Title
Statistical learning models of sensory processing and implications of biological constraints

Author
Dodds, Eric McVoy

Publication Date
2018

Peer reviewed|Thesis/dissertation

Statistical learning models of sensory processing and implications of biological constraints

by

Eric Dodds

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Physics

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Michael DeWeese, Chair
Professor Irfan Siddiqi
Professor Bruno Olshausen

Fall 2018

Statistical learning models of sensory processing and implications of biological constraints

Abstract

Statistical learning models of sensory processing and implications of biological constraints

by

Eric Dodds

Doctor of Philosophy in Physics

University of California, Berkeley

Professor Michael DeWeese, Chair

Despite progress in understanding the organization and function of neural sensory systems, fundamental questions remain about how organisms convert visual, auditory, and other sensory input into useful representations to understand the world and guide behavior. An important and fruitful line of work models the brain as an unsupervised statistical learner, examining how a sensory system may optimize for efficient representation of the natural environment or for explicit representation of useful structure in that environment. This dissertation explores efficient coding and sparse coding models of the visual and auditory systems, the data these systems process, and how these models are affected by the constraints imposed by implementation in biological neural systems. First, I show that both natural images and natural sounds have statistical structure amenable to a sparse coding model but that the sparse structure of these two types of natural data also differ in interesting ways that may be relevant to extending the success of sparse coding in describing primary visual cortex (V1) to analogous regions of the auditory system. I also discuss how a related model may shed light on how the neurons in these sensory systems are organized in space based on coding for related stimulus properties. Second, I show that a sparse coding model with biological constraints requires its inputs to be whitened in order to learn sparse features using synapse-local learning rules. This observation provides a novel explanation for the separation of sparse coding and spatial decorrelation into, respectively, V1 simple cells and preceding areas including retina. Third and finally, I turn back to the auditory system and extend existing work on efficient coding in the cochlea to account for the requirement of causality, i.e., determining a code without knowledge of the future of a signal.

to Rachel

Contents

Acknowledgments

Although nearly all the content of this document is the direct result of keystrokes from my own fingers, I could never have produced any such thing in isolation. I will try to thank some of the individuals whose support is most salient, for the my own benefit from the exercise of reflection and in case any should read this and enjoy my appreciation. I also acknowledge the many great scientists on whose work I have attempted to build.

My parents deserve a great deal of the credit and very little of the blame for everything I have done.. My brother Scott provided a powerful example of intellectual curiosity, even if he might think I still study lasers or something. I am also grateful for the love and support of my extended family.

I owe much of my knowledge, skills, understanding, and motivation to the many excellent teachers of whom I have been a more or less devoted student. Among those whose names I can recall (with sincere apologies for any mistakes) and whose contributions I can specify, I thank Mrs. Welker for encouraging me to write; Debbie Van Ryn for calling out laziness; Darlene Self and Kathy Scheff for teaching humility to a terrible student of it; Tim Morrison and John Jauss for showing me how interesting physical science could be; Scott Degitz for thankless toleration and calculus; Jane Ibur and Ted Ibur for nurturing my creativity; Shahriar Shahriari for linear algebra with fat strawberries; Stephan Garcia for linear algebra with penguins; Erica Flapan for making difficult math fun; Alfred Kwok for physics and quirks; Tom Moore for physics with clarity; Dwight Whitaker for physics with lasers and also clarity; and of course all my graduate school instructors for the depth and refinement of their knowledge.

I served as a graduate student instructor every semester (except my last) of my PhD, and I thank all the instructors I worked with for helping me learn how to communicate with and help students and to make efficient use of my time. I also learned more physics and statistics even than I wanted to from Andrew Charman, and the working relationships I built with Adrian Lee and Irfan Siddiqi also led to valuable mentorship as well as participation on my qualifying exam committee. I am also grateful to all the GSIs who worked with me while I served as a Head GSI, for their hard work and for letting me learn management on the fly.

I thank my qualifying exam committee for their valuable feedback, and I especially thank Stan Klein for filling in at the eleventh hour and providing his valuable perspective on the early stages of some of the work in this dissertation.

I also want to thank the physics department staff and especially Anne Takizawa, Donna Sakima, Joelle Miles, and Amanda Dillon.

Berkeley has surrounded me with excellent peers, and I am especially grateful for those who became personal friends. It is hard to imagine that I would have finished the PhD without complaing to Aaron Szasz, Lenny Evans and Ziqi Yan about it at lunchtime. I also want to acknowledge Jesse Livezey for his advice and collaboration, Alex Anderson for telling it like it was, and both of them as well as Dylan Paiton for helping me transition to the theoretical neuroscience in the middle of my PhD.

It was a great privilege to work among and know all the members of the Redwood Center for Theoretical Neuroscience, including but not limited to Charles Frye, Pratik Sachdeva, Neha Wadia, Ryan Zarcone, Mayur Mudigonda, Shariq Mobin, James Arnemann, Michael Fang, Yubei Chen, Vasha DuTell, Sylvia Madhow, Sophia Sanborn, Chris Warner, Brian Cheung, Spencer Kent, Jasmine Collins, Sean Mackesey, Steven Shepard, Saeed Saremi, Paxon Frady, Pentti Kanerva, Jeff Teeters, Fritz Sommer, and Kris Bouchard. Nothing else I have seen rivals the intellectual liveliness of a Friday Redwood group meeting, and I hope to continue to draw on the creativity and thoughtfulness of the group after I graduate. Besides his direct contributions to my work, I thank Bruno Olshausen for his leadership of the Redwood Center.

It is unlikely that I would have finished my PhD if Mike DeWeese had not welcomed me into his group in the middle of my time at UC Berkeley. Mike provided the opportunity and the direction to do this research, and his indefatigable enthusiasm helped me persist as much as his technical and professional mentorship guided my work and development.

I thank Aaron Szasz, Lenny Evans, Ziqi Yan, Ben Ponedel, and Maribel Sierra for many great meals and games; Zack Lasner, Alex Groth, Sarah Roh, Alexis Chuck – and more others than I could list – for their continued friendship.

Finally I dedicate this dissertation to Rachel Midori Chin, my partner in life.

Chapter 1

Introduction

There is no shortage of mysteries in neuroscience; this dissertation concerns itself with a particular class of questions around modeling neural sensory systems as performing statistical inference and learning. In this introductory chapter, I will provide a brief overview of the context and viewpoint on which the succeeding chapters are based and outline the structure of the rest of the dissertation.

1.1 Efficient coding, redundancy reduction, and perception as inference

This line of work can be traced back at least to the suggestions of Attneave [4] and Barlow [7] that "sensory relays recode sensory messages so that their redundancy is reduced but comparatively little information is lost" [7]. Here information is meant in the sense of Shannon [92] and redundancy refers to mutual information between relays, e.g., one neuron's action potential timings being predictive of another neuron's. Such redundancy could arise from the internal structure of the brain such as neurons with shared inputs, but regardless some redundancy is introduced in the inputs to the neural system. The natural world has statistical structure that leads to redundancy between, for example, the intensity of light striking adjacent photoreceptors in the retina.

The hypothesis of "redundancy reduction" or "efficient coding" in neural sensory systems provided a notion of optimality for a system confronted with structured inputs: the system's outputs should be statistically independent when its inputs come from the natural environment. Studying the statistical structure of the natural world has thus led to insights about and more specific hypotheses for the function of sensory systems. An important example to which we will return in Chapter 6 is the observation of the $1/f$ Fourier amplitude spectrum of natural images [30] and the theory that retinal ganglion cell receptive fields are optimized to account for this redundancy together with noise at high spatial frequencies [2].

Reflecting forty years later on his redundancy reduction hypothesis, Barlow wrote that "the idea was right in drawing attention to the importance of redundancy in sensory messages

because this can often lead to crucially important knowledge of the environment, but it was wrong in emphasizing the main technical use for redundancy, which is compressive coding" [6]. It may be that sensory systems sometimes use the statistical structure of their inputs simply to transmit the information to another population of neurons with greater fidelity using less resources – this is one view of the optic nerve, for example. But redundancy in sensory signals also provides important information about the environment, so modeling the structure that gives rise to redundancy can help the system determine the sources of sensory signals. The idea that sensory systems use their inputs to infer knowledge of the environment goes back to at least Helmholtz [37] and can be seen as the guiding principle behind a great deal of modern research into neural sensory systems, including this dissertation.

1.2 Sparse coding

The work in this dissertation builds on the hypothesis that natural signals, and specifically natural images and natural sounds, can be modeled as sparse combinations of a fixed dictionary of elementary signals. A system that knows this dictionary can then seek to infer which are present in a given input.

Sparse coding refers to a probabilistic model wherein each datum is a sparse linear combination of elements from a fixed dictionary plus some noise. Fitting such a model to patches of grayscale natural images, Olshausen and Field [76] found that the dictionary elements resemble the receptive fields that had been measured in primate primary visual cortex (V1) simple cells [40]. This resemblance suggests a model of these simple cells as encoding the sparsely present elements of natural images in the strengths of their afferent synaptic connections. That this model also reproduces so-called "extra-classical receptive field effects" [108] lends further plausibility to this simple model, and later efforts improved the fit to neural data [82, 73, 110].

1.3 Biological plausibility

Brains are not fabricated from silicon in a factory, nor do they operate in the imaginations of scientists theorizing about their function. We have already hinted at the need to consider how an abstract idea such as probabilistic inference maps onto the mechanisms of neurons and other brain matter in mentioning how the parameters of a sparse coding model may correspond to synaptic strengths in cortex. In this dissertation we will consider how implementation in systems of biological neurons may constrain or influence the algorithms used, their interaction with other brain areas, and how we interpret the parameters. From the perspective of Marr's levels of analysis [66], we will consider how reasoning at the hardware and mechanistic levels may need to influence our reasoning at the algorithmic level in the context of spare coding models of early vision and audition.

We will therefore discuss models that are more "biologically plausible" than a simple latent variable model mapped directly onto neural activations and connections. Specifically, we will consider the implications of using synaptic plasticity rules that use only local information. We will also consider the challenge of encoding a time-varying signal in a causal manner, without knowing the future of the signal.

1.4 Structure of this dissertation

Now that we have introduced the main ideas at a high level of abstraction, Chapter 2 will provide a technical exposition of specific models and algorithms that will feature in the chapters that follow, especially sparse coding. In Chapter 3 we focus on SAILnet, a sparse coding algorithm that respects certain biological constraints and that we will use in later chapters. The remaining chapters contain the bulk of the novel research in this dissertation, beginning with Chapter 4 on a comparison of the sparse structure of natural sounds and images. After a brief chapter (5) on a model of organization of auditory cortex by response properties that builds on the auditory models of Chapter 4, two chapters follow in the style of short research papers. Chapter 6 proposes an explanation for spatial whitening in the retina as a requirement for learning a sparse model of natural scenes in V1. Chapter 7 takes a closer look at a well-known efficient coding model of the auditory nerve and proposes a modification to account for a previously unappreciated discrepancy with the data by requiring the model to process sounds in a causal fashion.

Chapter 2

Preliminaries

In this chapter we describe some of the main technical ideas in a unified and pedagogical way. Some of this material will be recapitulated in the interest of making each chapter more-or-less self-contained, but I hope here to provide a reference and introduction that would have helped a younger me approach this material. This is a dissertation for a physics PhD, so the assumed background is the mathematics, probability, and statistics knowledge of a physicist; and the perspective and intuitions are those of a student of physics.

2.1 Sparse coding

A collection of numbers is said to be sparse if most of the numbers are zero or approximately so. For our purposes, this requirement will often be too strict. Take for example the output of a (nontrivial) linear filter convolved with a natural image. The output will almost never be zero, and the distribution of the outputs will in fact be approximately Gaussian over a region with roughly even illumination. But for certain special filters the outputs will approximately follow a Laplace distribution, which has more probability mass at large value and very small values. We will also call such distributions "sparse."

For us, "sparse coding" refers to a linear generative model[1]

$$x_i = \hat{x}_i + n_i = \sum_m a_m \Phi_{mi} + n_i \tag{2.1}$$

where we assume that n_i is iid Gaussian noise[2] with variance σ^2, conditioned on the set of coefficients a:

$$p(x|a;\Phi) \propto \prod_i e^{-(x_i - \sum_m a_m \Phi_{mi})^2/2\sigma^2} \tag{2.2}$$

[1] Like most technical terminology, several of these words have suffered some abuse. One point that may cause confusion: sparse *coding* sometimes refers only to finding sparse codes for data (the a in our notation), while the probabilistic model and fitting it (finding Φ) may be termed sparse modeling.

[2] iid is short for "independent and identically distributed." By noise we mean roughly "that which we are not modeling" and do not necessarily have a particular source in mind.

Sparseness arises from the assumption of a sparse and (often) independent prior for the coefficients a_m^μ, for example

$$p_a(a) = \prod_m \frac{1}{2\lambda} e^{-\lambda |a_m|}. \tag{2.3}$$

While this is a common example, we may also refer to models with other sparse distributions, including non-factorial distributions, as sparse coding models.

The model distribution can be expressed

$$p(x;\Phi) = \int p(x|a;\Phi) p_a(a) da. \tag{2.4}$$

Parameter estimation and inference

Given a dataset $\{x\}$ such as patches of natural images we usually want the values of the model parameters Φ (i.e., the dictionary elements) that maximize the likelihood of the data under the model. Usually we learn these parameters by a gradient-based method; the gradient is

$$\begin{aligned} \frac{\partial p(x;\Phi)}{\partial \Phi_{mi}} &= -\int \frac{\partial S(x|a;\Phi)}{\partial \Phi_{mi}} p(x|a;\Phi) p_a(a) da \\ &= -\int \frac{\partial S(x|a;\Phi)}{\partial \Phi_{mi}} p(a|x;\Phi) p(x) da \end{aligned} \tag{2.5}$$

where $S(x|a;\Phi) = -\log p(x|a;\Phi)$ is the surprise conditioned on a. In the second line I used Bayes's Theorem to rewrite the gradient as an expectation under the posterior distribution of a. This integral is in general intractable, so we proceed with some approximate method such as sampling from the posterior distribution of a. Sampling tends to be slow, so we usually just take a single sample at the mode of the distribution. This is called *maximum a posteriori* (MAP) estimation, and this MAP inference gives us a particular sparse code a^*. Using the MAP estimate is justified if $p(a|x;\Phi)$ is unimodal and sharply peaked.[3]

The gradient under this approximation is[4]

$$\frac{\partial p(x;\Phi)}{\partial \Phi_{mi}} \propto -\frac{\partial S(x|a^*;\Phi)}{\partial \Phi_{mi}}. \tag{2.6}$$

MAP inference is the same as minimizing the posterior surprise

$$S(a|x;\Phi) = -\log p(a|x;\Phi) = -\log p(x|a;\Phi) - \log p_a(a) + \log p(x) \tag{2.7}$$

[3] I am not aware of any study demonstrating that datasets of interest have this property, but comparing studies using MAP inference (e.g., [76]) to a study using a sampling approach [101] suggests that either it holds for whitened natural image patches or that the usual results of interest are robust to the inexactness of the MAP approximation.

[4] I'm dropping overall factors because they don't matter. We're going to scale the gradient by a learning rate anyway, so all that matters is the direction, i.e., how the gradient depends on m and i.

which up to an additive constant is

$$S(a|x;\Phi) = \frac{1}{2\sigma^2}||x - \sum_m a_m \Phi_m||_2^2 + \lambda \sum_m ||a||_1 \tag{2.8}$$

in the case of a factorial Laplacian prior. Since $S(a|x;\Phi)$ and $S(x|a;\Phi)$ only differ by terms that don't depend on Φ, we can summarize our approximate gradient descent procedure as alternating two steps: 1) minimization of $S(a|x;\Phi)$ with respect to a, and 2) a step down the gradient of the same function with respect to Φ with a fixed at the minimizer (i.e., the MAP estimate) a^*. In some contexts "sparse coding" is just defined by (2.8) as an objective function; the reasoning above shows that this is equivalent to the probabilistic model perspective with MAP inference.

Loose ends

The objective function $S(a|x;\Phi)$ can be averaged over the dataset. Usually we average over a random sample, say 100 data points, to compute the gradient for each step. This procedure is known in machine learning as "stochastic gradient descent" (SGD).

The noise variance σ^2 is usually swept under the rug since only the combination $\sigma^2\lambda$ actually affects inference and learning. We then refer to the combination as λ and think of it as the lone hyperparameter of the model. This parameter controls the tradeoff between (mean squared) reconstruction error and sparseness.

There is at least one major problem with the procedure as explained so far: we can always scale a down and scale Φ up to get the same reconstruction error but better sparsity. To deal with this we usually stop the dictionary elements from growing in some way. In the work presented in this dissertation, we simply set the norm of each dictionary element Φ_m to 1 after each learning step.

Sparse coding algorithms

Sparsenet

The algorithm in [76, 75] uses a direct version of the procedure outlined above, with the modification that the dictionary element norms are set to make the variance of the output of each element some fixed value. So the procedure is:

1. Draw a batch (say 100 points) of data.
2. Minimize the objective, averaged over the batch, with respect to the a_m.
3. Adjust the Φ_{mi} to descend the gradient of the objective at fixed a.
4. Update the norm of each Φ_m multiplicatively: new norm = (old norm)(variance goal / moving average variance)$^\eta$ where η might be 0.01 or so.

FISTA

The Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)[8] is a popular choice for sparse coding and other regularized linear inverse problems when speed is an important consideration. Using FISTA for inference and a standard optimization routine such as LBFGS is a good procedure from a performance standpoint on a standard computer, but our parochial interest in modeling the brain leads us to other algorithms.

LCA

LCA refers to a class of "locally competitive algorithms" that use a MAP inference procedure with an auxiliary variable u that gets thresholded to the coefficients a [86]. The choice of threshold function corresponds to a choice of prior on a. LCA is sufficiently flexible that it can be used with a variety of sparse priors and achieve L0 sparseness. It also has the advantage of being well suited for analog implementations that can run many times faster than digital simulations. Interpreting u_m as a membrane potential also gives LCA a more neural feel than Sparsenet.

We use LCA (with SGD learning) as our primary "conventional" sparse coding algorithm, so we provide some details on its implementation.

Details and implementation

Rozell et al. [86] show that sparse coding MAP inference can be implemented by assigning each dictionary element a unit with an internal variable u_i that evolves in time according to

$$\dot{u}_i(t) \propto -\frac{\partial E}{\partial a_i}. \tag{2.9}$$

The u variables are then put through a thresholding function to get the activities a, implementing the sparsity term in the objective function if

$$\lambda \frac{dC(a_m)}{da_m} = u_m - a_m = u_m - T_\lambda(u_m) \tag{2.10}$$

where C is the cost function (e.g. L1 norm) in the objective function and T_λ is the thresholding function. Rozell et al. emphasize that the dynamics are not directly performing gradient descent (since $\dot{u}_m \not\propto \partial E/\partial u_m$) but that as long as a_m is a monotonically increasing function of u_m the as will descend E.

The dynamical equation for the u variables is

$$\dot{u} = -u + \Phi^T X - (\Phi^T \Phi - I)a \tag{2.11}$$

where the as are thresholded us. For the $L1$ cost, the threshold function $T_\lambda(u)$ is 0 between $-\lambda$ and λ, and $u - \text{sign}(u)\lambda$ elsewhere. For the $L0$ cost, the threshold function is 0 between $-\lambda$ and λ and u elsewhere.

The usual procedure is to use a simple Euler method to solve this differential equation:

$$u(t+1) = (1-\eta)u(t) + \eta[\Phi^T X - (\phi^T \Phi - I)a(t)] \tag{2.12}$$

where η is a small number, typically 0.1 is small enough. The system takes several time constants to converge; 200 iterations with $\eta = 0.1$ (i.e., 20 time constants of evolution) seems to work well.

There is a trick to improve the convergence of this procedure. The threshold parameter λ is set to $\frac{1}{2}\max_m |\sum_i \Phi_{mi} X_i|$, then allowed to decay exponentially during the inference procedure until it reaches the nominal value, where it is kept constant. Multiplying λ by 0.98 at each time step works well with the parameters above. Since the us tend to overshoot their final values early on, this stops too many of them from clearing threshold and making the feedback complicated. (This is my post hoc explanation; I do not know why this was first done.)

SAILnet

Joel Zylberberg developed a Sparse And Independent Local network (SAILnet) as a more "biologically plausible" sparse coding algorithm with spiking neurons and synaptically local plasticity rules [110]. We will have more to say about SAILnet in later chapters.

Extensions and modifications

Sparse coding usually uses a factorial prior, but it is simple to extend the Sparsenet procedure to incorporate dependencies among the a_m.

Vanilla sparse coding uses a linear generative model that is symmetric about $a = 0$, but you can restrict the a_m to be nonnegative to gain on neural realism and/or fit asymmetric data.

Convolutional sparse coding may be desirable for reducing the number of parameters needed to learn a diverse dictionary.

Data preprocessing

Most sparse coding algorithms will not work well on just any data, even if that data has an underlying linear sparse structure. It is best to center the data and normalize to unit variance, and often you want to do more "preprocessing."

In the case of natural image patches, there are a number of "preprocessing" steps that are standard:

1. Do not use images with blur or other artifacts. Some images, like David Field's set of images from the Northwest [30, 76], have artifacts around the edges that should be trimmed.

2. Take a logarithm of the intensity. (This does not seem to be important, at least for images with a small range of intensity values such as the Van Hateren database [35].)

3. Set the mean to 0 and standard deviation to 1 for each image.

4. Whiten the resulting dataset. This is not strictly necessary but improves learning speed.

Practical issues

Choosing the sparseness parameter

So far I've offered no prescription for how to choose the sparseness parameter λ. Usually it is chosen between 0.1 and 2. One could optimize λ to maximize the likelihood, or for a task such as denoising. One simple prescription is to adjust λ during training so that the signal-noise ratio of the reconstructions stays around a preset value, often 15 dB [73].

Pretty dictionaries and low costs

Low values of the objective function do not always go hand-in-hand with nice-looking sparse coding dictionaries. Part of the problem seems to be that the usual results – e.g., Gabor functions for natural images – often only show up late in training, meaning that at least some of the gains in the objective function relative to random noise probably have nothing to do with these nice dictionary elements.

Training dynamics, model recovery, and the loss

It is straightforward to sample from the sparse coding probabilistic model with a factorial Laplace prior. In Chapter 6 we use data sampled from this model to see how two different algorithms perform at recovering a known sparse model with different "preprocessing" steps. Here I will briefly mention a set of observations about sparse model recovery that I have not seen elsewhere. These observations are summarized in Figure 2.1.

For the left half of this plot, we trained a complete sparse coding model on iid Gaussian noise. The model improves its reconstruction error dramatically on the first learning step; this improvement is due to making the dictionary orthogonal. An overcomplete dictionary cannot be orthogonal, but overcomplete models show similar though milder behavior when trained on noise.

On the right half of the plot, we train the same model on samples from a known sparse model. The benefit of an orthogonal dictionary persists (again, similar behavior is observed in the overcomplete case). Throughout the rest of training, the reconstruction error is small and nearly constant, while the L1 norm of the activations decreases substantially for as long as the model improves its fit.

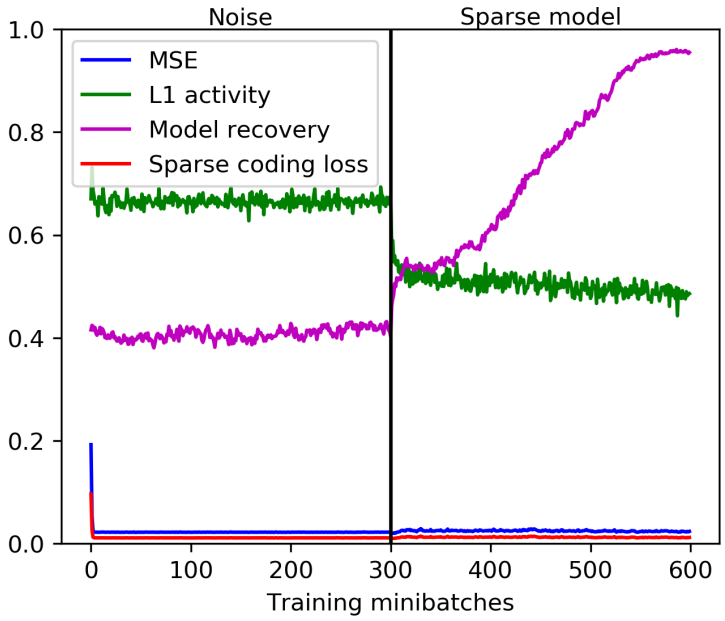

Figure 2.1: Sparse model recovery.

Appendix: clarity over brevity

The sparse coding model distribution can be written

$$p(x_i^\mu; \{\Phi_{mi}\}) = \int p(x_i^\mu | \{a_i^\mu\}_m; \{\Phi_{mi}\}) p_a(\{a_m^\mu\}) da_0 \cdots da_M. \tag{2.13}$$

where

$$p(x^\mu | \{a_m^\mu\}; \{\Phi_{mi}\}) \propto \prod_i e^{-(\hat{x}_i^\mu)^2 / 2\sigma^2} \tag{2.14}$$

and

$$p_a(\{a_m^\mu\}) = \prod_m \frac{1}{2\lambda} e^{-\lambda |a_m^\mu|}. \tag{2.15}$$

2.2 Whitening

Data is whitened (or sphered) if the covariance matrix is the identity:

$$\langle x_i x_j \rangle = \delta_{ij}, \tag{2.16}$$

meaning there are no pairwise bilinear correlations and the variance in each direction is the same. Sparse coding algorithms generally perform better on whitened data. Whitening is sometimes said to "reduce redundancy," but whitening is usually achieved by an invertible transformation, which cannot reduce total entropy and therefore does not reduce the

information-theoretic redundancy. Rather, whitening localizes information, increasing the entropy of each dimension/symbol at the expense of the entropy associated with pairwise correlations.

There are multiple ways to achieve this. One is to use a linear filter tailored to your data set; Olshausen and Field whitened their images using a filter designed to flatten the Fourier power spectrum up to a high frequency and then allow it to fall, considering the very high frequencies to be noise [76]. The result is only approximately whitened according to my definition of whitening.

Another method, which generalizes more simply to other datasets besides natural images, is to use principal components analysis (PCA). For natural images, the principal components are roughly Fourier modes, so flattening the principal component variances is roughly the same as flattening the Fourier power spectrum. That said, the difference between the Olshausen and Field 1996 whitening and PCA whitening can be significant, for instance altering dependencies among the sparse coding coefficients. See [45] for more on PCA whitening in the context of natural image statistics.

Chapter 6 discusses whitening and its relation to sparse coding at length.

Chapter 3

Sparse coding with spiking neurons and local learning rules

This chapter goes into greater depth on the Sparse and Independent Local Network mentioned in Chapter 2. Zylberberg et al. developed SAILnet as a model that explains in greater detail how sparse coding and learning could be implemented in a neural system, including spiking neurons and synaptically local learning rules. Here we discuss the model for its own sake, recapitulating some of the ideas in [110] but also providing some new insights and analysis. In later chapters we use SAILnet to better understand natural image and sound statistics and to offer an explanation for earlier stages of visual processing.

3.1 A neural implementation of sparse coding: LCA

Sparse coding can be implemented in a neural architecture by a Locally Competitive Algorithm (LCA) [86] as explained in Section 2.1. Each unit's internal variable u_m can be thought of as modeling the sub-threshold membrane potential of a neuron while the activations a_m correspond to firing rates and neurons have lateral connections according to the overlap of their feedforward connections. LCA performs well in practice, and Rozell et al. are able to prove convergence properties.

In terms of modeling the brain, however, LCA has a few shortcomings: the "neuron" outputs are analog values rather than spikes, and in order to compute the inference dynamics or weight update rule for a particular unit, we need to know the weights Φ and activities a of all other units.

A smaller issue with LCA as a model of sparse coding in the brain is that forcing the interactions between units in LCA to be inhibitory significantly impairs reconstruction performance. For good performance each unit inhibits some units and excites other units, which violates Dale's Principle.

3.2 Spikes and local learning: SAILnet

The Sparse and Independent Local Network (SAILnet) [110] addresses each of the shortcomings mentioned above. First, the thresholding function $T_\lambda(u_m)$ is replaced by a spiking operation: when the internal variable u_m crosses a threshold it spikes (incrementing a_m) and resets to 0. The a variables become spike counts (which can also be divided by inference time to get a spike rate). In [93] the authors present a more principled development of a spiking version of LCA, but I've found empirically that the naïve replacement just described can perform well in terms of descending E.

The fact that LCA's inference and learning rules are not synaptically local is more difficult to deal with. The solution Zylberberg et al. present is effectively to change the mean-squared-error term in the objective function:

$$\frac{1}{2}\sum_i \left(X_i - \sum_m \Phi_{mi} a_m\right)^2 \longrightarrow \frac{1}{2}\sum_{m,i}(X_i - \Phi_{mi}a_m)^2 \tag{3.1}$$

This change leads to the synaptically local learning rule

$$\Delta\Phi_{mi} \propto \langle a_m(X_i - a_m\Phi_{mi})\rangle. \tag{3.2}$$

The claim in [110] is that this learning rule approximates gradient descent well as long as the activities a are sufficiently sparse and uncorrelated. We will address when we can expect this approximation to be valid in Section 3.2. Before that, note that the dynamical equation for the internal variables also changes to become simply a leaky integrator:

$$\dot{u}_m = -u_m + \sum_i \Phi_{mi} x_i \tag{3.3}$$

which is local if boring.

In order to ensure that the activities a are actually sparse and decorrelated, Zylberberg et al. formulate a constrained optimization problem which may be expressed in terms of a Lagrange function

$$\mathcal{L} = \frac{1}{2}\sum_{ij}(X_i - \Phi_{ji}a_j)^2 + \sum_j \theta_j(a_j - p) + \frac{1}{2}\sum_{jk} W_{jk}(a_j a_k - p^2) \tag{3.4}$$

where the θ_j and W_{jk} are Lagrange multipliers enforcing the constraints that the terms they multiply are zero. Specifically, the activities are constrained to have mean p and no pairwise correlations, where p is a scalar input to the model. Interpreting $\mathcal{L}$ as a new objective function, we can derive dynamics for the inference circuit analogous to (2.11):

$$\dot{u}_m \propto -\frac{\partial L}{\partial a_m} = \sum_i \Phi_{mi} X_i - \sum_j \Phi_{ji}^2 a_i - \theta_i - \sum_{j\neq m} W_{jm} a_j. \tag{3.5}$$

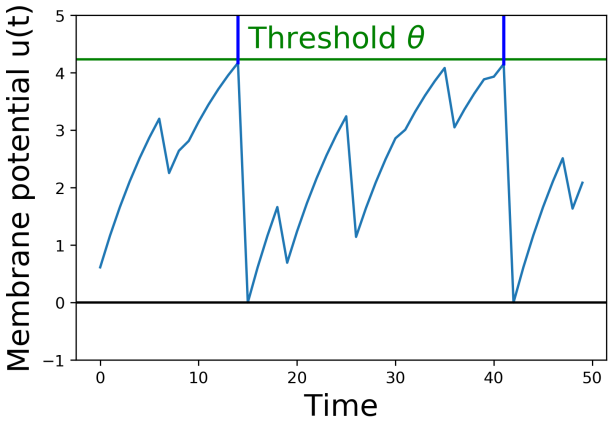

Figure 3.1: Example SAILnet neuron trajectory. Dark blue vertical lines mark spikes, after which the membrane potential is reset to its resting level. Sudden decreases occur when the neuron receives inhibitory input from other neurons spiking. Units are arbitrary but could be linearly mapped to real time (roughly 100ms total) and electrical potential (10s of mV) in a real neuron.

We can reasonably interpret the terms in this equation to give us a spiking circuit with dynamics

$$\dot{u}_m = \sum_i \Phi_{mi} X_i - u_i - \sum_{j \neq m} W_{jm} y_j, \tag{3.6}$$

which is the SAILnet inference procedure. The sum over feedforward weights in the leakage term is approximately 1 since these weights are approximately normalized, although in practice this factor can be as large as 6 or as small as 1/2 or so. The θ parameters we interpret as the thresholds for spiking, and $y_j = 1$ when unit j is spiking and 0 otherwise. This model corresponds to a leaky integrate-and-fire neuron. An example trajectory for one neuron is shown in Figure 3.1.

SAILnet does not descend its Lagrange function

In [110], Zylberberg et al. write down a Lagrange function similar to (3.4) and derive learning rules by gradient descent on that Lagrange function. Gradient descent on (3.4) gives the expected learning rule for the feedforward weights Φ, but for the Lagrange multipliers gradient descent gives

$$\begin{aligned} \Delta\theta_m &= -\frac{\partial \mathcal{L}}{\partial \theta_m} = -(a_m - p) \\ \Delta W_{mn} &= -\frac{\partial \mathcal{L}}{\partial W_{mn}} = -\frac{1}{2}(a_m a_n - p^2) \end{aligned} \tag{3.7}$$

which are the *opposite* of the SAILnet learning rules (up to the factor of 1/2 which I included to make the inference rule look nicer).

This is not to say that the results in [110] are wrong, per se. Zylberberg et al. flipped the signs on the Lagrange multipliers to make the learning rules come out right. But doing that puts minus signs in (3.5) which makes interpreting SAILnet inference in terms of that equation a stretch. In particular the thresholds appear as *excitatory* currents. Zylberberg et al. instead derive (3.6) by heuristic arguments.

SAILnet seeks a saddle point

We can resolve this confusion by noting that solutions to a constrained optimization problem occur at *critical points* of the corresponding Lagrange function, not necessarily minima. In the case of SAILnet, we want to minimize $\mathcal{L}$ with respect to Φ and a, but maximize with respect to θ and W: the solutions we want are saddle points of $\mathcal{L}$.

Empirically, this approach of minimizing with respect to some parameters while maximizing with respect to the others seems to work much of the time, for the reasons explained in [110]. As long as the learning rates for the Lagrange multipliers are large compared to the learning rates for the feedforward weights, the system stays close to the constraint surface.

SAILnet's method is similar to Dual Ascent, which is discussed as a precursor to ADMM in [14]. Dual Ascent is guaranteed to converge under certain assumptions.

Empirical observations of learning

SAILnet has a few failure modes.

1. Everything goes terribly if the "learning rates" for the Lagrange multipliers are too small. They generally need to be 10 times the dictionary learning rate or more. We can understand this requirement: SAILnet needs to stay close to the constraint surface for its dictionary learning rule to have any hope of working properly.

2. If the firing rate parameter p is too high relative to the number of units (say 0.25 for 100 units) the network learns nothing of interest and does a poor job descending the objective. The objective function may even increase without bound.

3. SAILnet has no mechanism to directly force units to learn different features from one another: the receptive field of one unit never enters directly into the inference or learning rule for any other unit. Sometimes a bunch of units learn the same thing, and then to keep correlations down they develop large inhibitory weights W. But they still each need to fire at a rate p, so if there are too many of them their thresholds have to be low. But these dynamics conflict; the result is that these units fire on the first or second time step of inference, which inhibits all of them in the next time step so much that they don't fire again. We do not know how to keep this from happening other than by setting p quite low.

4. Large violations of the constraints do not always seem to be bad. We have trained networks with the parameters in [110] and found that the Lagrangian gets somewhat

large ($\sim$10) even though the MSE term asymptotes down to $\sim$0.6 and the learned features are as expected in the image case.

Validity of the MSE approximation

Writing out the nonlocal learning rule derived from gradient descent on mean-squared error gives

$$\Delta\Phi_{mi} \propto \langle a_m X_i \rangle - \langle a_m^2 \rangle \Phi_{mi} - \sum_{n \neq i} \langle a_m a_n \rangle \Phi_{ni}. \tag{3.8}$$

If the SAILnet constraints are satisfied,

$$\Delta\Phi_{mi} \propto \langle a_m X_i \rangle - \langle a_m^2 \rangle \Phi_{mi} - p^2 \sum_{n} \Phi_{ni}. \tag{3.9}$$

The third term is the nonlocal part; [110] argues that this term is small compared to the other two in the sparse and uncorrelated limit so that SAILnet should approximately descend the mean-squared error objective.

To make this argument more precise, let $p = f\eta$ where η is the average activity ignoring zeros and f therefore typifies the fraction of stimuli that cause any particular unit to have nonzero activity. The mean of the square of the activities is therefore approximately $f\eta^2$. We can characterize the magnitude of the sum in the third term of (3.9) by the standard deviation of the sum of n_{dict} i.i.d. random variables with mean 0 and variance σ^2. Then a necessary condition for the approximation to be good is

$$f\eta^2 |\Phi_{ik}| \gg p^2 \sqrt{n_{\text{dict}}} \sigma / \sqrt{2\pi}. \tag{3.10}$$

Since $|\Phi_{ik}| \sim \sigma$, this condition can be expressed

$$f \ll \sqrt{2\pi} / \sqrt{n_{\text{dict}}}. \tag{3.11}$$

So the L0 sparseness factor needs to scale as one over the square root of the number of units. In [110], a 1536-unit network was trained with $p = 0.05$. $\sqrt{2\pi / n_{\text{dict}}} \approx 0.06$ in this case; since η is at least one (being an average of nonzero spike counts), the condition (3.11) was violated for this network. Nevertheless, the network seems to have functioned more or less as claimed, and I've found that violating the condition above by a factor of a few still allows the network to learn the usual features and decrease the mean-squared error function over training time.

