Temporal prediction as a principle underlying sensory representations



Yosef Singer

St Catherine's College, University of Oxford

Thesis submitted for the degree of Doctor of Philosophy

Trinity 2019

Abstract

Neurons in sensory pathways respond selectively to specific features of dynamic natural inputs. For example, in primary auditory cortex (A1), neurons respond selectively to sounds with specific spectral and temporal content. Retinal neurons prefer localized spots of light, primary visual cortical (V1) neurons moving bars, and those in higher cortical areas, such as middle temporal (MT) cortex, favour complex features like moving textures. Understanding the principles behind these specific forms of selectivity is a central aim of systems neuroscience. In this thesis I explore the idea that a key goal of sensory systems is to represent features of natural inputs that are efficiently predictive of future inputs. First, I show that a single hidden-layer artificial neural network that is optimized to predict the future of natural sound inputs from their past, produces receptive fields (RFs) with spectrotemporal tuning properties that both qualitatively and quantitatively resemble those of A1 neurons. Furthermore, I show that the better a network predicts the future, the more the receptive fields of its units resemble those recorded in A1. Next, I show that when the identical model is trained to predict the future of natural video inputs rather than sounds, its RFs resemble those of V1 simple cells in both their spatial and temporal tuning properties. Finally, I extend the temporal prediction model to a hierarchical form and show that it recovers the tuning properties of neurons as they change from the retina to higher visual cortex. The fact that these simple neural network models, when optimized to efficiently predict the immediate future of natural inputs, display the tuning properties of sensory neurons at so many levels of the visual and auditory hierarchies suggests that this principle may be a fundamental goal of sensory systems.

Acknowledgements

These past four years have been far more challenging than I ever would have imagined. They have also been amongst the best years of my life and I have many people to thank for that.

Firstly, to Nicol, Ben and Andy. I am so grateful for your supervision. Your endless patience and willingness to teach; the rigor and attention to detail with which you approach science; your far-reaching knowledge of sensory systems, computational models and many other things besides, have given me much to look up to and made me extremely lucky to have you as mentors. Thank you for the countless hours you have put into every aspect of the work that has resulted in this thesis. I'd also like to thank everyone in the ANG for your support, feedback and discussions that have helped to shape this project into what it is and made the lab a pleasant place to work.

I would never have been here in the first place if it had not been for Neville Passmore and Clare Digby. Raising the funding to study in Oxford was a daunting task when coming from South Africa. Thank you for helping me through the process. I am also extremely grateful to all of the organizations that funded me throughout my studies in Oxford – The Clarendon Fund, Skye Foundation, Oppenheimer Memorial Trust, Wits University and the Goodger & Schorstein Scholarship.

I have loved my time in Oxford and that is largely due to the incredible friends that I have made here. Alon, you have been a voice of sanity and an endless source of fun. Merton BOPs, floating down the river and many other times hanging out together form some of my best memories. I am truly lucky to have you as a friend. Tim, I have loved our conversations about the best places to shelter up in Oxford in the case of a zombie apocalypse (Magdalen, obviously), life, science and everything in between. Callum, thank you for the weekly beers,

(almost as frequent) cycles and many braais together. Luca, Liana, Alex, thank you for amazing times at the Blue Pearl and beyond - you gave me a family in Oxford.
Thank you to Caroline, Rory, Ben, Gilad, Hector, Ainslie, who have made these years in Catz and Oxford as amazing as they have been.

And I would like to thank Sarah. You have made my life immeasurably better by being a part of it.

Finally, I would like to thank my family. Your love and support have carried me through.

Contents

Chapter 1 | Introduction *1*

1.1 Representations in the visual system **1**

1.2 Representations in the auditory system **5**

1.3 Phenomenological models of sensory neurons **7**

1.4 Normative models of visual processing **9**

1.4.1 Efficient coding 9
1.4.2 Predictive coding 10
1.4.3 Sparse coding 11
1.4.4 Independent component analysis 13
1.4.5 Temporal coherence/slowness 13

1.5 Normative models of auditory processing **14**

1.6 Temporal prediction **15**

1.6.1 Motivation for temporal prediction 15
1.6.2 Predictive information bottleneck 16
1.6.3 The temporal prediction model 17
1.6.4 Temporal prediction and predictive coding 19
1.6.5 Incorporating time into normative models 20

1.7 Relationship to machine learning models **22**

1.8 Thesis overview **26**

1.9 Publications **26**

Chapter 2 | Temporal prediction model of natural sounds reproduces properties of primary auditory cortical neurons *27*

2.1 The temporal prediction model **27**

***2.2* Qualitative assessment of auditory receptive fields** **30**

***2.3* Qualitative comparison to other models** **31**

***2.4* Quantitative analysis of auditory receptive fields** **36**

2.5 Optimizing predictive capacity **39**

2.6 Variants of the temporal prediction model **42**

2.7 Discussion **43**

2.7.1 Relationship to other models 44
2.7.2 Auditory normative models 45
2.7.3 Conclusion 46

2.8 Methods **47**

2.8.1 Data used for model training and testing 47
2.8.2 Temporal prediction model 49
2.8.3 Sparse coding model 52
2.8.4 Auditory receptive field analysis 53

Chapter 3 | Temporal prediction model of natural video inputs reproduces properties of V1 simple cells *57*

3.1 **Qualitative assessment of visual receptive fields** ... **58**
3.2 **Qualitative comparison to other models** ... **59**
3.3 **Quantitative analysis of visual RFs** ... **64**
3.4 Variants of the temporal prediction model ... **67**
3.5 Discussion ... **68**
3.5.1 Relationship to other models ... 69
3.5.2 Critical evaluation of the temporal prediction model ... 72
3.6 Methods ... **75**
3.6.1 Data used for model training and testing ... 75
3.6.2 Temporal prediction model ... 76
3.6.3 Sparse coding model ... 76
3.6.4 Visual receptive field analysis ... 77
Chapter 4 | Hierarchical temporal prediction captures motion processing from retina to higher visual cortex ... ***82***
4.1 Introduction ... **82**
4.2 Results ... **84**
4.2.1 The hierarchical temporal prediction model ... 84
4.2.2 Temporal prediction of natural inputs produces retinal-like units ... 86
4.2.3 Hierarchical temporal prediction produces simple and complex cell tuning ... 87
4.2.4 Model units are tuned to two-dimensional features of visual motion ... 93
4.3 Discussion ... **97**
4.4 Methods ... **102**
4.4.1 Data used for model training and testing ... 102
4.4.2 Hierarchical temporal prediction model ... 103
4.4.3 Model unit spatiotemporal extent and receptive fields ... 106
4.4.4 In vivo V1 receptive field data ... 107
4.4.5 Receptive field size and polarity ... 107
4.4.6 Drifting sinusoidal gratings ... 108
4.4.7 Measuring end-stopping ... 110
4.4.8 Sparse noise stimuli and two-bar maps ... 110
4.4.9 Drifting plaid stimuli ... 111
Chapter 5 | Discussion ... ***112***
5.1 Summary of results ... **112**
5.2 Relating normative frameworks ... **112**
5.3 Strengths of the temporal prediction model ... **113**
5.4 Comparing computational models and making experimental predictions ... **114**
5.5 Limitations of the temporal prediction model and future developments ... **117**
5.5.1 Recurrent connectivity ... 117
5.5.2 Adding linear or non-linear processing stages to model layers ... 120
5.5.3 Using spike codes instead of rate codes ... 120
5.5.4 Modifying assumptions about the conditional distribution of futures ... 121
5.5.5 Time-scales of prediction ... 122
5.5.6 Using more biologically realistic learning rules ... 123
5.5.7 Incorporating other goals into the model ... 123
5.5.8 Modelling other sensory areas ... 125
5.6 Conclusion ... **127**
References ... ***128***

Supplementary figures..... *147*

Chapter 1 | Introduction

Neurons in successive stages of the brain's sensory pathways respond selectively to increasingly abstract features in their respective modalities and, ultimately, extract behaviourally relevant information from complex, dynamic scenes. At each stage of these hierarchies, neural representations take specific forms that are widely preserved across individuals and species. Examining whether there are general principles that can explain why these representations take the exact forms that they do has been the topic of much investigation in theoretical neuroscience (Simoncelli and Olshausen, 2001). In this thesis I examine the hypothesis that sensory neurons are optimized to represent features in the recent past that best predict the immediate future input.

