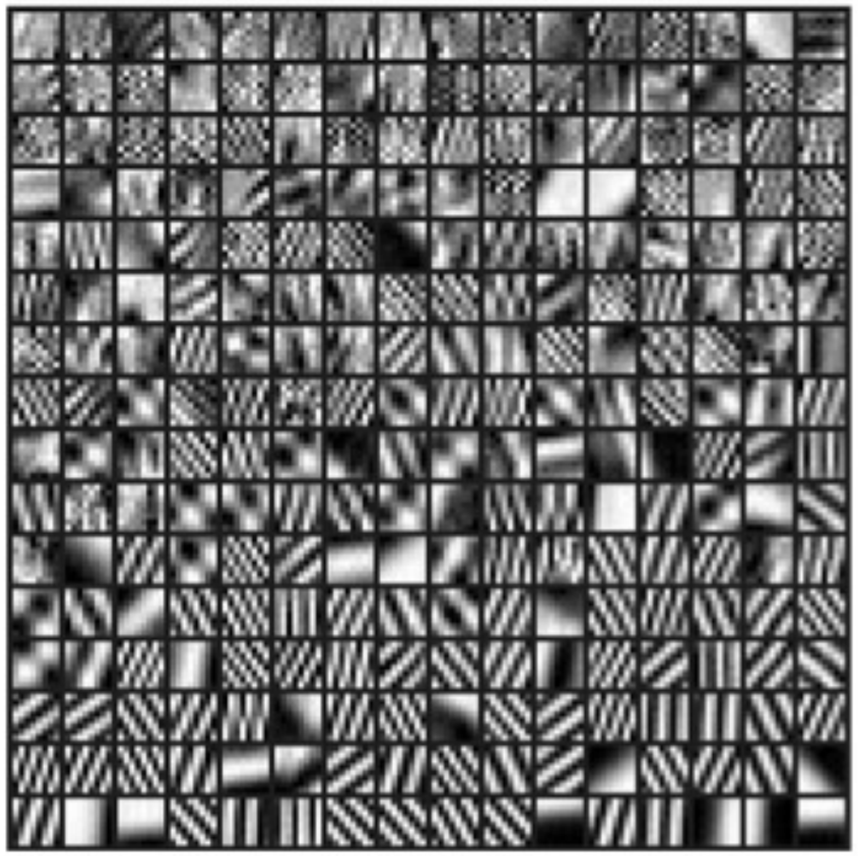


Elad 2006

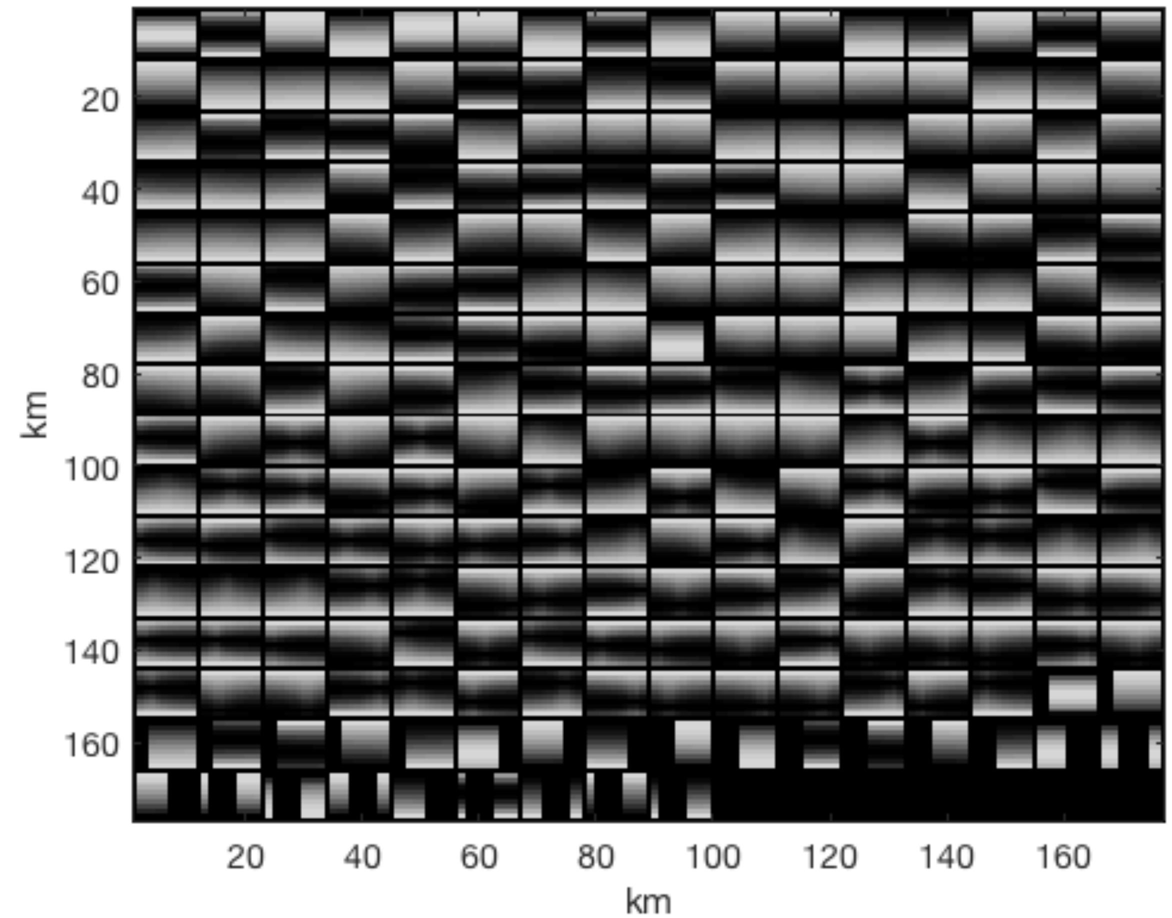
- Patch-based image denoising works by assuming that, at the local or 'patch' level within a digital image, the causes are sparse
- Example: each 8x8 pixel patch within image is represented using few atoms from dictionary trained on noisy image patches
- Iterative 2 step process: (1) local and (2) global solution

Seismic dictionaries?

Image dictionary (256 atoms, 8x8 pixels)



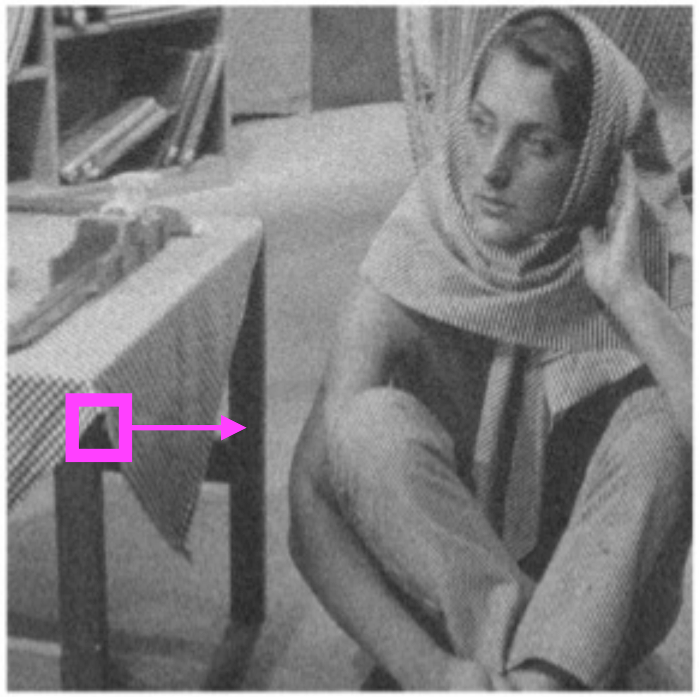
Potential seismic dictionary (267 atoms, 10x10 pixels)