Empirically, I have also found that with the parameters in [110], the neglected term is on average about the same size as (or a factor of 2 or 3 smaller than) the second term in the dictionary learning rule (though both are consistently a factor of 2 or 3 smaller than the Hebbian term). This average behavior is dominated by a few large "second terms"; a scatter plot of many examples shows that the dropped nonlocal term is usually *larger* than the kept local one (though both are usually smaller than the Hebbian term.) These observations

contrast with LCA, for which the local normalizing term is about a tenth as large as the nonlocal term, which is about half as large as the Hebbian term.

All this leads me to believe that what really matters is that the nonlocal term be small compared to the Hebbian term, which holds when

$$p\sqrt{n_{\text{dict}}}/\sqrt{2\pi D} \ll 1 \Longrightarrow p \ll \sqrt{2\pi}/\sqrt{\text{OC}} \tag{3.12}$$

where D is the dimension of the input space and OC is the overcompleteness of the dictionary. I assumed that the data has unit standard deviation in each dimension. This condition has the same scaling as (3.11) but is relaxed by a factor of the square root of the data dimension.

These conditions, or something more relaxed, are necessary but not sufficient for the approximation to be good. For instance, the dropped term might be small for each learning step but have a regular direction so that its effect adds up to become significant over time compared to the larger terms which may be more variable.

Inhibitory weights sometimes seek feedforward weight dot products when the data is sphered

With a few assumptions we can obtain a simple result for the lateral connections W. Let us assume that we are in a regime where SAILnet inference is dominated by the feedforward term and we can neglect lateral inhibition – call this the independent limit. Then for constant input we have

$$u_m(t) = (1 - e^{-t/\tau}) \sum_i \Phi_{mi} x_i. \tag{3.13}$$

If $\sum_i \Phi_{mi} x_i < \theta_m$, unit m will never spike. Otherwise, let us assume that $\sum_i \Phi_{mi} x_i$ is large enough that unit m spikes while $u_m(t)$ is approximately linear in t. Then the spike rate for unit m is given by

$$a_m = \frac{\sum_i \Phi_{mi} x_i}{\theta_m \tau}. \tag{3.14}$$

The thresholds are often all similar, so we have approximately

$$a \propto \sum_i \Phi_{mi} x_i. \tag{3.15}$$

The learning rule for the inhibitory weights in this approximation is (in matrix notation)

$$\Delta W \propto aa^T - p^2 \propto \Phi X X^T \Phi^T - \Xi p^2 \tag{3.16}$$

where Ξ is some constant. Ignoring that term for now, we see that the inhibitory connection matrix should move towards the Gram matrix of the feedforward weights *if and only if the data is white*, $XX^T \propto I$.

Evaluating assumptions

Future work could evaluate these assumptions to gain a better understanding of how our simple analysis relates to the true, complicated dynamics of SAILnet.

We totally neglected the term proportional to p^2 in the W learning rule. This may not matter when p is small. It may be the deviation of a from its average of p that is approximately proportional to the input current ΦX. Certainly one could instead work with a network with symmetric (instead of nonnegative) outputs. This provides one simple test of whether this approximation makes sense.

There is certainly a regime in which we can neglect the effect of the lateral connections during inference, when the network is undercomplete or at initialization with $W = 0$. Even in the mildly overcomplete regime, I expect that activities are strongly correlated with input current, based on experiments with LCA that I expect would hold up. One could run simulations to test this.

The approximation that neurons only fire while their dynamics are essentially linear amounts to assuming for each neuron either $\Phi_m X/\theta_m \gg \tau$ or $\Phi_m X < \theta_m$. This could be tested on a sample.

Chapter 4

On the sparse structure of natural sounds and natural images: similarities, differences, and implications for neural coding

Having established our primary tools and context, I now present in this chapter the first major new work in this dissertation. The success of sparse coding in explaining the receptive fields of neurons in primary visual cortex inspired similar investigations for other sensory modalities, especially audition. However, I am unaware of any direct comparison of the statistics of natural images with the statistics of natural stimuli from another modality that could guide modeling efforts to understand other modalities on similar terms. This chapter provides a comparison of the statistics of natural images and natural sounds using the tools of sparse coding generally and also SAILnet in particular that we have discussed in previous chapters.

In addition to the primary material, I have included a number of "supplementary" figures at the end of this chapter that provide additional comparisons and complete models. These figures are not essential to the main narrative of the chapter and are likely only of interest to experts, but I include them for completeness. This chapter is adapted from a paper authored by myself and Professor DeWeese that is in review at time of writing.

4.1 Abstract

Sparse coding models of natural images and sounds have been able to predict several response properties of neurons in the visual and auditory systems. While the success of these models suggests that the structure they capture is universal across domains to some degree, it is not yet clear which aspects of this structure are universal and which vary across sensory modalities. To address this, we fit complete and highly overcomplete sparse coding models to

natural images and spectrograms of speech and report on differences in the statistics learned by these models. We find several types of sparse features in natural images, which all appear in similar, approximately Laplacian distributions, whereas the many types of sparse features in speech exhibit a broad range of sparse distributions, many of which are highly asymmetric. Moreover, individual sparse coding units tend to exhibit higher lifetime sparseness for overcomplete models trained on images compared to those trained on speech. Conversely, population sparseness tends to be greater for these networks trained on speech compared with sparse coding models of natural images. To illustrate the relevance of these findings to neural coding, we studied how they impact a biologically plausible sparse coding network's representations in each sensory modality. In particular, a sparse coding network with synaptically local plasticity rules learns different sparse features from speech data than are found by more conventional sparse coding algorithms, but the learned features are qualitatively the same for these models when trained on natural images.

4.2 Introduction

An important goal of systems neuroscience is to discover and understand the principles that might govern sensory processing in the brain. Several principles have been proposed, such as reducing redundancy between neurons [4, 7, 2, 24, 21], representing statistical dependencies between objects and events to guide action [6], minimizing expended energy [57], maximizing entropy [89], and maximizing transmitted information [56, 26, 83, 9, 43, 49]. Each of these principles suggests that sensory systems should use the statistical structure of sensory data from the animal's environment to efficiently represent and process that data. Studying the statistics of natural sensory input and coding strategies specialized for those statistics has helped us understand neural sensory systems [90, 28, 10, 94, 95, 77, 103].

One principle that has provided insight into the structure of data from the natural environment and the way these data are represented by neural activity is *sparseness* [74]. We say that a fluctuating quantity is sparse if it is often zero (L0 sparseness), or if it is close to zero more often than a Gaussian random variable with the same variance (L1 sparseness). Natural visual scenes can be well-represented by sparse distributions [30], and coding strategies optimized for sparseness find local, oriented, bandpass features that match the receptive fields of simple cells in primary visual cortex (V1) [76, 82, 86, 110]. In the auditory domain, the filters that optimize a sparse coding scheme for the acoustic waveforms of natural sounds resemble cat auditory nerve filters, and they form a similar tiling of time-frequency space [97]. Interestingly, training this sparse coding model on speech rather than an optimized combination of recordings of environmental sounds yields just as good a fit to auditory nerve filters. Moreover, a sparse coding model of spectrograms of speech learns features that resemble spectro-temporal receptive fields (STRFs) measured at higher stages of auditory processing, such as the inferior colliculus, auditory thalamus, and primary auditory cortex (A1) [18]. Some similar features emerge in models of simulated cochlear responses [55, 51], and hierarchical models have found higher-level sparse structure [50, 100, 72]. Experiments

have uncovered sparse responses from neurons in visual cortex [104, 105] and auditory cortex [27, 39] as well as other brain regions [102], suggesting that the nervous system has evolved to take advantage of the sparse structure of its inputs. Furthermore, a sparse coding model of natural images exhibits many of the non-classical receptive field effects found in V1 neurons in addition to learning similar classical receptive fields [108].

These results suggest that the applicability of sparse coding to understanding sensory systems is not limited to a single modality such as vision, but that sparseness may be a more universal property of data from the natural environment. However, there are clear differences between visual and auditory data, which has affected the way they have been explored in past work. For example, sparse coding studies in vision have mostly focused on static images, while the time dimension is not as easily avoided for sounds. As another example, one model designed to separate form and motion in natural movies did manage to learn pairs of phase-shifted Gabor filters[17] but it did not learn phase-shifted auditory features, although an extension was used to model binaural sound coding [71]. Moreover, images exhibit some symmetries (e.g., a rotated natural image is still a natural image) without clear analogs in the auditory domain.

Our primary goal was to compare the statistical structure of natural visual scenes and of natural sounds through the lens of sparse coding. Our approach was to fit complete and highly overcomplete sparse coding models to spectrograms of speech and to natural image patches and then to compare the statistics of these models' representations. We have found that, while natural scenes and sounds can each be well-represented by sparse coding models, this structure differs in significant ways between the two modalities. Consistent with past work [18], we found a greater variety of sparse features in speech compared with natural images. Among these auditory features, we identify three types with distinctive, highly sparse and asymmetric distributions of projections in the dataset. Interestingly, elements in the full "dictionary" of auditory features learned by our sparse coding model vary greatly in their degree of sparseness and skewness. In contrast, we find that the sparse dimensions in the space of natural image patches, which mostly resemble Gabor functions in the complete case with a few other types appearing in the highly overcomplete regime, are much more uniform in their degree of sparseness and skewness.

In addition to these observations regarding "lifetime sparseness" of individual features, we also studied "population sparseness" of the two datasets — namely, sparseness across features for a given stimulus. We found that a typical speech spectrogram segment admits a sparser representation than a typical natural image patch, despite the fact that lifetime sparseness was typically greater for image features compared with speech features in the overcomplete regime.

We further demonstrate that the differences we find between the sparse structure of speech and that of images have significant consequences for coding schemes used to process these types of data, and therefore for neural models of vision and audition. In particular, we study the effects of the statistics of natural sounds and of natural images on a sparse coding network designed to match some important constraints imposed on real neural systems. The Sparse and Independent Local Network (SAILnet) [110, 109, 53] is the only algorithm we

are aware of with spiking neurons and synaptically local plasticity rules that can learn the diverse receptive field shapes of V1 simple cells when trained on natural image patches. Conventional sparse coding algorithms also learn these shapes but do not have the same constraints. Throughout this paper, "conventional sparse coding" refers to an L1-sparse linear reconstruction model with codes determined by the Locally Competitive Algorithm (LCA) [86] and parameters updated by gradient descent on mean-squared reconstruction error. This algorithm is used in, *e.g.*, [18] and [73]. The choice of LCA is not crucial, and we obtained qualitatively similar results with other coding algorithms such as gradient descent as in [76]. See Methods for details. We trained SAILnet models on spectrograms of speech sounds and on natural images, using the same preprocessing steps in both cases. While SAILnet learned similar features to those found using conventional sparse coding in the visual case, the SAILnet results were significantly different from conventional sparse coding for auditory data.

The divergence in results with SAILnet points to surprising differences between the sparse structure of natural images and natural sounds, with implications for both early development and sensory processing in the mature circuit in these different modalities.

4.3 Results

To compare the sparse structure of speech sounds to that of natural images, we fit sparse coding models to ensembles of each type of data. For speech, we adapted a preprocessing scheme introduced previously [18] in which segments of spectrograms of recordings of speech are first whitened and then reduced in dimensionality using principal components analysis (PCA). We followed the same procedure for image patches of the same dimensionality as the spectrogram segments in order to make a fair comparison between the two datasets. These preprocessing steps are illustrated in Fig. 4.1 and discussed in more detail in section 4.5. Note that although the preprocessing schemes for the two datasets differed in that we took spectrograms of the auditory data, the spectrogram is not an inherently lossy transformation [58].

After this preprocessing, we trained sparse coding models using an iterative scheme based on the locally competitive algorithm (LCA) [86] for inference (*i,e.*, determining the activity of each unit for representing a given sensory input) combined with stochastic gradient descent for learning (*i.e.*, setting the parameters of the model). (Note that we will use "activity" and "activation" interchangeably below.) Throughout this manuscript we use the term "conventional sparse coding" to refer to this particular scheme, and this is the primary model we used to generate most of the results we present here, but we obtained similar results using SPARSENET [76] and, where a comparison makes sense, Independent Components Analysis (ICA, [9, 46]).

Complete sparse representations

Before training a sparse coding model, one typically specifies the number of stimulus features (also referred to below as "elements" or "units") to include in the full "dictionary" of the model. The optimal dictionary learned by a sparse coding model can depend substantially on the size of that dictionary relative to the size of the data [73]. Intuitively, one might expect a greater diversity of stimulus feature classes with a larger dictionary, and this is often the case. We first studied the complete regime by fitting sparse coding dictionaries with 200 elements, which is the dimension of each of our datasets after PCA reduction; we will refer to this as the "complete" regime. While models with more dictionary elements than the dimension of the data may make a closer correspondence with the brain, we found that the complete regime elucidates some aspects of the datasets themselves that are less clear in the "overcomplete" regime. We also discuss the overcomplete regime in section 4.3. We used the L1-sparse locally competitive algorithm (LCA) [86] to compute sparse codes and stochastic gradient descent (SGD) to optimize the dictionaries (see section 4.5 for details).

Figure 4.2 illustrates several properties of the learned dictionaries and their representations of the data. The dictionary elements found by our sparse coding algorithm exhibit clear structure beyond the restriction to the subspace spanned by the first 200 principal components. When trained on image patches, the model recovers the Gabor functions and long edge filter-like elements that are known to emerge in sparse coding models of smaller image patches [76] (Fig. 4.2a, third column). In the spectrogram case, we recover the element types previously seen in sparse coding dictionaries, including acoustic features that resemble spectro-temporal receptive fields (STRFs) observed in the inferior colliculus and at various other stages of the mammalian ascending auditory pathway [18] (Fig. 4.2a, first column).

For both the visual and auditory case, the distribution of unit activations for every dictionary element was much sparser than is typically found for random directions in the data space. Log histograms of individual unit activities were consistently sharply peaked at 0, and they had fat tails, compared with the parabolic shape of the (log) activity distribution expected for Gaussian-distributed random vectors in the stimulus space (Fig. 4.2a, second and fourth columns).

While both the visual and auditory dictionaries were sparse, there were several striking differences between the sparse structure of their representations. To quantify these observations, we used the following sparseness score

$$S[\{y\}] = -\frac{\langle |y| \rangle}{\sqrt{\langle y^2 \rangle}} + \sqrt{2/\pi}, \tag{4.1}$$

where the angle brackets denote the expectation over the empirical distribution of y. The constant $\sqrt{2/\pi} \approx 0.80$ simply shifts the score so that a normal distribution has a sparseness score of zero. This measure of sparseness is less sensitive to outliers than is kurtosis [45], for example. Nevertheless, we found qualitatively similar patterns for all of our results using kurtosis (see Fig. 4.14).

Applying our measure to the distribution of activities for each unit in response to every stimulus in the dataset used for model training, we found that the sparseness score was always greater than zero. For each data set, we then calculated sparseness scores for the activity distributions for a dictionary with each element drawn iid from a normal distribution. These sparseness scores for random elements were small, with median values of 3.3×10^{-2} for spectrograms and 5.6×10^{-2} for images; these values correspond to a null hypothesis against which to compare the optimized dictionary elements. These control values are plotted as small points in Fig. 4.2b.

While most of the units in the image dictionary clustered around a particular value of sparseness score and appeared qualitatively similar to one another, the units in spectrogram space covered a wider range of sparseness scores, with several distinct clusters (note the plateaus on the left of the purple curves in Fig. 4.2b). These clusters correspond to qualitatively different classes of features: 1) harmonic stacks, 2) broadband onsets, and 3) broadband onsets preceded by high-frequency sound (Fig. 4.2a, first column). Examples of several other qualitatively different types are also shown, although these do not exhibit strong clustering of sparseness scores. The clusters we found resemble those described previously [18] in the usage frequency histogram across the units in a half-complete sparse coding network.

Another difference between the visual and auditory sparse coding dictionaries was that the auditory unit activations were typically much more asymmetrical compared to the visual units. We quantified this using the skewness, which is the normalized third moment of a distribution

$$\text{skewness}[\{y\}] = \frac{\langle y^3 \rangle}{\langle y^2 \rangle^{3/2}}, \tag{4.2}$$

for mean-centered data $\{y\}$ [1]. A symmetric data distribution has zero skewness, whereas a distribution with a longer tail on the right than the left has positive skewness. In fact, we computed the absolute value of the skewness since, like most sparse coding models, our network allows for both positive and negative activities, leading to degenerate representations of asymmetrical signals. Fig. 4.2c demonstrates that the skewness values for the image dictionary elements were much smaller than the majority of auditory elements. Note also that the three most distinct categories of auditory features cluster in their degree of asymmetry of activations, as measured by the skewness, just as what we found for sparseness.

We can understand the skewness of these elements in terms of properties of speech sounds as represented by power spectra: speech often contains harmonic structure — power concentrated at integer multiples of a fundamental frequency — but it rarely if ever contains the opposite of such structure, which would be broadband sound with power missing at regularly spaced frequencies. Speech, like other natural sounds, also tends to contain sharp onsets but only gradual decays into silence. Since our sparse coding scheme allows for both positive and negative coefficients (*e.g.*, unit activities), we multiplied the examples shown in Fig. 4.2 by the sign of their skewness before displaying them and their corresponding activation histograms, in order to show the acoustic feature that would be added with a positive coefficient to the network's representation of the input. The idea is that the long tail of a

skewed distribution of unit activity corresponds to the feature associated with large activity magnitudes; we obtain very similar results if we instead multiply each unit by the sign of its average activity.

The highly sparse and skewed distributions of unit activities onto these well-clustered acoustic feature classes share a distinctive shape exemplified by the first three log-histograms in Fig. 4.2a. In each case, a sharp peak around zero is accompanied by a long flat tail on the positive side, showing that, for example, harmonic stacks appear at a wide range of volumes or not at all. Most of the other activation distributions, for the auditory spectrogram case as for the image cases, have a more symmetric, Laplacian-like shape.

We wondered to what extent these results reflected the nonlinear inference process of our sparse coding algorithm with 200 interacting elements, as opposed to simply the one-dimensional statistics of the data projected linearly onto each dictionary element. For example, nonlinear processing in the retina has been found to be more responsible for decorrelation between retinal ganglion cell outputs than their center surround receptive field shapes, which were originally hypothesized to underlie this effect [79]. To address this, we examined the distributions of the training data projected onto individual elements from each of these complete sparse coding dictionaries. Since this is in the complete regime, with no more dictionary elements than there are independent dimensions in the preprocessed dataset, one might expect that the projection of the data onto any given dictionary element (*i.e.*, the distribution of inner products between the dictionary element and the collection of images or spectrograms) should be sparser than projections in random directions, provided our learning algorithm is effective and we have a reasonable model for the data being fit. However, since the dictionary was optimized for the sparseness of codes determined by a nonlinear function of the dictionary and the data (LCA, see section 4.5), it did not have to turn out that linear projections of the data onto every element had to be sparse even if sparse dimensions exist in the data.

Nonetheless, we found that the elements of our optimized complete sparse coding dictionaries did robustly correspond to sparse dimensions in the data (Fig. 4.7). As with the analysis of unit activations, we compared our results for linear projections with those for a dictionary composed of random directions. Specifically, for each data set, we calculated sparseness scores for the distributions of projections for each of 200 directions drawn iid from a normal distribution. These sparseness scores were small, with median values of 7.8×10^{-3} for spectrograms and 2.5×10^{-2} for images. For each dataset, the full range of sparseness scores for these 200 random dimensions is represented by the shaded region in Fig. 4.7b, which lies well below the corresponding curve of sparseness scores for nearly all of the dictionary elements learned by the model. As we found for the activity analysis, most of the units in the image dictionary clustered around a particular value of sparseness score and appeared qualitatively similar to one another, whereas the units in spectrogram space covered a wider range of sparseness scores, with several distinct clusters in the high sparseness tail. Moreover, the sparse coefficients determined by our nonlinear algorithm were highly correlated with linear projections onto the corresponding dictionary elements with Pearson's $r = 0.97$ for both datasets, and the sparseness statistics evaluated on unit activities correlated with

the same statistics evaluated on linear projections with $r > 0.99$. The distinction between activations and projections was therefore not important for this analysis for these datasets, in the complete regime. For a second point of comparison, we also studied dictionaries optimized for the sparseness of linear projections onto the dictionary elements using independent components analysis (ICA). The results are shown in Fig. 4.13 and are also very similar to Fig. 4.2.

Overcomplete sparse representations

In addition to our analysis of the complete regime, we also studied the sparse structure of speech and images in the highly overcomplete regime, defined as the case with many more dictionary elements than the dimensionality of the (preprocessed) data. This is particularly interesting from a biological perspective given the greater numbers of neurons in primary sensory cortical areas compared with the number of efferents from the sensory periphery.

In the overcomplete regime, the dictionary elements cannot be truly orthogonal to all other elements, so one might expect nonlinear interactions to be more pronounced during inference in order to achieve sparse representations. We fit models with 2000 elements, which is ten times the dimensionality of the preprocessed data given that we kept only the first 200 PCA components.

Figure 4.3 presents some statistics for the highly overcomplete dictionaries trained on spectrograms and on image patches. Unlike the complete regime, there is no longer obvious clustering of sparseness scores for either unit activations (Fig. 4.2) or linear projections onto the dictionary elements (Fig. 4.7) of the spectrogram dictionary. However, it is still the case that the spectrogram dictionary covers a wider range of sparseness scores than the image dictionary and it has a larger variety of activity distributions (see Fig. 4.9). Intriguingly, the distribution of L0 lifetime sparseness values (i.e., the fraction of stimuli eliciting no response) was nearly identical for the spectrogram and image dictionaries (Fig. 4.11a) unlike what we found for L1 sparseness, though the range of "L0 asymmetry" values (the fraction of positive minus the fraction of negative responses) was still much greater for the auditory model (Fig. 4.11b).

Since LCA uses a nonlinear process to determine a sparse representation for each data point and this nonlinearity becomes increasingly important for higher degrees of overcompleteness, we examined the sparseness of the activations of each unit in the LCA network and compared it to that of the linear projections for the corresponding unit. The activation of each unit depends on all the units, so we also compared our results to the behavior of a ten-times overcomplete dictionary of random elements (thin lines, Fig. 4.3a; shaded regions, Fig. 4.3b). We adjusted the sparseness parameter λ for each network to achieve the same reconstruction error on the appropriate dataset. For both the image and spectrogram models, the learned dictionary elements had sparser activations than the random dictionary elements of the same rank (Fig. 4.3a), just as we found for the complete regime. Similarly, linear projections were sparser for the learned dictionary elements compared with the random dictionaries for both the image and speech models (Fig. 4.3b). However, the unit

activations for the image dictionary were consistently sparser than those of the corresponding spectrogram units (Fig. 4.3a), whereas the sparseness of linear projections (Fig. 4.3b) displayed the same overall pattern we observed for the complete regime, with a larger range of sparseness scores across the spectrogram dictionary compared with a fairly constant middle value for the image dictionary. (Note that the rank (horizontal axis) in each panel of Fig. 4.3 is independently determined.)

Thus, unlike what we found for the complete regime, the sparseness of the linear projections of each element of either overcomplete dictionary was not closely correlated to the sparseness of that element's LCA activations: Pearson's r of -0.12 and -0.30 for spectrograms and image patches, respectively. Conversely, for both the image and spectrogram models, the skewness of the activations was better explained by the skewness of the linear projections, with Pearson's rs of 0.89 and 0.66 (Fig. 4.3c,d). Similar to what we found for the complete regime, the overcomplete spectrogram dictionary exhibited much greater skewness than the overcomplete image dictionary, which was true for both unit activations (Fig. 4.3c) and linear projections (Fig. 4.3d).

These results indicate that the L1 sparseness of the LCA activations in the highly overcomplete regime is strongly affected by interactions among the units and not directly by some aspect of the individual units, while the asymmetry of a unit's activations largely follows from the asymmetry of the corresponding data dimension. This contrasts with the complete regime, where each statistic is nearly the same for linear projections as for LCA activations. Interestingly, these nonlinearities increased the sparseness for the overcomplete image model more than for the auditory model (compare Figs. 4.3a and 4.3b).

Finally, repeating the analysis described above for L0 sparseness rather than L1 sparseness in the overcomplete regime, we found that most trends were unchanged. For example, both spectrogram- and image-trained networks had much sparser unit activations compared with the random controls (Fig. 4.11a), and the spectrogram activation distributions were more asymmetrical than the image activity distributions (Fig. 4.11b). However, the distributions of L0 sparseness values for images and spectrograms were nearly identical (Fig. 4.11a).

Population sparseness

The results described above focus on the sparseness of the activations (and linear projections) of a single unit across the dataset, which is directly related to the so-called lifetime sparseness of an individual unit — the distribution of a unit's activities at each moment over its lifetime. We also examined the sparseness of the distribution of simultaneous activations of all units, often called "population sparseness." These two notions of sparseness are distinct and not always related in an obvious way [106], so it is worth comparing the population sparseness of sound and image models in addition to the lifetime sparseness analyses above.

For each of our analyses, a typical speech spectrogram admitted representations with greater population sparseness than did comparably preprocessed images. Each panel of Fig. 4.4 presents a pair of histograms representing comparable distributions over the two datasets. Panels a and b show that the distribution of unit activations representing a given

spectrogram segment for an optimized sparse coding dictionary was typically sparser than the analogous distribution for an image patch. This trend was also evident for the projection analysis (Fig. 4.4d,e). Since LCA uses a thresholding procedure, most units had exactly zero activity for any given stimulus. We therefore also looked at the fraction of units active (a measure of L0 sparseness), which tended to be smaller for the spectrogram case (Fig. 4.4c). Thus, typical elements from the spectrogram dictionary had greater L0 population sparseness, in addition to having greater L1 population sparseness, compared with those from the image dictionary. All of these trends are summarized by the medians of the various histograms, represented by the lower vertical lines in each figure panel.

This observation is somewhat surprising given the opposite trend we found for lifetime sparseness (Fig. 4.3). Speech spectrograms typically admit sparser representations than those of images, even though individual units in the image network tend to have activations with greater sparseness across examples compared to individual auditory units. We emphasize that, while the population sparseness trends we have just described are true for the typical element of each distribution, the distributions for the image case in particular are not fully characterized by a single summary statistic. The means in each plot of Fig. 4.4 are represented by the top vertical lines and the differences are generally small: values of Cohen's d were 0.25, 0.16, 0.17, 0.20, and 0.020, for the pairs of distributions in the order of the panels in Fig. 4.4. Normalizing the differences in medians by the same pooled standard deviation as in Cohen's d gives magnitudes of 0.52, 0.58, 0.70, 0.49, and 0.35 for the median differences. The distributions for activations and for linear projections show similar differences between the two datasets. This suggests that the effect of the different data statistics on the population sparseness of an optimized sparse coding model is primarily driven by the statistics of the linear projections rather than by complicated nonlinear interactions between units during inference.

Implications for biologically plausible sparse coding

Sparse coding dictionaries that resemble the distributions of observed receptive fields of actual simple cells in the primary visual cortex have been obtained using several variations on the classic SPARSENET sparse coding model (*e.g.*,[76, 10]). Among these variations, the Sparse and Independent Local network (SAILnet; a sparse coding model with spiking neurons and synaptically local learning rules) has been shown to learn the variety of simple-cell receptive field shapes seen in primate primary visual cortex when trained on whitened natural image patches [110] just as well as the best existing sparse coding algorithms [82, 86, 73]. However, we have found that this more biologically plausible sparse coding model does learn a different representation than conventional sparse coding models on some datasets, and that this difference is more pronounced and more clearly relevant to the comparison with real neurons in the auditory case.

Fig. 4.5 presents examples of dictionary elements learned by conventional (LCA inference and gradient descent learning) overcomplete sparse coding as described above, each matched with a dictionary element learned by SAILnet on the same data with the same number of

dictionary elements. The SAILnet elements were selected automatically to minimize the angle with the corresponding conventional sparse coding element in the 200-dimensional space. The conventional sparse coding dictionary for spectrograms contains elements with no close matches in the SAILnet dictionary, and we were unable to find qualitatively similar elements by inspection in these cases. Full dictionaries are presented in the Supporting Information, Figs. 4.18, 4.19, 4.21, and 4.22. For example, SAILnet does not discover features with the distinct checkerboard structure seen in Fig. 4.5a, second and fifth from the left in the bottom row. These elements tend to have only moderately sparse and mostly symmetric distributions of linear projections on the data (*e.g.*, Fig. 4.2, example 6).

Although we present results for a particular learned dictionary for each dataset and each algorithm, the results do not change substantially for the same algorithm starting from other random initializations and/or using other random draws from the training sets during learning.

To understand the differences between the sparse coding dictionaries learned by SAILnet, we examined the sparseness of SAILnet activations after training on each dataset. Fig. 4.6a shows the sparseness of each SAILnet unit, similarly to Fig. 4.3a. Since SAILnet activations are nonnegative spike rates, we did not plot the asymmetry of these activations. The thicker lines in Fig. 4.6a represent the activations for trained networks, whereas the thinner lines represent values of sparseness for networks with random dictionary elements (feedforward weights in the SAILnet architecture) after optimizing the other SAILnet parameters at fixed mean spike rate. Interestingly, the trained network had greater sparseness than the network with random dictionary elements, despite the fact that the mean firing rate of each network was fixed to the same value. While some of the qualitative features in Fig. 4.6a agree with those in Fig. 4.3a, others differ. Detailed comparison between these results and those in Fig. 4.3 is hampered by the fact that the two SAILnet networks do not achieve the same reconstruction error, as was the case for the results in Fig. 4.3.

To understand the differences in the learned dictionary elements between conventional sparse coding and SAILnet, we therefore also examined the distributions of linear projections of the data onto the dictionary elements. We found that SAILnet tends to learn stimulus features corresponding to data dimensions that are highly sparse and, when possible, more asymmetrical. Fig. 4.6b,c show rank plots for the sparseness scores and skewness magnitudes of SAILnet dictionary elements projected onto the relevant dataset. These plots are similar in many ways to Fig. 4.3b,d, which show the same statistics for conventional sparse coding dictionaries. The strongest differences are for the spectrogram case: SAILnet learns fewer elements corresponding to data dimensions with low sparseness scores, and almost all of its elements correspond to data dimensions with higher values of skewness than that of any of the 2000 random directions.

The discrepancy between sparse representations for images vs. speech due to skewness can be partly addressed by modifying the SAILnet model to allow for negative spikes. We note that this model with positive and negative spikes is not as biologically-plausible as the original SAILnet model. A complete dictionary learned by this modified SAILnet model is shown in Fig. 4.25. While it learns a few elements with harmonic structure that abruptly

reverses sign, a feature found with conventional sparse coding but not the original SAILnet algorithm, this model still does not capture all the features shown in Fig. 4.5a. Furthermore, a dictionary trained with a rectified version of LCA that does not allow negative activities still learns these features. Such a dictionary is shown in Fig. 4.24 and may be compared with the conventional sparse coding dictionary in Fig. 4.18. Thus, the non-negativity of SAILnet can partly, but not entirely, explain the differences between the dictionary elements it learns and those learned in conventional sparse coding models. Full dictionaries for all the models discussed are shown in the Supporting Information.

There are multiple differences between conventional sparse coding models and SAILnet that may appear relevant to the sparse features the models learn. By repeating our basic analyses with other modifications of SAILnet, we determined that the spike-based coding scheme does not noticeably affect the results discussed above but that the local learning rules for the dictionary elements and lateral connections are the crucial difference.

4.4 Discussion

Guided by the principle of sparse coding, we have explored the statistical structure of natural stimuli from two different sensory modalities, vision and audition. Both natural images and natural sounds admit sparse linear representations, but we have found some clear differences.

For complete sparse coding models trained on natural image patches, the lifetime sparseness of individual features was nearly uniform across the learned dictionary, reflecting the uniform sparseness of the linear projections of the dictionary elements onto the image dataset. Complete dictionaries trained on spectrograms of speech, however, showed a much wider range of lifetime sparseness values, both in terms of unit activations and projections, although the average sparseness was comparable for the two models. Moreover, the spectrogram dictionary included many units with highly asymmetric distributions of activity (and projections) across the dataset, unlike the highly symmetric distributions displayed by the image dictionary elements. Most of these trends persisted in the highly overcomplete regime, but we found that the lifetime sparseness of unit activations was greater for the image dictionary, unlike population sparseness, which was typically greater for the spectrogram dictionary.

We then compared the distribution of visual features learned by a biologically-plausible sparse coding model trained on images with the distribution of acoustic features obtained when the model was trained on speech spectrograms. Despite the strong agreement between the visual features learned by this model and those learned by more conventional sparse coding models, the spectrogram dictionary produced by this model differed markedly from the set of acoustic features learned by conventional sparse coding models.

Previous studies have made comparisons between the statistics of natural visual and acoustic data and their implications for neural coding in these modalities. Well-known examples of this include the fact that natural scenes and sounds both exhibit power spectra with power law functional forms [30, 3] and natural scenes obey spatial translational invariance

just as natural sounds obey time translational invariance. One property shared by visual and auditory responses is the gain dependence modeled by divisive normalization [90]. Recent work has also shown that a model that minimizes neural wiring while efficiently representing stimuli learns various subcortical receptive fields in the visual and auditory systems [91].

However, while sparse coding has been remarkably successful at predicting the receptive fields of V1 simple cells based on the structure of natural scenes, there is not yet a comparable result for primary auditory cortex (A1), despite the apparent sparse structure of natural sounds. That a linear sparse coding model can represent natural scenes at all is perhaps surprising given the highly nonlinear processes, such as occlusion by opaque objects and cast shadows, that cannot be explicitly represented by linear summation models. Conversely, raw acoustic waveforms are actually very close to linear summations of different individual component sounds in the environment. Consistent with this, previous work has demonstrated success with sparse coding at subcortical stages of the auditory system. A sparse coding model trained on raw auditory waveforms learns to tile time-frequency space in the same way as cat auditory nerve fiber filters measured by reverse correlation [97], but this model applies to the auditory nerve — the earliest stage of auditory processing once acoustic signals are converted into spike trains. Sparse coding models of nonlinear spectrogram or cochleogram [61, 96] representations can learn sparse structure on longer time scales [55], and some of the learned dictionary elements resemble the diverse STRF shapes found at various stages of the ascending auditory pathway [18], including the inferior colliculus (ICC), the medial geniculate body (MGB) of auditory thalamus, and even some neurons recorded in A1, but across the dictionary the agreement is not as strong for any brain region as has been demonstrated for V1 [76, 86, 110].

This dichotomy in the ability of sparse coding models to fully capture response properties of neurons in V1 vs. A1 could reflect the possibility that A1 and V1 are not directly analogous, even if they are both primary sensory cortices. If we include the visual processing taking place in the retina, there are roughly equal numbers of processing stages in the visual and auditory pathways leading to A1 or V1, as quantified by the number of synaptic connections needed to reach each of these cortical areas (although the auditory system has more subcortical areas along the way). However, due to the greater dimensionality of visual input (the two optic nerves are comprised of roughly 10^6 axons and there are $\approx 10^8$ photoreceptors, whereas there are fewer than 10^5 fibers in the two cochlear nerves) and strong nonlinearities such as occlusion affecting visual input, it may be that more stages of processing are required for visual signals to reach the same level of refinement as auditory representations in A1. This is qualitatively consistent with the greater number of visual cortical areas compared with the number of auditory areas.

The aspects of the sparse structure of natural sounds that differ from the structure of natural images could guide our pursuit of better models of the relevant auditory brain regions. Our analysis points to some relevant considerations. One is that the asymmetry between greater and lesser sound intensity is important, especially for biologically realistic models restricted to have nonnegative activations. In addition, the sparseness of individual features optimized to form sparse representations of spectrograms of speech vary widely compared to

the relatively uniform sparseness of sparse visual features. Moreover, dependencies among the activities of units in overcomplete dictionaries — which are most relevant for biology — influence which dimensions in stimulus space are most useful for sparse coding. A concrete manifestation of this is that a network such as SAILnet, in which units cannot cooperate directly based on the knowledge of other units' contributions to the coding, will not learn some of the same acoustic features as a network such as an LCA-based scheme, in which such cooperation is explicitly incorporated. The inter-unit connections in SAILnet, learned with only information locally available at the synapse, are more biologically plausible, but they lead to different behavior. In the auditory case, the differences include learning a more limited sparse coding dictionary that does not match as many receptive fields measured in real neurons. This observation suggests that SAILnet may need to be modified to better account for auditory sparse coding. More generally, the dependence on the stimulus statistics we observe for a biologically plausible model suggests that some properties of neural coding need to be specialized for the auditory system, even though it may share the basic principle of sparse coding with the visual system. A biologically realistic mechanism for finding approximate solutions to an optimization principle may be effective for one type of data, but not for another.

Indeed, SAILnet was specifically designed to model learning and inference in V1. In particular, it treats different orientations within its two-dimensional input on an equal footing, which makes sense given that these are all spatial dimensions in the visual case. In fact, the algorithm does not assume any special relationship between the various pixels — one could scramble their locations or convert the pixel array into a vector with any ordering and SAILnet would find the same features when mapped back to the unscrambled space. There is typically some mild anisotropy present in natural images (*e.g.*, vertical and horizontal edges are often slightly over-represented compared with random orientations), but this could be learned by using the same learning rules in all orientations in the two-dimensional image space. Spectrograms, however, are strongly and inherently anisotropic, with time represented along one cardinal axis and (a nonlinear function of) acoustic frequency along the other. Perhaps this contributed in some way to the divergence between our SAILnet results for speech spectrograms and what we found using conventional sparse coding, but if so this is a subtle effect given that the LCA-based sparse coding algorithm we used also employs isotropic rules for learning and inference. We emphasize that, even though SAILnet may not treat time in a natural way for a biologically-realistic mechanistic model of auditory processing, it provides a useful tool for identifying aspects of the sparse structure of natural sounds that differ from those of natural scenes.

