An introduction to sparse coding and regularization methods for temporally coherent sparse codes

Philip Bachman

April 7, 2010

Controlling sparse codes over time

Three parts to this talk[1]:

1. An introduction to sparse coding
2. Sparse coding of temporal sequences
3. Some preliminary results

[1] Please ask questions and make suggestions.

1: Introduction to sparse coding

1. What are sparse codes?
2. How are sparse codes different from other codes?
3. How are sparse codes described mathematically?
4. What do sparse codes have to offer?
5. What do sparse codes encode?

1.1: What are sparse codes?

- Sparse codes are not local.

1.1: What are sparse codes?

- Sparse codes are not local.
- Sparse codes are not dense.

1.1: What are sparse codes?

- Sparse codes are not local.
- Sparse codes are not dense.
- Sparse codes are sparse!
- And there is more than one way to measure sparsity...

1.2: How are sparse codes different from other coding schemes?

- **PCA**: Finds an orthonormal set of bases that point in the directions of maximum (co)variance of the data.

1.2: How are sparse codes different from other coding schemes?

- **PCA**: Finds an orthonormal set of bases that point in the directions of maximum (co)variance of the data.
- **ICA**: Finds a set of maximally independent bases for describing the data.

1.2: How are sparse codes different from other coding schemes?

- **PCA**: Finds an orthonormal set of bases that point in the directions of maximum (co)variance of the data.
- **ICA**: Finds a set of maximally independent bases for describing the data.
- **DCTs, Wavelets, Etc**: Project the data directly onto a set of fixed basis functions with nice mathematical properties.

1.3: Mathematical formulations of sparse coding

Generally speaking, sparse coding methods seek a set of bases and coefficients that optimally reconstruct a set of inputs, or a distribution over inputs, with respect to some loss function defined for the reconstruction error and some sparsity inducing penalty on the reconstruction coefficients.

- $\min_x ||y - Ax||_2^2 + \lambda ||x||_0$

1.3: Mathematical formulations of sparse coding

Generally speaking, sparse coding methods seek a set of bases and coefficients that optimally reconstruct a set of inputs, or a distribution over inputs, with respect to some loss function defined for the reconstruction error and some sparsity inducing penalty on the reconstruction coefficients.

- $\min_x ||y - Ax||_2^2 + \lambda ||x||_0$
- $\min_x ||y - Ax||_2^2 + \lambda ||x||_1$

1.3: Mathematical formulations of sparse coding

Generally speaking, sparse coding methods seek a set of bases and coefficients that optimally reconstruct a set of inputs, or a distribution over inputs, with respect to some loss function defined for the reconstruction error and some sparsity inducing penalty on the reconstruction coefficients.

- $\min_x ||y - Ax||_2^2 + \lambda ||x||_0$
- $\min_x ||y - Ax||_2^2 + \lambda ||x||_1$
- $\min_x ||y - Ax||_2^2 + \lambda ||x||_2^2$??

1.3: Mathematical formulations of sparse coding

Generally speaking, sparse coding methods seek a set of bases and coefficients that optimally reconstruct a set of inputs, or a distribution over inputs, with respect to some loss function defined for the reconstruction error and some sparsity inducing penalty on the reconstruction coefficients.

- $\min_x ||y - Ax||_2^2 + \lambda||x||_0$
- $\min_x ||y - Ax||_2^2 + \lambda||x||_1$
- $\min_x ||y - Ax||_2^2 + \lambda||x||_2^2$??
- $\min_x ||y - Ax||_2^2 + \lambda_1||x||_1 + \lambda_2||x||_2^2$??

1.3: Mathematical formulations of sparse coding

Generally speaking, sparse coding methods seek a set of bases and coefficients that optimally reconstruct a set of inputs, or a distribution over inputs, with respect to some loss function defined for the reconstruction error and some sparsity inducing penalty on the reconstruction coefficients.

- $\min_x ||y - Ax||_2^2 + \lambda||x||_0$
- $\min_x ||y - Ax||_2^2 + \lambda||x||_1$
- $\min_x ||y - Ax||_2^2 + \lambda||x||_2^2$??
- $\min_x ||y - Ax||_2^2 + \lambda_1||x||_1 + \lambda_2||x||_2^2$??
- These problems are called Matching Pursuit, the Lasso, Ridge Regression, and the Elastic Net.