1.1 Representations in the visual system

The earliest stage of the visual hierarchy is the retina. Here, visual signals are detected by photoreceptors and are fed forward through bipolar cells and other neurons until they reach retinal ganglion cells (RGCs), whose axons in the optic nerve synapse onto neurons in the lateral geniculate nucleus (LGN). From the LGN, visual information passes to primary visual cortex (V1). In primates, there are three main subcortical pathways – the magnocellular (M), parvocellular (P) and koniocellular (K) pathways (explained in more detail below) – from the retina to V1. From V1, visual signals pass to higher cortical areas. These areas are organized into two parallel processing streams – the dorsal and ventral stream. The dorsal stream receives most inputs from M-dominated V1 signals, while the ventral stream receives

from P-dominated V1 signals (Nassi and Callaway, 2009). It should be noted however, that there is extensive mixing of M and P signals both within V1 and higher cortical areas (Nassi and Callaway, 2009, 2006). Nevertheless, each pathway is thought to have a distinct functional role, with the dorsal stream being responsible for processing features relating to motion and location (the 'where' stream) and the ventral pathway responsible for processing features relating to the identity and form of objects (the 'what' stream) (Ungerleider and Mishkin, 1982), although some studies suggest different functional roles for these pathways (Goodale and Milner, 1992; Rauschecker, 2018). Each stream is hierarchically organized, but with many reciprocal feedforward and feedback connections between levels, as well as extensive interactions between dorsal and ventral areas (Felleman and Van Essen, 1991). The main stages of the dorsal pathway are the middle temporal area (MT, also referred to as V5), which receives most of its input directly from V1 (Nassi and Callaway, 2009, 2006; Zeki, 1974), the medial superior temporal area (MST) and the posterior parietal area (PP) (Van Essen and Gallant, 1994). The main stages of the ventral pathway are V2, V4 and inferior temporal cortex (IT) (Van Essen and Gallant, 1994). Here I will provide a brief overview of some features of the representations used by the subsets of neurons in the visual areas that are most relevant for this thesis – namely the pathway from the eye through V1 and onto the dorsal pathway.

A concept that has been commonly used to understand neuronal representations is the receptive field (RF). A neuron's RF characterizes the mapping between a stimulus and the response it evokes in the neuron. The RF is often defined as a linear filter, and can be determined by presenting a large array of white noise inputs and calculating the correlation between the neural response and the preceding stimulus (Chichilnisky, 2001). Thus, in the visual system, the spatiotemporal RF can be seen as the mean linear combination of inputs over space and recent time that causes a neuron to spike (spike triggered average; STA). The

linear RF may also be determined by other methods, such as performing a linear regression between inputs and the neuronal responses they elicit. Neurons in early visual areas, such as RGCs, LGN neurons and V1 simple cells, have RFs with characteristic spatial and temporal profiles that provide moderately good descriptions of their selectivity for stimulus features. Neurons in later stages of the visual hierarchy, such as V1 complex cells and neurons in higher visual areas, require more nonlinear models to describe their selectivity for stimulus features.

RGCs and neurons in the LGN are selective for visual stimuli with a bright center and dark surround (ON cells; Fig. 1.1a) or a dark center and a bright surround (OFF cells). Parasol and midget cells comprise 80% of the population of RGCs in primates and synapse onto neurons in the magnocellular (M) and parvocellular (P) pathways respectively (Nassi and Callaway, 2009). A third subcortical pathway (the koniocellular pathway) receives most of its inputs from bistratified RGCs and projects in parallel to V1 but is less well understood (Casagrande, 1999; Nassi and Callaway, 2009). M cells tend to have large RFs with high temporal resolution that change polarity over time, while P cells have smaller RFs with low temporal resolution that do not change polarity. A similar division into X and Y cells exists in cats (Shapley and Perry, 1986) and other species (Price and Morgan, 1987).

In V1, two main functional cell types are simple and complex cells (Hubel and Wiesel, 1968, 1965, 1962, 1959). These neurons respond optimally to oriented line segments, such as drifting bars. Their response properties are commonly measured using drifting sinusoidal gratings (Movshon et al., 1978a, 1978b). Simple and complex cells are selective for orientation, bar width (spatial frequency), drift rate (temporal frequency) and spatial extent and some are tuned for direction of motion. Simple cells have spatial RFs with distinct bright (ON) and dark (OFF) regions (Fig. 1.1b,c) and will respond to a drifting bright or dark bar only when it overlaps with the corresponding region of the RF.

In addition to their spatial features, the RFs of RGCs, LGN neurons and V1 simple cells have characteristic temporal profiles, concentrating most of their power at time points near the present and decaying into the past (Fig. 1.1d-g). The RFs of V1 simple cells may switch polarity over time (Fig. 1.1e,f) or shift smoothly over space, as in the case of direction-selective neurons (Fig. 1.1g).