Motivated by previous work [59, 97], we analyzed speech data as a proxy for a more complete collection of "natural sounds." As recapitulated here, spectrograms of speech by themselves have a rich sparse structure, with several distinct feature types that our models use for their sparse codes; some of these features resemble STRFs measured in inferior colliculus and other brain regions [18]. Using speech is particularly convenient, since using ensembles of recorded sounds has been shown to yield good agreement between sparse coding predictions and auditory nerve response properties, such as the same time-frequency

trade-off, only when the relative proportions of three different types of recorded sounds are empirically adjusted to fit the model [59, 97]. Thus, using speech data is in some sense a more principled approach, since it removes two adjustable parameters from the model.

This picture is somewhat complicated by the fact that the filters learned in such models depend on the sound class used even when the time-frequency tiling properties match [59]. It is also unlikely that speech captures the structure of natural sounds that occurs on the longer time-scales of our spectrogram-based models. To address this, we fit sparse coding models to the ensemble of natural sounds described in [97]. The sparse structure captured by our models in that data is less rich than, and largely redundant with, what we found for speech. We have included a dictionary and sparseness rank plot in Figs. 4.26 and 4.27.

More broadly, what constitutes the relevant ensemble of "natural scenes" or "natural sounds" is not clear to us; these notions may not be well-defined or independently determinable in a way that does not rely on fitting neural response properties. Another question is whether or not one can determine definitively if a given type of natural signal is truly sparse with the sort of analysis employed here. In particular, preprocessing using PCA or some other dimensionality reduction technique necessarily changes the structure of the data for any realistic scenario (*i.e.*, unless the raw signal is strictly L0 sparse, with the relevant dimensions contained within the space spanned by those PCA dimensions retained for later analysis). There are persuasive arguments challenging the notion that natural scenes or sounds are truly sparse, in the L0 sense, for the sort of linear generative models we have considered here [64, 38]. In addition, the oriented filters learned by sparse coding models represent a shallow optimum, as representations of natural scenes using center-surround filters, for example, are almost as sparse [11, 29]. There are good reasons to question [64] why the local oriented filters predicted by sparse coding models trained on natural images would first appear in V1, skipping past the highly nonlinear retina and lateral geniculate nucleus (LGN), if indeed sparseness is the correct normative principle for the visual system. Moreover, the sparseness of successive sensory representations at higher stages of processing in the ascending auditory [22] and visual [87, 88] pathways does not always appear to increase.

There exist alternative choices for the models as well as the data, and different models yield different results. The results from SAILnet learning differ enough from those from gradient descent learning to be of interest, but we did not observe any substantial differences between the results of gradient descent learning using different algorithms to compute the sparse codes. Furthermore, while a greater variety of sparse features of natural images than found in early sparse coding work (e.g., [76]) has been shown using various methods, we are not aware of any work showing sparse features of natural images that do not have qualitative matches in a 10-times overcomplete conventional sparse coding network. We believe that our approach at least captures the known sparse structure of natural images in terms of feature diversity and so can be taken as representative of the subtly varying results that different sparse coding and learning algorithms uncover.

In this work, we have taken a pragmatic approach to our model selection and data choices. Undoubtedly, the specific sparse coding models we have employed here are imperfect approximations to whatever model would best fit ensembles of natural scenes and sounds as

defined by our datasets, but by applying these models to both images and sounds, we have been able to identify several similarities and differences between the statistical structure of these natural signals. We are, of course, motivated by the fact that the sparse coding models we consider here can predict receptive fields in V1 and several cell types at various stages of the ascending auditory pathway, even if these models do not entirely capture the statistics of natural signals. It will be interesting in future studies to explore more fully the structure of natural stimuli, and its implications for neural coding.

Beyond the particular results presented in this work, we have shown that it is possible and fruitful to compare the sparse structure of natural data from different modalities. The principle of sparse coding appears to have applicability to auditory data as well as visual data, supporting the idea that sparseness is, to at least some degree, a universal property of natural data. Nonetheless, we have found that there are aspects of sparse structure that are clearly not universal. Understanding these differences offers insights into the structure of natural stimuli and into the ways in which neural systems represent it.

4.5 Methods

Data

We performed our primary analyses on three sets of natural data; Fig. 4.1 illustrates the preparation of the two primary datasets we compared. The same preprocessing steps were taken where possible, in order to reveal the effects of the structure inherent in the data rather than differences in how the data were presented to the sparse coding algorithms. In addition to these two comparably prepared datasets, we used an image dataset preprocessed by methods common in the literature to reveal some effects of this processing. Results from this alternative image dataset are discussed in Section S1.

Following previous work [55, 97, 18], we focused on human speech as a rich class of natural sounds. Speech data were taken from the TIMIT continuous speech corpus [33] and preprocessed as in [18]. Specifically, we divided each waveform by 10 times its variance and removed any DC value. We then used MATLAB's [68] spectrogram function to calculate the discrete Fourier transform (DFT) of Hamming-windowed segments of 16 ms (256 samples) of sound, with neighboring segments overlapping by half their length. The DFT was sampled at 256 frequencies logarithmically spaced between 100 Hz and 4 kHz. We trimmed the power spectrograms to remove periods of silence and then took the logarithm of the results. We divided these spectrograms into overlapping 25-timepoint (216 ms edge-to-edge) segments, yielding about 3×10^5 spectrogram segments. While this procedure is not a precise model of early auditory processing, previous work has found better agreement with experimental data using spectrograms than with preprocessing meant to emulate the cochlea [18]. Spectrograms also provide a representation often used for generating stimuli and visualizing spectro-temporal receptive fields in the experimental literature, e.g., [70, 81, 32, 85, 103]. Although using only the (log) power obscures the phase structure, the original sound waveforms can in fact

be reconstructed from power spectrograms using implicit phase structure from overlapping windows[58].

Natural image data was taken from a subset of the van Hateren database [35] with minimal blur and other artifacts (see [73]). We extracted approximately 3×10^5 80-by-80 pixel patches from the images and took the logarithm of the intensity at each pixel. The mean log-intensity was removed from each patch.

The speech spectrogram segments and the natural image patches were both 6400-dimensional, and we used PCA to reduce the dimensionality to 200. We also divided each principal component by its variance, achieving a "whitened" or "sphered" representation in which the empirical covariance matrix was equal to the identity matrix [52]. The PCA step discarded about 7% of the variance in each of the two raw datasets. Another 18% of the original variance in the images was removed by the patch-wise mean subtraction described above. No comparable effort was made to remove the dimension of largest variance in the spectrogram data, following [18]. After whitening, this dimension had the same variance as the others and therefore did not strongly affect our results.

Our choices were driven by the need to make the two datasets comparable, so our preprocessing differed from that employed in much of the literature. We repeated our analyses on a third dataset, containing the same natural images as they were preprocessed in [73] and other sparse coding work. There were two key differences: first, [73] used small image patches of 16x16 pixels while we used larger patches of 80x80 pixels. Since [73] first downsampled by a factor of 2, the scale of our images is better compared to 32x32 patches. Since natural images have less variance in higher spatial frequencies, our dimensionality reduction also discarded the information destroyed by this downsampling.

The other crucial difference between these two image datasets is due to the whitening step. Olshausen and Field [76, 75, 73] whitened their raw images using a filter that flattens the Fourier spectrum at low frequencies while allowing the variance of very high frequencies, which is largely noise, to remain small. In contrast, we exactly equalized the variance of the first 200 principal components and removed the other components entirely. Results with the images preprocessed as in [73] are discussed in Section S1.

Reconstructions of original data from our reduced representations are shown in Fig. 4.28.

Sparse coding

Sparse coding is a probabilistic model

$$p(x;\Phi) = \int p(x|a)p_a(a)da, \tag{4.3}$$

where x denotes a data vector with components x_i and a denotes a set of latent variables a_m. The conditional distribution $p(x|a)$ is an isotropic Gaussian of fixed variance centered on a linear reconstruction of the data in terms of the dictionary elements Φ_m:

$$p(x|a) \propto \exp\left[-\frac{1}{2\sigma^2}\sum_i (x_i - \sum_m \Phi_{mi}a_m)^2\right] \tag{4.4}$$

and the prior distribution of the coefficients a_m is factorial with each factor given by the same sparse distribution (in this work, a Laplace distribution):

$$p_a(a) \propto \prod_m e^{-\lambda|a_m|}, \tag{4.5}$$

where λ is a parameter that determines the width of the distribution and therefore how strongly the prior favors sparse sets of a_m.

The a_m are determined by maximum *a posteriori* (MAP) inference given input data x:

$$a^{\text{MAP}} = \arg\max_{\{a_m\}} p(x|a_m)p_a(\{a_m\}) = \arg\max_{\{a_m\}} e^{-\sum_i (x_i - \sum_m \Phi_{mi} a_m)^2/2\sigma^2} \prod_m e^{-\lambda|a_m|}. \tag{4.6}$$

The parameters σ^2 and λ now only affect the model through the combination $\lambda\sigma^2$, so for simplicity of notation we set $\sigma = 1$.

The a_m^{MAP} are often referred to as the activity of the mth unit, and the dictionary elements Φ_m are often compared to receptive fields of neurons. The analogies to neurons suggested by these terms are not exact, but a unit's dictionary element is approximately the same as the linear receptive field that would be measured for that unit with an activity-triggered average [76].

The dictionary elements Φ_m are learned by descending the estimate of the gradient provided by differentiating the model log-likelihood with respect to Φ with a fixed at the MAP value:

$$\Delta\Phi_{mi} \propto -\frac{\partial}{\partial \Phi_{mi}} \left[-\frac{1}{2} \sum_j (x_j - \sum_m \Phi_{mj} a_m^{\text{MAP}})^2 \right]. \tag{4.7}$$

For each step, this gradient with respect to Φ is averaged over a minibatch of 100 data examples.

The use of MAP inference requires that we constrain the norms of the Φ_m to prevent solutions with small a_m and large, meaningless Φ_m. We therefore divide each Φ_m by its norm after each gradient step. Using the MAP estimate to compute gradients for learning is not guaranteed to result in the same learned dictionary Φ, but a method that uses more samples from the posterior learns familiar Gabor functions on whitened natural image patches [101].

Locally Competitive Algorithm

We used the L1-sparse locally competitive algorithm (LCA) [86] to perform MAP inference. LCA uses a dynamical system with auxiliary variables that are thresholded to obtain estimates of a^{MAP}. Typically most of the auxiliary variables are below threshold and the a_m^{MAP} estimates are exactly zero for most m. The threshold is set by the sparseness parameter λ. We dynamically adjusted this parameter to achieve reconstructions with 15 dB signal-noise ratio while training the models, allowing direct comparison to the results of [73].

The choice of coding algorithm is not crucial to our results, and learning using alternative inference schemes yields similar dictionaries. This is particularly true for dictionaries that

are not overcomplete, as demonstrated by the similarity of the results in Fig. 4.13, which used Independent Components Analysis (ICA) [9], to Fig. 4.2, which used LCA and stochastic gradient descent on the mean-squared error.

SAILnet

We used the Sparse and Independent Local network (SAILnet) model [110] to study how the statistics of different stimuli interact with biologically realistic constraints. SAILnet uses spiking neurons and synaptically local plasticity rules to achieve sparse codes. Mathematically, SAILnet can be understood as optimizing the Lagrange function

$$\mathcal{L} = \frac{1}{2}\sum_{mi}(X_i - \Phi_{mi}a_m)^2 + \sum_m \theta_m(a_m - p) + \frac{1}{2}\sum_{mn} W_{mn}(a_m a_n - p^2). \tag{4.8}$$

Here the first term approximates the mean-squared error in the sparse coding log-likelihood in the limit that the a_m are sparse and uncorrelated. Maximizing with respect to the Lagrange multipliers θ_m and W_{mn} constrain the a_m to be sparse, with average activity $p \ll 1$, and uncorrelated. The a_m are the firing rates of leaky integrate-and-fire circuits with thresholds θ_m and inhibitory connections between neurons with strengths W_{mn}. The dynamics of this circuit approximately seek firing rates a_m that minimize $\mathcal{L}$. As in conventional sparse coding, the dictionary elements Φ_{mi} are updated at fixed a_m using; the Lagrange multipliers are updated at the same time but with greater rates to ensure the constraints are satisfied during learning.

The SAILnet Lagrange function, and in particular the approximation to mean-squared error in the first term of Eq. 4.8, allow the gradient descent update for each connection to be computed using only information available at that connection, e.g., one only needs to know a_1 and a_2 to update W_{12}. The cost of this locality is that SAILnet units do not directly learn to cooperate to represent the data.

Although SAILnet has been shown to learn the expected dictionary Φ on whitened natural images, in some ways it behaves differently from a conventional sparse coding algorithm such as LCA with gradient-descent based learning. Here we have focused on how SAILnet interacts with differing input statistics.

Model implementation

We implemented soft-thresholded LCA [86] in TensorFlow [67] to learn the overcomplete sparse coding dictionaries. We implemented SAILnet in Python. Code for these implementations may be found online at github.com/emdodds/DictLearner and github.com/emdodds/SAILnet. For the ICA results shown in Fig. 4.13 we used the FastICA [42] implementation in scikit-learn [78].

Alternative image preprocessing

We focused on comparing the intrinsic sparse structure of speech sounds to that of natural images. This required making some preprocessing decisions that are not standard in the literature, as discussed in Data. Here we discuss the results of our analyses on the data from [73], preprocessed as described in that work. These image patches are 16x16 pixels and have been approximately whitened by applying a filter to the original images.

Fig. 4.7 shows sparseness and skewness plots and example distributions for complete sparse coding on the filter-whitened images, expanding Fig. 4.2 to include results on this third dataset. Random directions in the space of 16x16 images whitened as in [75] are fairly sparse, with median sparseness score 0.13. This fact is well known and largely accounted for by variation in the local variance of natural images [5, 63]. The trend of excess sparseness of the sparse coding dictionary closely follows the trend for the other image dataset.

For the PCA-reduced datasets, we used 10-times overcomplete sparse coding, while for the 16x16 image patches we used a network that was nominally 8-times overcomplete, making it about 10-times overcomplete given that some dimensions are essentially noise. Our results on this dataset closely match those shown in [73]. A dictionary is shown in Fig. 4.20, and sparseness and skewness rank plots are shown in Fig. 4.10.

The sparseness of an overcomplete dictionary element's linear projections is not closely correlated to the sparseness of that element's LCA activations for this dataset: Pearson's r of -0.16. The skewness of the activations is better explained by the skewness of the linear projections, with Pearson's r 0.59. These observations qualitatively echo what we saw for the other datasets.

The fact that the 16x16 image patches are not fully whitened hampers meaningful comparisons among the various population sparseness results. Plots are shown in Fig. 4.12. The filter-whitened images generally admit sparser representations, but the effect is driven by the whitening scheme and not by the intrinsic structure of the data. This is already suggested by the fact that PCA-whitened natural images yield very different results.

A simple argument also demonstrates why whitening should matter, particularly for the sparseness of optimized sparse codes as we computed them. In an extreme case, the data variance may be so much greater along one dimension than all others that it is possible to achieve 15 dB SNR reconstructions with only the special dimension nonzero. Then only one unit need be active, in which case the population sparseness of the representation is approximately 0.8 by our measure (the precise value depends on the dictionary size) for most data examples. The same data, after whitening, does not permit this trick since no direction has more variance than any other. The filter-whitened images are not exactly whitened, and the residual variation in the variance of different dimensions allows a weaker version of this trick to work. Imperfect whitening can also strongly affect the features found by SAILnet — an interesting topic for future work.

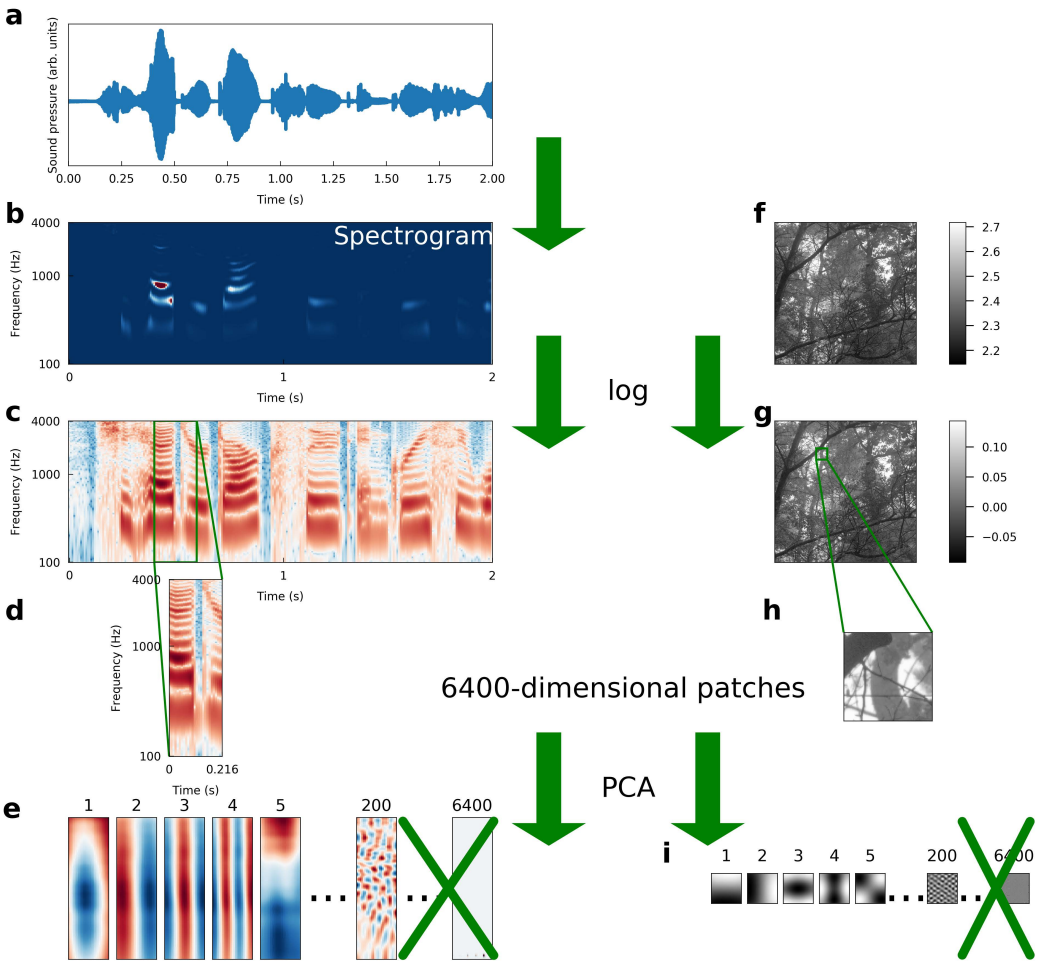

Figure 4.1: **Schematic illustration of preprocessing.** We preprocessed a set of natural images and a set of speech sounds using steps as similar as possible to allow for a meaningful comparison of the intrinsic structure of the datasets. (**a**) The raw auditory data consisted of recordings of speech from the TIMIT corpus [33]. The blue curve is the sound pressure waveform of an isolated speaker uttering the first two seconds of "She had your dark suit in greasy wash water all year." (**b**) Spectrograms were computed from the waveforms (see Methods for details). The color of each pixel represents the intensity (red is more intense, blue is less) of sound at a particular frequency and a particular time. (**c**) We took the logarithm of each intensity spectrogram. (**d**) The spectrograms were divided into overlapping segments of 25 time points each, derived from 216 ms of audio. Since 256 frequencies were sampled at each time point, these spectrogram segments were each 6400-dimensional. (**e**) Principal components analysis was used to whiten the data and reduce its dimensionality to 200. (**f**) Raw image data were taken from the Van Hateren natural image dataset [35]. The lightness of each pixel represents the intensity of light at that location. (**g**) We took the logarithm of each intensity spectrogram and each intensity image. (**h**) Patches of 80 pixels on each side were taken from the log-intensity images to make 6400-dimensional image patches. (**i**) We repeated the PCA procedure we used for spectrograms exactly on the set of image patches.

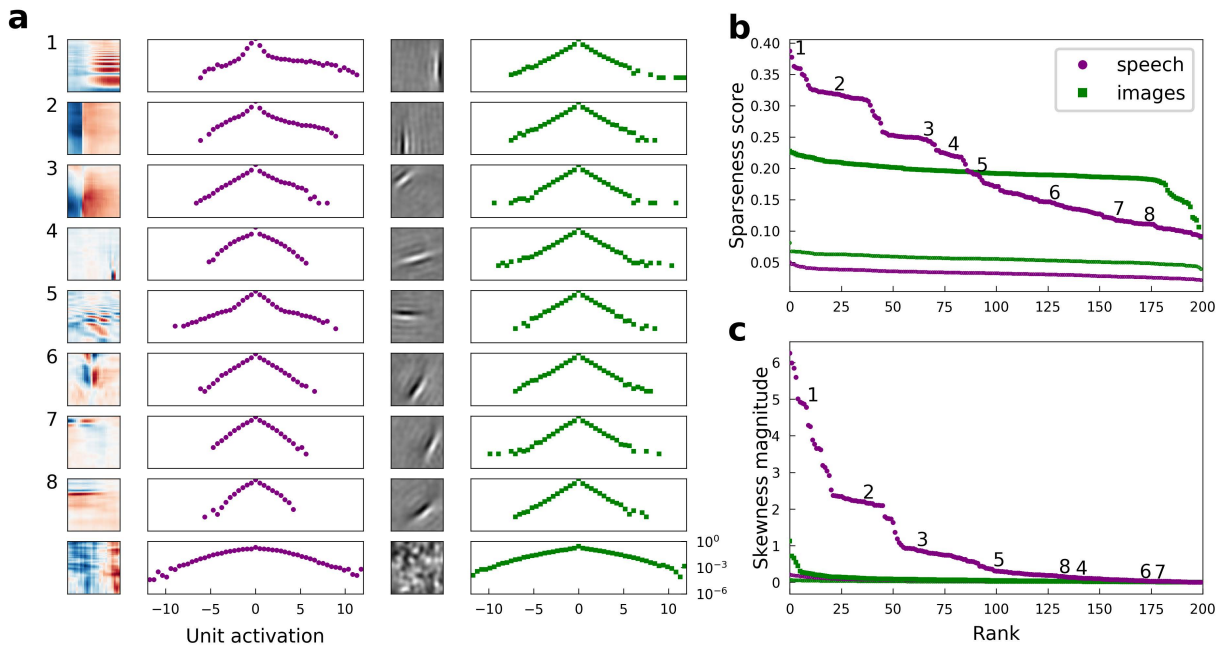

Figure 4.2: **Natural images and speech each exhibit sparse structure, but with clear differences.** (**a**) Log-histograms of unit activations for individual elements (units) of complete dictionaries show more skewness and a greater range of sparseness for representations of speech (left pair of columns) than for representations of images (right pair of columns). Each square contains the spectrogram (left column) or image patch (third column) representing one sparse coding dictionary element fit to the corresponding dataset. For the spectrogram elements, white regions have no effect on the element's activity while red denotes positive weights and blue negative weights. For the image patches, gray represents zero and lighter pixels represent positive weights. Each element has been multiplied by the sign of the skewness of its activations to show the element as it is used by the sparse coding network. The log-histogram next to each element shows how often the unit had a given level of activity. No marker is shown when a histogram bin is empty. The horizontal scale is consistent for each dataset, and the vertical scale is the same for all histograms. The last example for each dataset is not from a trained network but from a network where all elements are independently and identically distributed (iid) Gaussian noise in each principal component after whitening. As expected, these Gaussian noise control cases yield approximately parabolic curves, compared with the sparser distributions for the learned dictionary elements. (**b**) Rank plots for the sparseness score of the distributions of activations of each unit. Each point corresponds to one dictionary element. The numbered points correspond to the numbered examples in panel (a). Since the activation of one element cannot be determined without the other elements due to the non-linear nature of LCA, these statistics can be meaningfully compared to their values for a full dictionary of random elements (thin curves at bottom). For these curves, we generated 200 random directions in PCA space and evaluated the sparseness scores in the same way as for the learned dictionary elements. (**c**) Rank plots analogous to (b) with the magnitude of skewness (Eq. 4.2) substituted for sparseness.

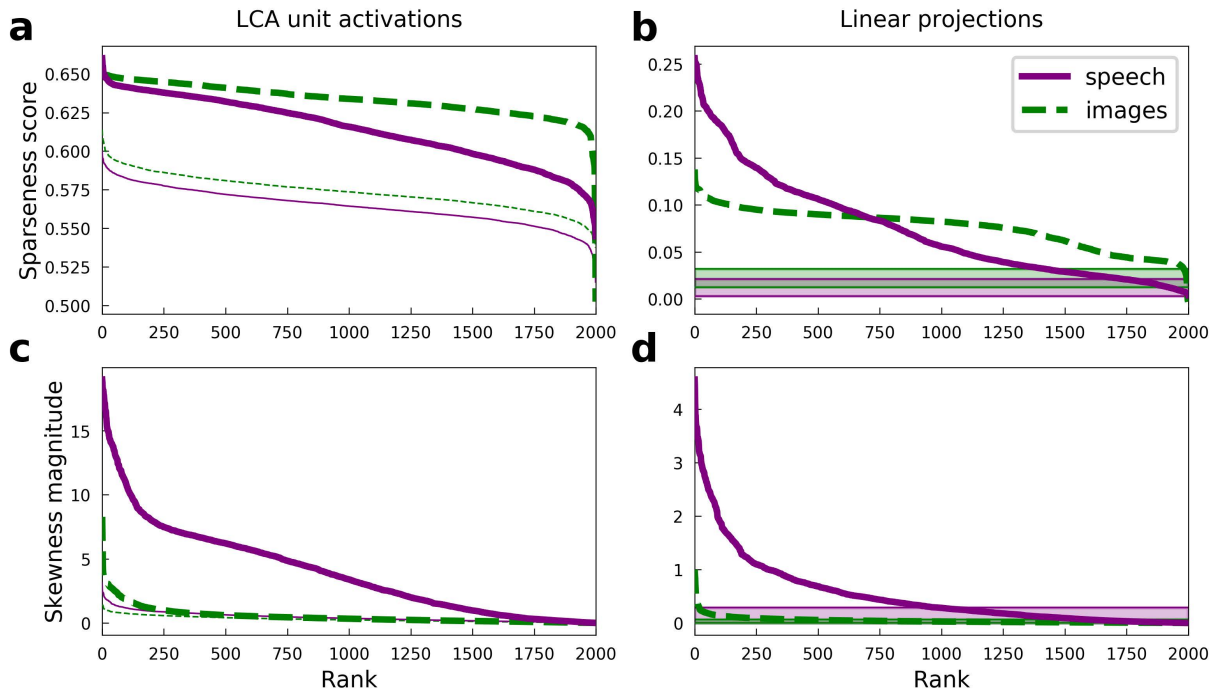

Figure 4.3: **Differences in the sparse structure of natural images and speech are also present in overcomplete representations.** As in Fig. 4.2, the various measures presented here are closely related to the lifetime sparseness of individual elements. Ten-times overcomplete sparse coding dictionaries were trained on either the spectrogram dataset or the natural image dataset. Each curve represents 2000 discrete points, with each point representing one element of the corresponding dictionary. (**a**) Sparseness scores were calculated for the activations of the sparse coding dictionary elements using the same L1-sparse LCA algorithm that was used to train the models. To provide meaningful comparisons, the thin curves represent the same analysis applied to 2000 random directions in the PCA space of the corresponding dataset. As for the complete case shown in Fig. 4.2b, training on speech sounds produced a wider range of sparseness scores across elements. Unlike the complete case, however, the median sparseness was clearly greater for elements from the model trained on images than for the model trained on speech. (**b**) Sparseness scores were calculated for the distribution of each dictionary element's linear projections onto the dataset. Each sparseness score is a statistic of the corresponding dimension of the data space, independent of the sparse coding learning algorithm that was used to find that unit's dimension. As in panel (a) and Fig. 4.2b, the speech dictionary showed a greater range of sparseness values, but unlike panel (a), there was no systematic difference in the sparseness scores for speech and images. The shaded regions at the bottom show the range of sparseness values one might achieve by chance in the corresponding dataset. Sparseness scores for each dataset were calculated for 2000 random Gaussian-distributed vectors instead of dictionary elements; the minimum and maximum scores determined the bounds of the shaded region. (**c**) Similar to (a) but with skewness magnitudes instead of sparseness scores. (**d**) Similar to (b) but with skewness magnitudes instead of sparseness scores.

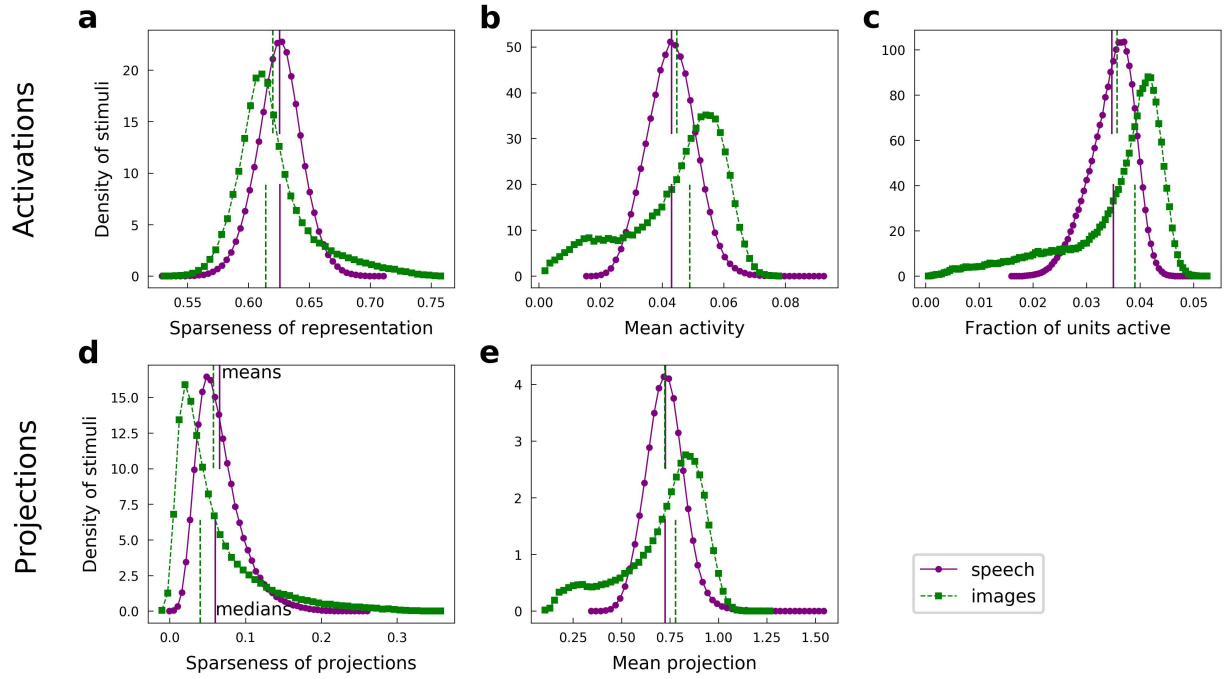

Figure 4.4: **Speech and natural scenes exhibit different distributions of population sparseness for overcomplete dictionaries.** (**a**) Sparseness scores (Eq. 4.1) were evaluated on the sparse codes (unit activations) generated by LCA for each spectrogram or image patch in the corresponding dataset. This is one measure of "population sparseness," which quantifies the sparseness of the full network's representation of individual images or sounds. The purple circles form a histogram of the sparseness scores across the speech dataset, with a linear interpolation plotted in solid purple for clarity. The vertical solid purple lines represent the mean (top) and median (bottom) of the distribution. The green squares constitute the analogous histogram for the image dataset, with a dotted green interpolation curve and vertical dashed lines indicating the mean (top) and median (bottom). Note that, though the means of the two histograms are quite similar, the medians are well separated, indicating qualitatively that a typical speech sound from our auditory dataset tends to project onto fewer dictionary elements than a typical image patch from our visual dataset. (**b**) Similar to (a), but with the mean activity level replacing the sparseness score. (**c**) The LCA-generated representations had most units completely inactive for any given input (*i.e.*, the L0 sparseness was high). Here we plot histograms of the fraction of LCA units active while representing each image patch or histogram from the relevant dataset. (**d**) Sparseness scores were evaluated on the set of projections of each image patch or spectrogram segment onto the corresponding sparse coding dictionary. These histograms follow the same conventions as in panel (a). (**e**) Similar to (d), but with the mean absolute value of the projection replacing the sparseness score.

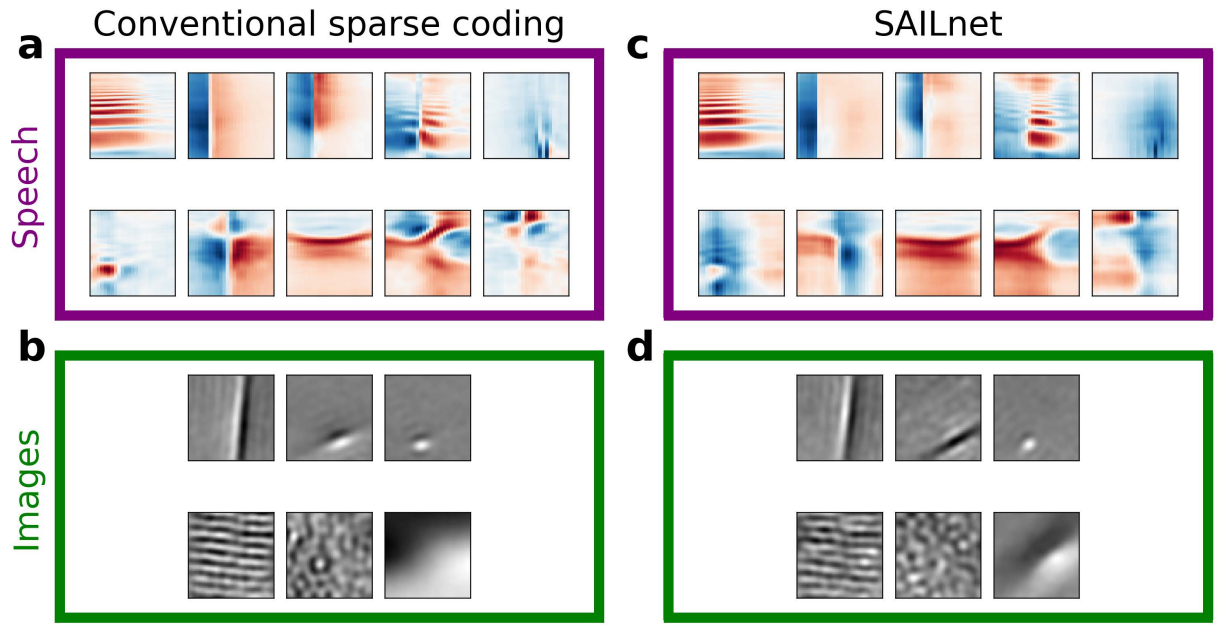

Figure 4.5: **SAILnet and conventional sparse coding learn similar representations when trained on natural images, but not speech.** Each box shows elements from a ten-times overcomplete dictionary learned with conventional sparse coding (left) or with SAILnet (right) on one of the datasets. (**a**) For a ten-times overcomplete sparse coding dictionary trained on spectrogram segments, we handpicked elements that show qualitatively different structure. These element types do not occur with equal frequency in the dictionary. (**b**) Elements selected from a dictionary trained on image patches. There are apparently fewer distinct classes of elements in this dictionary than in the speech-trained dictionary. (**c**) SAILnet dictionary elements were selected so as to minimize the angle to each hand-picked sparse coding element. While this yielded similar elements in some cases, there are no elements in the SAILnet dictionary that match several of the dictionary element types seen in the conventional sparse coding dictionary for speech data. (**d**) The SAILnet dictionary trained on images includes good qualitative matches to every element from the corresponding conventional sparse coding dictionary. Full dictionaries are shown in Figs. 4.18, 4.19, 4.21, and 4.22.