1.3: A probabilistic interpretation of sparse coding

$$\Rightarrow \max_x [p(x|y) = \frac{p(y|x)p(x)}{p(y)}]$$

And $p(y)$ is the same for all x, so:

$$\Rightarrow \max_x [-log(p(x|y))] = \max_x [-log(p(y|x)) - log(p(x))]$$

Which, assuming that $p(y|x)$ is normally distributed and $p(x) \propto e^{-\lambda|x|}$ becomes:

$$\Rightarrow \min_x ||y - Ax||_2^2 + \lambda||x||_1$$

1.3: Sparse coding as constrained optimization

Consider the following optimization problem:

$$\min_x ||y - Ax||_2^2 \text{ s.t. } \sum_i |x_i| \leq c$$

The constraint on the L_1 norm of x turns this problem into an L_1-regularized least-squares problem when we introduce Lagrange multipliers to change to an unconstrained optimization problem. For each value of c in this problem, there exists an L_1 regularization weight for the unconstrained problem such that the solutions produced are the same.

1.4: What do sparse codes have to offer?

- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian visual cortex (e.g. V1/V2).

1.4: What do sparse codes have to offer?

- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian visual cortex (e.g. V1/V2).
- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian auditory cortex.

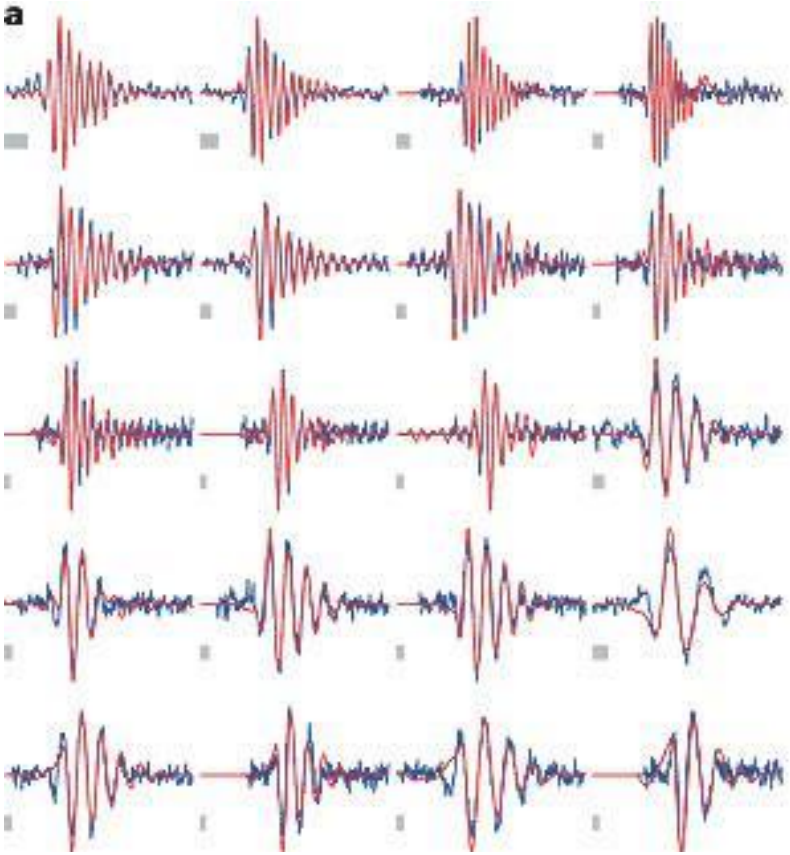
1.4: What do sparse codes have to offer?

- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian visual cortex (e.g. V1/V2).
- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian auditory cortex.
- Sparse codes are an energy efficient form of data representation.

1.4: What do sparse codes have to offer?

- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian visual cortex (e.g. V1/V2).
- Sparse coding produces sets of bases that resemble measured receptive fields in parts of mammalian auditory cortex.
- Sparse codes are an energy efficient form of data representation.
- Sparse coding has shown promise as a method for unsupervised feature extraction and selection.

1.5: What do sparse bases look like?