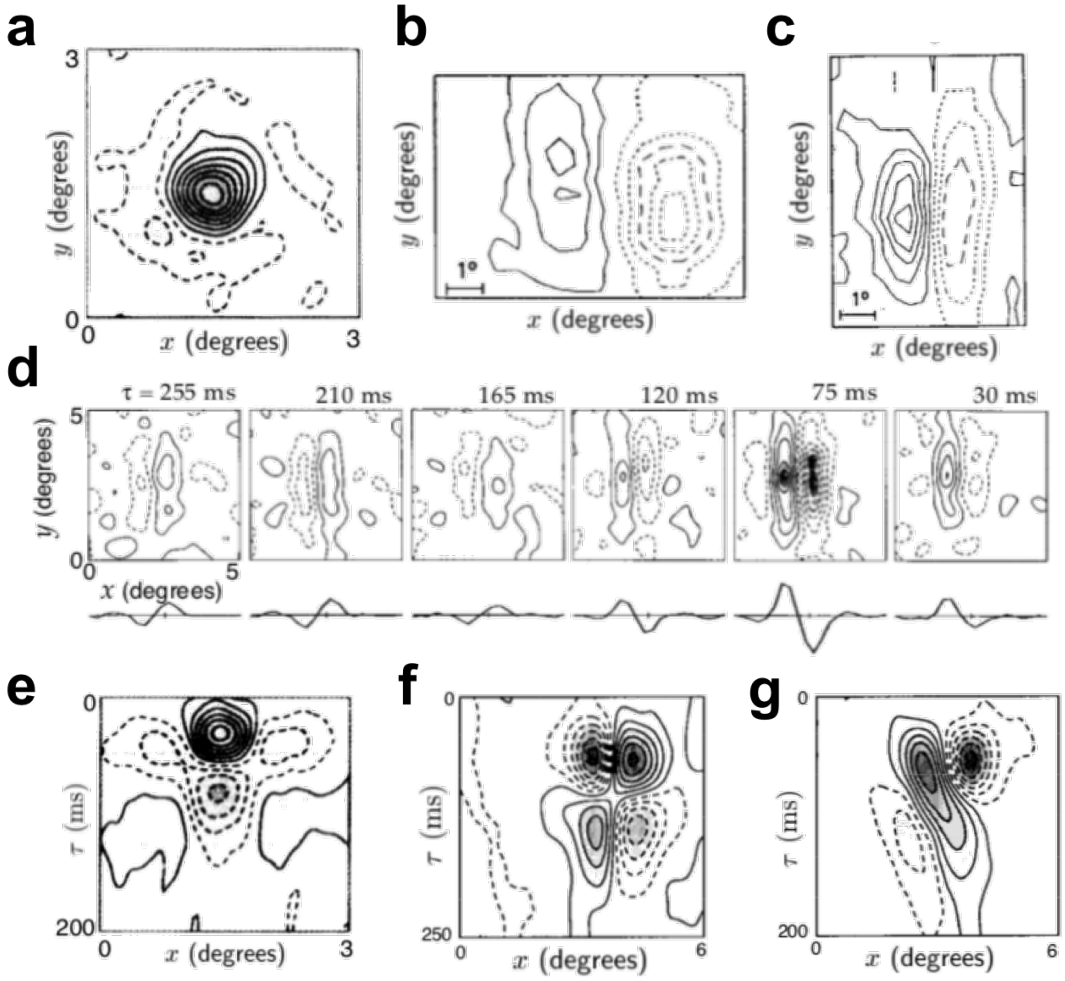


Figure 1.1 | Example LGN and V1 simple cell RFs. In all panels, solid contour lines indicate ON (bright) regions and dashed contour lines indicate OFF (dark) regions. **a**, Spatial RF of an example ON-center LGN neuron. **b**, **c**, Spatial RFs of example V1 simple cells. **d**, 3-dimensional (space-space-time) spatiotemporal RF of example simple cell, where τ indicates time prior to neuronal response. In each time step, the spatial RF is shown and below it the same RF summed over the 'y' dimension. 2-dimensional spatiotemporal (space-time) RFs can be generated by summing along the 'y' spatial dimension, which lies parallel to the orientation of the RF. **e-g**, 2-dimensional (space-time) spatiotemporal RFs of example ON-center LGN neuron (e), non-direction-selective simple cell (f)

and direction-selective simple cell (g). Adapted from Dayan and Abbott (2001), **a**,**d-g** originally from DeAngelis et al. (1995), **b**,**c** originally from Jones and Palmer (1987a).

Complex cells typically lack spatial structure in their RFs (they do not have distinct ON and OFF regions) and will respond to a drifting bar regardless of where it is within the RF (Hubel and Wiesel, 1968, 1962). Thus, for complex cells and for neurons in higher visual areas, the linear RF no longer accurately describes neuronal selectivity for stimulus features. With complex cells selectivity for these stimulus features may be estimated with drifting bars or sinusoidal gratings. A similar strategy is employed in MT, where response selectivity may be measured using drifting gratings, plaids or moving dots. Here, most neurons are selective for direction of motion (Wang and Movshon, 2016). Unlike cells in V1, many MT neurons signal relatively global rather than local features of motion (Movshon et al., 1985), and are unaffected by the aperture problem. The aperture problem occurs when an extended edge is viewed through a small aperture. Here, the direction of motion of the edge is ambiguous and only components of motion perpendicular to the orientation of the edge are distinguishable. In contrast to V1 neurons, MT cells are able to signal the direction of motion of a complex stimulus such as a drifting plaid (composed of two superimposed sinusoidal gratings) unambiguously. These neurons therefore represent a higher processing level in the dorsal visual motion pathway. This is examined in greater detail in Chapter 5.

1.2 Representations in the auditory system

In the auditory system sounds are converted from pressure waveforms into neuronal signals in the inner ear. Here the cochlea decomposes sounds into their constituent frequency components by means of the basilar membrane. This long membrane is stiff at the base, where it extracts high frequency components. As one moves towards the apex it becomes less stiff and hence extracts progressively lower sound frequency components. Receptor

cells known as hair cells form part of the Organ of Corti, which sits on top of this membrane. For each frequency-specific region of the basilar membrane, the overlying hair cells transduce the vibrations, changing their membrane potential, which modulates the release of excitatory neurotransmitter at the synapse with fibres of the auditory nerve. In turn, the auditory nerve transmits this frequency decomposed (spectrotemporal) representation of sound to the brain. The output of the cochlea is processed through a series of nuclei in the brainstem before arriving at the inferior colliculus (IC) in the midbrain. The IC projects to the medial geniculate body in the thalamus and from there, signals pass to primary auditory cortex (A1).

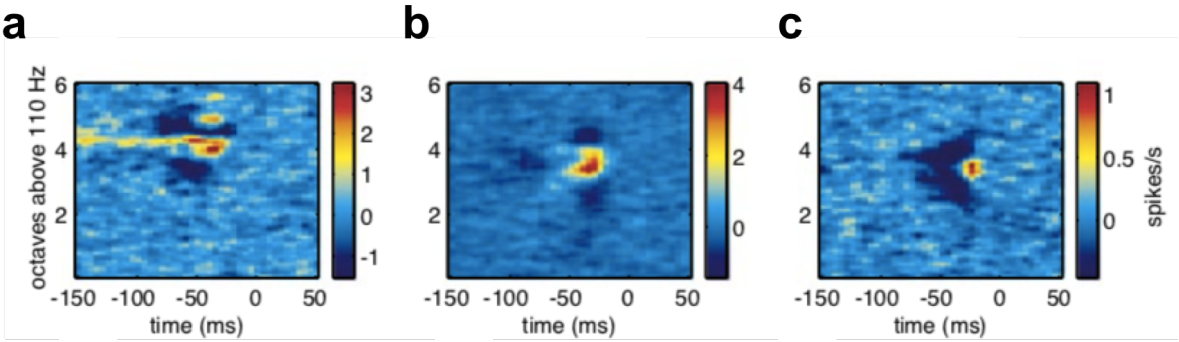


Figure1.2 | Example spectrotemporal receptive fields of neurons recorded in primary auditory cortex of primates. The profile of each receptive field shows one or more excitatory region, surrounded by flanking (**a-c**) and/or lagging (**b-c**) inhibitory regions. Red and yellow-excitation, light blue-zero (i.e. no change), dark blue-inhibition. Adapted from deCharms et al. (1998).

Spectrotemporal RFs are the linear approximation of the mapping from the spectrotemporal representation of the sound (the spectrogram or 'cochleagram') to the neural response and can be measured by the spike triggered average or other methods. They can be seen as the pattern of sound that best excites (or inhibits) the neuron (under linear assumptions). Most studies of the auditory system have focused on neurons in the IC and A1, where spectrotemporal RFs typically consist of an excitatory region around a 'best frequency' surrounded by one or more inhibitory regions, although neurons with more than

one excitatory region are also common (deCharms et al., 1998; King and Nelken, 2009; Schnupp et al., 2001; Fig. 1.2). RFs with lagging inhibition (Fig. 1.2 b-c) are most common, but many neurons also display flanking inhibition (Fig. 1.2 a-c) or more disordered profiles (Carlin and Elhilali, 2013). RFs of IC and A1 neurons also have characteristic temporal profiles with most energy being concentrated in the most recent time steps and decaying into the past.

1.3 Phenomenological models of sensory neurons

Phenomenological models (sometimes also referred to as functional models) provide a concise description of the transformation between stimulus and neural response. These models are typically fitted to recorded neural data with the aim of accurately predicting neurons' firing rates in response to arbitrary stimuli. The linear-nonlinear (LN) model has been widely used for this purpose (Carandini et al., 2005; Simoncelli et al., 2004). Here, a neuron's firing rate in response to inputs is described by a cascade of processing stages. A linear stage describes the average stimulus that drives the neuron (i.e. its linear RF). The linear stage is followed by a static nonlinear function such as a sigmoid, which accounts for other aspects of the neuron's response sensitivity, including response threshold or saturation. LN models are commonly used to describe the response properties of RGCs, LGN neurons and V1 simple cells (Fig. 1.3a,b), as well as auditory neurons in the IC and A1. As one moves up the visual hierarchy, response properties become increasingly complex and can no longer be well approximated by simple LN models and therefore require other linear or nonlinear stages. For example, the energy model of complex cells (Adelson and Bergen, 1985) involves squaring and summing the outputs of two phase-shifted linear subunits (Fig. 1.3c). In order to describe the tuning of pattern-selective MT cells, functional models employ further LN stages. One commonly used model, developed by Simoncelli and Heeger (1998)

involves applying a divisive normalization step to the outputs of many direction-tuned complex cells (each of which incorporates the energy model outlined above). These normalized outputs then pass through further linear and nonlinear stages to produce the firing rate outputs of MT pattern selective cells (Fig. 1.3d).