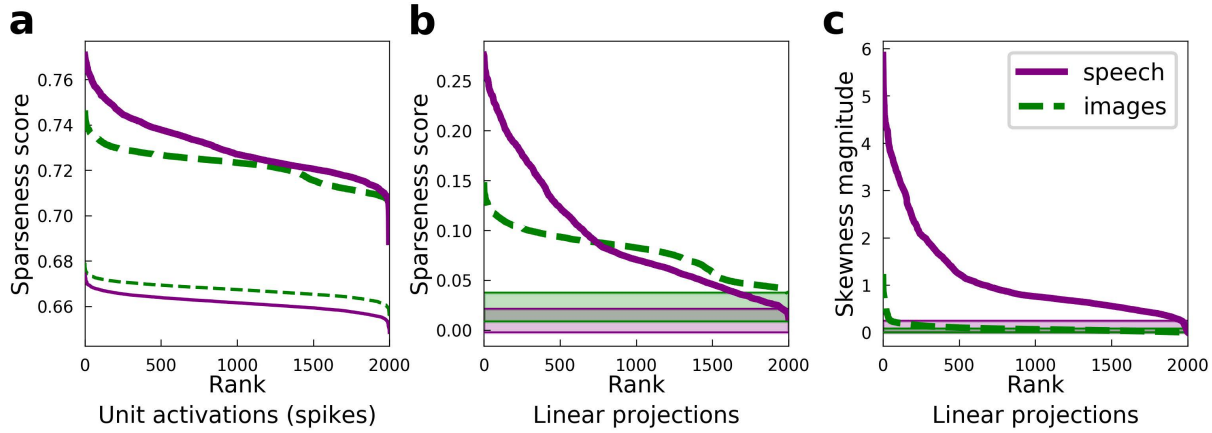

Figure 4.6: **Statistics of overcomplete SAILnet representations are similar to conventional sparse coding for natural images, but they differ for speech.** Plots as in Fig 4.3a,b,d, for ten-times overcomplete dictionaries learned by SAILnet rather than LCA-based learning. (**a**) SAILnet activations are extremely sparse. This plot is analogous to Fig. 4.3a but a direct comparison of the learned dictionaries through these plots is confounded by the differences between LCA and SAILnet inference. (**b**) The sparseness score rank plots qualitatively resemble those for conventional sparse coding (compare with Fig. 4.3b). For the spectrogram-trained dictionary, the lower-rank tail contains somewhat higher sparseness scores than for the conventional sparse coding dictionary. This observation is consistent with SAILnet not utilizing some of the element types conventional sparse coding does (see Fig 4.5a), since the data's projections onto these element types tend to have relatively low sparseness scores. The shaded regions at the bottom show the range of sparseness values one might observe by chance in the corresponding dataset. Sparseness scores for each dataset were calculated for 2000 random Gaussian-distributed vectors instead of dictionary elements; the minimum and maximum scores determined the bounds of the shaded region. (**c**) Almost all the SAILnet dictionary elements in the spectrogram case correspond to directions in the data space with large skewness. This is consistent with SAILnet not learning some of the element types shown in Fig 4.5a, which tend to have symmetric distributions (*e.g.*, Fig. 4.2, example 6). The shaded regions at the bottom indicate the range of possible skewness values one might observe by chance based on the same skewness analysis applied to 2000 random Gaussian-distributed directions in PCA space for the corresponding dataset.

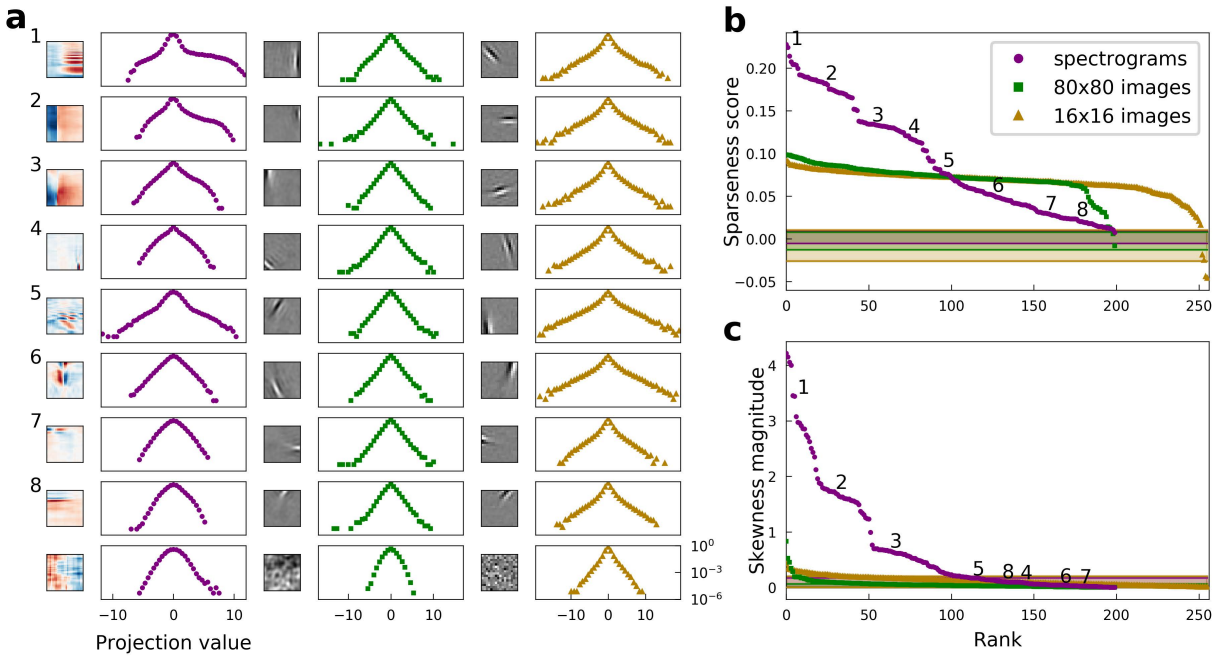

Figure 4.7: **Histograms of projections exhibit the same sparse structure as unit activities for both natural images and speech in the complete regime.** This figure is identical to Fig. 4.2, but with unit activities replaced by linear projections of individual dictionary units on the corresponding dataset, and the addition of a third dataset consisting of filter-whitened 16x16 image patches (rightmost pair of columns in panel (a), beige curves in panels (b) and (c)). Note that these results look very similar to those in Fig. 4.2, demonstrating that projections and activities exhibit very similar statistics in the complete regime, even for different preprocessing of the image dataset. The shaded regions at the bottom of panels (b) and (c) show the range of sparseness or skewness values one might observe by chance in the corresponding dataset; sparseness scores (skewness values) on each of the three datasets were calculated for 200 Gaussian-distributed random vectors instead of dictionary elements, and the minimum and maximum scores (values) determined the bounds of the shaded region for the corresponding dataset. All results in this figure are for linear projections and not sparse codes represented by unit activities generated by a nonlinear coding algorithm. However, the results are similar for dictionaries learned with independent components analysis (ICA), in which case the activity corresponding to each dictionary element is itself a linear projection of the stimulus (Fig. 4.13).

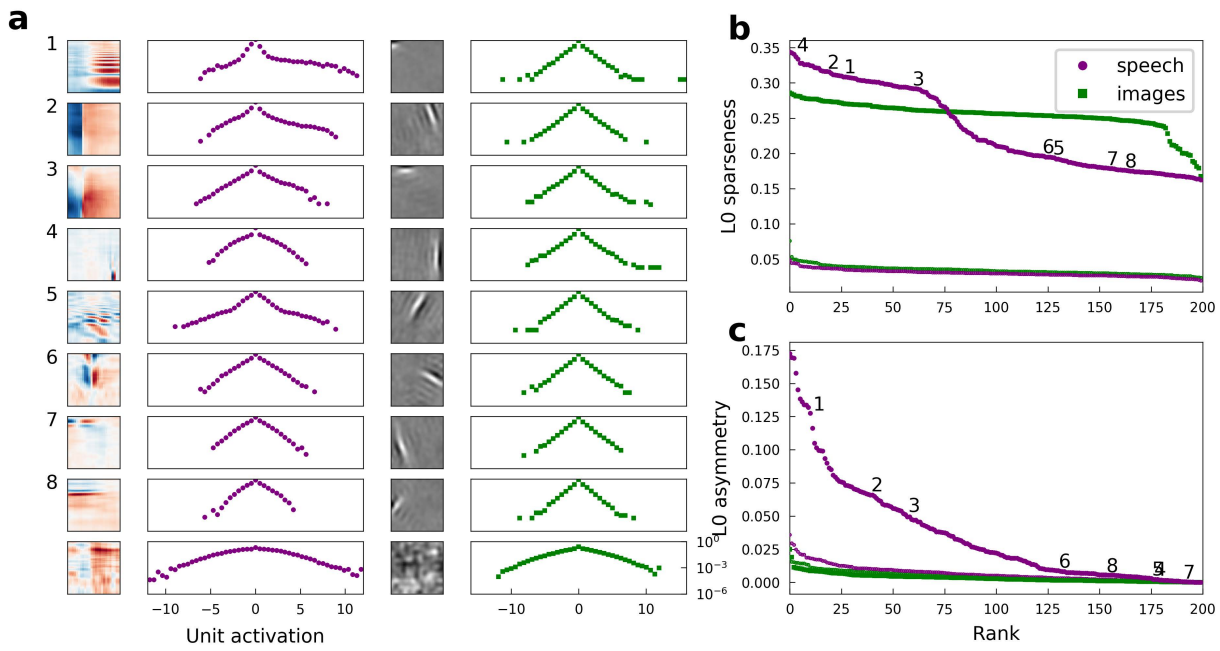

Figure 4.8: **L0 lifetime sparseness rank plots for complete sparse coding** This figure is identical to Fig. 4.2 except that panels b and c show statistics that disregard the magnitude of activations. The L0 sparseness of a unit is the fraction of stimuli that did not elicit a nonzero activation from the unit. The L0 asymmetry is the absolute value of the difference between the fraction of stimuli that elicited a positive response and the fraction that elicited a negative response.

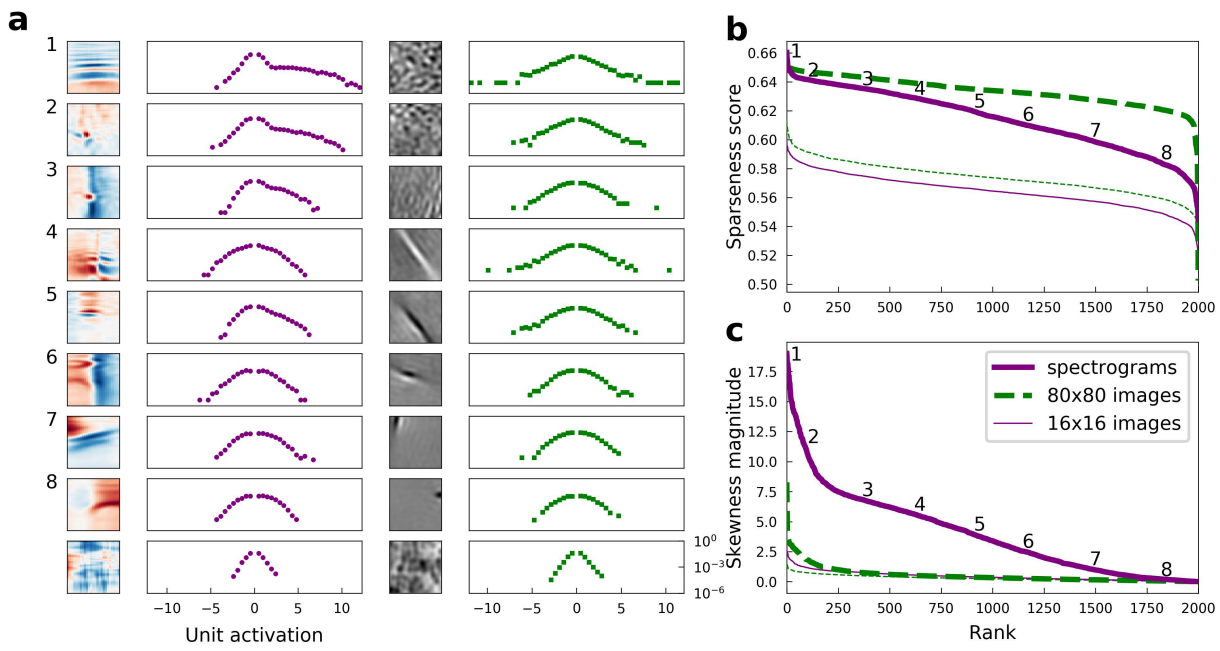

Figure 4.9: **Overcomplete models exhibit some of the same contrasts observed in complete models.** Same statistics as in Fig. 4.2

evaluated for 10-times overcomplete models. Although there is no clear clustering of auditory feature types as observed in the complete regime, the auditory features continue to show a greater diversity of sparseness values and activity distribution shapes. Note that the full auditory dictionary is shown in Fig. 4.18 and that the examples in panel **a** do not represent all the feature types in the overcomplete spectrogram dictionary.

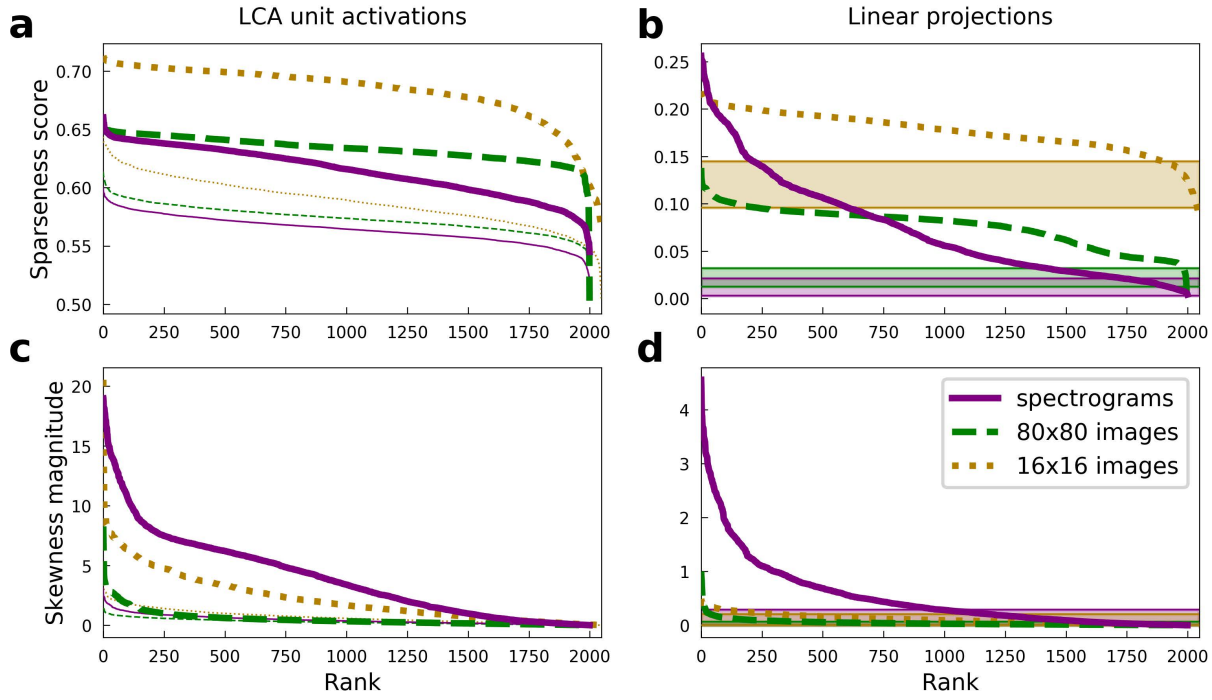

Figure 4.10: **Statistics of highly overcomplete sparse representations for spectrograms and images with both preprocessing schemes.** This figure is identical to Fig. 4.3 except for the addition of a third dataset consisting of filter-whitened 16x16 image patches (beige).

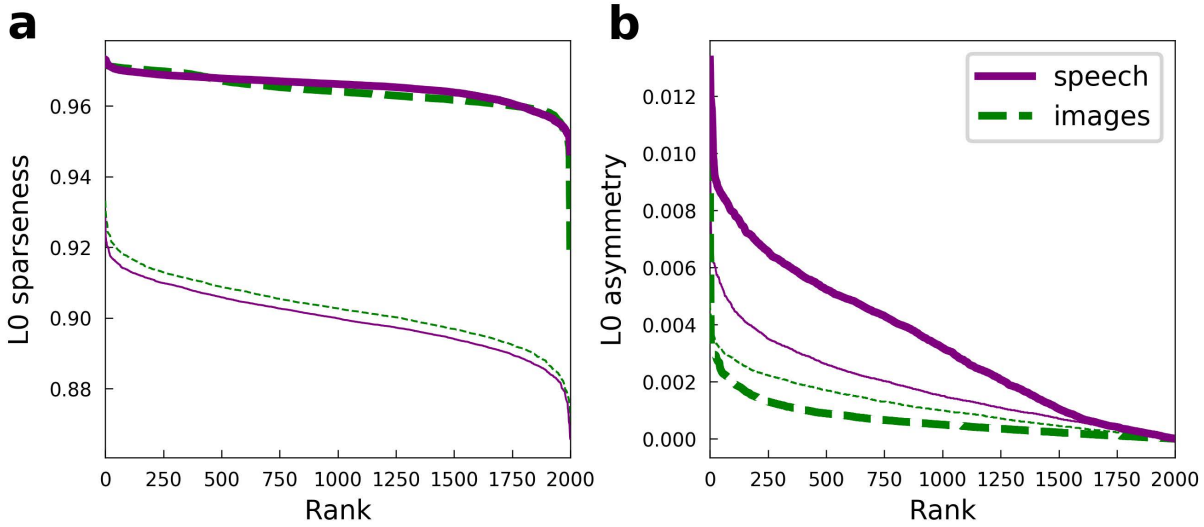

Figure 4.11: **Statistics of highly overcomplete sparse representations for spectrograms and images: using L0 norm.** This figure is identical to Fig. 4.3a,c except that here L0 sparseness rather than L1 sparseness is plotted in panel a, and the asymmetry in panel b is quantified as the absolute value of the difference between the fraction of inputs that gave a positive response for a given unit and the fraction that gave a negative response. Note that, unlike what we found for L1 sparseness, the range of values for L0 sparseness are nearly identical for the image and speech models. However, the "L0 asymmetry" we plot in panel b is still much greater for typical speech units compared with image units, just as we found using other measures of asymmetry.

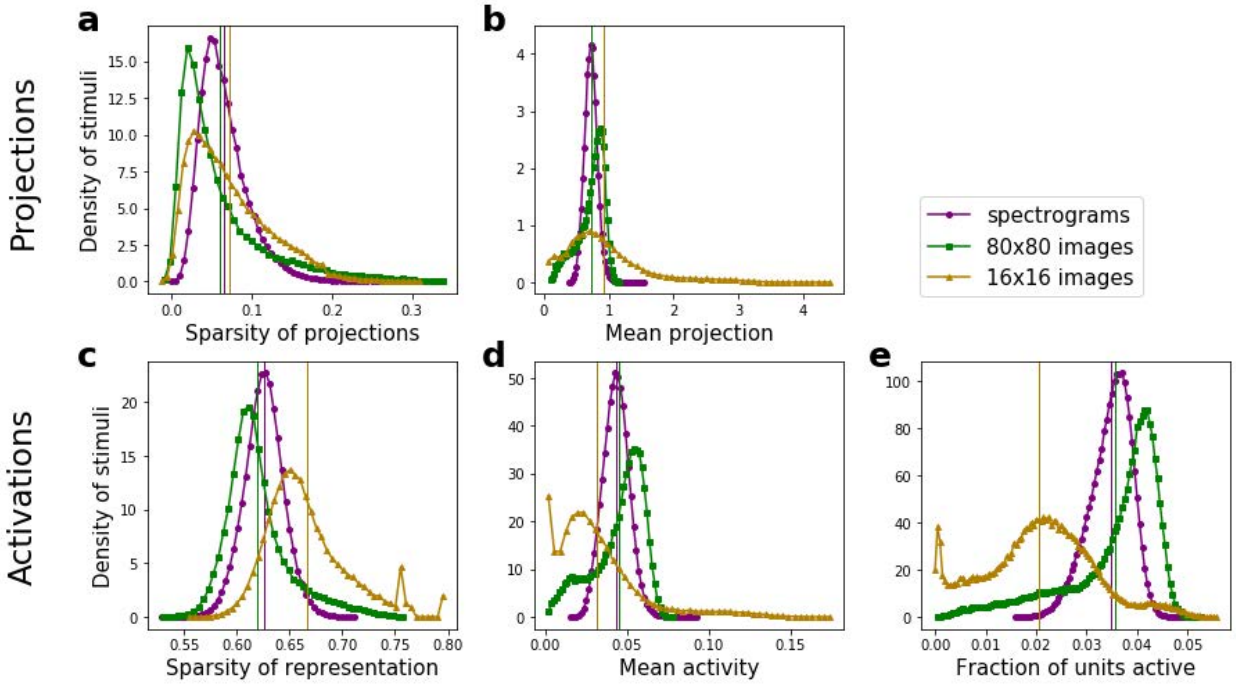

Figure 4.12: **Distributions of population sparseness statistics across each dataset, including filter-whitened images** This figure is identical to Fig. 4.4 except for the addition of a third dataset consisting of filter-whitened 16x16 image patches (beige).

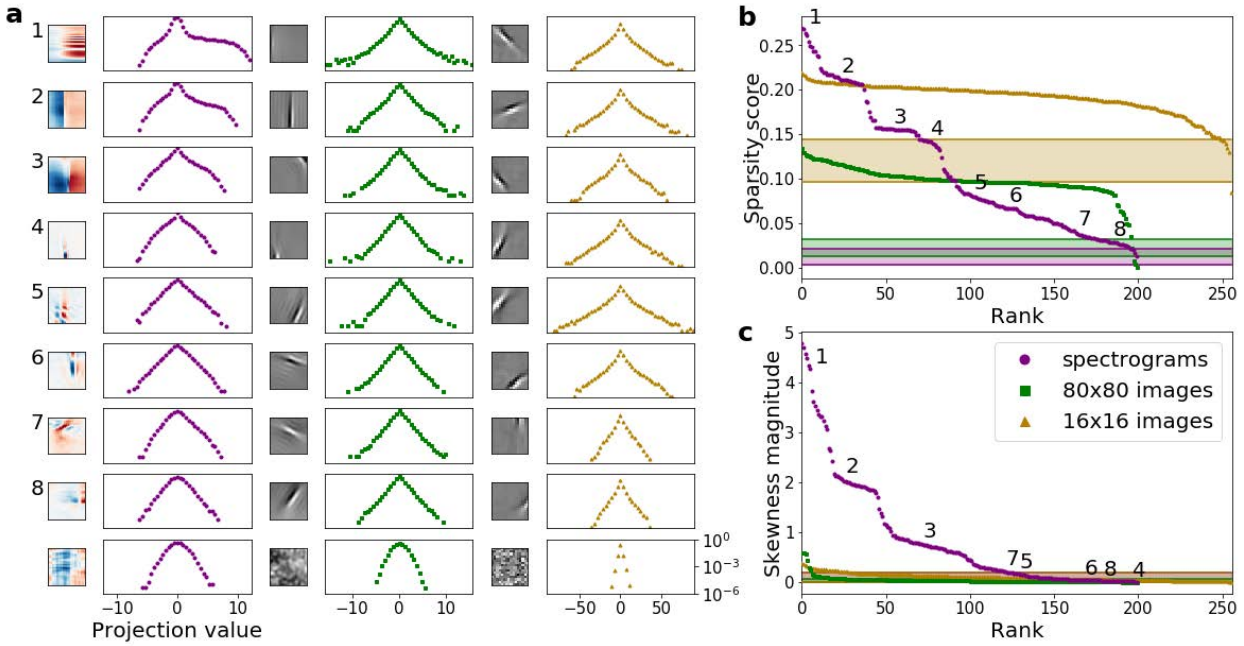

Figure 4.13: **ICA learns a similar representation to complete sparse coding.** This figure is identical to Fig. 4.7 except that the dictionaries have been learned with ICA.

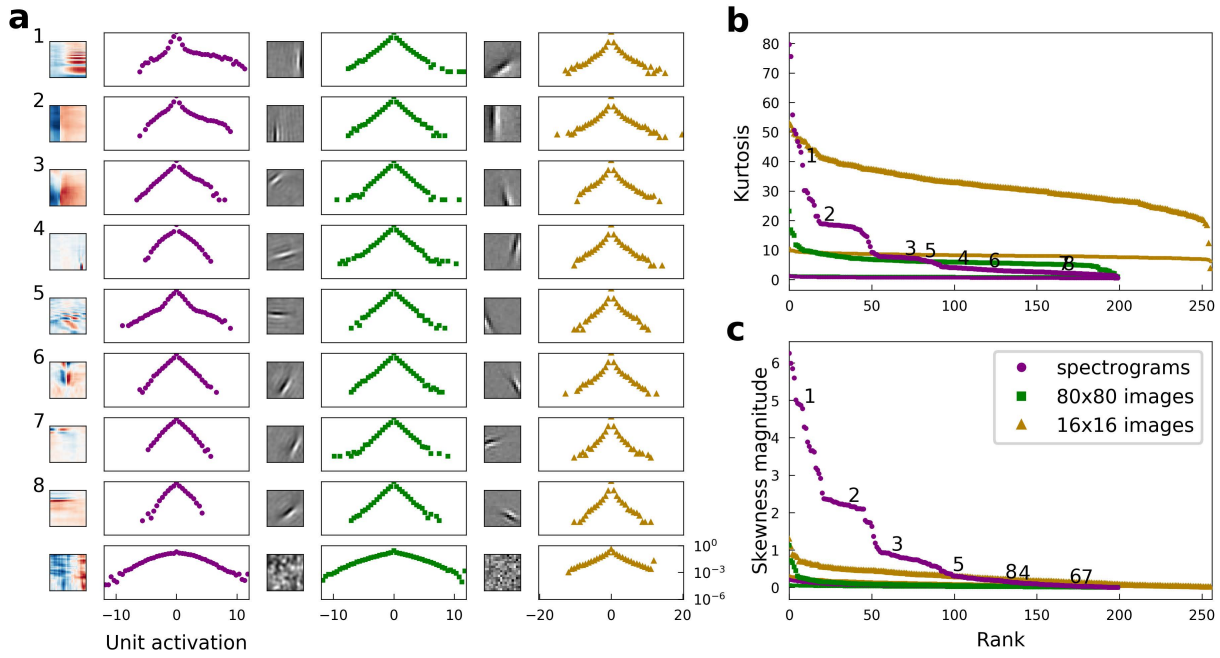

Figure 4.14: **Using kurtosis to measure sparseness gives qualitatively similar results** This figure is identical to Fig. 4.2 except that in panel **b** the sparseness score is replaced by the normalized fourth moment, *i.e.*, kurtosis.

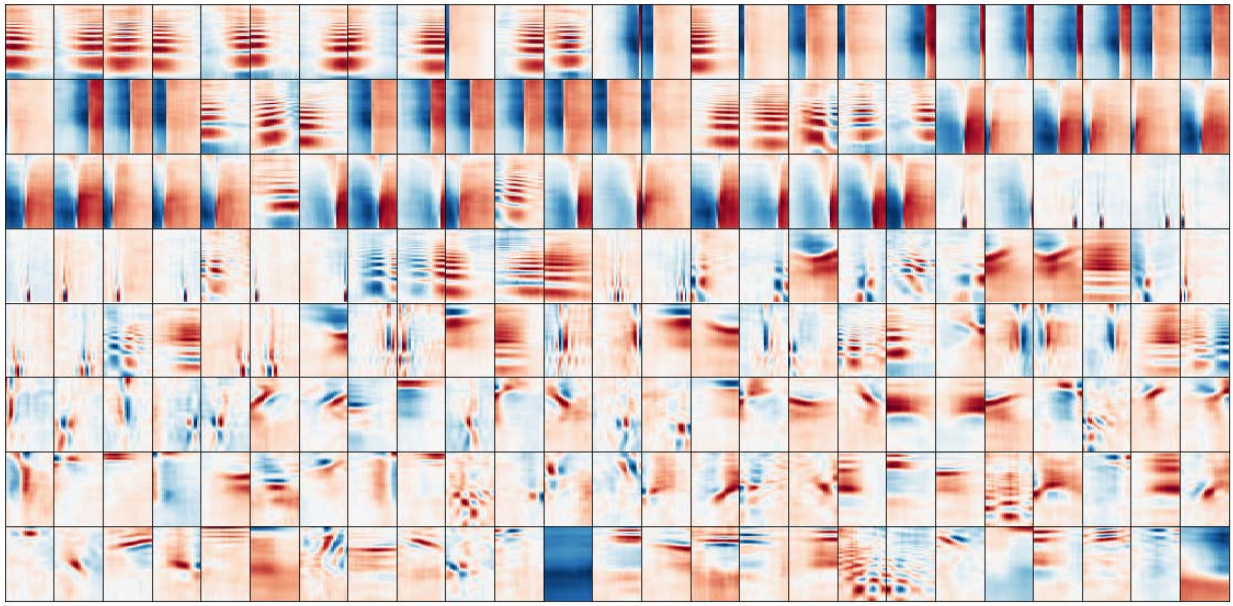

Figure 4.15: **Complete sparse coding dictionary for the speech spectrogram dataset.** There are 200 elements, spanning the 200-dimensional space of retained principle components. The elements are ordered from most sparse (upper left) to least sparse (lower right), evaluated with linear projections. Color and scaling conventions are the same as in the main figures; each element is scaled independently before plotting.

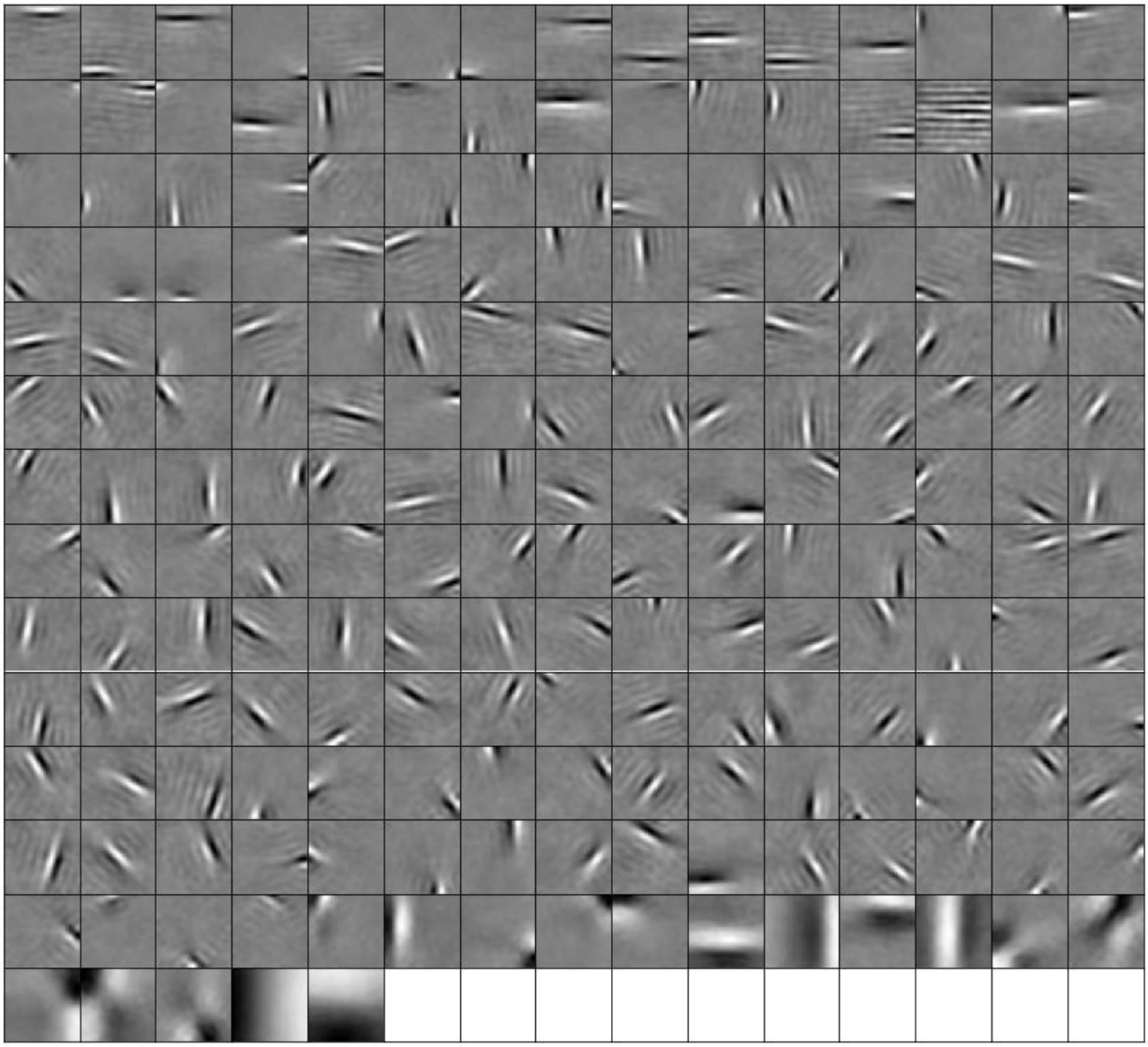

Figure 4.16: **Complete sparse coding dictionary for the 80x80 pixel PCA-whitened image dataset.** Sorted as in Fig. 4.15.

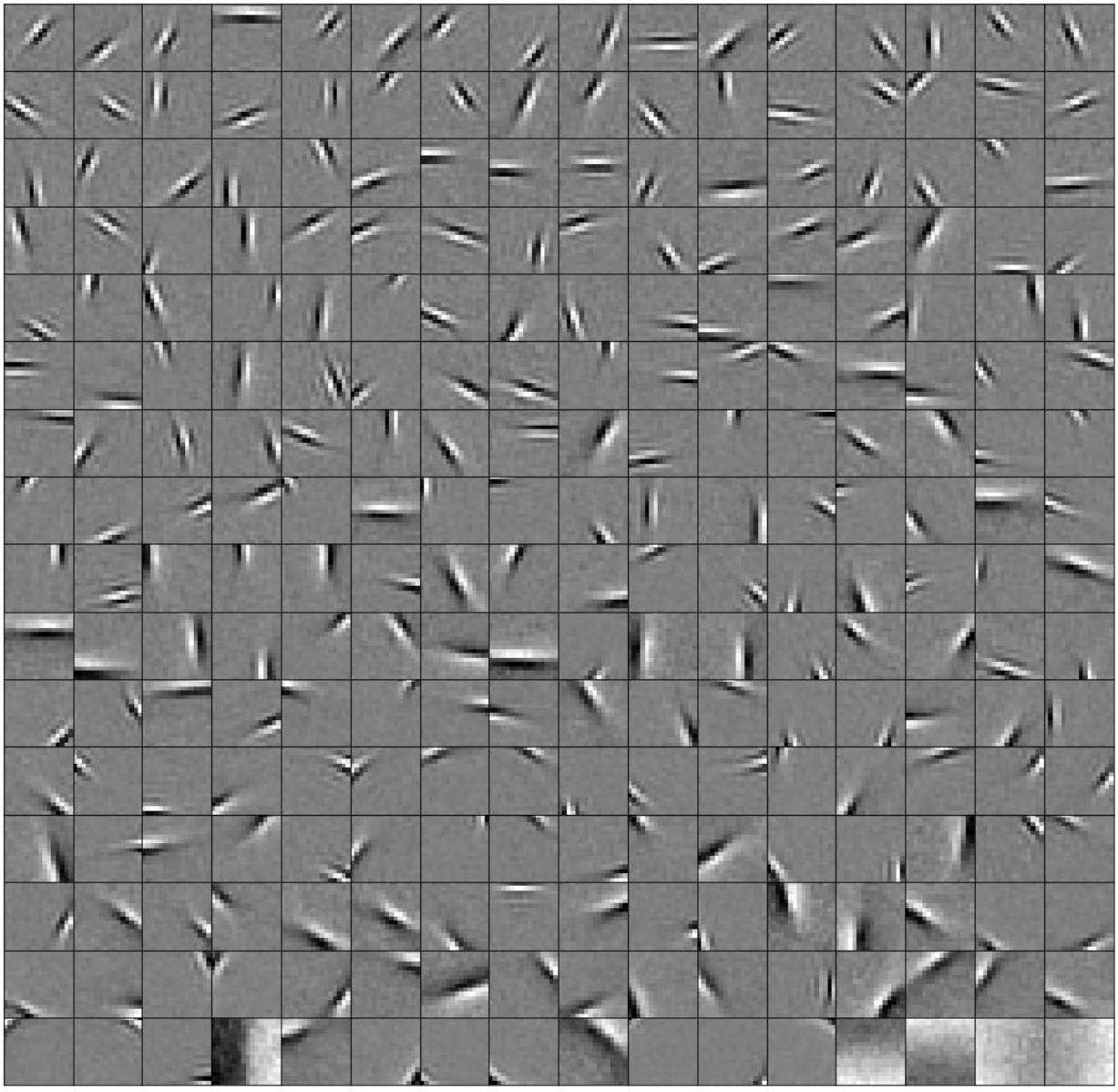

Figure 4.17: **Complete sparse coding dictionary for the 16x16 pixel filter-whitened image dataset.** Sorted as in Fig. 4.15.