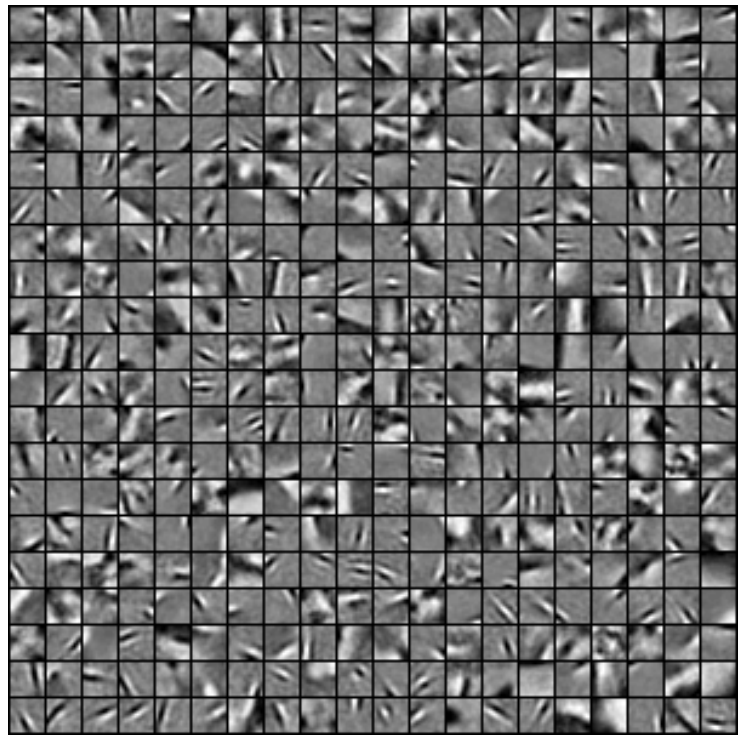


Figure: Left: Sparse bases learned for natural audio signals. Right: Sparse bases learned for natural images.

2: Approaches to sparse coding of temporal sequences

1. Two approaches to temporal smoothing of sparse codes
2. Algorithms for regularizing temporal coherence of sparse codes
3. Behavior of temporally regularized sparse codes

2.1: Two approaches to temporal smoothing of sparse codes

Goal: generate sequences of coefficients that are easier to predict, and maybe easier to approximate, both of which should be facilitated by smoother code trajectories.

2.1: Two approaches to temporal smoothing of sparse codes

Goal: generate sequences of coefficients that are easier to predict, and maybe easier to approximate, both of which should be facilitated by smoother code trajectories.

- Use the implicit supervisory signal provided by temporal proximity.

2.1: Two approaches to temporal smoothing of sparse codes

Goal: generate sequences of coefficients that are easier to predict, and maybe easier to approximate, both of which should be facilitated by smoother code trajectories.

- Use the implicit supervisory signal provided by temporal proximity.
- Encourage coefficients for temporally adjacent inputs to have either small L_1 or L_2 distance.

2.1: Two approaches to temporal smoothing of sparse codes

Goal: generate sequences of coefficients that are easier to predict, and maybe easier to approximate, both of which should be facilitated by smoother code trajectories.

- Use the implicit supervisory signal provided by temporal proximity.
- Encourage coefficients for temporally adjacent inputs to have either small L_1 or L_2 distance.
- These correspond to assumptions about a prior distribution over *changes* in the coefficients.

2.1: Two approaches to temporal smoothing of sparse codes

Goal: generate sequences of coefficients that are easier to predict, and maybe easier to approximate, both of which should be facilitated by smoother code trajectories.

- Use the implicit supervisory signal provided by temporal proximity.
- Encourage coefficients for temporally adjacent inputs to have either small L_1 or L_2 distance.
- These correspond to assumptions about a prior distribution over *changes* in the coefficients.
- Additionally, minimizing L_2 distance simultaneously maximizes correlation.

2.2-L_1: Algorithms for regularizing temporal coherence of sparse codes

Regularizing with an exponential prior, corresponding to penalizing the L_1-norm of changes in our coefficients, gives the following optimization problem:

$$\min_x ||y - Ax_t||_2^2 + \lambda(\alpha||x_t||_1 + (1-\alpha)||x_t - x_{t-1}||_1),$$

for which an efficient algorithm can be derived by generalizing the Feature-Sign algorithm presented in "Efficient Sparse Coding Algorithms" by H. Lee et. al. Briefly, the noted algorithm and its generalization both rely on the piecewise-linear and convex nature of the regularization function to search through locally convex, differentiable spaces for an optimal solution.