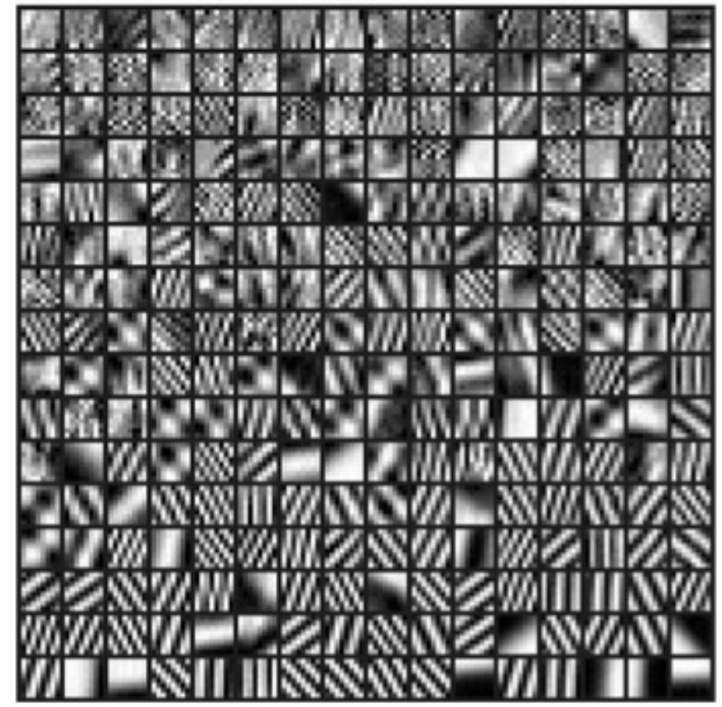
- Since true seismic wave speed maps have smooth and discontinuous features, good candidates for locally sparse priors, similar to natural images
- Questions: (1) can we estimate a dictionary of local seismic patch priors from seismic data, and (2) could this improve results

Some details of patch based image denoising

Noisy Image (22.1307 dB, σ=20)



Learned dictionary (256 atoms, 8x8 pixels)



Alternate between local and global solutions until convergence

- Local solution:
  - Overlapping patches are raster-scanned from image and vectorized for dictionary learning: become training set **Y**
  - Solved by dictionary learning (K-SVD)

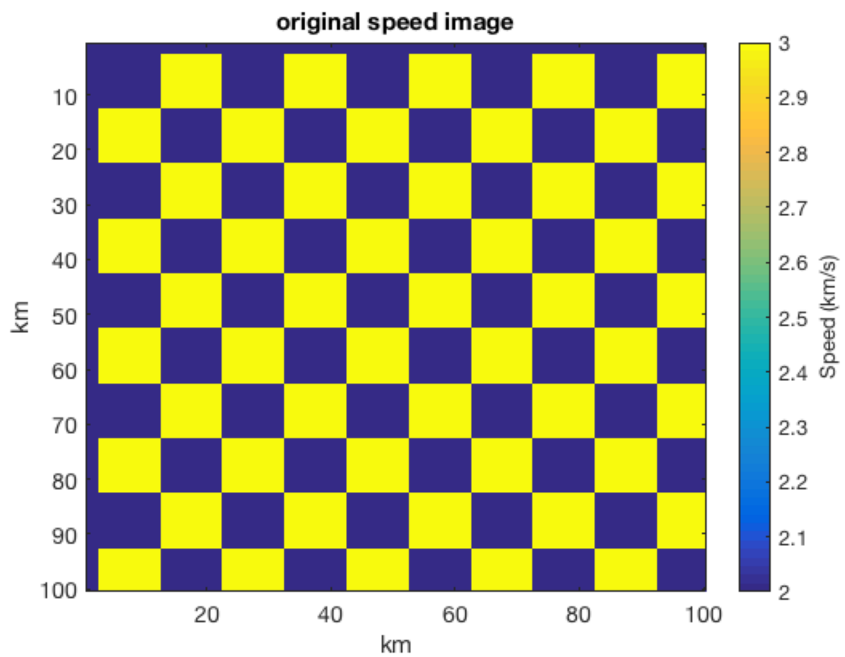
$$\min_{\mathbf{Q}}\{\min_{\mathbf{X}}\|\mathbf{Y}-\mathbf{Q}\mathbf{X}\|_F^2 \ \text{ subject to } \ \|\mathbf{x}_m\|_0 \leq T\}$$