In some cases, the linear portion of these models may be well approximated with a mathematical function. Examples of this are the difference of Gaussians model (Marr and Hildreth, 1980), which describes the center-surround tuning of RFs in the retina and LGN and the Gabor filter, which approximates simple cell RFs (Jones and Palmer, 1987b). The advantage of using such functions is that the RF can be well described using very few parameters.

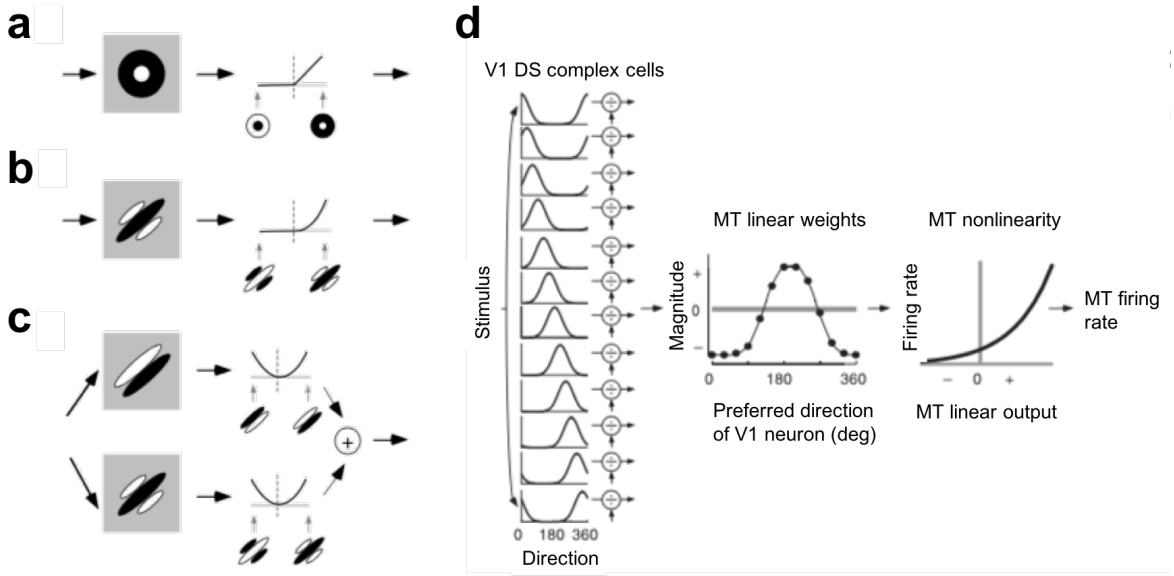


Figure 1.3 | Example phenomenological models of visual neurons. a, Model of RGC or LGN neuron. The linear stage has the characteristic center-surround tuning of these neurons and is followed by a thresholding nonlinearity. **b,** Model of V1 simple cell. The linear stage resembles a Gabor filter and is followed by a half-wave rectifying nonlinearity. **c**, Energy model of complex cell. There are two linear filters that are shifted in phase relative to each other. The outputs of these filters are then squared and summed. **d,** A version of the Simoncelli-Heeger model (Simoncelli and Heeger, 1998) of a pattern selective MT neuron. The input is passed through a number of direction-tuned complex cells. The response of each complex cell is divisively normalized. The outputs of the

complex cells are then linearly combined using a weighting function and pass through an instantaneous nonlinear function to produce a firing rate. **a-c** adapted from Carandini et al. (2005), **d** adapted from Rust et al. (2006).

1.4 Normative models of visual processing

Unlike phenomenological models, normative models are not fitted to neural data but instead aim to derive the form of sensory representations from a set of first principles. They aim to explain not only how information is represented in a given region of the brain, but why neurons respond to the specific features that they do. This typically starts with an examination of the statistics of natural inputs. A clear set of optimization criteria can then be set by defining a goal (or set of goals) that a set of sensory neurons is thought to fulfil given the statistical properties of the inputs (Simoncelli and Olshausen, 2001). The explanatory power of these principles can then be evaluated by optimizing a network model to fulfil these criteria given a set of natural inputs and examining whether the model produces a similar transformation of the input to that performed by real sensory neurons. Here we will present a brief overview of a few key normative models of visual processing.

1.4.1 Efficient coding

Attneave (1954) and Barlow (1961) suggested that the goal of the early sensory neurons is to represent natural inputs efficiently by eliminating statistical redundancy. This idea has formed the theoretical basis for several normative frameworks including predictive coding and sparse coding.

1.4.2 Predictive coding

Predictive coding provides a concrete interpretation of the efficient coding hypothesis. It puts forward the idea that sensory neurons reduce statistical redundancy, which is present in natural scenes, by removing predictable elements of signals and feeding forward only unpredicted elements (Huang and Rao, 2011; Rao and Ballard, 1999; Srinivasan et al., 1982).

Srinivasan et al. (1982) proposed this framework to explain the center-surround structure of retinal neurons' spatial and temporal RFs. Given that natural scenes are correlated over short distances, the intensity value at the center of a retinal RF may be predicted by a linearly weighted sum of intensity values in the surrounding region. By transmitting only the residual error between the predicted value at the center and the measured value, retinal neurons may reduce redundancy in their outputs. Similarly, given the temporal correlations in natural scenes, redundancy may be reduced by transmitting only the difference between the current intensity value and the value predicted by a linearly weighted sum of past values. Srinivasan et al. (1982) show that a filter that does this optimally exhibits similar spatial and temporal profiles to those of retinal neurons, including their characteristic lagging inhibition.

Rao and Ballard (1999) extended this framework to a hierarchical form to account for processing in V1. In their model, neurons in a higher layer estimate the activity they expect to receive from neurons in the layer below. These estimates are then fed back to the lower layer using top-down connections. Neurons in the lower layer subtract the top-down predictions from their outputs, feeding forward only the residuals (prediction errors). Thus, each layer forms a set of basis vectors whose joint activity predicts the activities of the units in the layer below, with the very bottom layer predicting the input. Rao and Ballard (1999) trained a three-layer hierarchical predictive coding model on thousands of natural image patches. Examining the receptive fields of the model units after training, they found that

some units in the lowest layer had RFs with oriented tuning resembling V1 simple cells. Many units also exhibited end-stopping, responding most strongly to short bars and less to longer bars.

1.4.3 Sparse coding

Sparse coding also stems from the efficient coding framework. It postulates that early sensory areas are optimized to accurately represent current or past inputs using a sparse, distributed code. In this scheme, all neurons in the code have an equal probability of being active over the full class of natural images, but a low probability of being active for any one image (Field, 1994). Here, especially when an overcomplete code is used (that is, when there are more neurons than input dimensions), it is efficient despite not having a reduction in the dimensionality of the code because only few of those neurons are active to represent any given image.