2.2-L_2: Algorithms for regularizing temporal coherence of sparse codes

Regularizing with a Gaussian prior, corresponding to penalizing the L_2-norm of changes in our coefficients, gives the following optimization problem:

$$\min_x ||y - Ax_t||_2^2 + \lambda(\alpha||x_t||_1 + (1-\alpha)||x_t - x_{t-1}||_2^2),$$

for which an efficient algorithm can be derived through the simple problem transformation shown below:

$$\text{Let : } A' = \begin{bmatrix} A \\ \alpha I \end{bmatrix}, \; y' = \begin{bmatrix} y \\ \alpha x' \end{bmatrix},$$

Then, choosing α appropriately gives:

$$||y' - A'x||_2^2 + \lambda_s||x||_1 = ||y - Ax||_2^2 + \lambda_s||x||_1 + \lambda_c||x - x'||_2^2$$

2.3: Behavior of temporally regularized sparse codes

In the results being presented, all image sequences and regularization regimes made use of the same set of bases. This was done in an attempt to avoid confounding the effects of coherence regularization on basis learning with its effects on coding.

2.4: Results

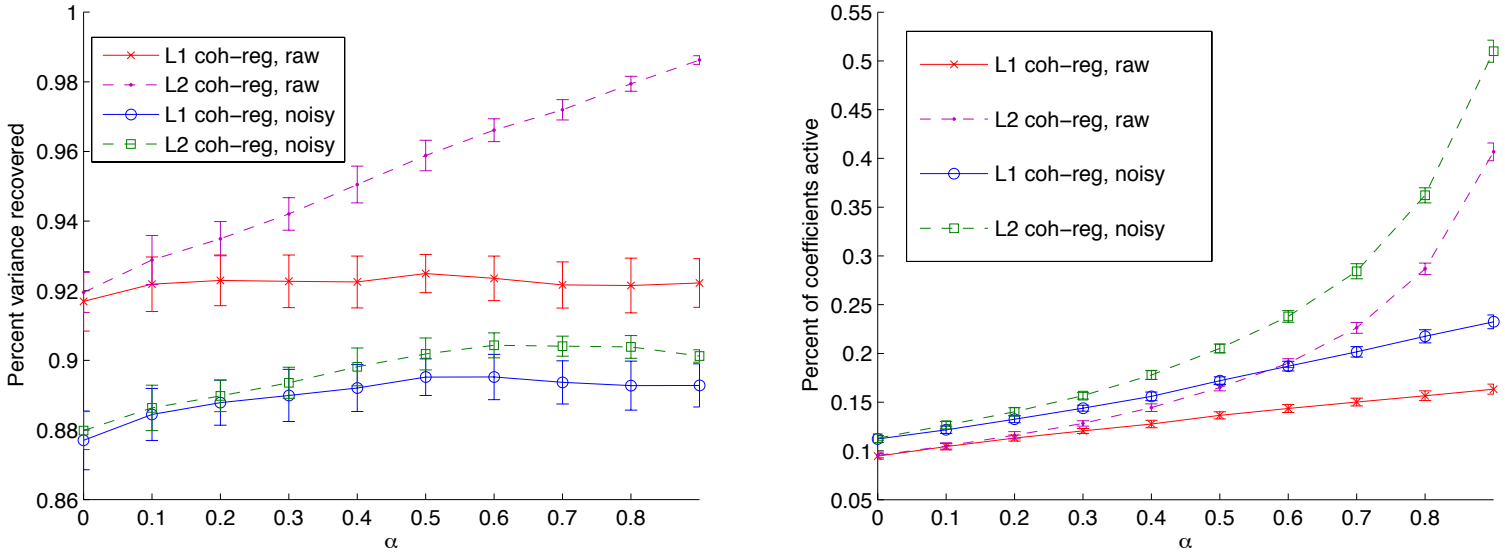


Figure: Left: Accuracy, Right: Sparsity. The measures were taken using either L_1 or L_2 coherence regularization for encoding image sequences both with and without noise. The parameter α controls the relative weighting of instantaneous versus temporal sparsity.