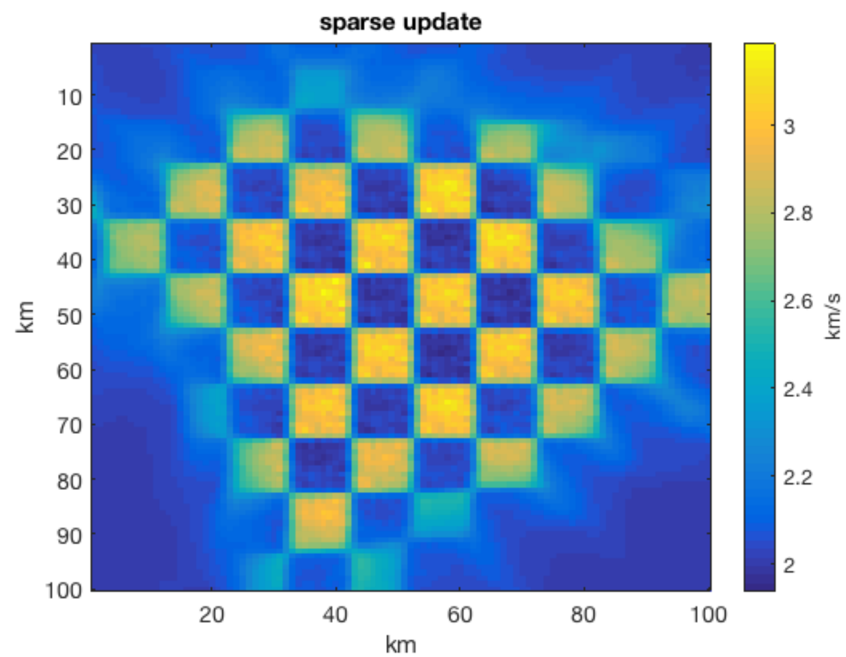
- Global solution: denoised patches are effectively averaged by solving

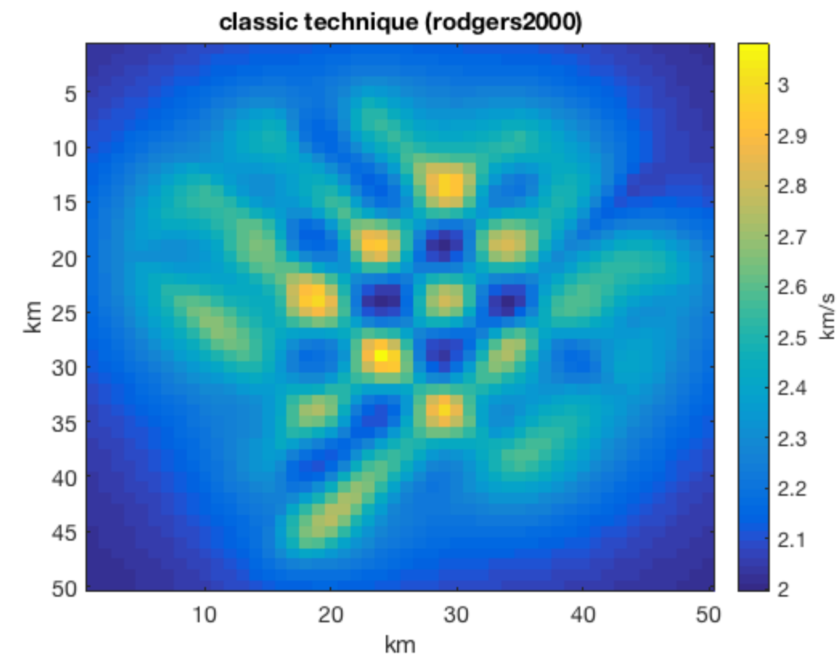
$$\widehat{\mathbf{Z}} = \arg\min_{\mathbf{x}_i,\ \mathbf{Z}} \lambda\|\mathbf{Z}-\mathbf{Y}\|_2^2 + \sum_i \|\mathbf{Q}\widehat{\mathbf{x}}_i - \mathbf{R}_i\mathbf{Z}\|_2^2$$