In their pioneering study, Olshausen and Field (1996) built a model of V1 simple cells based on this idea. They trained a generative model with a single overcomplete layer of linear units to accurately reconstruct thousands of natural image patches using a sparse code. Their generative model (Fig. 1.4a) assumes that inputs can be explained by means of a linear superposition of basis functions weighted by a set of amplitudes (activations). For any given input, the model undergoes a recurrent settling process whereby it determines the set of activations that, when multiplied by the basis functions, provide the best match for the current input. The model cost function contains two key terms – one encouraging an accurate reconstruction of the input image and one constraining the activities of the model units to be sparse. Here, sparseness means that the distribution of the responses of a population of neurons to a batch of image patches is heavy-tailed. That is, most responses will be close to zero except for a few. After training, the receptive fields of the units showed the key spatial

tuning properties of V1 simple cells, being localized, oriented and spanning a range of spatial frequencies (Fig. 1.4b).

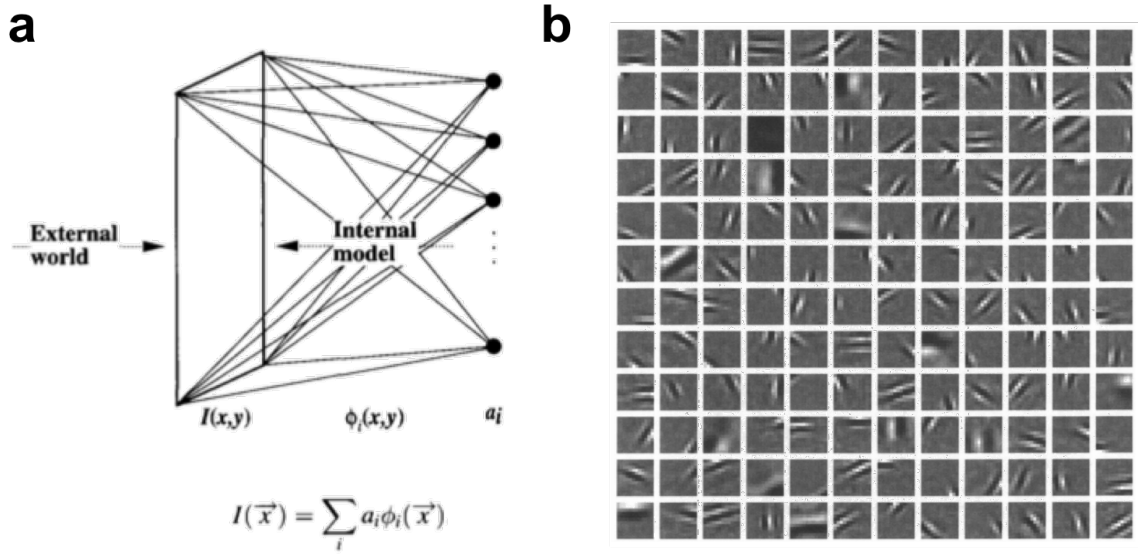


Figure 1.4 | Sparse coding model and results after training. a, Schematic of generative sparse coding model. Here the input image, I(x,y) is assumed to be composed of a linear superposition of basis functions ϕ_i(x,y) mixed together with amplitudes (activations) a_i. In sparse coding, a sparse prior is placed on the amplitudes a_i such that for any given image, there are only few non-zero amplitudes. **b**, Resulting basis functions (projective fields) after training sparse coding model on a large set of natural image patches. The basis functions show spatial tuning resembling V1 simple cells. Adapted from (Olshausen and Field, 1997).

When reduced to a single layer, Rao and Ballard's (1999) predictive coding model is closely related to Olshausen and Field's (1996) sparse coding model (Huang and Rao, 2011; Olshausen and Field, 1997). Both are generative models of natural image patches. The main difference is that the predictive coding model places a Gaussian (rather than a sparse) prior on the activities of the hidden units. Indeed, when a sparse prior is placed on the activities of the units in the first layer of the predictive coding model, the RFs of the units in this layer closely resemble those produced by sparse coding (Rao and Ballard, 1999). However, with the original Gaussian prior on the activities of the hidden units, the RFs of the predictive

coding model units must be explicitly constrained in space in order to be spatially localized (Olshausen and Field, 1997; Rao and Ballard, 1999).

1.4.4 Independent component analysis

Independent component analysis (ICA) is a framework that is closely related to sparse coding. It is also based on the idea that the function of early sensory systems is to use a code that reduces redundancy in natural inputs, but here a critically complete code (having the same number of basis functions as input dimensions) is used and the aim of the model is to find maximally independent features of inputs, i.e., those which are distributed in a non-Gaussian (heavy-tailed) manner (Bell and Sejnowski, 1997). Bell and Sejnowski's (1997) ICA model is formally equivalent to Olshausen and Field's (1996, 1997) sparse coding model, except that it assumes a critically complete code (Olshausen and Field, 1997). When trained on thousands of natural image patches, ICA produces filters that are quite similar to those produced by sparse coding. There are many variants of the original ICA model using different learning algorithms to find independent components (see Hyvärinen et al. (2001) for a comprehensive review). However, the general principle and the nature of the results achieved with these models remain similar.

1.4.5 Temporal coherence/slowness

A different coding objective to efficient coding is temporal coherence (or slowness) (Földiák, 1991; Hinton, 1989; Wiskott and Sejnowski, 2002). This suggests that features of interest in the world, such as those relating to moving objects, change on a relatively slow time-scale compared to sensory signals (such as the change in activity of photoreceptors in the retina that is caused by the movement of the object in the environment). Thus, features of interest can be recovered from sensory signals by looking for features that vary slowly

over time. Models built on this idea include slow subspace analysis (SSA; Kayser et al., 2001) and slow feature analysis (SFA; Berkes and Wiskott, 2005; Wiskott and Sejnowski, 2002). These models are typically optimized to minimize the change in each unit's response to a natural visual input over time (i.e. minimize the mean squared derivative of the response with respect to time). Additional constraints are also applied to ensure that responses have zero mean, unit variance and are decorrelated between model units. When trained on natural video patches, these models have been shown to produce basis functions with tuning resembling V1 complex and hypercomplex cells (showing end- and side-inhibition).

1.5 Normative models of auditory processing

Normative models of auditory processing have largely derived from the efficient, sparse coding and temporal slowness frameworks. Smith and Lewicki (2006) trained an ICA model on natural sound inputs and recovered filters that resembled the frequency tuned 'gammatone' filters of the cochlea. Carlson et al. (2012) applied a sparse coding model to spectrograms (short-windowed Fourier transforms) of natural sounds and produced model neurons that capture some of the frequency tuning properties of IC neurons. Other authors have used related methods to model A1 neurons (Klein et al., 2003; Kozlov and Gentner, 2016; Zhao and Zhaoping, 2011). Carlin and Elhilali (2013) explored the use of a model related to SFA, whose units' firing rates were optimized to vary slowly over time and be mutually decorrelated between units. This model produced RFs with spectral properties somewhat resembling those of A1 neurons. These models are examined in more detail in Chapter 2, section 2.7.2.

1.6 Temporal prediction

In this thesis we develop the temporal prediction framework. This normative approach is based on the hypothesis that representations in sensory systems are optimized for the efficient prediction of future inputs from their recent past. This stands in contrast to many other normative approaches, such as efficient coding, sparse coding and predictive coding, which aim to efficiently represent all the current incoming input, rather than just those features that are predictive of the future. The idea that prediction is an important component of perception dates at least as far back as Helmholtz (Helmholtz, 1867; Sutton and Barton, 1981), although what is meant by prediction and the purpose it serves is quite varied between models incorporating it (section 1.6.4; Chalk et al., 2018; Salisbury and Palmer, 2016). Several theoretical studies have proposed prediction of future input as a primary function of neural systems (Hawkins and Blakeslee, 2004; O'Reilly et al., 2014; Softky, 1996). However, models incorporating temporal prediction that explicitly attempt, with varying degrees of success, to explain features of sensory representations from natural inputs have only very recently started to emerge (Chalk et al., 2018; Lotter et al., 2018; Ocko et al., 2018; Palm, 2012; Singer et al., 2019, 2018).