2.4: Results

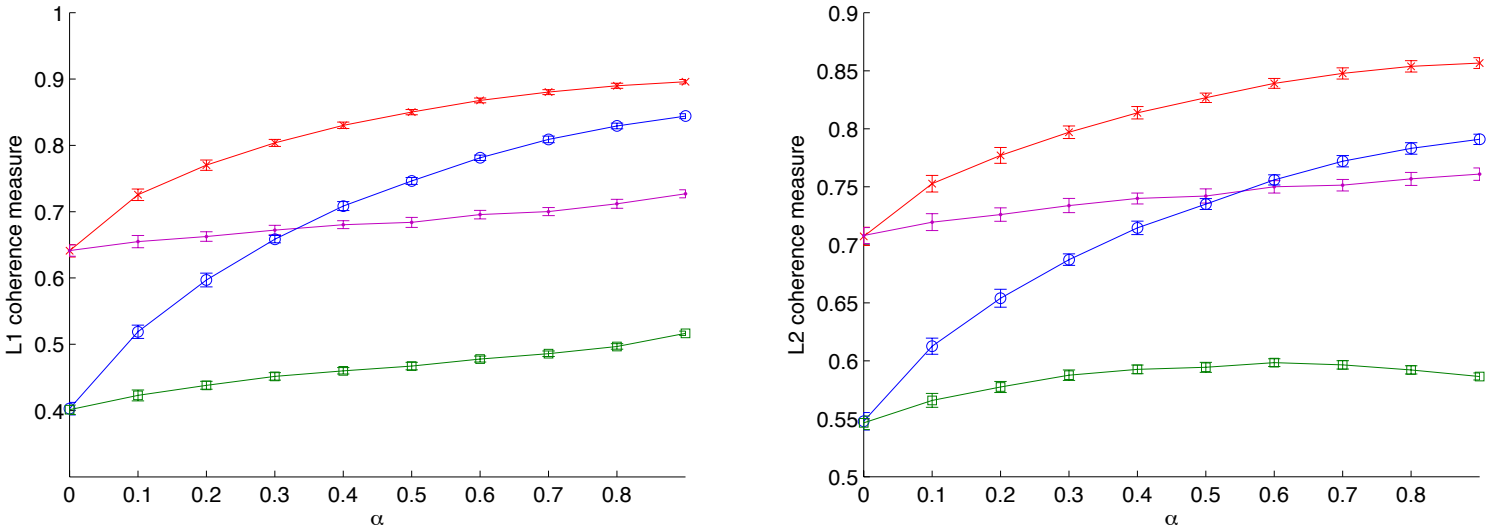


Figure: Left: L_1 coherence, Right: L_2 coherence. These measures represent the degree to which codes for temporally adjacent images were similar with respect to the L_1 and L_2 norms. The measures were taken using either L_1 or L_2 coherence regularization for encoding image sequences both with and without noise. The parameter α controls the relative weighting of instantaneous versus temporal sparsity.