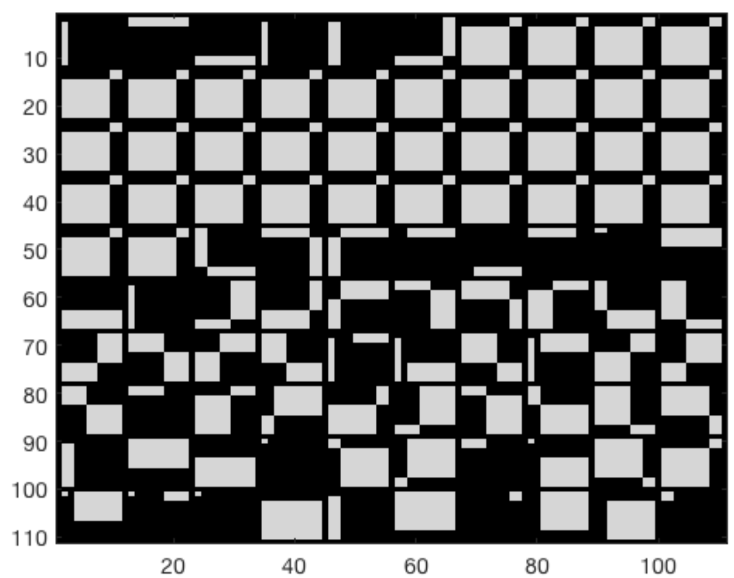
Dictionary learning from seismic data: checkerboard

(unknown dictionary, to be estimated)