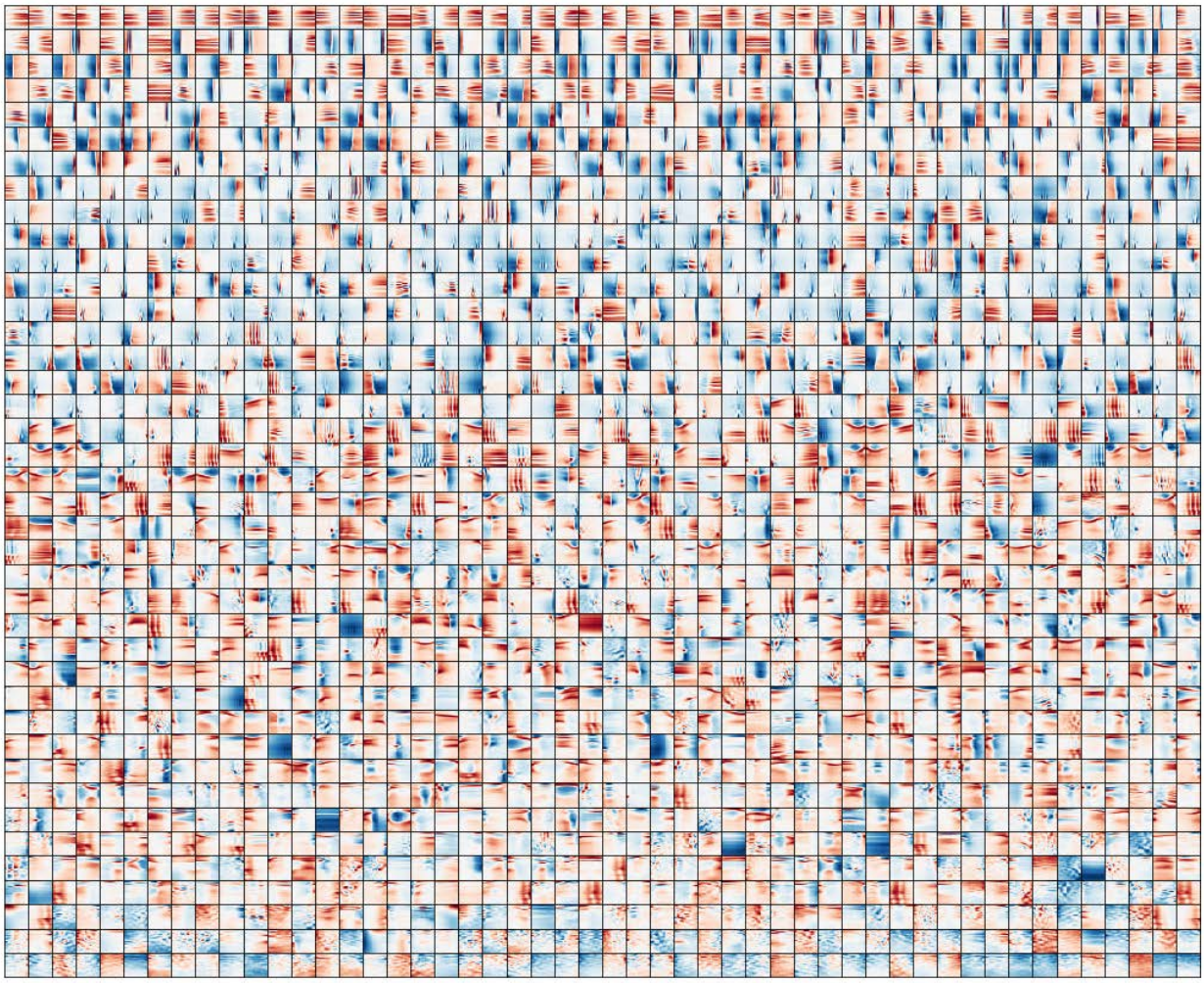

Figure 4.18: **10-times overcomplete sparse coding dictionary for the speech spectrogram dataset.** There are 2000 dictionary elements, 10 times the number of retained principle components. These elements are not all mutually orthogonal. Sorted by sparseness of dot products with the data, as in Fig. 4.15. Note that this is not the same as sorting by LCA activities.

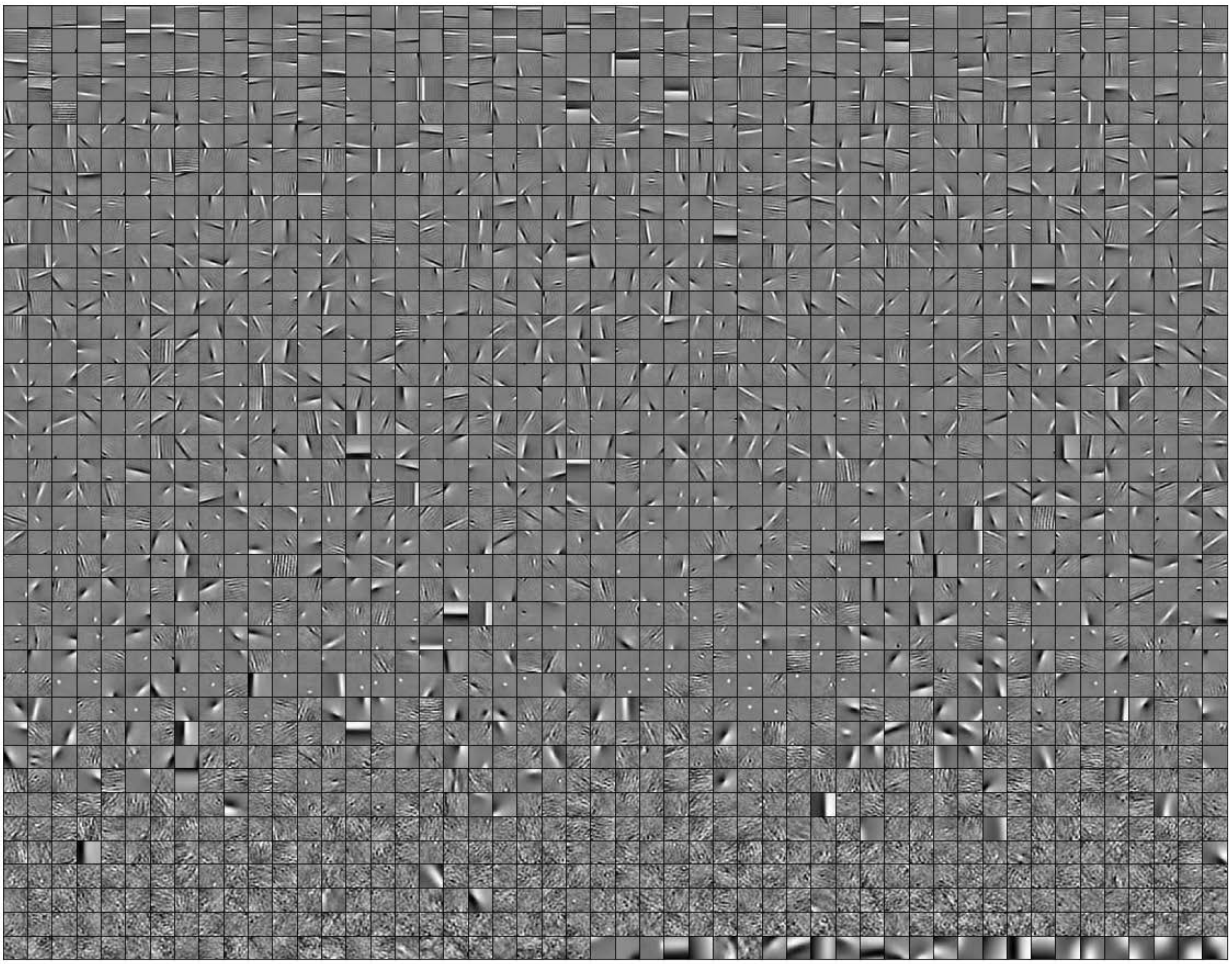

Figure 4.19: **10-times overcomplete sparse coding dictionary for the 80x80 pixel PCA-whitened image dataset.** There are 2000 dictionary elements, 10 times the number of retained principle components. These elements are not all mutually orthogonal. Sorted as in Fig. 4.18.

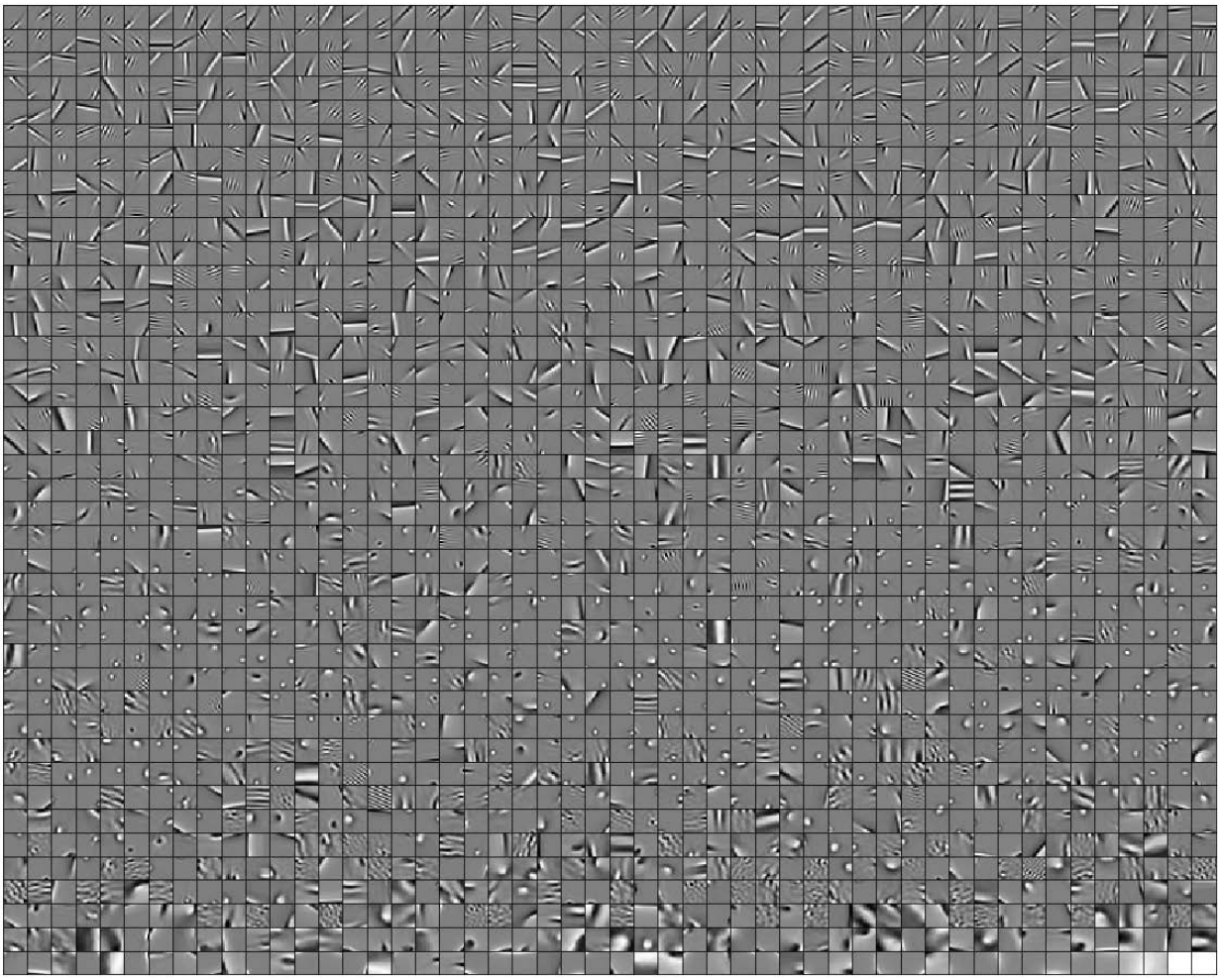

Figure 4.20: **Approximately ten-times overcomplete sparse coding dictionary for the 16x16 pixel filter-whitened image dataset.** There are 2048 dictionary elements, 8 times the number of pixels (256). While the dictionary is therefore nominally 8-times overcomplete, lowpass filtering cut the number of significant dimensions to about 200, making this dictionary approximately 10-times overcomplete. The elements are not all mutually orthogonal. Sorted as in Fig. 4.18.

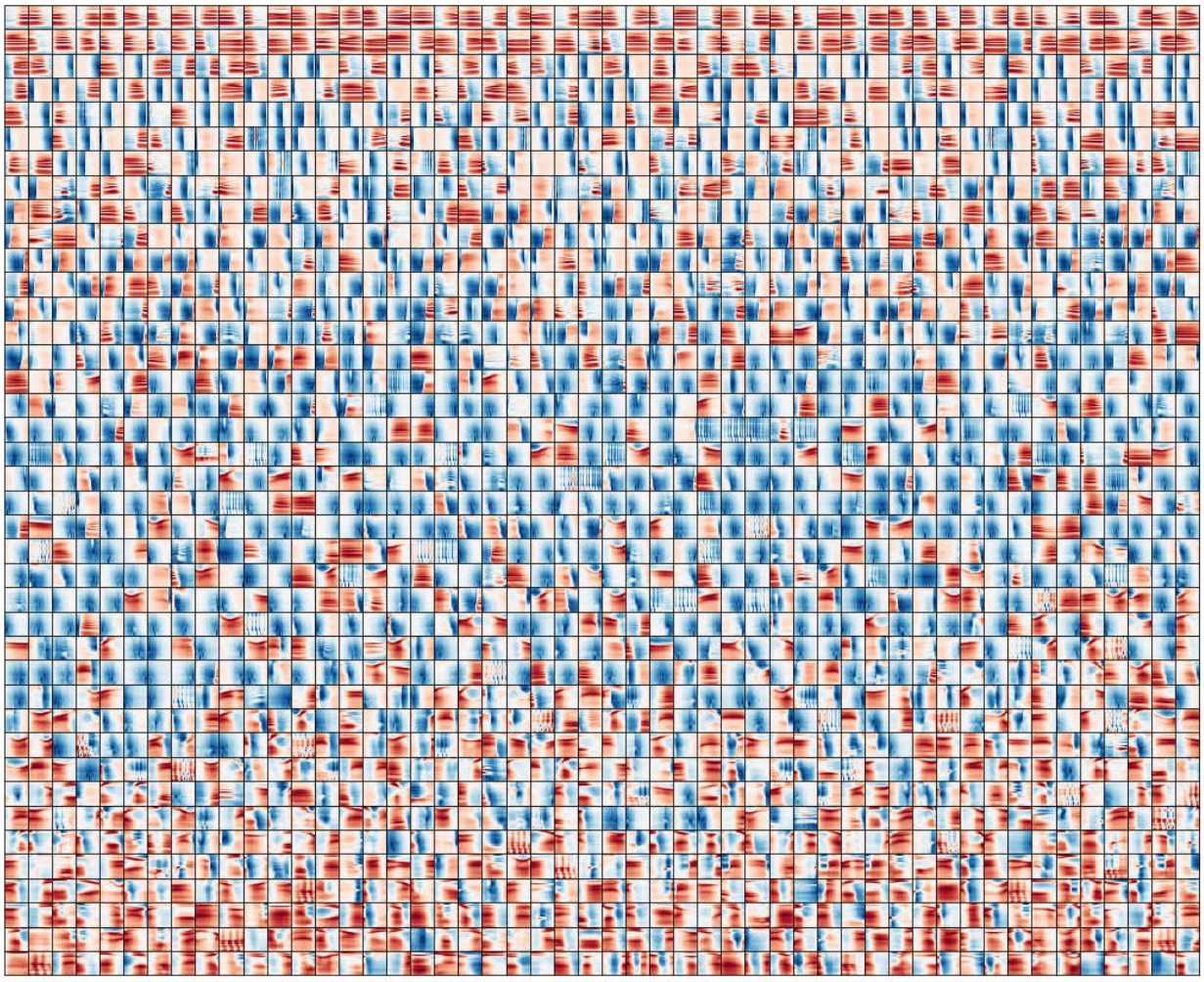

Figure 4.21: **Ten-times overcomplete SAILnet dictionary for the speech spectrogram dataset.** SAILnet has additional learned parameters, which act to enforce sparseness and decorrelation. These parameters are not shown but have an effect on the learning process for the dictionary. Sorted as in Fig. 4.18.

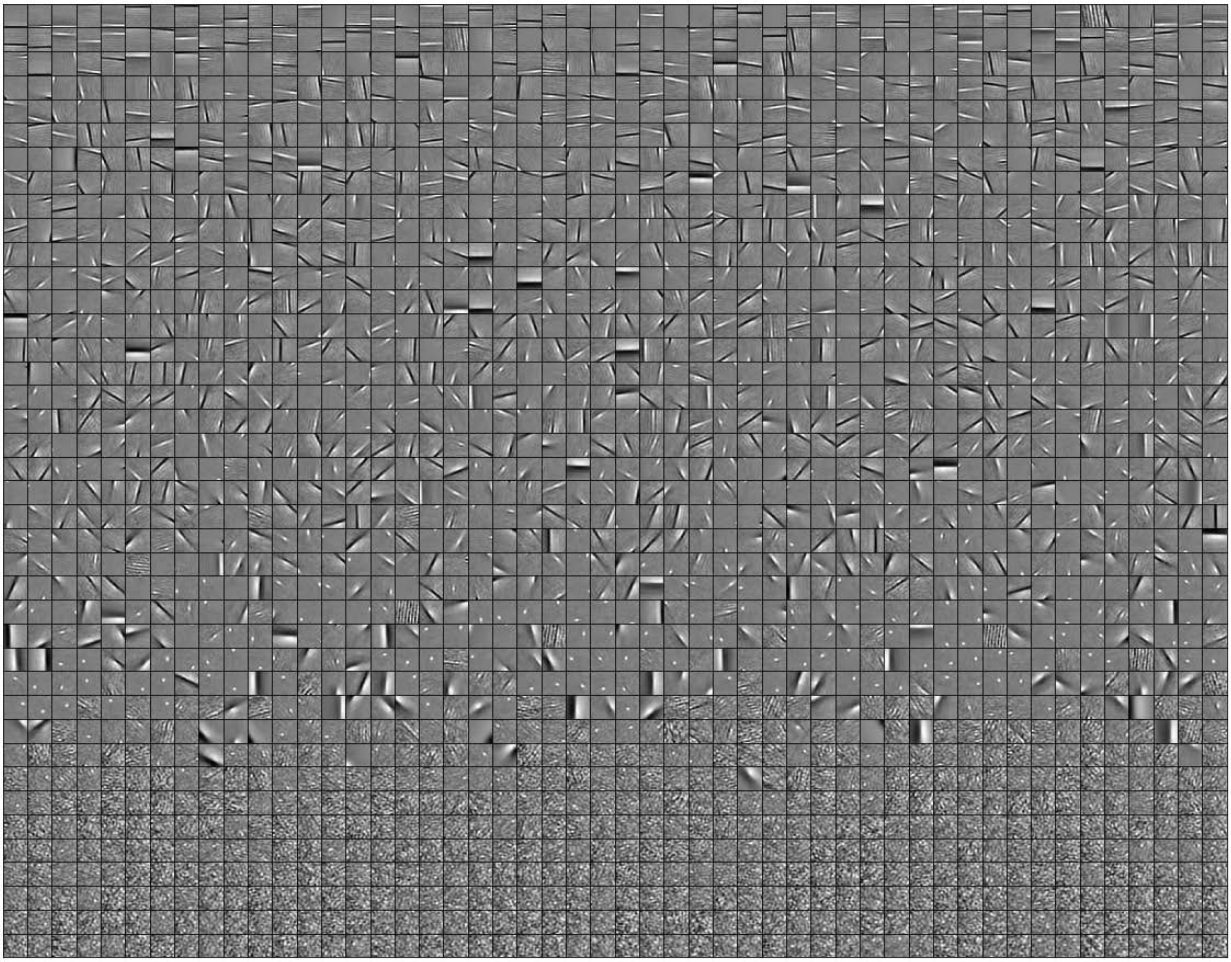

Figure 4.22: **Ten-times overcomplete SAILnet dictionary for the 80x80 pixel PCA-whitened image dataset.** Sorted as in Fig. 4.18.

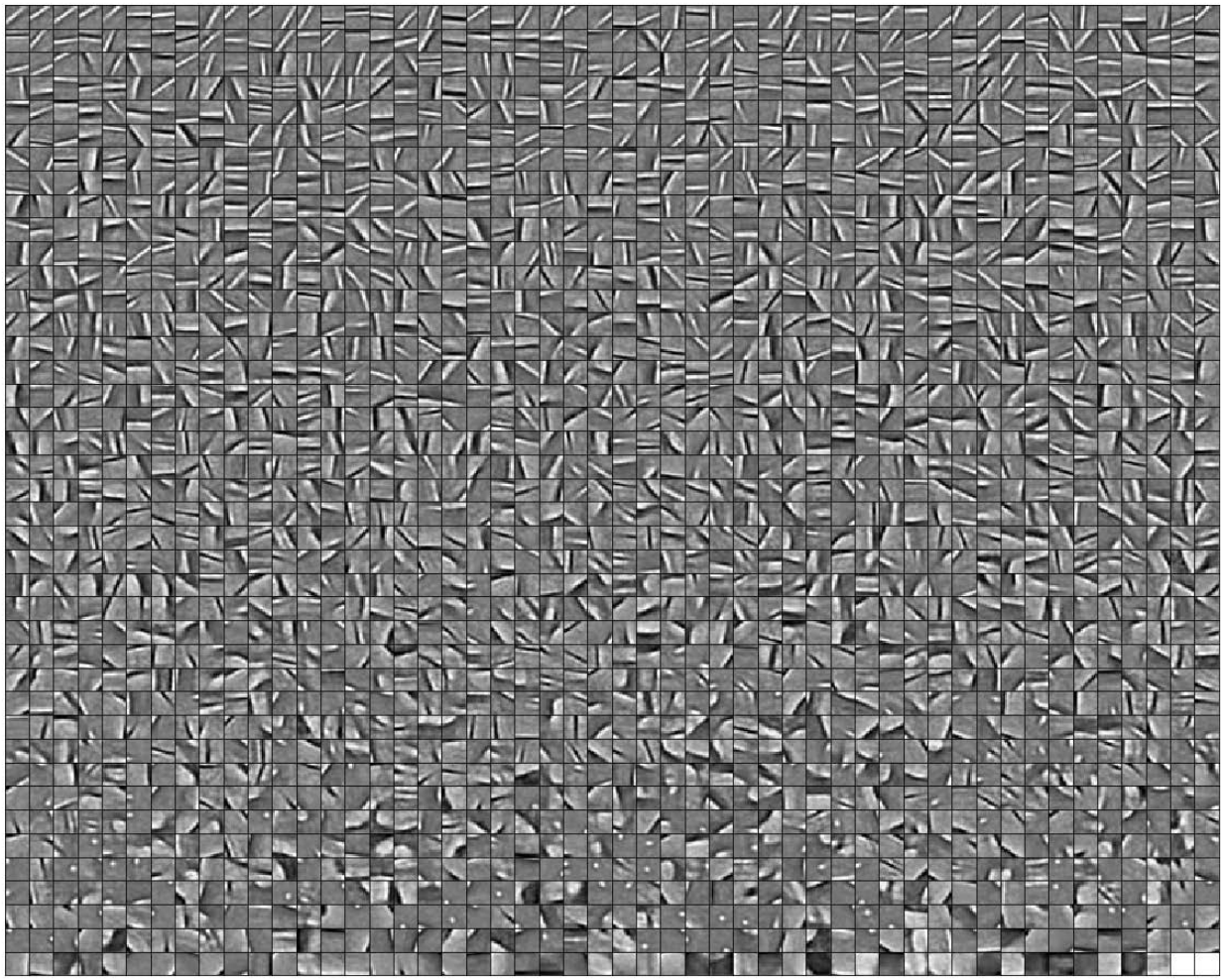

Figure 4.23: **Approximately ten-times overcomplete SAILnet dictionary for the 16x16 pixel filter-whitened image dataset.** Sorted as in Fig. 4.18.

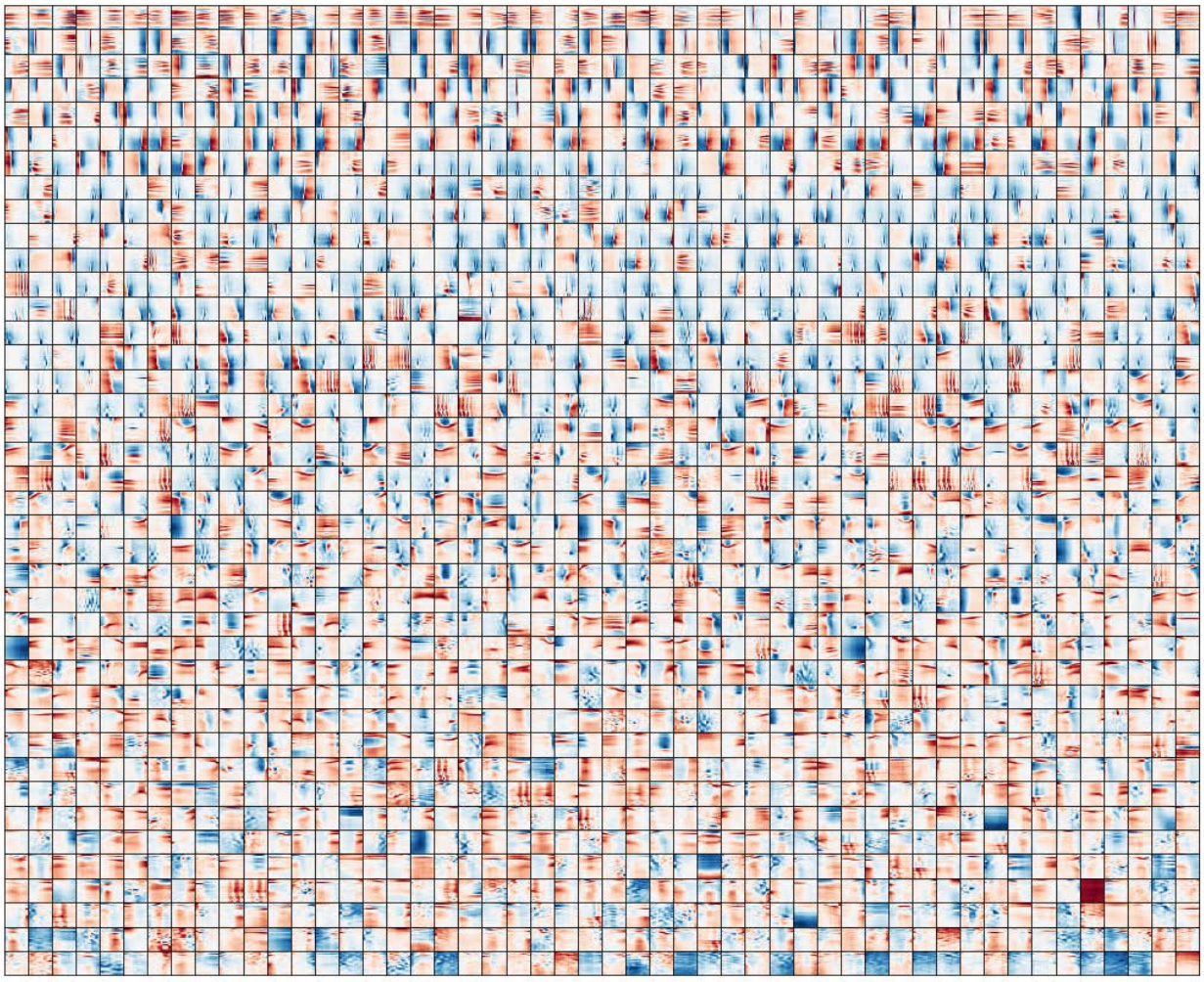

Figure 4.24: **Ten-times overcomplete sparse coding dictionary for the speech spectrogram dataset, learned with non-negative activities.** This dictionary was learned by the same procedure as the one in Fig. 4.18, but with the LCA thresholding function rectified so that negative activities instead set to zero. Sorted as in Fig. 4.18.

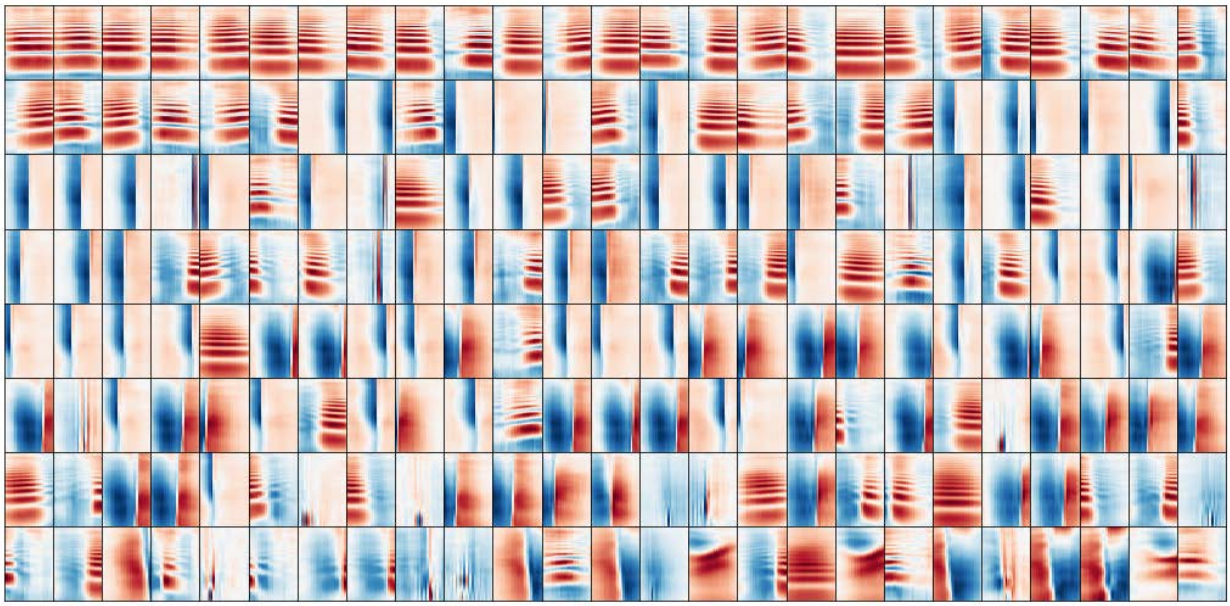

Figure 4.25: **Complete dictionary for the speech spectrogram dataset, learned with modified SAILnet.** This dictionary was learned with a version of SAILnet modified to allow negative spikes (which still count positively toward the average firing rate p). The dictionary has a different distribution of types with this modification, including the appearance of several elements with harmonic structure that changes sign abruptly in time. This type is not present in conventional SAILnet dictionaries, even at high degrees of overcompleteness. Sorted as in Fig. 4.18.

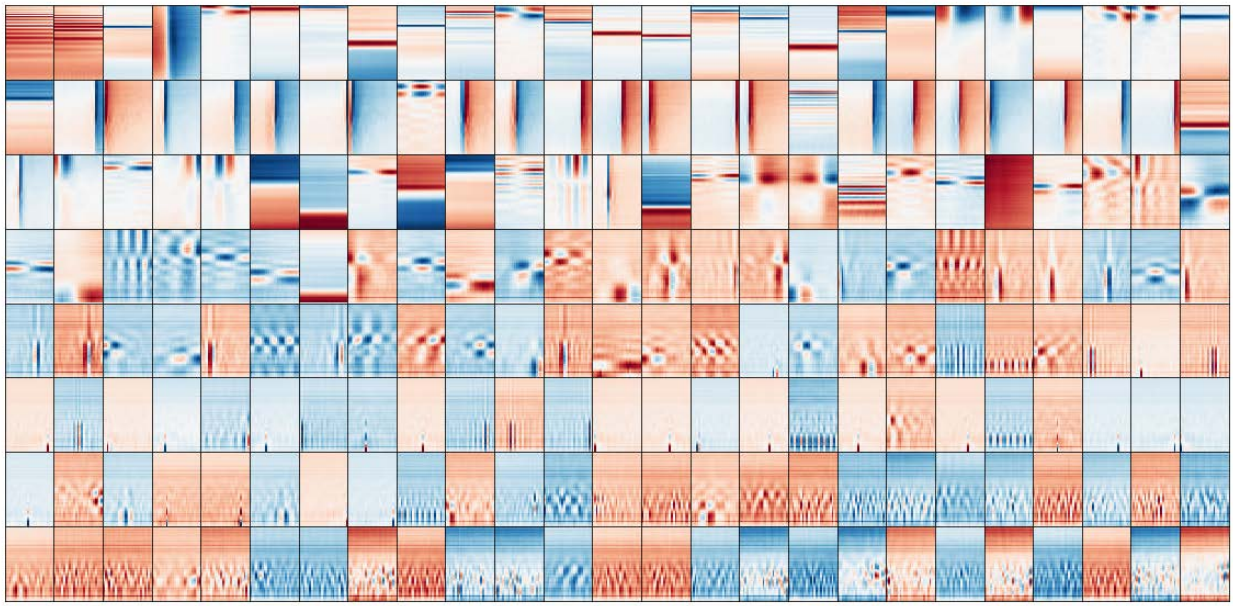

Figure 4.26: **ICA dictionary fit to "natural sounds" dataset** This dictionary was fit to a combination of natural sounds including ambient sounds recorded in various locations, animal vocalizations. It includes a few sparse elements reminiscent of harmonic stacks, single frequency detectors, and checkerboard-like features also seen in the speech-trained dictionaries. The majority of the elements, however, are no more sparse than random directions and have no easily discernible structure. Overcomplete sparse coding dictionaries trained on the same data show similar structure. Sorted as in Fig. 4.15.

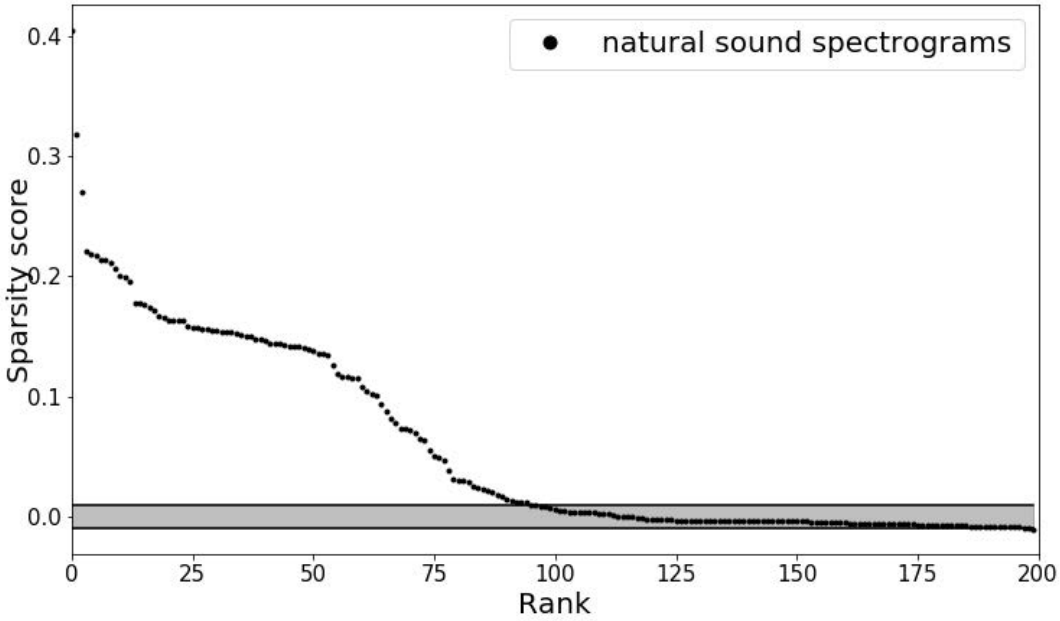

Figure 4.27: **Sparseness score rank plot for natural sounds dictionary** Sparseness scores for the dictionary elements in Fig. 4.26. Many of the sparseness scores are within the same range as random directions.

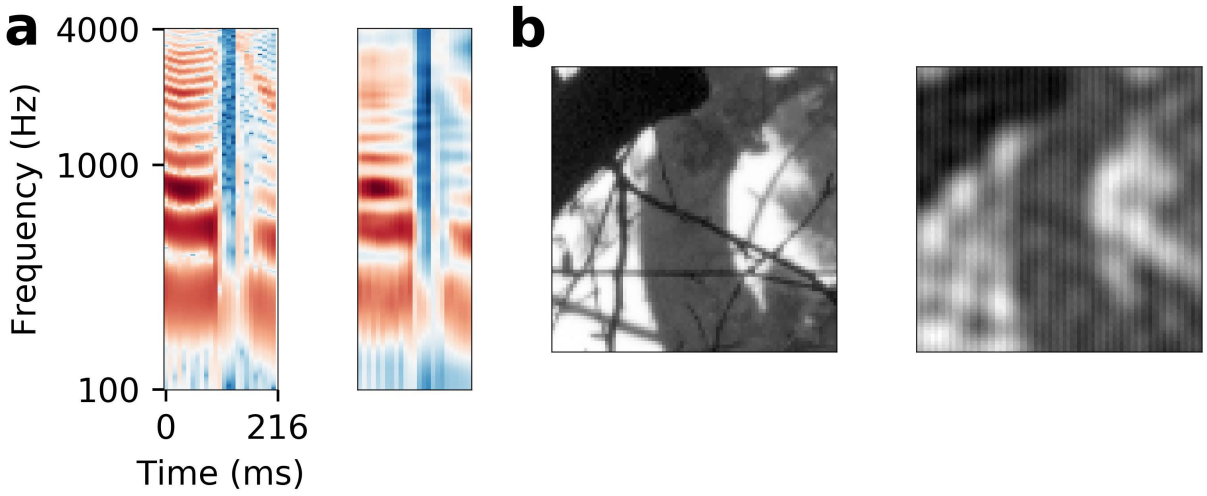

Figure 4.28: **Data reconstruction from 200 principal components.** While reconstructions from the dimensionally-reduced data are clearly distinguishable from the original data, they retain much of the original structure, particularly at coarse scales. The speech in the audio reconstructions is distorted but comprehensible.

Chapter 5

Topographic sparse coding of speech: towards a model of organization in the primary auditory pathway

This short chapter describes the application of topographic sparse coding to sound as a possible model of organization of cells in the primary auditory pathway by coding properties. While the focus here is not on a comparison to the visual modality, this work also attempts to apply a model that found success in the visual system to the auditory domain.

This line of work can be seen in the light of the main theme of this dissertation as attempting to glean insight from applying to an abstract statistical learning model the biological constraint of being implemented on a physical, essentially two-dimensional sheet of neurons in cortex.