1.6.1 Motivation for temporal prediction

There are a number of motivations for positing temporal prediction as a principle behind the form and function of sensory neural processing. Notably, sensory inputs guide actions, but such actions necessarily lag behind these inputs due to delays caused by sensory transduction, axonal conduction, synaptic transmission, and muscle activation (Camalier et al., 2012; DiCarlo and Maunsell, 2005; Maunsell and Gibson, 1992; Schmolesky et al., 1998). Hence estimation of the future state of the sensory world at the time of action is likely

to be essential. To strike a cricket ball, for example, one must estimate its future location, not where it is now (Nijhawan, 1994). Prediction has other fundamental theoretical advantages: a system that parsimoniously predicts future inputs from their past, and that generalizes well to new inputs, is likely to contain representations that reflect their underlying causes (Bialek et al., 2006, 2001). This is important because ultimately, we are interested in these causes (e.g. the position or speed of the cricket ball), not the raw images or sound waves incident on the sensory receptors. Furthermore, much of sensory processing involves discarding irrelevant information. Discarding that which is not predictive of the future may provide a good first stage in processing, to arrive at a representation of what is important in the environment for guiding action (Bialek et al., 2006, 2001).

1.6.2 Predictive information bottleneck

The information bottleneck method provides a way of finding reduced representations of a random variable (such as the past sensory input) that provide maximal Shannon information about a second random variable (such as the future sensory input) by compressing the former through an intermediate variable (Tishby et al., 2000). In the context of prediction of sensory input, this implies finding a compressed internal representation of the past stimulus that preserves maximal information about the future of the stimulus.

Much theoretical work by Bialek and colleagues (2001, 2006) has shown using the predictive information bottleneck that, under certain conditions, solving the problem of efficiently representing predictive information provides a way of learning underlying parameters from observed noisy variables. However, the information bottleneck method becomes computationally intractable for high dimensional and/or non-Gaussian data and so has largely been used for data with a small number of discrete states or for Gaussian variables (Chalk et al., 2016). Only recently have more tractable versions of the information bottleneck

been developed to form normative models of neural responses to (non-Gaussian) natural stimuli and been shown to produce RFs with features resembling those of V1 simple cells (Chalk et al., 2018). These results are discussed more in section 3.5.1.

1.6.3 The temporal prediction model

Our model (which we refer to as the temporal prediction model) is related to the predictive information bottleneck in that it prioritizes representations which are predictive of future inputs. However, it is simpler and easier to work with, using mean squared error between the model's prediction and the true future as the measure of predictive capacity rather than mutual information.

We instantiate temporal prediction using a very simple feedforward neural network model that updates the input and output weights to a set of hidden units using stochastic gradient descent over a large set of natural inputs. We envisage the activity of these hidden units as the neural representation. The activity of each hidden unit arises from a linear-nonlinear mapping of the inputs over several time steps into the past. The output weights perform a linear mapping between the hidden unit responses and the predicted future. The model is trained by minimizing the mean squared error between the model's prediction and the true future input for a given past over a diverse range of natural inputs. We also constrain the input and output weights in the model to be sparse, akin to a constraint on neural wiring. This is in contrast to sparse coding, where the unit activities are constrained to be sparse.

In Chapters 2 and 3, we use a single hidden-layer temporal prediction model and compare its representations to A1 and V1 simple cell RFs. In Chapter 4 we extend this model to a hierarchical form, under the intuition that it may be useful to stack temporal prediction models, each stack in turn predicting the future activity of the hidden unit representation of the stack below. We compare the representations of the hierarchical model's units to those

of neurons at multiple levels of the visual hierarchy. An illustration of the operations performed by the temporal prediction model are shown for a single stack in Fig. 1.5a and across stacks of the hierarchical model in Fig. 1.5b. Formal details of the temporal prediction model architectures and training procedures are outlined in more detail in sections 2.1, 2.8.2, 3.6.2, 4.2.1, and 4.4.2.

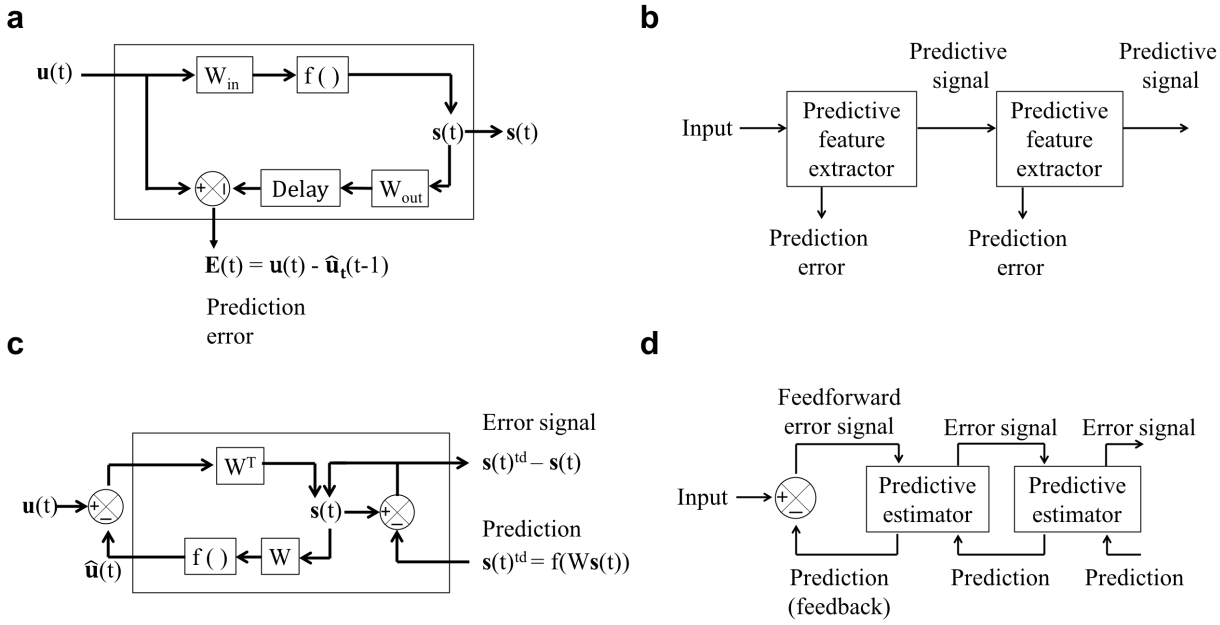


Figure 1.5 | Comparison between temporal prediction and predictive coding frameworks. a, Components of the temporal prediction model. The model receives a time-varying input **u**(t). The input passes through feedforward weights W_{in} and undergoes a nonlinear transformation f( ) to generate hidden unit activity **s**(t). The model is optimized so that a linear transformation W_{out} generates an accurate estimate $\hat{\mathbf{u}}_t$ of its future input (at time t) based on hidden unit activity generated by the past input (at time t-1). Because of the delay (which is shown at the end of processing but is likely distributed throughout the system) the predicted input is in the future relative to the input that generated the hidden unit activity. A prediction error, which is an optional readout of the system, is used to train the input and output weights W_{in} and W_{out}. The prediction error can equivalently be written as $\mathbf{E}(t+1) = \mathbf{u}(t+1) - \mathbf{u}_{t+1}(t)$. **b,** Architecture of the hierarchical temporal prediction model. The operations performed in each stack (labelled 'predictive feature extractor') are shown in **a.** Each stack of the model receives input from the stack below and finds features of its input that are predictive of its future input, with the lowest layer predicting the raw natural inputs. The predictive signal is fed forward to the next layer. A single hidden-layer model would comprise of only a single stack predicting natural inputs. **c**, Operations performed at each level of Rao and Ballard's (1999)

hierarchical predictive coding model. The input **u**(t) undergoes a linear transformation W^T to generate hidden unit representation **s**(t). This is used to generate an estimate $\hat{\mathbf{u}}$(t) of the input at the same time step. An error neuron computes the difference between the hidden representation **s**(t) and its estimate $\mathbf{s}(t)^{td}$ from the layer above. This error is passed forward as input to the next layer. **d**, General architecture of the predictive coding model. At each stage, the model generates an estimate of the input which is fed back down to the previous layer. Top-down predictions from the layer above are subtracted from the current estimate and the error signal is fed forward to the next layer. A key difference between the two frameworks is that the predictive coding framework generates a prediction of the entire current input at a single time step, while the temporal prediction framework generates a prediction of the future input based on past input. In the predictive coding framework, the error signal is passed forward to the next layer, while in the temporal prediction framework, the signal containing predictive features of the input are passed forward. **c**, **d** adapted from Rao and Ballard (1999).