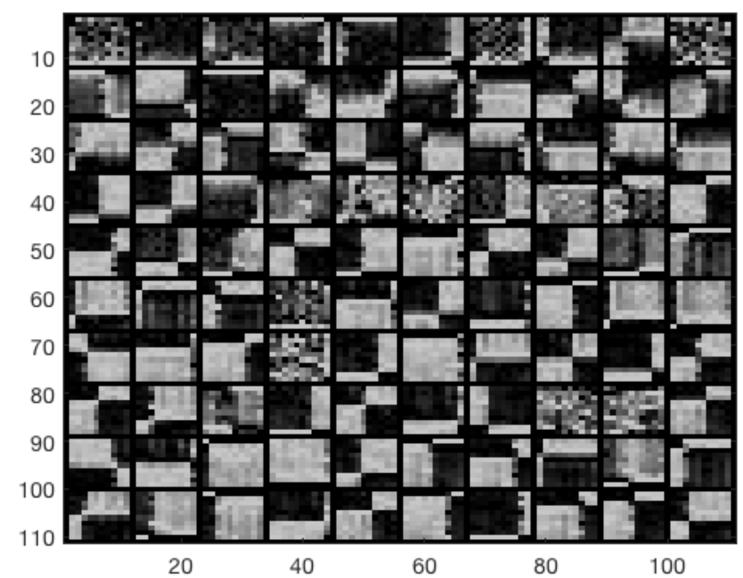




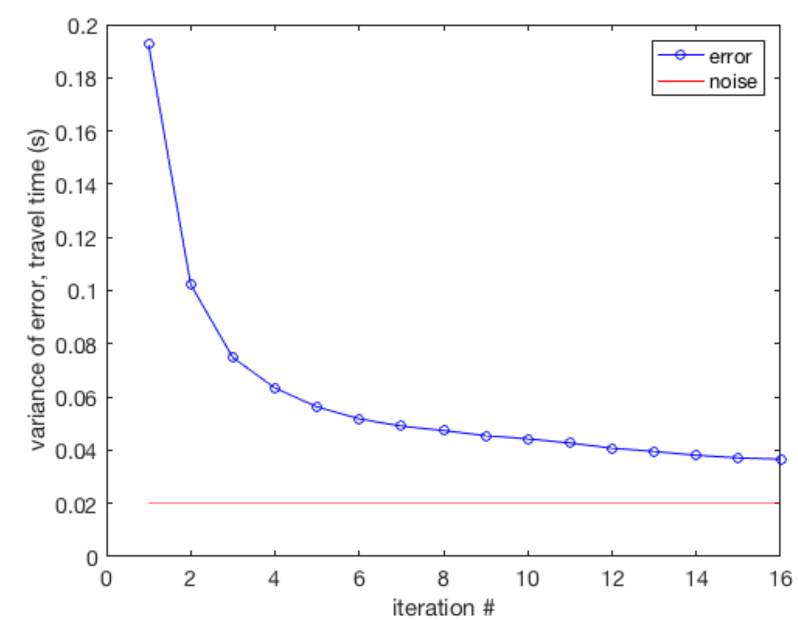




Dictionary learned directly from image data (K-SVD, T=1)



Dictionary learned from simulated seismic data (K-SVD, T=2)