5.1 Abstract

Sparse coding of natural stimuli has been shown to predict receptive field properties of neurons in early sensory systems, especially primary visual cortex (V1) [76]. Models such as topographic independent components analysis (ICA, closely related to sparse coding) that pool the activity of nearby units also show the topographic structure that has been observed in V1 [44, 43]. The relevance of sparse coding to computation in the auditory system has been less deeply explored, but some studies have shown qualitative agreement between sparse codes for spectrograms of speech and spectro-temporal receptive fields (STRFs) in Inferior Colliculus (IC), medial geniculate body (MGB), and primary auditory cortex (A1) [18]. Here we apply a topographic sparse coding model to spectrograms of speech as a step toward understanding how units may be organized spatially by properties of their receptive fields if an area such as IC optimizes for sparseness. Our model units learn similar STRFs to an ordinary sparse coding model, and nearby units tend to learn STRFs with related features due to their activities being correlated. We observe pooling of STRF types and

smooth variation of some aspects of the STRFs in our model. Studying the organization of such a model when trained on natural stimuli may help us understand the tonotopy of early auditory areas as well as other potential organizing factors such as modulation frequency that are not yet well understood.

5.2 Significance and related work

Sensory regions of cortex display a range of topographic organization, including the retinotopic and orientation maps found in V1 [12]. The organization and function of neurons in auditory processing areas is not yet well understood. Many studies have found tonotopic organization [69], and there is also evidence for multiple, overlapping maps of other features including sound amplitude and tuning sharpness in cortex [47, 80]. The degree of order of these maps and their functional significance remains unclear, and theoretical work may help explain and predict the topography of auditory areas.

Terashima and Okada applied a topographic sparse coding model to try to understand the order (or disorder) in A1 by comparing to results of the same model trained on natural images [100]. Their model, however, learns only a small variety of STRF shapes compared to our methods (see Chapter [TODO: reference]), not including some shapes that have been found in experimental data. We believe this is related to certain preprocessing steps including the method of generating spectrograms and coarse-graining. Carlson, Ming, and DeWeese [18] showed that after the preprocessing we use, a sparse coding model learns a wider variety of STRF shapes that qualitatively resemble STRFs that have been reported in cat IC, MGB, and A1, suggesting that the principle of sparse coding may be important for understanding a population of neurons in one or more of these regions. In this work, we use the preprocessing of Carlson et al. in order to train a topographic sparse coding model that learns a variety of shapes and orders them topographically.

5.3 Methods

We trained our topographic sparse coding model on segments consisting of 25 time points (216 ms total) of log-power spectrograms of speech from the TIMIT corpus. Each spectrogram samples 256 frequencies logarithmically spaced from 100 Hz to 40 kHz. The spectrograms were generated using a short-time Fourier transform with overlapping 16 ms Hamming windows. These spectrograms have enough resolution and range to capture much of the spectrotemporal structure of speech. To reduce dimensionality and speed up training, we kept only the first 200 principal components of the data and equalized the variance along these 200 dimensions.

Our model is based on Sparsenet (see Chapter 2). We add an additional term in the objective function that can be thought of as sparsifying the activities of a second layer of units that pool the squares of the activities of the first layer (i.e., the coefficients). The

objective function of our model is

$$E(X, a_m; \Phi_m) = \frac{1}{2}||X - \sum_m a_m \Phi_m||^2 + \lambda_1 \sum_m |a_m| + \lambda_2 \sum_{m,k} g_{mk} a_m^2$$

where X is a vector representing a spectrogram segment, a_m is the activity of the mth unit in response to X, Φ_m is dictionary element associated with unit m, and λ_1 and λ_2 are parameters weighting the relative importance of fidelity, sparsity, and group sparsity. The matrix g_{mk} is fixed and specifies a topographic structure. For the results in Figure 5.1 we used a binary matrix specifying overlapping circular blocks of related units.

Note that since each a_m can be positive or negative, the sign of the corresponding dictionary element Φ_m is not directly meaningful.

5.4 Results

Figure 5.1 shows a topographic sparse coding dictionary with size 30x30 units. This sample shows some of the typical variations seen among nearby units, including time-shifts of a particular shape and small deformations. These units are on the boundary between a region dominated by harmonic stacks and a region dominated by sharp broadband onsets and terminations of sound. These results show that while our model only explicitly organizes units by their activities, nearby units learn related STRFs. Further study with different topographical structure and/or a model that learns the topographic structure from the data may help us understand how the organization of units with related activity corresponds to organization by spectrotemporal modulation properties.

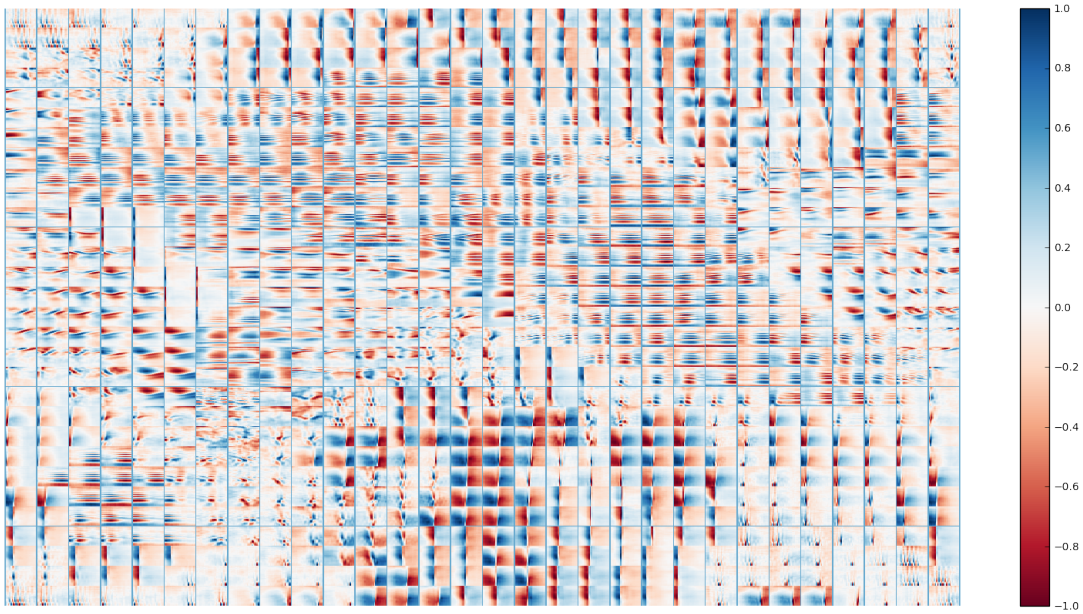

Figure 5.1: Each of the 30x30 rectangles represents a single unit's feedforward weights. Sprectrogram plotting conventions are the same as the figures in Chapter [TODO: reference].

Chapter 6

Spatial whitening in the retina may be necessary for V1 to learn a sparse representation of natural scenes

We have so far focused on sparse coding and related models of low-level sensory processing, but there have of course been other important advances in modeling and understanding the computations of neural sensory systems. Retinal ganglion cells are thought to counter the spatial correlation structure in natural images, whether by the center-surround linear receptive fields or by more complicated nonlinear computation. In this chapter we consider the interaction between this understanding of retinal processing and the theory of sparse coding in primary visual cortex. Specifically, we find that a biologically realistic model of learning for sparse coding of natural scenes, like SAILnet, appears to require the sort of spatial decorrelation thought to be accomplished in retina.

This chapter is based on a paper I coauthored with Jesse Livezey and Professor DeWeese, which is not yet published.

6.1 Abstract

Retinal ganglion cell outputs are less correlated across space than are natural scenes. However, sparse coding, a successful computational model of primary visual cortex, can be achieved by most sparse coding algorithms without a preceding decorrelation stage. We propose that sparse coding with biologically plausible local learning rules does require decorrelated inputs, providing a possible explanation for why whitening may be necessary early in the visual system.

6.2 Introduction

Following proposals that the brain seeks to reduce redundancy in signals from the natural environment[4, 7], such as natural visual scenes [30], Atick and Redlich proposed that the center-surround receptive fields of retinal ganglion cells serve to decorrelate natural visual input to obtain a representation with less redundancy among the outputs of individual neurons, while also suppressing noise [2]. Atick and Redlich emphasized advantages of a code that uses statistically independent elements, such as simple computation of joint probabilities. The removal of pairwise linear correlations, or "whitening", is then seen as a first step towards such a representation. In fact, there is strong experimental evidence for decorrelation at the earliest stages of the visual pathway [23], including nonlinear retinal processing to remove spatial correlations [79].

We propose that it may in fact be *necessary* to whiten visual input before circuitry in primary visual cortex (V1) can achieve a sparse representation of natural scenes if plasticity mechanisms at each synapse can only access local information.

A popular normative theory for V1 simple cells postulates that these neurons are optimized for forming sparse representations of natural visual stimuli [31, 76]. Olshausen and Field demonstrated that a sparse coding model trained on photographs can learn visual features that resemble receptive fields measured in primate V1 simple cells [76]. Related algorithms obtain similar results [10] or even closer agreement with experiment [82, 110]. The "Sparse and Independent Local Network" (SAILnet) [110], a sparse coding network with more biologically realistic spiking neurons and local learning rules, obtains similarly strong results when trained on whitened natural image patches. A modified version of SAILnet with separate populations of excitatory and inhibitory neurons also produces reasonable receptive field shapes [54].

Whereas sparse coding studies often use whitening as a preprocessing step, pre-whitening is not necessary for conventional methods, such as those of [76, 10, 82]. In fact, some authors have pointed to the existence of nonlinearities in the retina and thalamus as an argument against sparseness as a normative theory for coding in V1 [64]. SAILnet, however, does not learn V1-like receptive fields if the image data is not pre-whitened (Fig. 6.1A-D). SAILnet differs from conventional sparse coding in several ways, but the essential feature that makes pre-whitening necessary is the network's synaptically local learning rules.

We call a learning rule "synaptically local" if the strength of a synapse is updated using information known to be available at that synapse: the number and timing of recent spikes in the pre-synaptic and post-synaptic neurons and the present strength of the synapse.

Importantly, sparse coding models with *non*-local plasticity can perform well on unwhitened data such as raw natural images. We demonstrate this with an algorithm that finds sparse codes using the Locally Competitive Algorithm (LCA) [86] to perform inference and stochastic gradient descent (SGD) on mean squared error, a non-local learning rule, to perform learning. While this network learns somewhat more accurately and quickly on whitened data than on unwhitened data, pre-whitening is not necessary to generate codes of high fidelity and sparseness or to learn visual features resembling V1 receptive fields.

6.3 Results and Discussion

We demonstrated the contrast between learning with SAILnet versus conventional sparse coding on whitened and unwhitened data in two complementary ways, both shown in Fig. 6.1. First, we trained models on raw natural image patches and also on the same patches after whitening. We trained SAILnet with many different settings of its hyperparameters; results corresponding to the optimal hyperparameters are shown in the figure. No values of these hyperparameters led to SAILnet learning qualitatively different receptive fields in the unwhitened case.

Second, we generated synthetic data consisting of sparse (Laplace-distributed) linear combinations of a fixed, randomly-generated set of vectors; each of these vectors was an instantiation of frozen white noise. Both conventional sparse coding and SAILnet were able to recover these vectors from the data. We then applied to this synthetic data the singular value spectrum of our natural image patch dataset — in effect, "unwhitening" the dataset. We trained models on this type of synthetic data with distributions of singular values interpolated between that of natural images and the original whitened data, searching many values of hyperparameters for SAILnet in each case. We found that conventional sparse coding continued to perform well for all spectra we tested, whereas SAILnet's performance collapsed to chance-level for data with the singular value spectrum of natural images (Fig. 6.1E).

We have been unable to find, either in the literature or by our own efforts, a sparse coding algorithm using synaptically local learning rules of the type discussed above that performs well on unwhitened natural images. We believe that any such algorithm optimizing for sparse coding will tend to be pulled towards directions of high variance just as SAILnet is.

More specifically, a network with local learning rules needs to overcome two related challenges: 1) learning lateral connections to facilitate cooperation in coding without each neuron having direct access to the stimulus features that other neurons represent; and 2) learning feedforward connections to minimize future coding errors without each neuron having direct access to the part of the stimulus represented by other neurons. SAILnet's inhibitory connection learning rule solves the first problem by leveraging the fact that neuron activity correlations are closely tied to similarity of the features the neurons represent, but strong correlations in the stimulus can distort this relationship. Feedforward synapses onto a given neuron lack direct access to other neurons' spikes, so they must learn to reduce coding error using only the stimulus and the neuron's spikes. If some directions in stimulus space have higher variance than others, these synapses can best reduce error alone within the spike budget imposed by the sparseness constraint by aligning with the high-variance directions, overwhelming the advantage of a new stimulus direction of sufficiently lower variance.

Following work on biologically plausible coding with excitatory-inhibitory balance [15][25] we also developed sparse coding learning rules for which each synapse also has access to a neuron-wide "membrane potential" variable. We find that these rules can learn a sparse representations from unwhitened data (see Supplementary Information). However, dendrites in real neurons are typically electrically compartmentalized [36], making it unlikely that there is a well-defined variable (*i.e.*, membrane potential) that is uniform across all synapses

throughout the dendritic tree of any given neuron.

It has also been shown that a sparse coding network can learn successfully from unwhitened inputs by comparing sparse codes for the same stimulus generated by two different configurations of the network [60]. Whereas the learning rules in this method use information local to each synapse, we do not think it is biologically plausible that each synapse could acquire, store, and compare the spike rates for two completely different network configurations at each moment in time.

While we believe it is unlikely, more complicated models that make use of precise relative spike timing could in principle obviate the need for pre-whitening while using local learning rules to learn a sparse code. It is not clear how such a scheme could overcome the challenges described above.

Our proposal compliments the prevailing notion that retinal whitening provides a more efficient representation for transmission through the limited capacity of the optic nerve before the creation of an overcomplete, sparse representation in primary visual cortex, and it suggests a second compelling reason for decorrelation to be performed by a distinct population of neurons at the earliest stages of the visual system.

6.4 Methods

Data and whitening

We used a set of 3×10^5 square patches of 256 pixels each drawn from the van Hateren image dataset[35]. The full images, but not the patches, were mean-centered and normalized by the standard deviation across pixels. We then subtracted the mean patch and divided by the standard deviation across all pixels and patches.

Whitening or sphering refers to a linear transformation that results in a covariance matrix proportional to the identity. In this work we use PCA whitening. First we multiply the data by the matrix of eigenvectors of the covariance matrix, and then we divide each component by its standard deviation. Thus,

$$X_{\text{raw}}^T X_{\text{raw}} = USU^{-1} \longrightarrow X = X_{\text{raw}} US^{-1}, \tag{6.1}$$

where S is diagonal and U is unitary. A 2D visualization is shown in Fig. 6.2.

The distribution of standard deviations of the principal components, also known as the singular values of the data, characterizes the asphericity of the data. In the case of natural images (and other data with approximate translational invariance), the singular values correspond closely with the Fourier amplitude spectrum.

Sparse coding

We first define a probabilistic model

$$p(x; \Phi) = \int p(x|a) p_a(a) da, \tag{6.2}$$

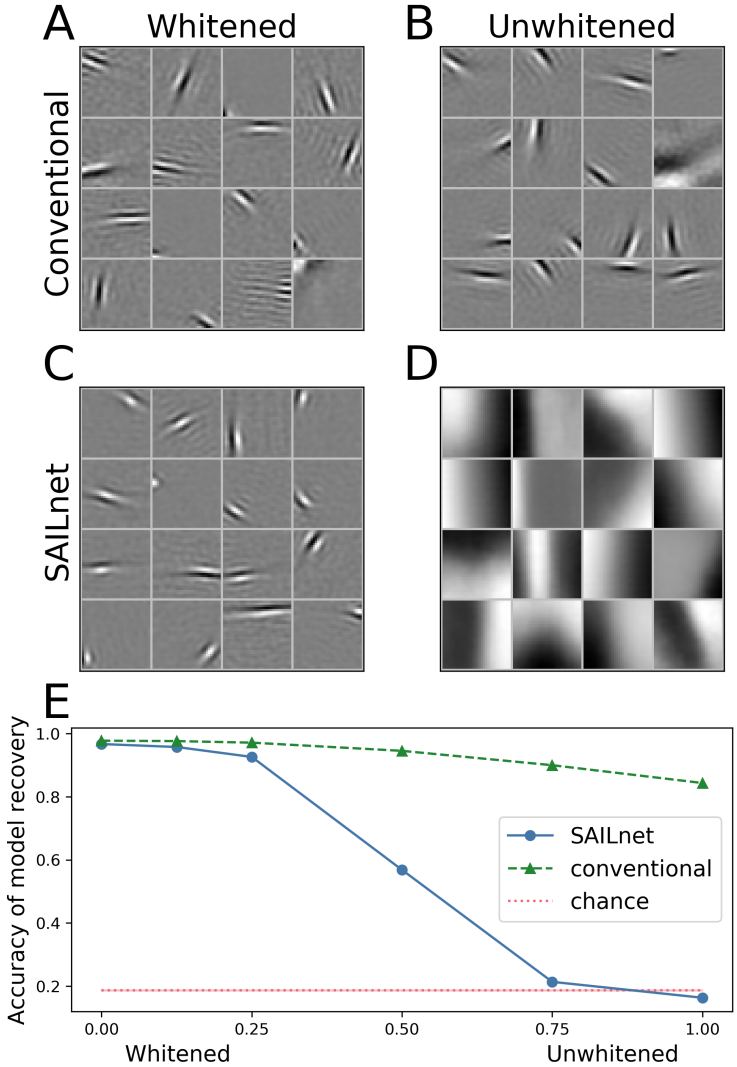

Figure 6.1: **SAILnet learning requires pre-whitening while conventional non-local learning does not.** (A) Receptive fields of conventional sparse coding model neurons resemble V1 simple cell receptive fields after the network has been trained on a set of whitened grayscale natural image patches. The network had 256 units; a random sample of 16 units are shown. (B) Conventional sparse coding learns qualitatively similar features on raw, unwhitened natural image patches. (C) SAILnet receptive fields after training on a set of whitened natural image patches resemble conventional results and V1 simple cells. (D) SAILnet receptive fields after training on unwhitened image patches are markedly different. (E) Results for SAILnet and conventional sparse coding on a sparse model recovery task where the data has been de-whitened to some degree, with 1 on the horizontal axis corresponding to the same pairwise correlation structure as natural images. Whereas conventional methods degrade only slightly with de-whitening, SAILnet performs no better than a randomly generated dictionary (dotted red line) for synthetic data as far from white as natural images. All simulations were run five times. Means are plotted; standard deviations are too small to display as error bars.

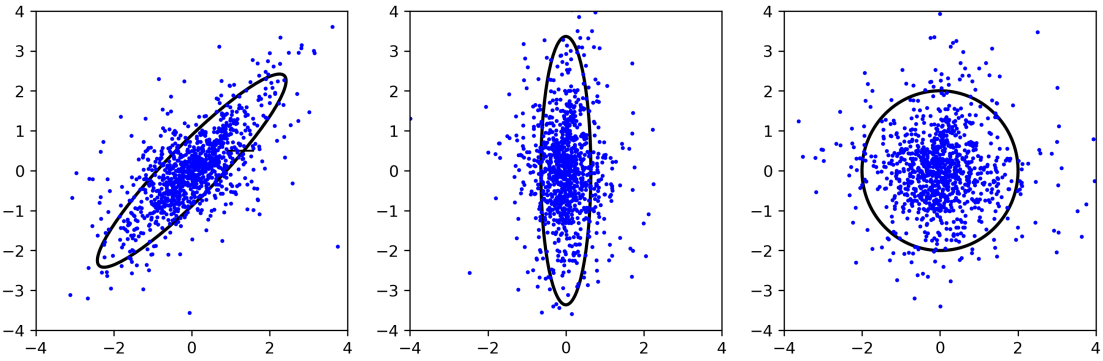

Figure 6.2: **PCA whitening for 2D data.** Data can be whitened by transforming to the principal component basis and then scaling the axes (that is, the principal components). The result and all rotations of the result are whitened.

where x denotes a data vector with components x_i and a denotes a set of latent variables a_m. The conditional distribution $p(x|a)$ is an isotropic Gaussian of fixed variance centered on a linear reconstruction of the data in terms of the dictionary elements Φ_m:

$$p(x|a) \propto \exp\left[-\frac{1}{2\sigma^2}\sum_i (x_i - \sum_m \Phi_{mi} a_m)^2\right], \tag{6.3}$$

and the prior distribution of the coefficients a_m is factorial with each factor given by the same sparse distribution (*e.g.*, a Laplace distribution):

$$p_a(a) \propto \prod_m e^{-\lambda|a_m|}, \tag{6.4}$$

where λ is a parameter that determines the width of the distribution and therefore how strongly the prior favors sparse sets of a_m.

Fitting this model usually involves estimating the gradient of the likelihood (or log-likelihood) with respect to the parameters Φ_{mi}, but this calculation involves an intractable integral over the latent variables a_m. A common approximation is to set the a_m by maximum *a posteriori* (MAP) inference given input data x:

$$a^{\text{MAP}} = \arg\max\{a_m\} p(x|a_m) p_a(\{a_m\}) = \arg\max\{a_m\} e^{-\sum_i (x_i - \sum_m \Phi_{mi} a_m)^2/2\sigma^2} \prod_m e^{-\lambda|a_m|}. \tag{6.5}$$

The parameters σ^2 and λ now only affect the model through the combination $\lambda\sigma^2$, so to simplify the notation we set $\sigma = 1$.

The quantity a_m^{MAP} is typically referred to as the activity of the mth unit, and the dictionary elements $\{\Phi_m\}$ are often compared to receptive fields of neurons. The analogies to

neurons suggested by these terms are not exact, but a unit's dictionary element is approximately the same as the linear receptive field that would be measured for that unit with an activity-triggered average [76].

The dictionary elements Φ_m are conventionally learned by descending the estimate of the gradient provided by differentiating the model log-likelihood with respect to Φ with a fixed at the MAP value:

$$\Delta\Phi_{mi} \propto -\frac{\partial}{\partial\Phi_{mi}}\left[-\frac{1}{2}\sum_j (x_j - \sum_m \Phi_{mj} a_m^{\mathrm{MAP}})^2\right]. \tag{6.6}$$

In this work, this gradient with respect to Φ is averaged over a minibatch of 100 data examples.

The use of MAP inference requires that we constrain the norms of the Φ_m to avoid solutions with small a_m and large, meaningless Φ_m. We therefore divide each Φ_m by its norm after each gradient step.

Locally Competitive Algorithm

We used the L1-sparse locally competitive algorithm (LCA) [86] to perform MAP inference in our "conventional sparse coding" model. LCA uses a dynamical system with auxiliary variables that are thresholded to obtain estimates of a^{MAP}. Typically most of the auxiliary variables are below threshold and the a_m^{MAP} estimates are exactly zero for most m. The threshold is set by the sparseness parameter λ.

The choice of coding algorithm is not crucial to our results, and learning using alternative inference schemes yields qualitatively similar dictionaries.

SAILnet and related models

SAILnet uses leaky integrate-and-fire neurons to form a sparse spike-count code and local rules to updates its synaptic strengths. The structure of the network is illustrated in Fig. 6.3

The LIF unit dynamical equation is

$$\dot{u}_m = -u_m + \sum_i \Phi_{mi} x_i - \sum_n W_{mn} y_n, \tag{6.7}$$

where the spike variable y_n is 0 unless the membrane potential u_m just crossed the threshold θ_m (after each spike, u_m is reset to 0). (Here and elsewhere we use time units such that the resulting time constant is 1.) An example LIF unit's evolution is shown in Fig. 6.4. (Replacing the LIF neurons with continuous-valued units similar to LCA units makes a quantitative but not qualitative difference to learning.) These dynamics give approximately optimal codes a_m, proportional to the spike counts, if the lateral weights are $W^*_{mn} = \sum_j \Phi_{mj}\Phi_{nj}$.

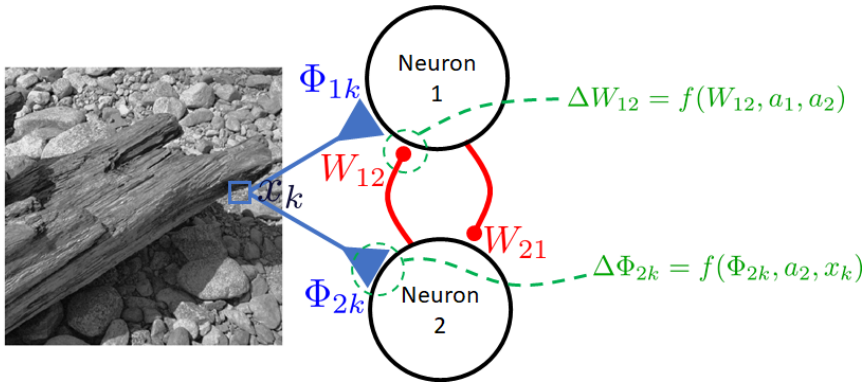

Figure 6.3: **SAILnet and similar models use feedforward and lateral connections updated with local learning rules.**

SAILnet updates its parameters according to

$$\Delta\theta_m \propto a_m - p \tag{6.8}$$

$$\Delta W_{ml} \propto a_m a_l - p^2 \tag{6.9}$$

$$\Delta\Phi_{mk} \propto a_m(X_k - a_m\Phi_{mk}), \tag{6.10}$$

where a_m is the (possibly normalized) spike count for unit m. The first equation imposes a constraint on the firing rate of each neuron. The second attempts to impose the constraint that the neurons' activities be decorrelated. The third drives each neuron to reconstruct the stimulus (alone).

Synthetic data

Our synthetic data was generated by sampling from the sparse coding probabilistic model (6.2) with a known, fixed dictionary. For N sources in D dimensions, we generated the dictionary Φ^* by sampling N directions in D-dimensional space uniformly at random. Then each data vector x^μ was determined by N samples $a_1^\mu, \ldots, a_N^\mu$ from an exponential distribution with scale λ:

$$x_i^\mu = \sum_n \Phi_{ni}^* a_n^\mu. \tag{6.11}$$

z This data generation process is spherically symmetric, but any particular dictionary and dataset will not be perfectly white. We used $D = 256$ and $N = 256$ for the experiments in the main text, for as close correspondence with the natural image experiments as possible.

To de-whiten the synthetic data, we multiplied by the matrix of singular values of our natural image patch dataset, raised to some power ψ:

$$\sum_i x_i^\mu S_{ij}^\psi. \tag{6.12}$$

The parameter ψ is the horizontal axis of Fig. 1E in the main text.

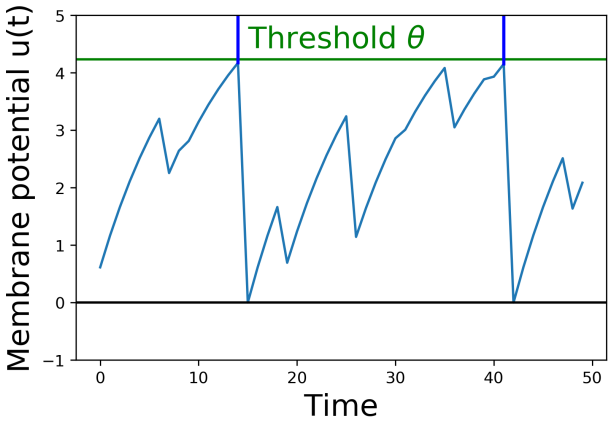

Figure 6.4: **SAILnet uses leaky integrate-and-fire (LIF) model neurons.** Constant feedforward inputs from the data drive the unit's internal "membrane potential" variable u, while inhibitory inputs from other units decrease u whenever the inhibiting units spike. When u hits the unit's threshold θ, the unit emits a spike and resets u to zero at the next timestep. Each unit has its own learned threshold θ.

6.5 Supplementary material

Complete dictionaries of learned models

We believe the sample dictionary elements in the main text represent the learned dictionaries faithfully for the purposes of our argument, but we present here the complete dictionaries. In each case the dictionary elements, which are trained on principal component representations, are projected back to image space for presentation with no other alteration. In particular, the whitening matrix is not used in this transformation.

Models using membrane potential and spike times

There are other notions of locality and "biological plausibility" that do not fit within the framework described in the main text. One possibility is to allow learning based on information that may be available during the "inference" process but that isn't in the final activations a_m. Inspired by the work of [15][25] on "learning to represent signals spike by spike," we developed a SAILnet-like sparse coding model in which the post-synaptic membrane potential is used in addition to spikes to determine synaptic strength updates. This model, which we describe below, learns the expected Gabor-like features on natural image patches regardless of whether the data is pre-whitened, and recovers a ground-truth model when the data has been de-whitened (Fig 6.9).

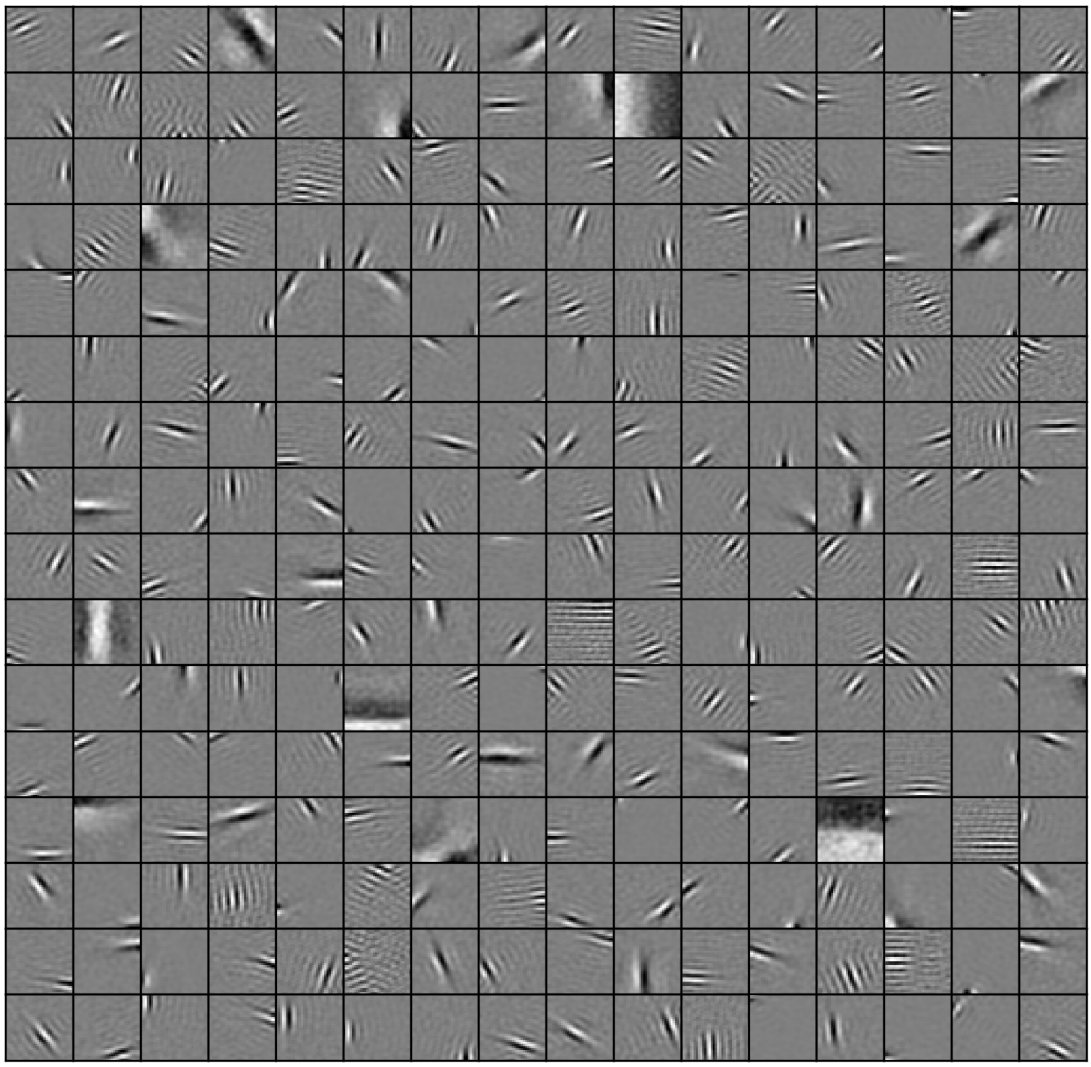

Figure 6.5: **All dictionary elements from a complete sparse coding dictionary learned with stochastic gradient descent and LCA inference on whitened natural image patches.**

Sparse coding spike by spike

In the supplementary information of [15], Brendel et al. provide a set of learning rules for an efficient autoencoding network that can be trained on unwhitened data. In our notation and framework the feedforward weight rule is

$$\Delta\Phi_{mi} = a_m x_i \left(1 - \sum_j \Phi_{mj} x_j\right). \tag{6.13}$$

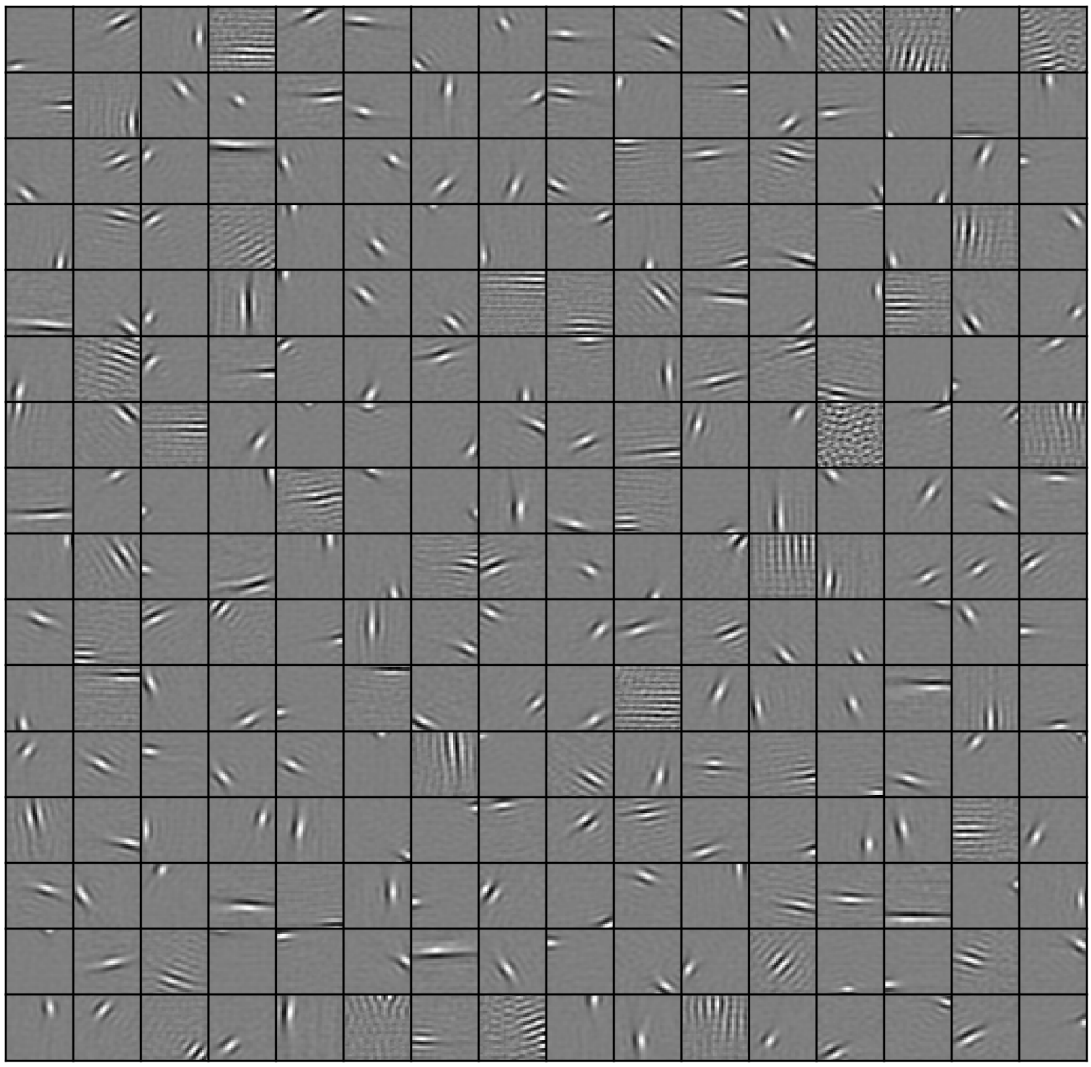

Figure 6.6: **All dictionary elements from a complete SAILnet model trained on whitened natural image patches.**

SAILnet with this rule and the membrane potential-based W rule can learn using unwhitened data.

This rule as written does not fit our notion of biological plausibility: each synapse needs to know an aggregate of many other synapses' inputs that is not simply captured in the membrane potential.