1.6.4 Temporal prediction and predictive coding

With regards to perception and prediction, two contrasting but interrelated frameworks have been distinguished (Chalk et al., 2018; Salisbury and Palmer, 2016). In the "predictive coding" framework (Friston, 2003; Huang and Rao, 2011; Rao and Ballard, 1999; Srinivasan et al., 1982), prediction is used to remove statistical redundancy in order to provide an efficient representation of the entire stimulus. Some models of this type use prediction as a term for estimation of the current or a static input (such as images) from latent variables (Rao and Ballard, 1999), whereas others have also considered the temporal dimension of the input (Rao, 1999; Srinivasan et al., 1982). Sparse coding models can be related to this framework (Huang and Rao, 2011; Olshausen and Field, 1997). In contrast, the "predictive information" framework (Bialek et al., 2001; Palmer et al., 2015; Salisbury and Palmer, 2016), which our approach relates to more closely, involves selectively encoding those features of the stimulus that predict future input. Hence, the predictive coding approach seeks to find a compressed representation of the entire input, whereas the predictive information

approach selectivity encodes only predictive features (Chalk et al., 2018; Salisbury and Palmer, 2016). Our temporal prediction model relates to the predictive information approach in that it is optimized to predict the future from the past, but it has a combination of characteristics, such as a non-linear encoder and sparse weight regularization, which have not previously been explored for such an approach. In Fig. 1.5 we present a high-level comparison between our temporal prediction model and Rao and Ballard's (1999) predictive coding model.

1.6.5 Incorporating time into normative models

Efficient coding models, such as sparse coding and ICA have been developed to account for the spatial properties of static natural inputs (images) and have been shown to successfully to capture the spatial tuning properties of V1 simple cells. However, it is essential to also consider the temporal aspects of inputs and how these affect neural representations, especially when extending these models to the auditory domain. Sparse coding and ICA models have been extended for this purpose by training them to represent features of spatiotemporal visual inputs (videos) (Olshausen, 2003; van Hateren and Ruderman, 1998) and spectrotemporal features of natural sounds (Carlson et al., 2012; Cusack and Carlyon, 2004; Klein et al., 2003; Kozlov and Gentner, 2016; Zhao and Zhaoping, 2011).

However, simply extending the existing models that have been designed to account for the statistics of natural images has several drawbacks. These frameworks do not account for the directionality of time, and instead simply treat the temporal dimension in the same way as visual space or sound frequency. Because of this, the temporal RFs of these models tend to be either unlocalized or randomly localized in time (Carlin and Elhilali, 2013; Carlson et al., 2012; Cusack and Carlyon, 2004; Klein et al., 2003; Kozlov and Gentner, 2016; Zhao and Zhaoping, 2011). For both vision (DeAngelis et al., 1993; Ohzawa et al.,

1996) and audition (deCharms et al., 1998), the envelopes of real neuronal RFs tend to be asymmetric in time, with greater sensitivity to very recent inputs compared to inputs further in the past. In contrast, the RFs predicted by these models typically show symmetrical temporal envelopes, with either approximately flat envelopes over time or a balanced falloff of the envelope over time either side of a peak. They also lack greater sensitivity to very recent inputs. Furthermore, the temporal scale of these RFs is also arbitrary, and typically matches the scale of the inputs being used to train the model. Finally, sparse coding and some related models, such as predictive coding, typically incorporate a recurrent settling process to explain each input (Carlson et al., 2012; Olshausen and Field, 1997; Rao and Ballard, 1999). While this process may be fast enough to handle still images, it may not be fast enough to account for natural inputs that are constantly and rapidly changing. This problem is further compounded when considering hierarchical architectures, which would be needed for these models to account for the response properties of neurons in higher sensory areas of real neural systems, which are hierarchical in nature.

It is possible that these model architectures could be modified to account more explicitly for the temporal aspects of natural inputs (as is done for example by Rao (1999) and Srinivasan et al. (1982)). However, to date no study has demonstrated that such a model is able to account for the spatiotemporal tuning of V1 neurons, the spectrotemporal tuning of A1 neurons and the tuning properties of neurons in other areas of the visual hierarchy such as retina and MT. In this thesis, I show, using qualitative and quantitative comparisons, that for both visual and auditory RFs these shortcomings are largely overcome by the temporal prediction approach.

1.7 Relationship to machine learning models

Understanding how neurons in sensory systems compute and represent information is of fundamental importance in its own right – this is necessary if we are to explain how these neurons use raw information from the outside world to form our perceptions and inform our actions. In addition, developments in computational neuroscience have served as a basis for machine learning models that attempt to carry out tasks that our sensory systems perform with ease. Ideas and tools flow in both directions between researchers working on computational models of sensory systems, which aim primarily to provide explanations for neural function, and those in machine learning, whose primary aim is to develop models that perform well on practical tasks.

Two areas of machine learning that have shown some of the greatest advances are computer vision and natural language processing, which have achieved and surpassed human-level performance on tasks such as object classification (Krizhevsky et al., 2012), image scene labelling (Farabet et al., 2013) and speech recognition (Hinton et al., 2012). This has been accomplished through the use of deep supervised learning models (LeCun et al., 2015), which consist of hierarchical networks of artificial neurons inspired by the brain that are optimized for the performance of specific tasks using large datasets of labelled input-output pairs.

Before the success of deep supervised learning, most task-optimized neural network models involved some unsupervised learning (using no labelled data) and included aspects based on frameworks such as sparse coding, ICA and other algorithms developed largely in an attempt to understand how sensory processing is achieved in the brain (Bengio, 2012; Bengio et al., 2012). After initializing these networks in an unsupervised manner, they could then be fine-tuned for specific tasks using a small amount of labelled training data. A

significant development by Hinton et al. (2006), followed closely by Bengio et al. (2006), showed how to stack single layer networks (trained one layer at a time in an unsupervised manner) to form deep networks with multiple layers of nonlinear neurons that could then be optimized for task performance.

Due primarily to improvements in computer hardware and the increased availability of labelled training data (for example, through the curation of ImageNet (Deng et al., 2009)), deep supervised models, where all layers are trained simultaneously on large labelled datasets (e.g. Krizhevsky et al. (2012)), have surpassed unsupervised models in performance. The advent of convolutional networks (LeCun et al., 1998, 1990), loosely inspired by the local receptive fields of V1 simple and complex cells that tile visual space, have also led to a significant improvement in the performance of these models. Currently, most state-of-the-art deep learning models are trained using supervised learning. However, for many problems, acquisition of large labelled datasets is not feasible. Hence, it is expected that unsupervised learning algorithms, such as the one explored in this thesis, will have a more significant impact in the long term if we are to develop intelligent machines that are capable of generalizing beyond the few tasks they are currently being optimized for (LeCun et al., 2015).