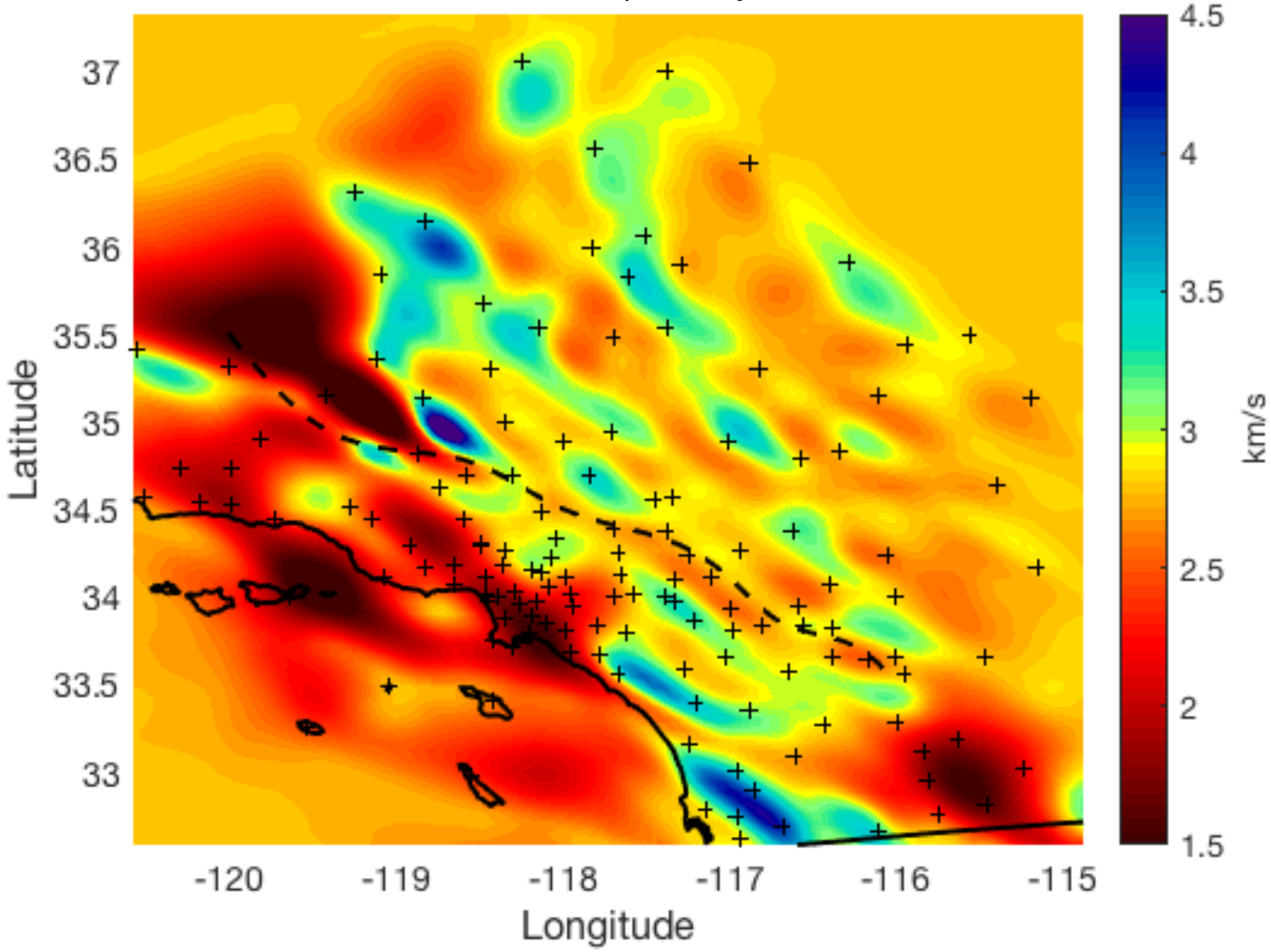
Iteration error

With noise, std of 1% of mean travel time