However, an approximation using the membrane potential in place of this term also works:

$$\Delta\Phi_{mi} = x_i(a_m - \langle u_m(t)\rangle_t) \tag{6.14}$$

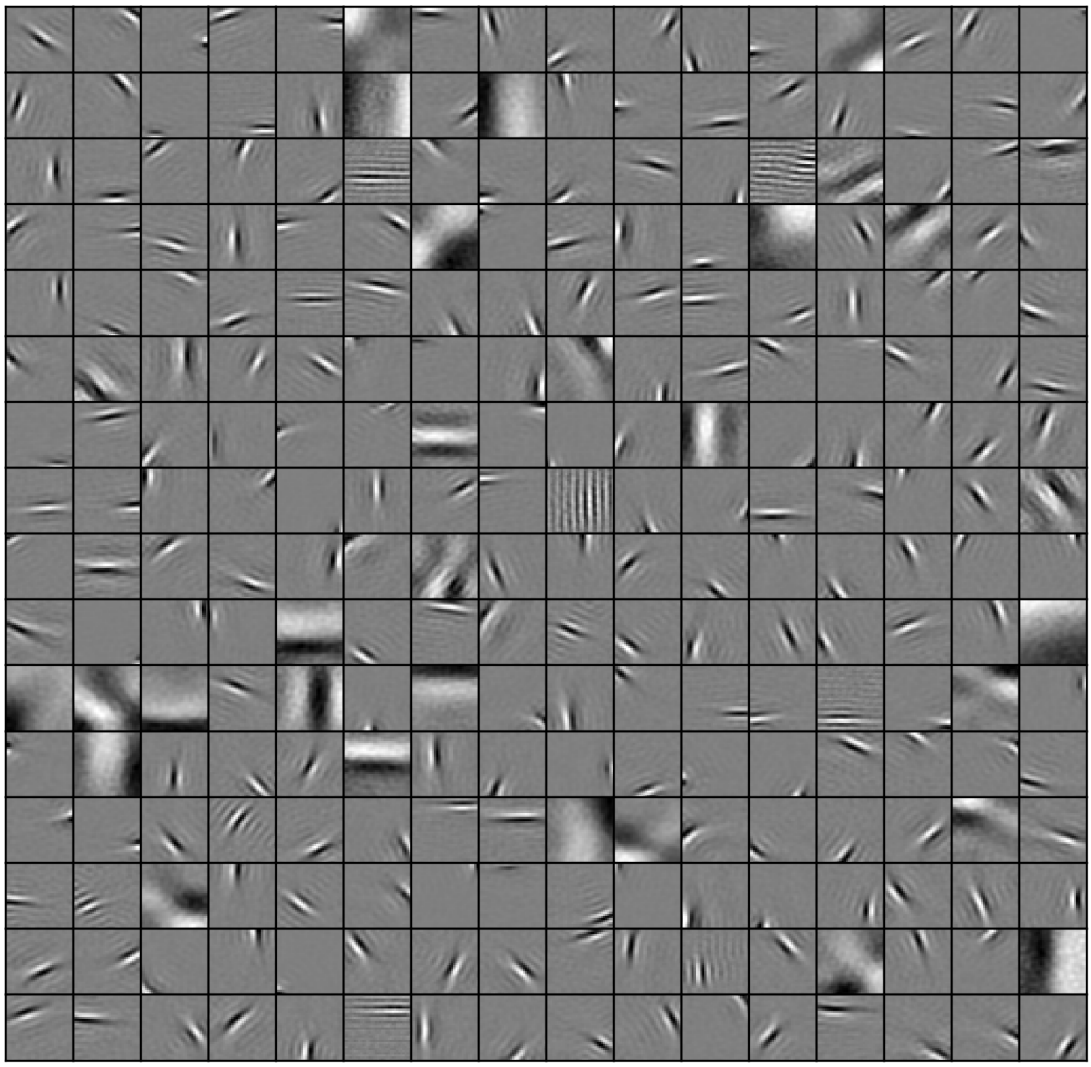

Figure 6.7: **All dictionary elements from a complete sparse coding dictionary learned with stochastic gradient descent and LCA inference on natural image patches that were not pre-whitened.**

where the angle brackets denote averaging over inference time. To achieve sparse codes, we can use this learning rule together with the inhibitory weight learning rule of [15] and the SAILnet spiking threshold learning rule of [110]. For completeness, the inhibitory weight learning rule in our notation is

$$\Delta W_{mn} = \langle u_m(t) - W_{mn} \rangle_{\{t:y_n(t)\neq 0\}}, \tag{6.15}$$

where the angle brackets denote an average over the post-synaptic neuron's spike times. We call the resulting network MP-SAILnet, using the rules in Eqs. 6.14 and 6.15 plus the

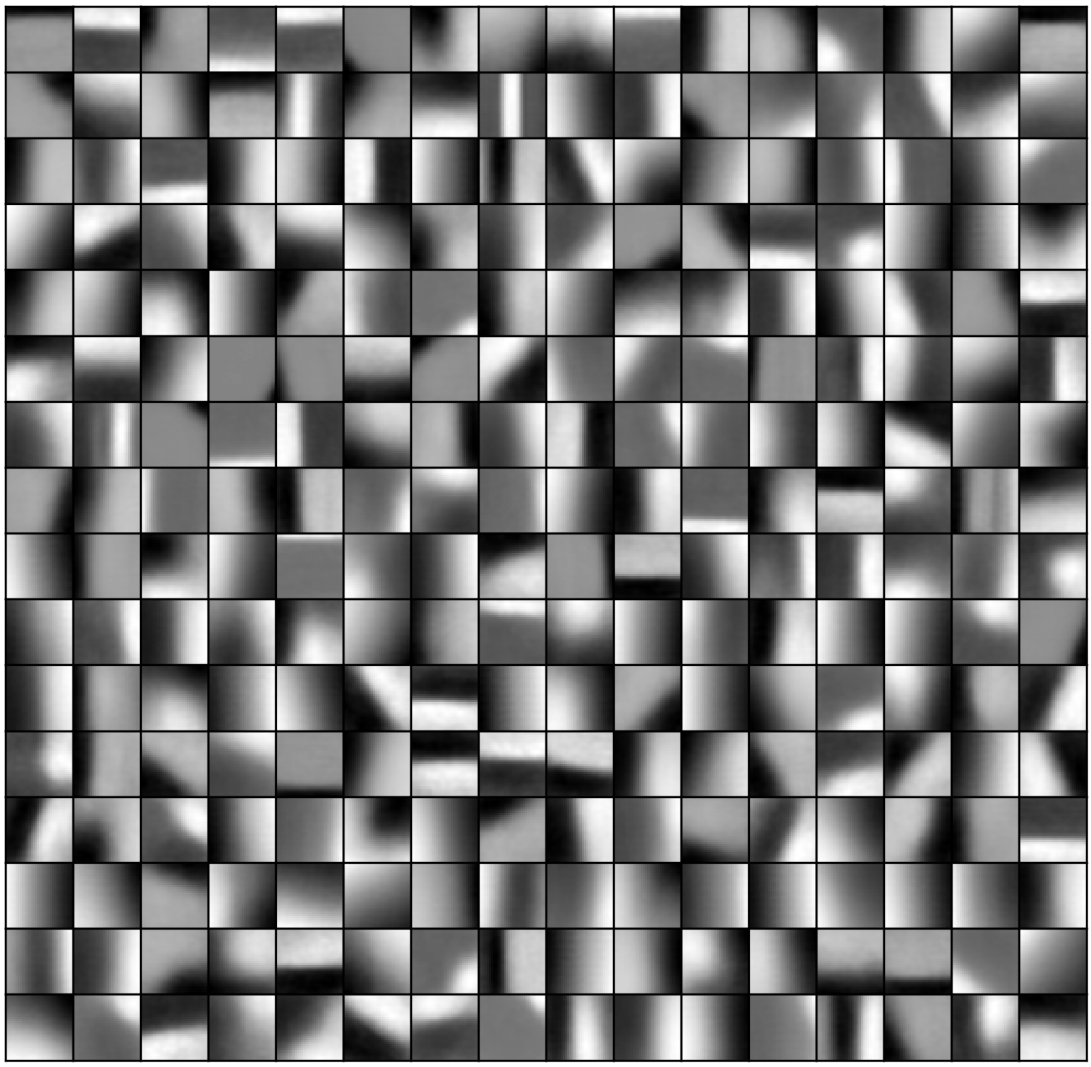

Figure 6.8: **All dictionary elements for a complete SAILnet model trained on natural image patches that were not pre-whitened.**

SAILnet threshold update. MP-SAILnet can learn the expected sparse features of natural image patches without pre-whitening, at the cost of requiring a neuron-global "membrane potential" to be available to the feedforward synapses while determining their weight updates. We discuss this issue further and clarify the comparison to SAILnet and other models in section 6.5.

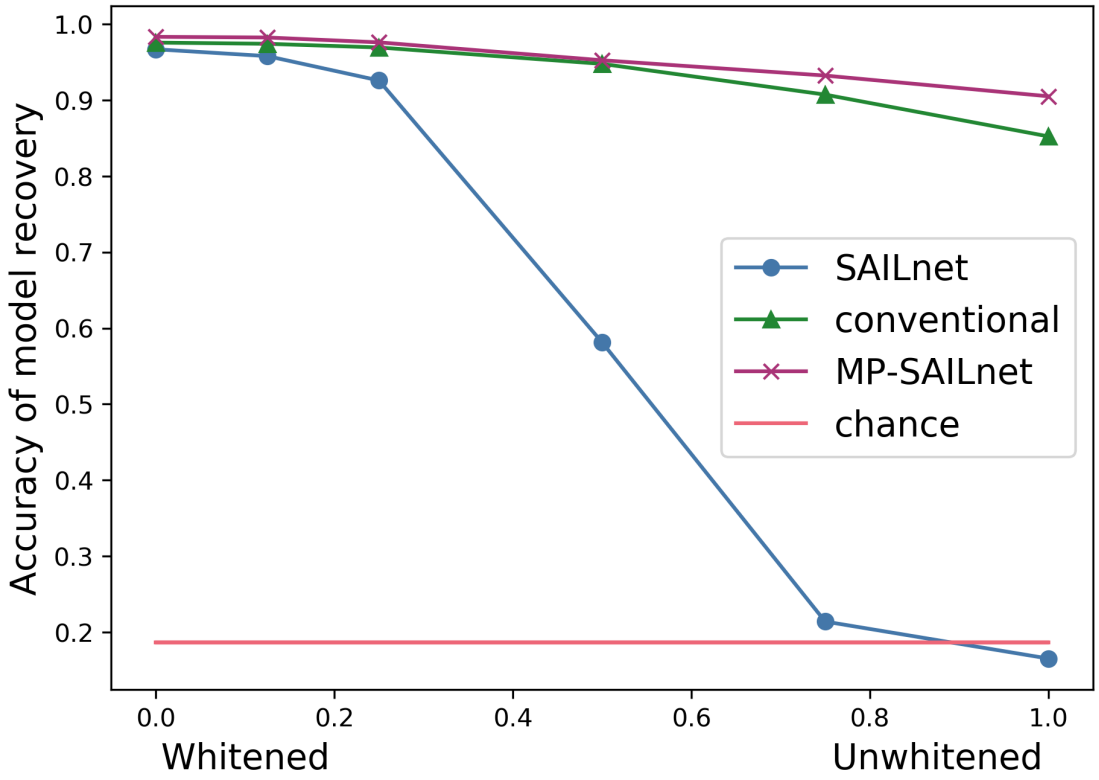

Figure 6.9: **A modified SAILnet with learning rules that depend on post-synaptic membrane potential recovers a sparse model even on unwhitened data.** This figure is identical to Fig 1E in the main text, except for the addition of data for our modified SAILnet model (MP-SAILnet).

Model variations and relation to whitening

A number of neurally-inspired sparse coding models have been proposed to learn the parameters of sparse linear generative models. The properties of these models are "biologically plausible" to different degrees and in different ways, and the various algorithms have various strengths and weaknesses in perfomance. Here we attempt to clarify the type of biological plausibility we have focused on in this work and how models that are more or less plausible in this sense can or cannot learn effectively from data that is not pre-whitened.

The primary aspect of biological plausibility we are concerned with here is locality of the learning rules. One possibility is that learning rules should be "neuron-local", meaning that they should only depend on quantities that a neuron would have access to. For instance, a neuron has access to its own dendritic synapse strengths, Φ_{ij}, its membrane potential and spike status (activity), u_i and a_i, and the spike status of its neighbors, $a_{\setminus i}$. In contrast, it is not likely that neuron i would have access to the dendritic synapse strengths in other

neurons: $\Phi_{(\backslash i)j}$.

Although these quantities may be available at some location in the neuron, it is not clear how many of them may be available at the dendritic synapses for learning. A different criterion for biologically plausible learning rules, and the one we adopted in the main text, is for the rules to be "synaptically local". Tabulating what neural quantities are available at the dendrites for learning is still an open area of investigation. For instance, it has been shown that action potentials can backpropagate into the dendritic arbor from the soma [99], so those spike times could be known to all synapses. However, due to electrical compartmentalization throughout the dendrites of typical neurons, it is not clear that the membrane potential at any specific point in the neuron, *e.g.* the soma, is continusously communicated to every synapse in the dendritic tree.

The learning rules for various models require different sets of values to update the dendritic synaptic strengths. For several models, the features required to learn the dendritic synaptic weight for neuron i and presynaptic element j are listed in Table 6.1. Here X_j is the jth element of the feedforward activity and $X_{\backslash j}$ are the other elements (not available directly at synapse ij). A dendritic synapse may have access to the net filtered input of the feedforward activity, $\sum_j \Phi_{ij} X_j$, the current membrane potential of its soma, u_i, or its own spiking activity and the spiking activity of its neighbors, a_n. Finally, gradient descent on the mean-squared reconstruction error requires the dendritic synapse to have access to the sum of the dendritic synapses of all neurons weighted by their spiking activity, $\sum_m \Phi_{mj} a_m$.

Feature	SAILnet[110]	Brendel et al.[15]	MP-SAILnet	mse GD	NL	SL
X_j	✓	✓	✓	✓	✓	✓
$X_{\backslash j}$	×	×	×	×	✓	×
$\sum_j \Phi_{ij} X_j$	×	✓	×	×	✓	×
a_n	✓	✓	✓	✓	✓	✓
u_i	×	×	✓	×	✓	×
$\sum_m \Phi_{mj} a_m$	×	×	×	✓	×	×
Neuron-local	✓	✓	✓	×		
Synaptically local	✓	×	×	×		
Unwhitened inputs?	×	✓	✓	✓		

Table 6.1: Features required for learning reconstruction feature weight, Φ_{ij}, for neuron i and presynaptic element j for learning rules associated with four models. NL and SL indicate whether the learning rule or quantities is neuron-local (NL) or synaptically-local (SL). Learning rules with a check mark in the "Unwhitened inputs?" row are capable of learning from significantly non-white data.

Chapter 7

Efficient causual auditory coding

In the last chapter we considered how locality, a constraint associated with implementation in a biological neural system, may require certain computations to occur in separate stages to facilitate optimization for stimulus statistics. In this chapter we consider a different constraint, causality, that any system must account for in order to process data in real time with minimal delays. Within the framework of convolutional sparse coding, we will see that how a system deals with causality (or does not) affects the optimal parameters of that system.

7.1 Abstract

Efficient coding models of the early auditory system explain the frequency-bandwidth relationship in auditory nerve filters as measured by reverse correlation, as well as some basic properties of the shape of these filters. Although the model of Smith and Lewicki appeared to explain the detailed temporal structure of these filters as well, we show that a proper comparison with the revcor filters shows that the model more strongly matches the *time-reverse* of these filters than the filters themselves. We propose that a *causal* efficient coding model will tend to favor filters with a similar shape in time to the auditory nerve filters, and we provide preliminary evidence with particular causal model closely related to the non-causal model of Smith and Lewicki.

7.2 Introduction

Efforts to understand neural sensory processing in terms of high-level principles have provided substantial insight into the function of the early auditory system. The idea that early stages in a sensory pathway, such as the auditory nerve, should be optimized for efficient transmission of information to later processing stages [**Ricke1997**, 4, 7, 2, 24, 22, 49, 56, 26, 9, 43, 76] led to the seminal work of Smith and Lewicki that explained important features of auditory nerve cells in terms of efficient coding of natural sounds[59, 98, 97]. Specifi-

cally, Smith and Lewicki developed a representation of acoustic signals consisting of discrete "spikes," each coding for a feature at a particular time [98]. They used a convolutional matching pursuit algorithm to find efficient spike codes, meaning codes with low reconstruction error and few spikes. To obtain predictions of the properties of auditory nerve cells, they optimized the acoustic features in the model for coding a set of sounds from the natural environment. The distribution of frequencies and bandwidths of the optimal features closely matched the corresponding distribution for reverse-correlation filters measured in the auditory nerve of several mammals, suggesting that the auditory nerve may be optimized for efficient coding[97].

While this work on efficient auditory coding provided substantial insight into early auditory processing, it also raised important further questions and a careful examination shows that some significant details of the reverse-correlation filters remain to be explained. Although there exists an interesting line of work trying to describe and understand cochlear processing beyond the simplified linear-filter description supplied by measuring reverse-correlation filters (*e.g.*, [13, 19, 61, 48, 62, 107]), in this work we focus on theoretical explanations of the properties of these filters.

The optimal features for Smith and Lewicki's spike coding model of natural sounds are asymmetric sinusoids in time, which appear to closely resemble the filters measured by reverse correlation in auditory nerve cells [13, 19]. However, this resemblance is misleading in part: the model filters most closely resemble the time-reversed auditory nerve filters (often reported as the "impulse response" by analogy to linear systems), and the asymmetric envelope of the model filters is backwards from that of the reverse-correlation ("revcor") filters. We elucidate this point in section 7.3 and figure 7.1.

Since the asymmetric envelopes of measured revcor filters are not optimal for coding natural sounds under the model of Smith and Lewicki, it is worth considering whether this asymmetry may have an explanation under a different model. Their model departs from biological realism in several ways, and it may be that a more realistic model will benefit from the observed asymmetry.

In this work we propose that the time-asymmetry of auditory nerve fiber revcor filters may be explained by causality. A system is *causal* if its response depends only on the past, never the future. Biological systems appear to be causal, but it is possible that causality could be irrelevant to models of neural processing in some situations if the system can simply wait for whatever information it might need from the future before responding to an event in the past. However, organisms benefit from being able to respond quickly to stimuli, and neurons at the auditory periphery are sensitive to stimuli within a few milliseconds [13, 19], as is clear from auditory nerve revcor filters. We present a simple model that is closely related to that of Smith and Lewicki but that generates codes causally, and we show that optimal filters in this model have the same qualitative shape – including the time-asymmetry – as auditory nerve revcor filters, although the correspondence is not perfect.

In section 7.3 we establish terminology and clarify why filters in a model like that of Smith and Lewicki should be compared to the time-reverse of reported impulse responses. Then in section 7.4 we review previous methods and describe our own simple causal version

of matching pursuit. We show our primary results in section 7.5, including a replication of Smith and Lewicki's results and results from our causal model. Finally we discuss in section 7.6 our findings and some of what we believe remains to be explained about cochlear processing.

7.3 Preliminaries

Linear, time-invariant systems

The cochlea is not a linear, time-invariant system. However, linear, time-invariant systems are simple to analyze and provide a simple first-order approximation and a convenient language that turns out to be more applicable to nonlinear systems like the cochlea than one might expect.

A linear, time-invariant system can be completely characterized by its impulse response $h(t)$, which we define as the response $y(t)$ of the system to an impulse $x(t) = \delta(t)$. The output of the system in response to a general stimulus is

$$y(t) = (x * h)(t) = \int_{-\infty}^{\infty} x(t-\tau)h(\tau)d\tau \tag{7.1}$$

$$= \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau \tag{7.2}$$

where the second line follows from the commutativity of convolution[1]. From this equation we can recover our definition of the impulse response:

$$y(t)|_{x(t)=\delta(t)} = \int_{-\infty}^{\infty} \delta(\tau)h(t-\tau)d\tau = h(t). \tag{7.3}$$

We can also see from equation (7.2) that the output is a weighted average of the stimulus at every time point, with the weights given by the *time-reversal of the impulse response* $h(-\tau)$, shifted by the time t at which the output is evaluated. In particular, a stimulus will induce the largest response if its shape matches the system's kernel $\Phi(\tau) = h(-\tau)$. The system will in general respond more weakly to a stimulus with the same power and the shape of the impulse response $h(\tau)$, as shown in figure 7.1.

We say a linear, time-invariant system is *causal* when $h(t)$ is zero for $t < 0$, *i.e.*, when the output gives no weight to the future of the signal.

[1] A warning/reminder for readers steeped in machine learning: the "convolutions" in software like TensorFlow and PyTorch as of 2018 are really cross-correlations, without the minus sign. The cross-correlation is not commutative.

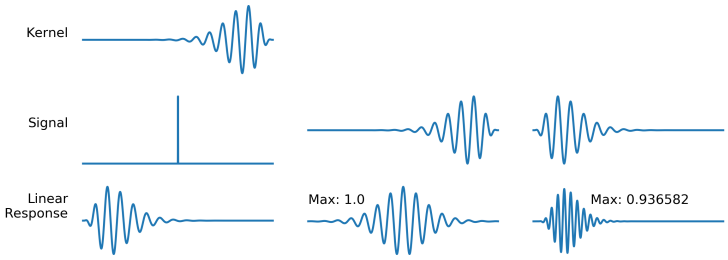

Figure 7.1: A linear time-invariant system and its responses to three example signals. The kernel is the time-reverse of the response to a unit impulse (left). The system is maximally sensitive to a signal proportional to the kernel (center) and responds less to a signal proportional to the impulse response (right) if the kernel is not symmetric.

Reverse correlation (revcor) filters

Reverse correlation estimates the filter h for a linear time-invariant system. When used with a system that is not linear and/or time-invariant, it can still provide some information about the function of that system. If the system is nonlinear but time-invariant, for example, the reverse correlation filter can be thought of as the linear term in a series expansion of the true response function. In the special case of a system that generates its output as a nonlinear function of a linear functional of its input, it can be shown that reverse correlation with white noise recovers the linear functional (i.e., h) up to a multiplicative constant [16].

Reverse correlation has been used to study retinal ganglion cells and primary visual cortex simple cells, among others[84]. Most relevantly for our purposes here, the reverse correlation ("revcor") filters of fibers in the cochlear nerve have been measured [13, 34, 19] and are often modeled by asymmetric sinusoidal "gammatone" functions [19].

7.4 Methods

Time-relative spike coding

We adopt the spike coding model described in [98], with each "spike" coding for an instance of its corresponding kernel at the time of the spike. This model can be formulated as the convolution of the kernels $\Phi_m(t)$ with their spikes $a_m(t)$:

$$x(t) = \sum_m \int_{-\infty}^{\infty} a_m(\tau)\Phi_m(t-\tau)d\tau + \epsilon(t). \tag{7.4}$$

where $\epsilon(t)$ is "noise" unexplained by the model. With time discretized and finite kernels, we have

$$x(t) = \sum_m \sum_\tau a_m(\tau)\Phi_m(t-\tau) + \epsilon(t). \tag{7.5}$$

The coefficients $a_m(t)$ may be real numbers; we use the analogy to spikes loosely here. This formulation is most interesting to us when the coefficients are usually zero with occasion nonzero "spikes." Although decoding is linear (the map from a to x is linear), we cannot achieve such sparseness in general with a linear encoding function.

Matching pursuit

Smith and Lewicki used a convolutional matching pursuit[65] algorithm to generate spike codes from acoustic signals [97]. At each step, the algorithm adds to the spike code the one spike that most reduces the mean-squared error $\langle \epsilon^2 \rangle$ for the representation in equation (7.5) of a given signal. Figure 7.2A shows one step of this process. The encoding algorithm terminates when the next spike's magnitude would fall below a preset threshold.

Since each spike is determined using the whole signal and all previously determined spikes – including those placed at future times – matching pursuit is not a causal algorithm.

Causal matching pursuit

There are many possible algorithms to obtain a sparse spike code (for another example, see [20]). Here we focus on a causal scheme closely related to matching pursuit that we call causal matching pursuit. One step of causal matching pursuit is shown in figure 7.2B. Rather than scan over the entire residual signal for the next best spike, causal matching pursuit simple takes the best spike at the current time – or commits to having no spike at that time if no spike exceeds the fixed, preset threshold. After a fixed number of iterations at the current time (just one iteration for the results presented here), we move on to the next time point. To determine the spike code at each time step, therefore, causal matching pursuit only requires the signal and spikes that precede that time as long as each kernel is causal.

Learning

We follow Smith and Lewicki in using gradient descent on mean-squared reconstruction error to optimize kernels for efficient coding [97]. For reproducing their results, we also follow their prescription for allowing the kernels to grow and shrink in length. For our causal matching pursuit model, we instead simply use a fairly long (800 sample) kernel size; some learned kernels are near zero for a substantial fraction of their extent.

Since Smith and Lewicki found good agreement with auditory nerve properties by training their model on a set of speech sounds without the careful adjustments necessary with other sets of "natural sounds" [97], we used speech sounds as a proxy for natural sounds generally in this work. Specifically, we used the TIMIT speech corpus [33].

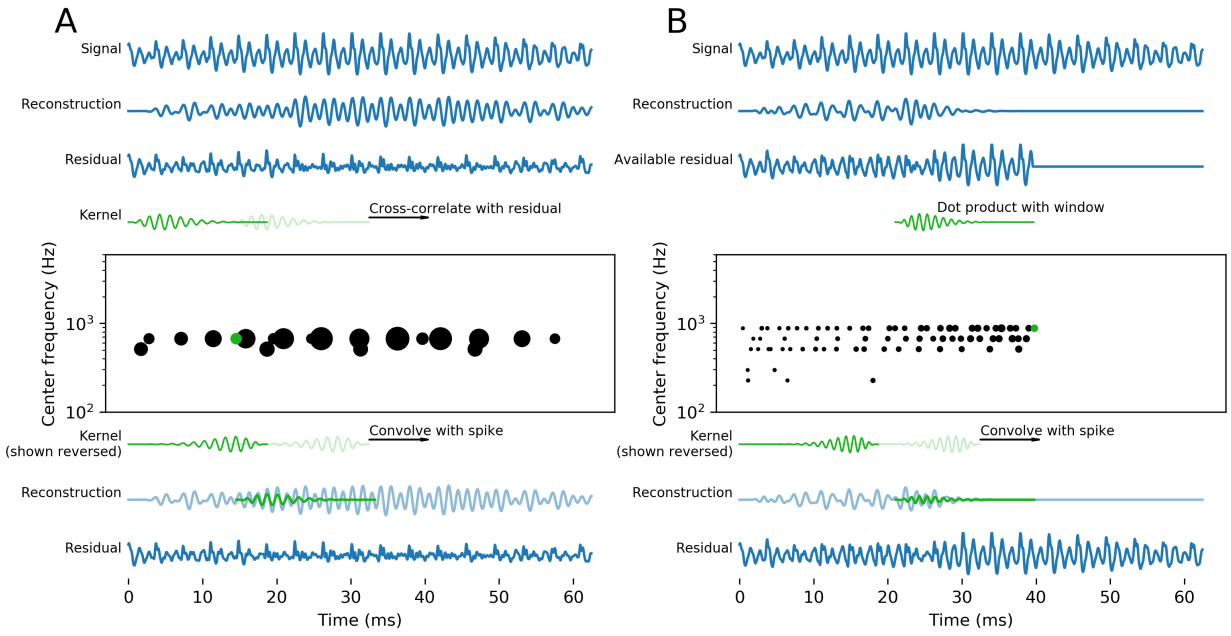

Figure 7.2: **One step of matching pursuit and causal matching pursuit encoding.** (**A**) The current residual signal is the difference between the given signal and the reconstruction from the current code. To compute the next spike in the code, the kernels are cross-correlated with the entire residual and the largest value across kernels and time is chosen as a new spike (highlighted in the spikegram). The new spike's contribution to the new reconstruction is given by the convolution of the kernel with the spike, i.e., the kernel itself placed at the spike's time. The residual is updated and the process repeats until a stopping condition. (**B**) In a causal method only the signal up to the current time is available, so only the residual up to that time is considered. In causal matching pursuit, we take dot products between each kernel and a window of the residual ending at the current time. The largest dot product is accepted as a spike if it exceeds a preset threshold. The reconstruction and residual are updated in the same way as matching pursuit, and then the next time step is considered.

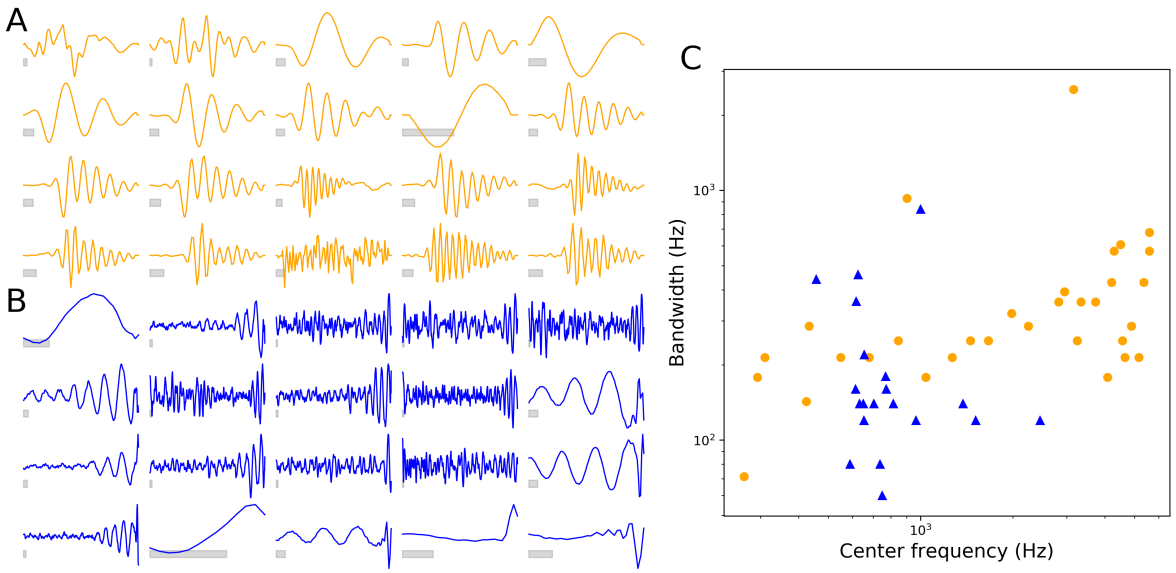

Figure 7.3: (**A**) Kernels (orange lines) optimized for matching pursuit; compare to [97] figure 3A. Gray scale bars represent 0.5 ms (8 samples). Kernels are sorted by center frequency from low to high. (**B**) Kernels (blue lines) optimized for causal matching pursuit are also asymmetric sinusoids, but the asymmetry is in the opposite direction. Plotting conventions are as in (A), except that each kernel is trimmed to show 90% of the total power in the kernel. (**C**) The kernels exhibit a range of center frequencies and bandwidths (defined here as the frequency range over which the kernel has at least half its maximum power). Each orange circle represents one kernel in (A); each blue triangle represents one kernel in (B). The kernels optimized for matching pursuit follow a similar distribution in frequency-bandwidth space to that observed in auditory nerve, as shown in [97].

7.5 Results

We attempted to replicated Smith and Lewicki [97] as closely as possible in order to understand their results and make a clear comparison to the results of our causal model. Details on our implementation and our explorations of closely related models are given in the supplementary material. We also attempted to present the kernels learned by the two models as similarly as possible. Results for both models are shown in figure 7.3.

We replicate the qualitative distribution of the learned kernels in center frequency-bandwidth space (Figure 7.3A) and the qualitative gammatone-like shapes of many of the kernels in the model optimized for ordinary, non-causal matching pursuit (Figure 7.3C, orange circles). We believe the discrepancy between our results and those of [97] are due to difficulties in optimization, which future work should correct.

Preliminary results with our causal model show asymmetric sinusoids with slow attacks and fast decays (Figure 7.3B), but detailed agreement with auditory nerve revcor filters

is lacking. The frequency-bandwidth distribution of the kernels optimized for our causal encoding model is significantly different. These results depend on threshold parameters in the model and on the optimization strategy, as discussed in section 7.8.

7.6 Discussion

Smith and Lewicki [97] found and we have confirmed that the kernels optimized for matching pursuit-based convolutional sparse coding of speech sounds bear a strong resemblance to the impulse responses of auditory nerve fibers measured by reverse correlation (revcor). The impulse response is the time-reverse of the kernel corresponding to the linear approximation implied by a revcor measurement, so the optimal kernels presented in [97] and in our Figure 7.3A are directly comparable to the time-reverse of the revcor impulse responses. Indeed, reverse correlation with white noise using the matching pursuit model yields impulse responses very close to the time-reverse of the kernels(see [97] supplementary discussion). We conclude that the model predicts the time-reverse of the auditory nerve data.

The strong resemblance between a theoretical prediction and the time-reverse of the data the theorists hoped to explain strikes us as begging for an explanation. Although we believe the correspondence remains strong on close inspection, there are a few subtleties that explain away the result to some degree. First, the model of [97] involves a number of *a priori* somewhat arbitrary choices whose justification lies primarily in the strength of their resulting predictions. Other approaches to sparse coding of sound waveforms yield different results, often without any time-asymmetry (see, *e.g.*, [59] and our supplementary material). In exploratory investigations of related models, we found that the L0-sparse nature of the matching pursuit encodings was crucial in obtaining asymmetric sinusoids. We also found that providing significantly more zero-padding in the kernels during learning led to some kernels of greater symmetry and longer extent in addition to the gammatone-like kernels. Second, the results of [97] depend critically on the choice of natural sound ensemble, a point obscured by our *post hoc* emphasis on results using clean recordings of individual humans speaking sentences.

These subtleties aside, we believe the explanation is likely to lie in an improved understanding of cochlear processing and the sense in which it is optimized for natural sounds. This paper provides a potential explanation already hinted at in [59] as well as some preliminary evidence in favor of that explanation. However, we hope that our work has made clear that more remains unexplained about why the auditory nerve has the response properties it does than was previously appreciated, and that new ideas may be required to further our understanding. We consider briefly a few possible directions for future work.

There are other possible efficient coding models that respect causality, aside from the one we focused on here. In the supplementary material we discuss a model based on maximum *a posteriori* inference in the convolutional sparse coding probabilistic model, conditioned on the past of the code. One could also explore deep neural network models, which are likely flexible enough to learn to efficiently encode a set of natural sounds.

Some other constraint or effect that could be added to an efficient coding model, besides causality, may explain the shape of the envelope of the cochlear revcor filters. A model that accounts for the time scale of transmission of information among units – *e.g.*, as determined by the time scale of feedback from outer hair cells through the basilar membrane – might have different optimal kernels, for example. The models we have considered here have tens of analog-output units, whereas a mammalian auditory nerve may contain tens of thousands of neurons that receive inputs from inner hair cells and fire stereotyped action potentials[41]. It is unclear how this significant difference impacts the results of efficient coding models. Although reverse correlation with white noise gives a result that corresponds very closely with the generative kernels of all the models we have considered in this work, it is not necessary that auditory nerve revcor filters correspond to any underlying "kernel." Indeed, measurements at different sound levels show somewhat different revcor filters [19], and cochlear processing is highly non-linear.

As pointed out in [97], speech is more likely adapted to the cochlea than vice versa. It may be that the cochlea is organized by frequency for a mechanistic reason unrelated to the detailed statistical structure of natural sounds, and that efficiency at the level we have attempted to model is too small a consideration to explain major properties of coding in the auditory nerve.

Efficient coding has yet to offer a complete explanation for auditory nerve revcor data, let alone for the full, complex response properties of auditory nerve cells. However, the observation that a simple model of efficient coding of natural sounds appears to explain some aspects of the revcor filters suggests that further inquiry may yield new insights and/or a stronger correspondence between theoretical models and measurements.

7.7 Alternative methods

Convolutional sparse coding

Sparse coding can be viewed as a probabilistic model[76, 75] based on linear reconstruction of a stimulus vector:

$$x_i = \hat{x}_i + n_i = \sum_m a_m \Phi_{mi} + n_i \tag{7.6}$$

where we assume that n_i is iid Gaussian noise with variance σ^2, conditioned on the set of coefficients a:

$$p(x|a;\Phi) \propto \prod_i e^{-(x_i - \sum_m a_m \Phi_{mi})^2/2\sigma^2} \tag{7.7}$$

Sparseness arises from the assumption of a sparse and (often) independent prior for the coefficients a_m^μ, for example

$$p_a(a) = \prod_m \frac{1}{2\lambda} e^{-\lambda|a_m|}. \tag{7.8}$$

It is straightforward to extend this model to the convolutional representation discussed in the main text, with

$$p(x|a;\Phi) \propto \prod_t e^{-(x(t)-\sum_m \sum_\tau a_m(\tau)\Phi_m(t-\tau))^2/2\sigma^2}. \tag{7.9}$$

The encoding scheme of Smith and Lewicki [97] can be seen as an approximation to maximum *a posteriori* inference in this model with an $L0$ prior $p_a(a)$. With an L1-sparse prior it is possible to find at least a local maximum of the posterior distribution using an optimization scheme such as gradient descent, as done in [76]. We experimented with this convolutional sparse coding strategy, for which the optimized kernels were mostly localized symmetric sinusoids as found in [59] with ICA.

Causal convolutional sparse coding

Maximum *a posteriori* inference in the convolutional sparse coding model described above does not suggest a causal encoding scheme. To make the procedure causal, we instead find the most probable latent variables $a_m(t)$ *conditioned on the past of the signal and code*:

$$\max_{a_m(t\geq T)} p\left(a_m(t \geq T)|a_m(t < T), x(t < T)\right). \tag{7.10}$$

Applying this procedure iteratively for $T = 0, 1, 2\ldots$ gives a code $a_m(t)$ with no acausal dependencies.