2.4: Results

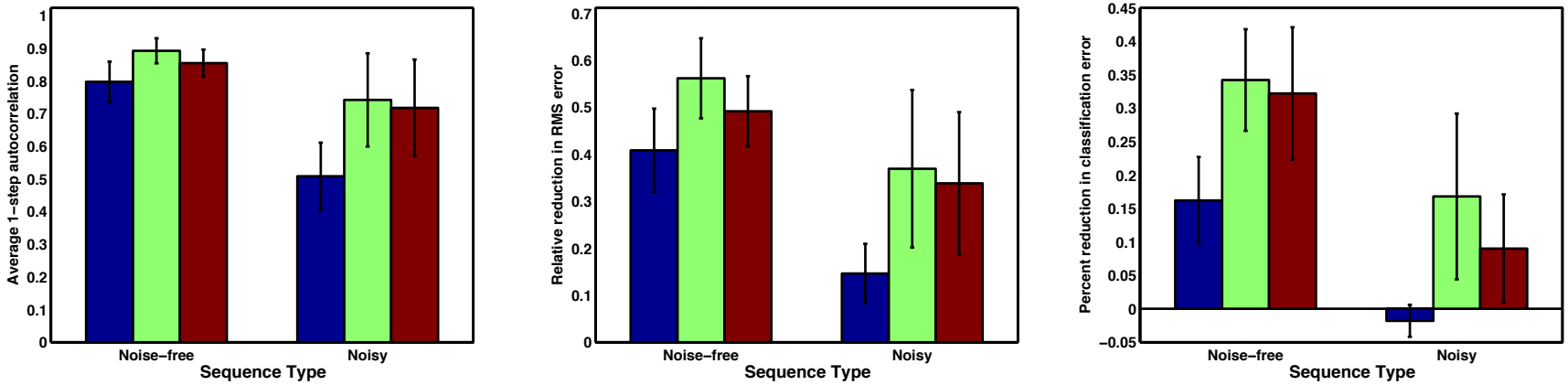


Figure: Left to Right: Linear regression correlation, linear regression prediction accuracy, logistic regression prediction accuracy. Left bar sets are for noise-free sequences and right bar sets are for noisy sequences. In each bar set, the regularization types are (left to right): no coherence regularization, L_1 coherence regularization, L_2 coherence regularization. Measurements all represent improvements offered by the various code types, versus a naive classifier.

2.5: Brief Aside

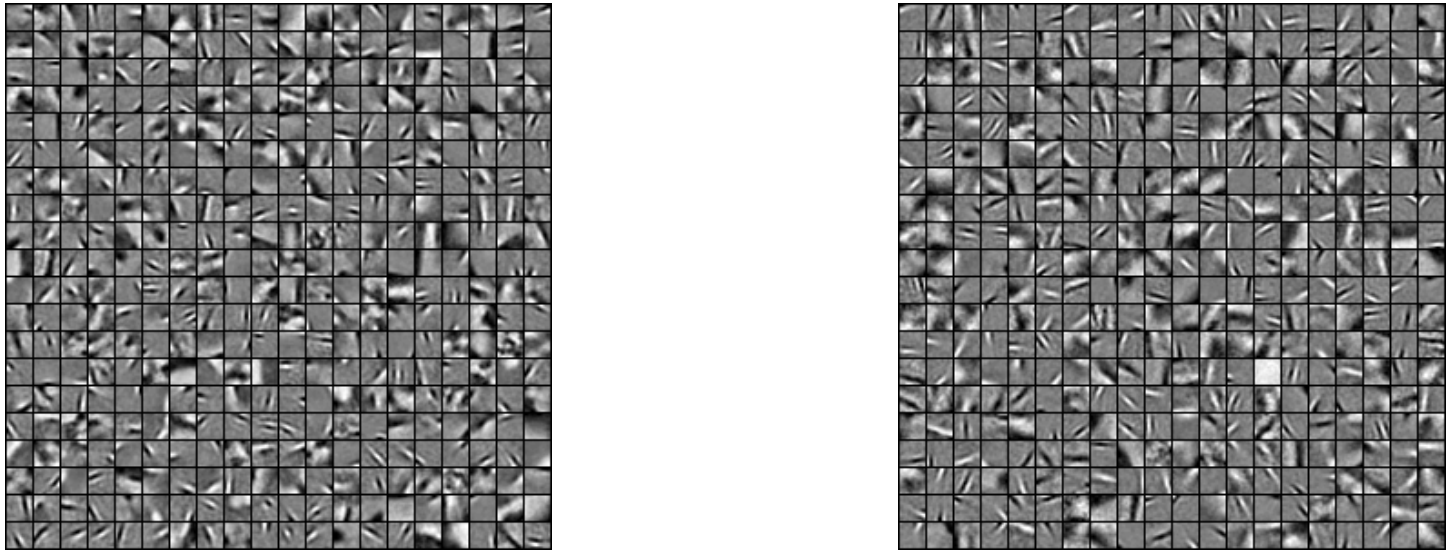


Figure: Left: Bases learned while regularizing solely for instantaneous or population sparsity. Right: Bases learned while regularizing solely for temporal or lifetime sparsity. Notice that optimizing for either population sparsity or lifetime sparsity produces qualitatively similar bases.