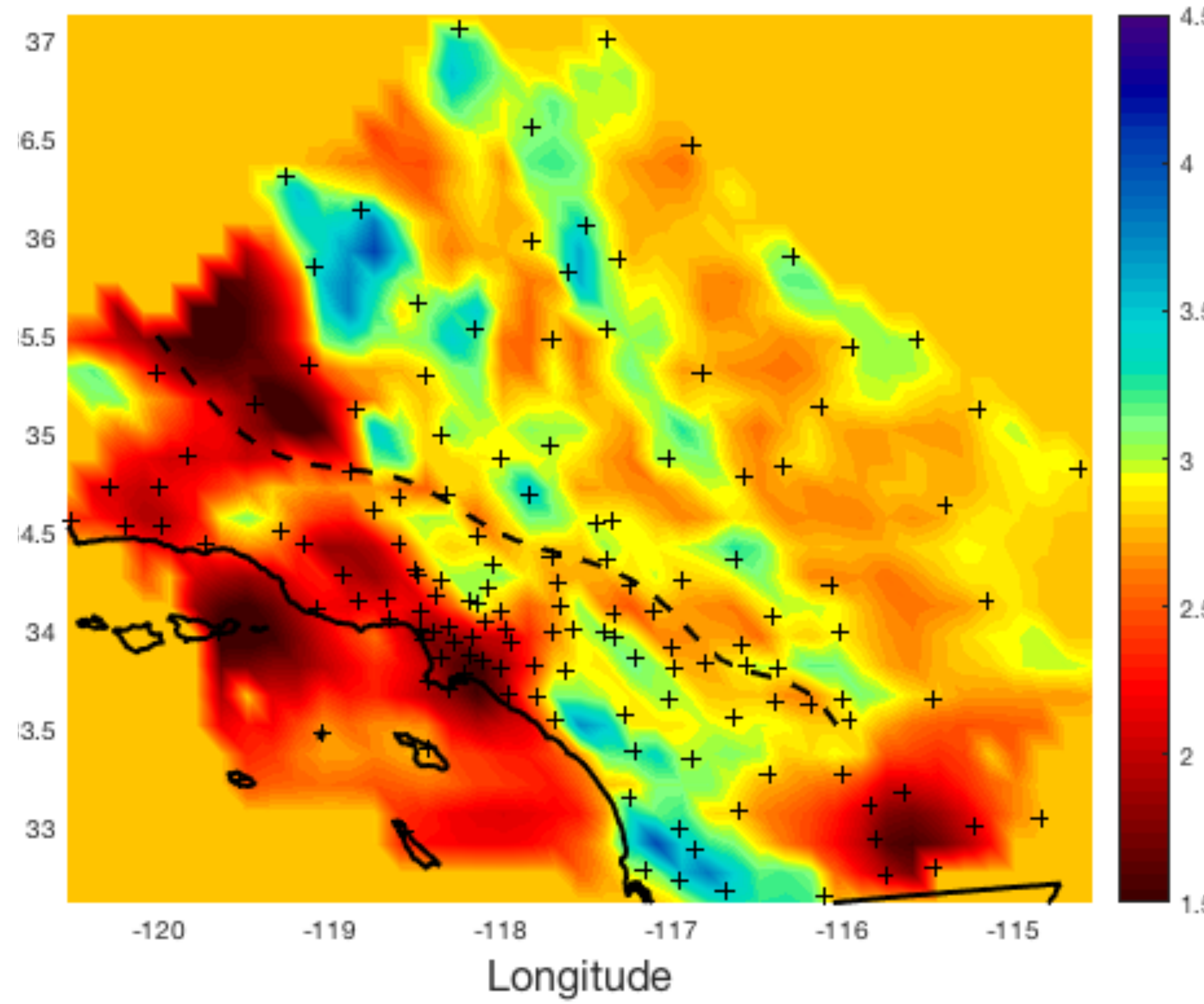
Rayleigh wave tomographic inversion:

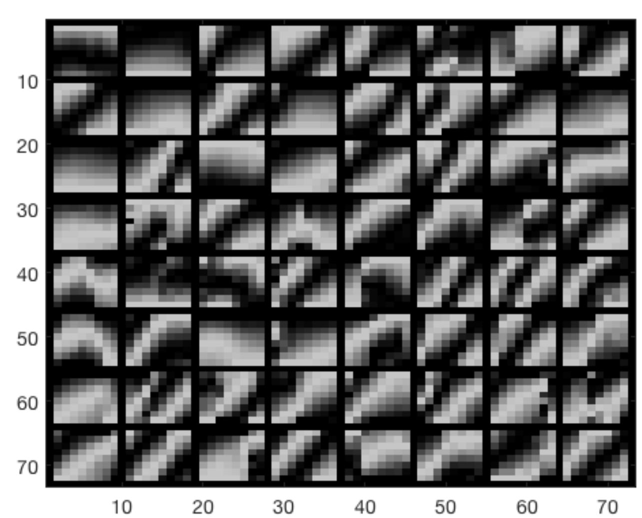
microseism observations on the Southern California Seismic Array

Patch sparsity

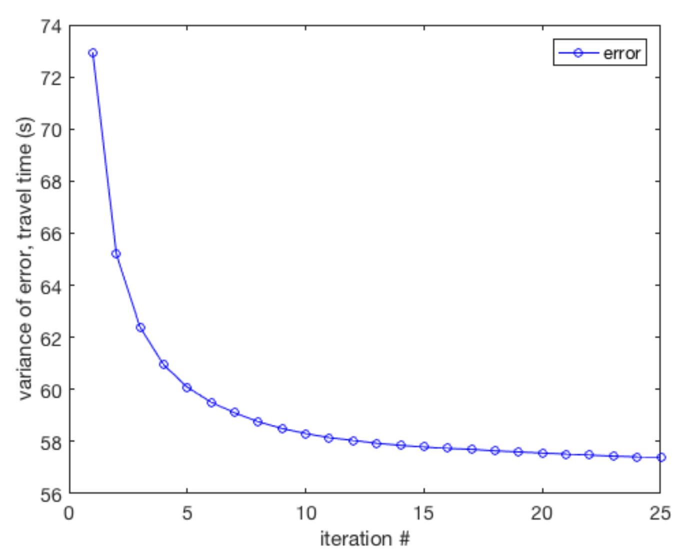


Classic method





Dictionary learned directly from seismic data (K-SVD, T=2)



Iteration error

- 1 month of data
- 151 stations
- SNR_{min} = 15dB
- ~5000 ray paths

Conclusions: Adaptive patch-based seismic inversion

- Seismic inversion regularized with adaptive dictionaries appears to give improved results over at least classic methods
- This method avoids complications associated with MCMC based techniques
- Method currently does not give *a posteriori* error distributions for model estimate