Using Bayes's Theorem and assuming the prior $p(a_m)$ is factorial,

$$p\left(a_m(t \geq T)|a_m(t < T), x(t < T)\right) = \frac{p(a_m(t \geq T))p(x(t < T)|a_m)}{p(x(t < T))}. \tag{7.11}$$

The denominator is a constant for the purposes of our optimizations. The numerator is a product of two terms: first, the prior for present and future activations, which we assume to be simple and factorial; and second, the likelihood of the data conditioned on the activations, which is Gaussian. Taking a logarithm, we can write the function we want to minimize at a given T as

$$L(T) = \sum_{t\leq T} ||x(t) - \sum_\tau \sum_m a_m(\tau)\Phi_m(t-\tau)||^2 \tag{7.12}$$

and we perform the minimization with respect to all $a_m(s)$ with $s \geq T$, keeping those with $s < T$ fixed from previous iterations. We initialize the optimization at time T using the values of $a_m(s \geq T)$ from the optimization of that fixed $a_m(T)$.

Given a code, we can compute the gradient of the model log-likelihood with respect to the kernels with the code fixed and update the kernels as in ordinary sparse coding. Results using convolutional sparse coding and the causal variant we have just discussed are shown in figure 7.4. The causal variant's results should be considered preliminary as the model has not converged due to time constraints. While some kernels appear to be gammatone-like with

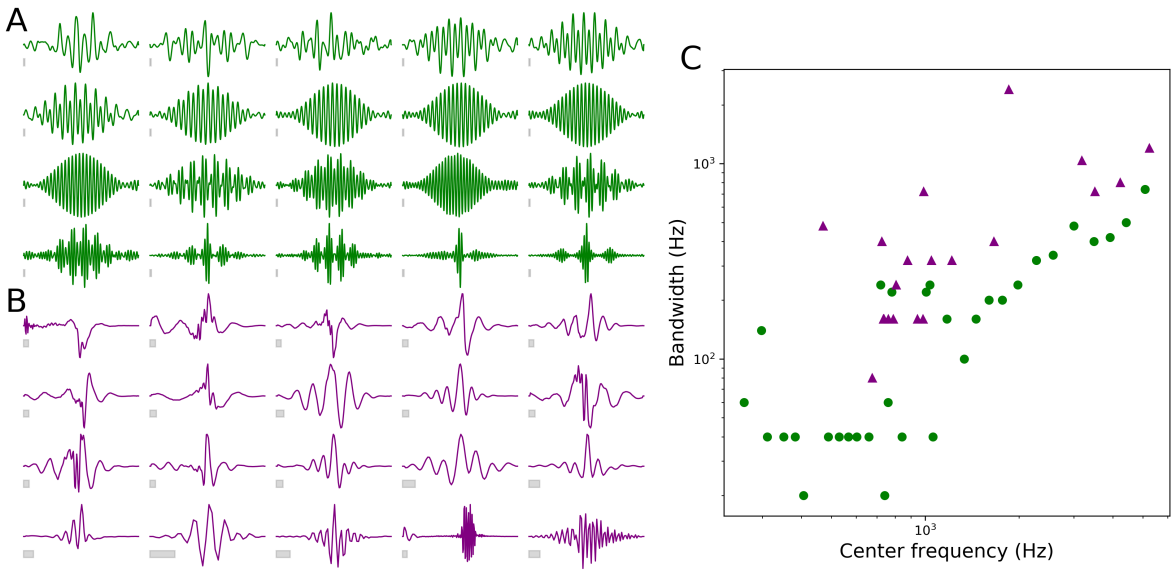

Figure 7.4: This figure is identical to Figure 7.3 except that the kernels in panel A were optimized for convolutional sparse coding with a factorial Laplace prior and the kernels in panel B were (incompletely) optimized for our causal variant of convolutional sparse coding.

the asymmetry expected from revcor filters, it is not clear whether fully-optimized kernels will have these properties.

The primary difficulty with causal convolutional sparse coding in our experience is its computational cost. Using PyTorch to run the model on an NVIDIA Tesla K40 GPU, each gradient step takes about 30 minutes to compute for a single signal of length 20000 samples (1.25 seconds of audio). We expect training to take 1000s of gradient steps, for a training time on the order of weeks to months. Future work could reduce this time in several ways, including but not necessarily limited to: a faster GPU, better optimization of convolution calculations, and more efficient optimization at each T than gradient descent. It may also be possible to subsample the times at which the optimization is performed without too much degradation.

7.8 Causal matching pursuit and optimization

Beyond the difficulties in efficiently coding a signal on-the-fly, there are challenges in finding the optimal kernels for a given coding strategy. Causal matching pursuit uses a heuristic to avoid using many spikes in its codes, and a learning rule based only on reducing coding error may exploit the shortcomings of the heuristic to reduce error at the expense of efficiency. One solution where this difficulty is particularly clear is shown in Figure 7.5. A single kernel with all its weight on the last time bin can be used to code with perfect fidelity, with the

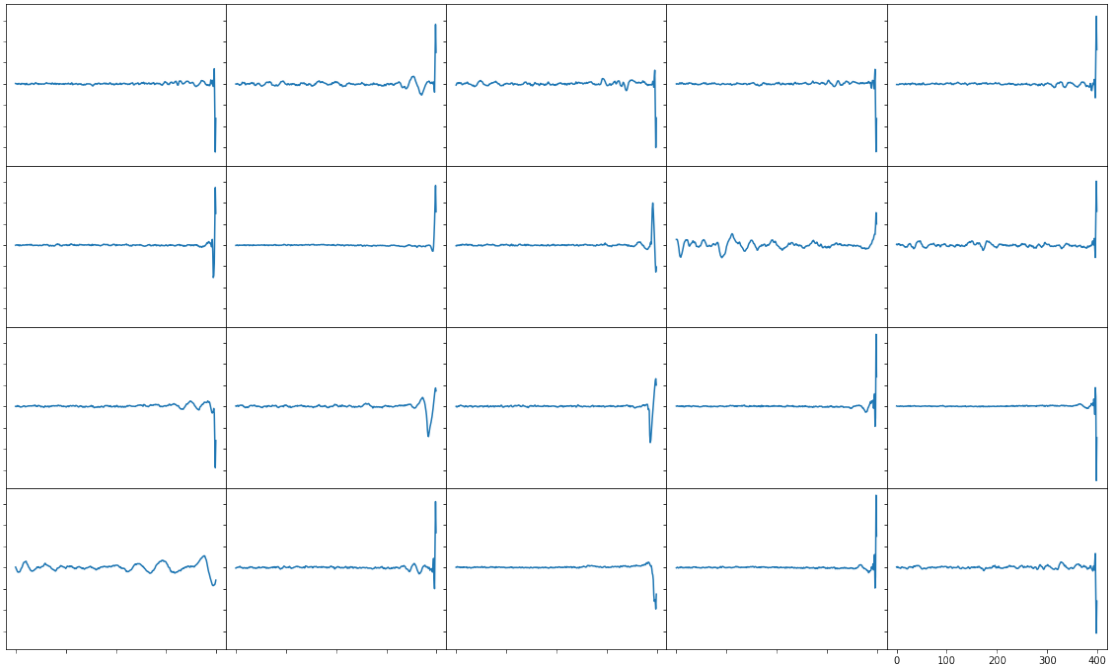

Figure 7.5: A set of kernels that allow causal matching pursuit to achieve low error but at the cost of a high spike rate.

code identical to the signal.

In an attempt to find kernels that optimize a combination of coding accuracy and sparseness, we also attempted to optimize the model considering the code to be a function of the kernels. In this alternative optimization scheme, the gradient of the loss includes terms due to $\partial a_m / \partial \Phi_m$ rather than considering the a_m to be fixed. The sparseness term in the sparse coding loss function thereby contributes to the gradient. Preliminary results with this scheme are shown in Figure 7.6. We speculate that our implementation of this scheme, which neglects terms arising from the dependence of the residual signal on the code, may not provide an adequate approximation to the true gradient.

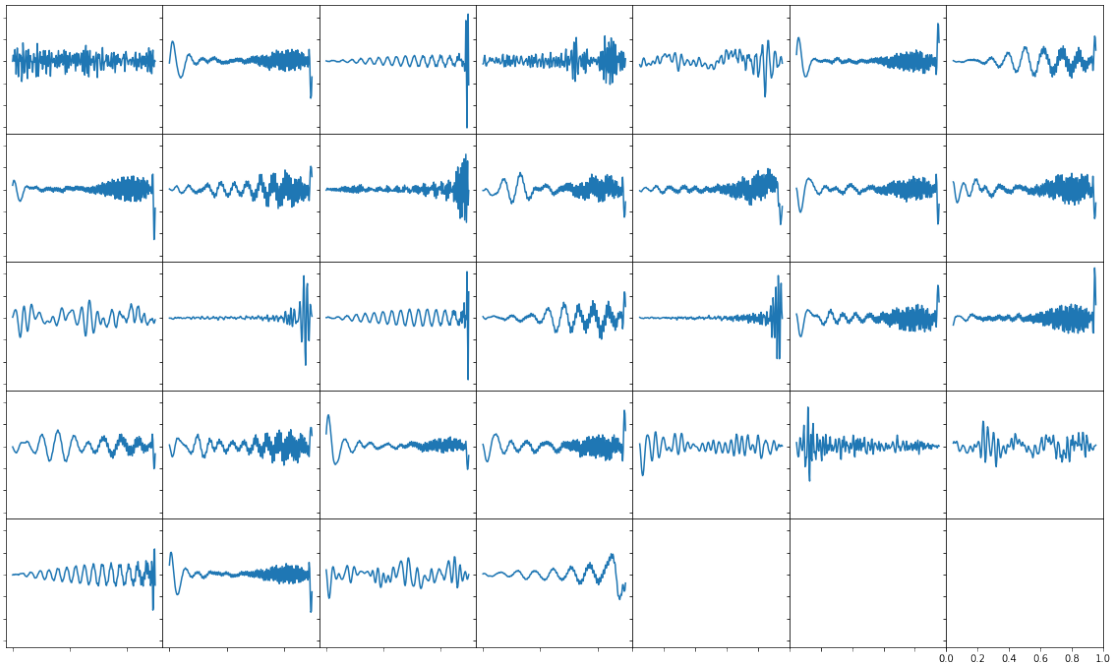

Figure 7.6: A set of kernels for causal matching pursuit optimized by treating the coding scheme as a function of the kernels and signal.

Bibliography

[1] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. 9th printing. New York: Dover, 1972.

[2] Joseph J Atick and A Norman Redlich. "What Does the Retina Know About Natural Scenes?" In: *Neural Computation* 210 (1992), pp. 196–210. ISSN: 0899-7667. DOI: 10.1162/neco.1992.4.2.196.

[3] H Attias and C E Schreiner. "Temporal Low-Order Statistics of Natural Sounds". In: *Advances in Neural Information Processing Systems* 9 (1997), pp. 27–33. DOI: 10.1.1.53.201. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.53.201%7B%5C&%7Drep=rep1%7B%5C&%7Dtype=pdf.

[4] Fred Attneave. "Some informational aspects of visual perception." In: *Psychological Review* 61.3 (1954), pp. 183–193. ISSN: 0033-295X. DOI: 10.1037/h0054663. URL: http://content.apa.org/journals/rev/61/3/183.

[5] Roland Baddeley. "Searching for filters with 'interesting' output distributions: an uninteresting direction to explore?" In: *Network: Computation in Neural Systems* 7.2 (1996), pp. 409–421.

[6] H. Barlow. "Redundancy reduction revisited". In: *Network: Computation in Neural Systems* 12.3 (2001), pp. 241–253. ISSN: 0954-898X. DOI: 10.1088/0954-898X/12/3/301. URL: http://www.informaworld.com/openurl?genre=article%7B%5C&%7Ddoi=10.1088/0954-898X/12/3/301%7B%5C&%7Dmagic=crossref%7B%5C%%7D7C%7B%5C%%7D7CD404A21C5BB053405B1A640AFFD44AE3.

[7] Horace Barlow. "Possible principles underlying the transformations of sensory messages". In: *Sensory communication* 6.2 (1961), pp. 57–58. ISSN: 15459624. DOI: 10.7551/mitpress/9780262518420.003.0013. URL: http://www.trin.cam.ac.uk/horacebarlow/21.pdf.

[8] Amir Beck and Marc Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". In: *SIAM journal on imaging sciences* 2.1 (2009), pp. 183–202.

[9] Anthony J. Bell and Terrence J. Sejnowski. "An Information-Maximization Approach to Blind Separation and Blind Deconvolution". In: *Neural Computation* 7.6 (1995), pp. 1129–1159. ISSN: 0899-7667. DOI: 10.1162/neco.1995.7.6.1129. arXiv: arXiv:1511.06440v1. URL: http://www.mitpressjournals.org/doi/10.1162/neco.1995.7.6.1129.

[10] Anthony J. Bell and Terrence J. Sejnowski. "The 'independent components' of natural scenes are edge filters". In: *Vision Research* 37.23 (1997), pp. 3327–3338. ISSN: 00426989. DOI: 10.1016/S0042-6989(97)00121-1. arXiv: 9809069v1 [arXiv:gr-qc].

[11] Matthias Bethge. "Factorial coding of natural images: how effective are linear models in removing higher-order dependencies?" In: *Journal of the Optical Society of America. A, Optics, image science, and vision* 23.6 (2006), pp. 1253–68. ISSN: 1084-7529. DOI: 10.1364/JOSAA.23.001253. URL: http://www.ncbi.nlm.nih.gov/pubmed/16715144.

[12] Gary G Blasdel. "Orientation selectivity, preference, and continuity in monkey striate cortex". In: *Journal of Neuroscience* 12.8 (1992), pp. 3139–3161.

[13] E. de Boer. "On cochlear encoding: Potentialities and limitations of the reverse-correlation technique". In: *The Journal of the Acoustical Society of America* 63.1 (1978), p. 115. ISSN: 00014966. DOI: 10.1121/1.381704. URL: http://scitation.aip.org/content/asa/journal/jasa/63/1/10.1121/1.381704.

[14] Stephen Boyd et al. "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers". In: *Foundations and Trends® in Machine Learning* 3.1 (2010), pp. 1–122. ISSN: 1935-8237. DOI: 10.1561/2200000016.

[15] Wieland Brendel et al. "Learning to represent signals spike by spike". In: (), pp. 1–15. arXiv: arXiv:1703.03777v2.

[16] Julian Jakob Bussgang. "Crosscorrelation functions of amplitude-distorted Gaussian signals". In: (1952).

[17] Charles F. Cadieu and Bruno A. Olshausen. "Learning Intermediate-Level Representations of Form and Motion from Natural Movies". In: *Neural Computation* 24.4 (2012), pp. 827–866. ISSN: 0899-7667. DOI: 10.1162/NECO{_}a{_}00247.

[18] Nicole L. Carlson, Vivienne L. Ming, and Michael Robert DeWeese. "Sparse Codes for Speech Predict Spectrotemporal Receptive Fields in the Inferior Colliculus". In: *PLoS Computational Biology* 8.7 (2012), e1002594. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1002594. URL: http://dx.plos.org/10.1371/journal.pcbi.1002594.

[19] Laurel H Carney and Tom C T Yin. "Temporal coding of resonances by low-frequency auditory nerve fibers: single-fiber responses and a population model." In: *Journal of neurophysiology* 60.5 (1988), pp. 1653–1677. ISSN: 0022-3077.

[20] Adam S. Charles, A. Kressner, Abigail, and Christopher J. Rozell. "A Causal Locally Competitive Algorithm for the Sparse Decomposition of Audio Signals". In: *Proceedings of the IEEE Digital Signal Processing and Signal Processing Education Workshop, Sedona, USA, January 4-7.* (2011), pp. 265–270. DOI: 10.1109/DSP-SPE.2011.5739223. URL: https://docs.google.com/file/d/0B3N-C9xfG%7B%5C_%7DCPb2w4SVhlQ01RU0E/edit.

[21] Gal Chechik et al. "Reduction of Information Redundancy in the Ascending Auditory Pathway". In: *Neuron* 51.3 (2006), pp. 359–368. ISSN: 08966273. DOI: 10.1016/j.neuron.2006.06.030.

[22] Gal Chechik et al. "Reduction of information redundancy in the ascending auditory pathway". In: *Neuron* 51.3 (2006), pp. 359–368.

[23] Y Dan, Joseph Atick, and R C Reid. "Efficient coding of natural scenes in the lateral geniculate nucleus: experimental test of a computational theory." In: *The Journal of neuroscience : the official journal of the Society for Neuroscience* 16.10 (1996), pp. 3351–3362. ISSN: 0270-6474.

[24] John G. Daugman. "Entropy Reduction and Decorrelation in Visual Coding by Oriented Neural Receptive Fields". In: *IEEE Transactions on Biomedical Engineering* 36.1 (1989), pp. 107–114. ISSN: 15582531. DOI: 10.1109/10.16456.

[25] Sophie Denève, Alireza Alemi, and Ralph Bourdoukan. "The Brain as an Efficient and Robust Adaptive Learner". In: *Neuron* 94.5 (2017), pp. 969–977. ISSN: 10974199. DOI: 10.1016/j.neuron.2017.05.016. arXiv: 1705.08031.

[26] Michael DeWeese. "Optimization principles for the neural code." In: *Network* 7 (1996), pp. 325–331. ISSN: 0954-898X. DOI: 10.1088/0954-898X/7/2/013.

[27] Michael R DeWeese and Anthony M Zador. "Binary Coding in Auditory Cortex". In: *Advances in Neural Information Processing Systems 15* 15 (2003), pp. 101–108.

[28] Dawei W Dong and Joseph J Atick. "Temporal Decorrelation: A Theory of Lagged and Nonlagged Responses in the Lateral Geniculate Nucleus". In: *Network* 6536.September (1995), pp. 159–178.

[29] Jan Eichhorn, Fabian Sinz, and Matthias Bethge. "Natural Image Coding in V1: How Much Use Is Orientation Selectivity?" In: *PLoS Computational Biology* 5.4 (2009). ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1000336. arXiv: 0810.2872.

[30] D. J. Field. "Relations between the statistics of natural images and the response properties of cortical cells." In: *Journal of the Optical Society of America. A, Optics and image science* 4.12 (1987), pp. 2379–94. ISSN: 0740-3232. DOI: 10.1364/JOSAA.4.002379. URL: http://www.ncbi.nlm.nih.gov/pubmed/3430225.

[31] P Foldiak. "Forming sparse representations by local anti-Hebbian learning". In: *Biological Cybernetics* 64.2 (1990), pp. 165–170.

[32] Jonathan Fritz, Mounya Elhilali, and Shihab Shamma. "Active listening: Task-dependent plasticity of spectrotemporal receptive fields in primary auditory cortex". In: *Hearing Research* 206.1-2 (2005), pp. 159–176. ISSN: 03785955. DOI: 10.1016/j.heares.2005.01.015.

[33] John S. Garofolo et al. *TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1*. Philadelphia, 1993.

[34] R. V. Harrison and E. F. Evans. "Reverse correlation study of cochlear filtering in normal and pathological guinea pig ears". In: *Hearing Research* 6.3 (1982), pp. 303–314. ISSN: 03785955. DOI: 10.1016/0378-5955(82)90062-4.

[35] J H van Hateren and A Van Der Schaaf. "Independent component filters of natural images compared with simple cells in primary visual cortex". In: *Proceedings of the Royal Society of London B* 265 (1998), pp. 359–366.

[36] Michael Häusser, Nelson Spruston, and Greg J Stuart. "Diversity and Dynamics Signaling of Dendritic Signalling". In: *Science* 290.5492 (2000), pp. 739–744. ISSN: 00368075. DOI: 10.1126/science.290.5492.739.

[37] Hermann von Helmholtz. *Treatise on Physiological Optics*. Dover Books on Physics. Dover Publications, 2013. ISBN: 9780486174709. URL: https://books.google.com/books?id=cSjEAgAAQBAJ.

[38] O J Hénaff et al. "The local low-dimensionality of natural images". In: *Int'l. Conf. on Learning Representations (ICLR2015)*. Available at http://arxiv.org/abs/1209.5006. San Diego, CA, May 2015. URL: http://arxiv.org/abs/1412.6626.

[39] Tomáš Hromádka et al. "Sparse Representation of Sounds in the Unanesthetized Auditory Cortex Figure S7 Neuronal responses are heterogeneous". In: *PLoS Biology* 6.1 (2008), pp. 4–5. ISSN: 1545-7885. DOI: 10.1371/journal.pbio.Citation.

[40] David H Hubel and Torsten N Wiesel. "Receptive fields and functional architecture of monkey striate cortex". In: *The Journal of physiology* 195.1 (1968), pp. 215–243.

[41] A James Hudspeth et al. *Principles of neural science*. McGraw-Hill, Health Professions Division, 2013.

[42] A. Hyvärinen and E. Oja. "Independent component analysis: Algorithms and applications". In: *Neural Networks* 13.4-5 (2000), pp. 411–430. ISSN: 08936080. DOI: 10.1016/S0893-6080(00)00026-5.

[43] Aapo Hyvärinen and Patrik O. Hoyer. "A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images". In: *Vision Research* 41.18 (2001), pp. 2413–2423. ISSN: 00426989. DOI: 10.1016/S0042-6989(01)00114-6.

[44] Aapo Hyvärinen, Patrik O Hoyer, and Mika Inki. "Topographic independent component analysis." In: *Neural computation* 13.7 (2001), pp. 1527–1558. ISSN: 0899-7667. DOI: 10.1162/089976601750264992. URL: http://www.ncbi.nlm.nih.gov/pubmed/11440596.

[45] Aapo Hyvärinen, Jarmo Hurri, and Patrik O Hoyer. *Natural image statistics*. London: Springer-Verlag, 2009. ISBN: 978-1-84882-490-4. DOI: 10.1007/978-1-84882-491-1. arXiv: arXiv:1011.1669v3. URL: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2941908%7B%5C&%7Dtool=pmcentrez%7B%5C&%7Drendertype=abstract.

[46] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.

[47] Kazuo Imaizumi et al. "Modular functional organization of cat anterior auditory field." In: *Journal of neurophysiology* 92.1 (2004), pp. 444–57. ISSN: 0022-3077. DOI: 10.1152/jn.01173.2003. URL: http://www.ncbi.nlm.nih.gov/pubmed/15014102.

[48] T Irino. "Time-domain, level dependent auditory lter: the gammachirp". In: *The Journal of the Acoustical Society of America* 101.January 1997 (1997), pp. 412–419. ISSN: 00014966. DOI: 10.1121/1.417975.

[49] Y. Karklin and E. P. Simoncelli. "Efficient coding of natural images with a population of noisy Linear-Nonlinear neurons". In: *Advances in Neural Information Processing Systems (NIPS)* (2011), pp. 1–9.

[50] Y Karklin and M Lewicki. "A Hierarchical Bayesian Model for Learning Nonlinear Statistical Regularities in Nonstationary Natural Signals". In: *Neural Computation* 17.2 (2005), pp. 397–423. ISSN: 0899-7667. DOI: 10.1162/0899766053011474. URL: http://www.mitpressjournals.org/doi/abs/10.1162/0899766053011474%7B%5C%%7D5Cnpapers3://publication/doi/10.1162/0899766053011474.

[51] Yan Karklin, Chaitanya Ekanadham, and Eero P Simoncelli. "Hierarchical spike coding of sound". In: *Adv. NIPS 25*. 2012, pp. 3041–3049. ISBN: 9781627480031. URL: http://books.nips.cc/papers/files/nips25/NIPS2012%7B%5C_%7D1393.pdf.

[52] Agnan Kessy, Alex Lewin, and Korbinian Strimmer. "Optimal whitening and decorrelation". In: *Statistics* May (2015), p. 12. DOI: 10.1080/00031305.2016.1277159. arXiv: 1512.00809. URL: http://arxiv.org/abs/1512.00809.

[53] Paul D King, Joel Zylberberg, and Michael R DeWeese. "Inhibitory interneurons decorrelate excitatory cells to drive sparse code formation in a spiking model of V1". In: *Journal of Neuroscience* 33.13 (2013), pp. 5475–5485.

[54] Paul D King, Joel Zylberberg, and Michael R DeWeese. "Inhibitory interneurons decorrelate excitatory cells to drive sparse code formation in a spiking model of V1". In: *Journal of Neuroscience* 33.13 (2013), pp. 5475–5485.

[55] David J. Klein, Peter König, and Konrad P. Körding. "Sparse Spectrotemporal Coding of Sounds". In: *Journal on Advances in Signal Processing* 2003 (2003), pp. 659–667. ISSN: 1687-6172. DOI: 10.1155/S1110865703303051.

[56] Simon Laughlin. "A Simple Coding Procedure Enhances a Neuron's Information Capacity". In: *Zeitschrift Fur Naturforschung C-a Journal of Biosciences* 36.9-10 (1981), pp. 910–912. ISSN: 0939-5075.

[57] Simon B Laughlin. "Energy as a constraint on the coding and processing of sensory information". In: *Current Opinion in Neurobiology* 11 (2001), pp. 475–480.

[58] Jonathan Le Roux et al. "Fast Signal Reconstruction Frommagnitude Stft Spectrogram Based on Spectrogram Consistency". In: *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10), Graz, Austria , September 6-10, 2010* (2010), pp. 1–7.

[59] Michael S. Lewicki. "Efficient coding of natural sounds". In: *Nature Neuroscience* 5.4 (2002), pp. 356–363. ISSN: 10976256. DOI: 10.1038/nn831. URL: http://www.nature.com/doifinder/10.1038/nn831.

[60] Tsung-Han Lin and Ping Tak Peter Tang. "Dictionary Learning by Dynamical Neural Networks". In: *CoRR* abs/1805.08952 (2018). arXiv: 1805.08952. URL: http://arxiv.org/abs/1805.08952.

[61] R F Lyon. "A computational model of filtering, detection, and compression in the cochlea". In: vol. 12821285. Paris, France, 1982, pp. 1282–1285.

[62] Richard F. Lyon, Andreas G. Katsiamis, and Emmanuel M. Drakakis. "History and future of auditory filter models". In: *ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems* (2010), pp. 3809–3812. DOI: 10.1109/ISCAS.2010.5537724.

[63] S. Lyu. "Dependency Reduction with Divisive Normalization: Justification and Effectiveness". In: *Neural Computation* 23.11 (2011), pp. 2942–2973. ISSN: 0899-7667.

[64] S Lyu and E P Simoncelli. "Nonlinear extraction of 'Independent Components' of natural images using radial Gaussianization". In: *Neural Computation* 21.6 (June 2009), pp. 1485–1519. DOI: 10.1162/neco.2009.04-08-773.

[65] Stéphane Mallat and Zhifeng Zhang. *Matching pursuit with time-frequency dictionaries*. Tech. rep. Courant Institute of Mathematical Sciences New York United States, 1993.

[66] D Marr and T Poggio. *From Understanding Computation to Understanding Neural Circuitry*. 1976. DOI: AIM-357.

[67] Martin Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Tech. rep. 2015. URL: http://tensorflow.org/.

[68] MathWorks. *MATLAB 2016b*. Tech. rep. Natick, Massachusetts, 2016.

[69] M M Merzenich et al. "Representation of cochlea within primary auditory cortex in the cat." In: *Journal of neurophysiology* 38.2 (1975), pp. 231–249. ISSN: 0022-3077. DOI: 10.1121/1.1920046. URL: http://ftp.utdallas.edu/%7B~%7Dkilgard/02a%20merzenich%20A1%20tonotopy%201975.pdf.

[70] Lee M Miller et al. "Spectrotemporal receptive fields in the lemniscal auditory thalamus and cortex." In: *Journal of Neurophysiology* 87.1 (2002), pp. 516–527. ISSN: 0022-3077.

[71] Wiktor Młynarski. "The Opponent Channel Population Code of Sound Location Is an Efficient Representation of Natural Binaural Sounds". In: *PLoS Computational Biology* 11.5 (2015), pp. 1–31. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1004294.

[72] Wiktor Młynarski and Josh H McDermott. "Learning Mid-Level Auditory Codes from Natural Sound Statistics". In: (2017), pp. 1–26. arXiv: 1701.07138v2.

[73] Bruno A Olshausen. "Highly overcomplete sparse coding". In: *IS&T/SPIE Electronic Imaging* 8651.510 (2013), 86510S–86510S–9. ISSN: 0277786X. DOI: 10.1117/12.2013504. URL: http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1668827%7B%5C%%7D5Cnhttp://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2013504%7B%5C%%7D5Cnhttp://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1668827.

[74] Bruno A. Olshausen and David J. Field. "Sparse coding of sensory inputs". In: *Current Opinion in Neurobiology* 14.4 (2004), pp. 481–487. ISSN: 09594388. DOI: 10.1016/j.conb.2004.07.007.

[75] Bruno A. Olshausen and David J. Field. "Sparse coding with an overcomplete basis set: A strategy employed by V1?" In: *Vision Research* 37.23 (1997), pp. 3311–3325. ISSN: 00426989. DOI: 10.1016/S0042-6989(97)00169-7.

[76] Bruno A Olshausen and David J Field. "Emergence of simple-cell receptive field properties by learning a sparse code for natural images." In: *Nature* 381.6583 (1996), pp. 607–609. ISSN: 0028-0836. DOI: 10.1038/381607a0.

[77] Bruno A Olshausen and Michael S Lewicki. "What Natural Scene Statistics Can Tell Us about Cortical Representation". In: *The New Visual Neurosciences*. Ed. by John S Werner and Leo M Chalupa. 2013, p. 26. ISBN: 9780262019163. URL: https://redwood.berkeley.edu/bruno/public/TNVN-chapter.pdf.

[78] F Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[79] Xaq Pitkow and Markus Meister. "Decorrelation and efficient coding by retinal ganglion cells". In: *Nature Neuroscience* 15.4 (2012), pp. 628–635. ISSN: 1546-1726. DOI: 10.1038/nn.3064.

[80] Daniel B Polley et al. "Multiparametric auditory receptive field organization across five cortical fields in the albino rat". In: *Journal of neurophysiology* 97.5 (2007), pp. 3621–3638. ISSN: 0022-3077. DOI: 10.1152/jn.01298.2006.

[81] Anqi Qiu. "Gabor Analysis of Auditory Midbrain Receptive Fields: Spectro-Temporal and Binaural Composition". In: *Journal of Neurophysiology* 90.1 (2003), pp. 456–476. ISSN: 0022-3077. DOI: 10.1152/jn.00851.2002. URL: http://jn.physiology.org/cgi/doi/10.1152/jn.00851.2002.

[82] Martin Rehn and Friedrich T. Sommer. "A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields". In: *Journal of Computational Neuroscience* 22.2 (2007), pp. 135–146. ISSN: 0929-5313. DOI: 10.1007/s10827-006-0003-9. URL: http://link.springer.com/10.1007/s10827-006-0003-9.

[83] Fred Rieke et al. *Spikes: Exploring the Neural Code*. MIT Press, 1997.

[84] Dario Ringach and Robert Shapley. "Reverse correlation in neurophysiology". In: *Cognitive Science* 28.2 (2004), pp. 147–166. ISSN: 03640213. DOI: 10.1016/j.cogsci.2003.11.003.

[85] Francisco A Rodríguez et al. "Neural modulation tuning characteristics scale to efficiently encode natural sound statistics." In: *J Neurosci* 30.47 (2010), pp. 15969–15980. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.0966-10.2010. URL: http://dx.doi.org/10.1523/JNEUROSCI.0966-10.2010.

[86] Christopher J Rozell et al. "Sparse coding via thresholding and local competition in neural circuits." In: *Neural Computation* 20.10 (2008), pp. 2526–63. ISSN: 08997667. DOI: 10.1162/neco.2008.03-07-486. URL: http://www.ncbi.nlm.nih.gov/pubmed/18439138%7B%5C%%7D5Cnhttp://redwood.berkeley.edu/bruno/papers/rozell-sparse-coding-nc08.pdf.

[87] N. C. Rust and J. J. DiCarlo. "Balanced increases in selectivity and invariance produce constant sparseness across the ventral visual pathway". In: *Journal of Vision* 9.8 (2009), p. 738.

[88] N. C. Rust and J. J. DiCarlo. "Balanced Increases in Selectivity and Tolerance Produce Constant Sparseness along the Ventral Visual Stream". In: *Journal of Neuroscience* 32.30 (2012), pp. 10170–10182.

[89] Elad Schneidman et al. "Weak pairwise correlations imply strongly correlated network states in a neural population". In: *Nature* 440.7087 (2006), pp. 1007–1012.

[90] Odelia Schwartz and Eero P Simoncelli. "Natural signal statistics and sensory gain control." In: *Nature neuroscience* 4.8 (2001), pp. 819–25. ISSN: 1097-6256. DOI: 10.1038/90526. URL: http://www.ncbi.nlm.nih.gov/pubmed/11477428.

[91] Honghao Shan, Matthew H. Tong, and Garrison W. Cottrell. "A Single Model Explains both Visual and Auditory Precortical Coding". In: *ArXiv e-prints* (2016), pp. 1–32. arXiv: 1602.08486. URL: http://arxiv.org/abs/1602.08486.

[92] Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Communication*. Urbana and Chicago: University of Illinois Press, 1949.

[93] Samuel Shapero et al. "Optimal Sparse Approximation With Integrate and Fire Neurons". In: *International Journal of Neural Systems* 24.05 (2014), p. 1440001. ISSN: 0129-0657. DOI: 10.1142/S0129065714400012. URL: http://www.worldscientific.com/doi/abs/10.1142/S0129065714400012.

[94] Eero P Simoncelli and Bruno A Olshausen. "Natural image statistics and neural representation". In: *Annual Review of Neuroscience* 24 (2001).

[95] Nandini C Singh and Frédéric E Theunissen. "Modulation spectra of natural sounds and ethological theories of auditory processing." In: *J Acoust Soc Am* 114.6 Pt 1 (2003), pp. 3394–3411. ISSN: 00014966. DOI: 10.1121/1.1624067.

[96] Malcom Slaney. *Auditory toolbox version 2*. Tech. rep. Interval Research Corporation, 1998.

[97] Evan C. Smith and Michael S. Lewicki. "Efficient auditory coding". In: *Nature* 439.7079 (2006), pp. 978–982. ISSN: 0028-0836. DOI: 10.1038/nature04485. URL: http://www.nature.com/doifinder/10.1038/nature04485.

[98] Evan Smith and Michael S. Lewicki. "Efficient coding of time-relative structure using spikes." In: *Neural Computation* 17.1 (2005), pp. 19–45. ISSN: 0899-7667. DOI: 10.1162/0899766052530839.

[99] G Stuart et al. "Action potential initiation and back propagation in neurons of the mammalian central nervous system". In: *Trends in Neurosciences* 20.3 (1997), pp. 125–131. ISSN: 0166-2236. DOI: 10.1016/S0166-2236(96)10075-8. URL: http://discovery.ucl.ac.uk/21791/.

[100] Hiroki Terashima and Masato Okada. "The topographic unsupervised learning of natural sounds in the auditory cortex". In: *Advances in Neural Information Processing Systems* (2012), pp. 1–9. ISSN: 10495258.

[101] Lucas Theis, Jascha Sohl-Dickstein, and Matthias Bethge. "Training sparse natural image models with a fast Gibbs sampler of an extended state space". In: *Advances in Neural Information Processing Systems 25* (2012), pp. 1133–1141. ISSN: 10495258. URL: http://books.nips.cc/papers/files/nips25/NIPS2012%7B%5C_%7D0540.pdf.

[102] Frédéric E. Theunissen. "From synchrony to sparseness". In: *Trends in Neurosciences* 26.2 (2003), pp. 61–64. ISSN: 01662236. DOI: 10.1016/S0166-2236(02)00016-4. arXiv: S0166-2236(02)00016-4.

[103] Frédéric E Theunissen and Julie E Elie. "Neural processing of natural sounds." In: *Nature reviews. Neuroscience* 15.6 (2014), pp. 355–66. ISSN: 1471-0048. DOI: 10.1038/nrn3731. URL: http://www.ncbi.nlm.nih.gov/pubmed/24840800.

[104] William E Vinje and Jack L Gallant. "Sparse Coding and Decorrelation in Primary Visual Cortex During Natural Vision". In: *Science* 287.5456 (2000), pp. 1273–1276. ISSN: 00368075. DOI: 10.1126/science.287.5456.1273. URL: http://www.sciencemag.org/cgi/doi/10.1126/science.287.5456.1273.

[105] Michael Weliky et al. "Coding of Natural Scenes in Primary Visual Cortex". In: *Neuron* 37.4 (2003), pp. 703–718. ISSN: 0896-6273 (Print). URL: http://www.sciencedirect.com/science/article/B6WSS-4CC2YG4-4F/2/2f2faac6d71976420ea1be0fea5567ef.

[106] Ben D B Willmore, James A Mazer, and Jack L Gallant. "Sparse coding in striate and extrastriate visual cortex". In: *J Neurophysiol* April 2011 (2011), pp. 2907–2919. ISSN: 1522-1598. DOI: 10.1152/jn.00594.2010..

[107] Xuedong Zhang et al. "A phenomenological model for the responses of auditory-nerve fibers: I. Nonlinear tuning with compression and suppression". In: *The Journal of the Acoustical Society of America* 109.2 (2001), pp. 648–670.

[108] Mengchen Zhu and Christopher J. Rozell. "Visual Nonclassical Receptive Field Effects Emerge from Sparse Coding in a Dynamical System". In: *PLoS Computational Biology* 9.8 (2013), pp. 1–15. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1003191.

[109] Joel Zylberberg and Michael Robert DeWeese. "Sparse coding models can exhibit decreasing sparseness while learning sparse codes for natural images". In: *PLoS computational biology* 9.8 (2013), e1003182.

[110] Joel Zylberberg, Jason Timothy Murphy, and Michael Robert DeWeese. "A sparse coding model with synaptically local plasticity and spiking neurons can account for the diverse shapes of V1 simple cell receptive fields". In: *PLoS Computational Biology* 7.10 (2011), pp. 1–33. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1002250. arXiv: 1109.2239.