Computational neuroscience has benefited enormously from the tools developed for machine learning. Tools that are used in the models developed for this thesis and in many other computational models of the brain include backpropagation (Rumelhart et al., 1986) (examined in more detail in the following section), convolutional neural networks (LeCun et al., 1998, 1990), software libraries (for example http://deeplearning.net/software/theano/) and computational hardware. In addition, ideas such as training networks one layer at a time and stacking them to form deep hierarchical models were first developed for machine learning (Bengio et al., 2006; Hinton et al., 2006).

1.7.1 Backpropagation

Artificial neural networks are most commonly trained using the error backpropagation algorithm (LeCun et al., 2015; Rumelhart et al., 1986; Whittington and Bogacz, 2019). With this algorithm, learning occurs in two steps: first the input is propagated in a forward pass through the network. Next a backward pass is performed, where the error between the network's output and the desired output is computed and the derivative of this error is backpropagated through the network. The network weights are then adjusted so as to minimize the contribution of that neuron to the error at the output of the network. This is explained in more detail in Figure 1.6.

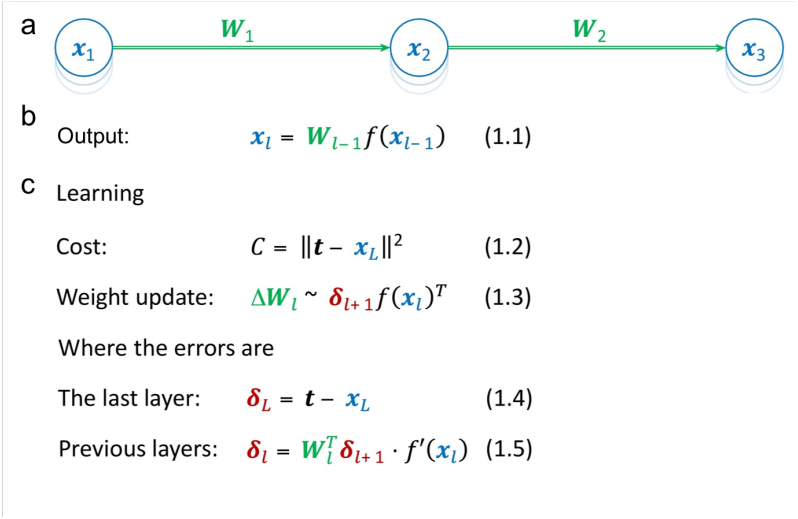


Figure 1.6 | Illustration of the backpropagation algorithm. a, Illustration of network. Each set of blue circles represents a layer of neurons in the network, which receives a set of weighted inputs from the neurons in the previous layer. Biases are omitted for simplicity. **b,** Forward pass by which an input is propagated through the network (Eqn 1.1). The activation of the units in layer l is denoted by the vector x_l and is computed by multiplying the nonlinearly transformed activations $f(x_{l-1})$ of the units in layer l-1 by the weight matrix (W_{l-1}) connecting layers l-1 and l. **c,** Learning in the network is performed using error backpropagation. First the error between the network's output x_L and the desired target value t are computed. This is quantified using the cost function, an example of which is the L2-norm (mean squared error), denoted by the $\|\ \|^2$ operator shown in Eqn 1.2. During learning, the weights connecting the units in the network are adjusted so as to minimize the contribution of each unit to the error at the output. For a given layer l, the required weight update is calculated by multiplying a vector of error derivatives (δ_{l+1}) associated with the neurons in the subsequent layer l+1 by the output $f(x_l)$ at layer l (Eqn 1.3). In order to update the weights of the network, error

derivative vectors must be computed for every layer. For the output layer L, the error derivative vector (δ_L) is simply the derivative of the cost function (Eqn 1.4). For all preceding layers, the error derivative vector (δ_l) is computed by taking the sum of the errors of the layer above (l+1), weighted by the strength of the connections between the two layers (W_l). This is then scaled by derivative $f'(x_l)$ of the output $f(x_l)$ at layer l (Eqn 1.5). In Eqn 1.5 the $\cdot$ operator signifies element-wise multiplication. In this way, the error is backpropagated from the output layer through all previous layers of the network. At each stage the computed error is used to update the weights of the network so as to minimize the error at the output. The forward and backward passes are repeated for many different inputs until the cost function is minimized for the dataset. Figure adapted from (Whittington and Bogacz, 2019).

Despite the success of this learning procedure for the training of artificial neural networks, its usefulness for explaining the function of biological neural networks has long been questioned. This is because there are several aspects of backpropagation that are biologically implausible (Whittington and Bogacz, 2019). First, the error computation and subsequent weight update of every neuron depends on the activity and errors of all downstream neurons in the network. However, biological neurons typically only have access to local error signals. Second, backpropagation necessitates the existence of symmetric input and output weights between each set of connecting neurons. This is a feature that is not common in biological neural networks. Finally, while the derivative of a neuron's output with respect to its input can be easily computed for the rate-coding neurons (with continuous outputs) that are commonly used in artificial neural networks, the derivative becomes difficult to approximate for biological neurons that use spikes.

Throughout this thesis all networks consist of continuous output (rate coding) neurons that are trained using backpropagation. In this study we are interested primarily in the weights of the networks after training and how the networks' representations (as determined by their synaptic weights) are affected by the normative principle of temporal prediction and how these representations compare with the biology. We are less concerned with the details of how these representations are learned, and simply use backpropagation to

find these representations rather than claim that this is how the brain learns them. However, some ways in which the networks and learning rules used in this study could be modified to account for more features of biological networks are explored in Section 5.5 of the Discussion.

1.8 Thesis overview

In Chapter 2, I present the temporal prediction model and show that when trained on natural auditory inputs it develops receptive fields resembling those of A1 neurons in both their frequency and temporal tuning. In Chapter 3, I train the identical model using natural video inputs and show that, in this case, the model units' receptive fields resemble those of V1 simple cells in both their spatial and temporal tuning properties. In Chapter 4, I extend the temporal prediction model to a hierarchical form and show that when trained on natural video inputs, the model units display many of the tuning properties of the neurons along the dorsal visual hierarchy, with units in consecutive layers of the model resembling retinal ganglion cells, V1 simple cells, V1 complex cells and MT pattern selective neurons.

1.9 Publications

Much of the work presented in Chapters 2 and 3 has been published as:

Singer Y, Teramoto Y, Willmore BDB, King AJ, Schnupp JWH, Harper NS. 2018. **Sensory cortex is optimized for prediction of future input**. *Elife* **7**:e31557.

Much of Chapter 4 is in preparation for publication and is online as a preprint article:

Singer Y, Willmore BDB, King AJ, Harper NS. 2019. **Hierarchical temporal prediction captures motion processing from retina to higher visual cortex**. *bioRxiv* 575464.

Chapter 2 | Temporal prediction model of natural sounds reproduces properties of primary auditory cortical neurons

2.1 The temporal prediction model

To determine what type of sensory RF structures would facilitate predictions of the imminent future, we built a feedforward network model with a single layer of nonlinear hidden units, mapping the inputs to the outputs through weighted connections (Fig. 2.1). Each hidden unit's output results from a linear mapping (by input weights) from the past input, followed by a monotonic nonlinearity, much like the classic linear-nonlinear model of sensory neurons. The model then generates a prediction of the future from a linear weighting of the hidden units' outputs. This is consistent with the observation that decoding from the neural response is often well approximated by a linear transformation (Eliasmith and Anderson, 2003).

We trained the temporal prediction model using backpropagation (see Section 1.7.1) on extensive corpora of soundscapes, modelling A1 neurons. The network was trained by optimizing its synaptic weights to most accurately predict the immediate future of the stimulus from its very recent past. We trained the network to predict the next three time steps (15 ms) of cochleagrams of natural sounds based on the 40 most recent time steps (200 ms). Cochleagrams resemble spectrograms but are adjusted to approximate the auditory nerve representation of sounds (see Methods). We set the temporal resolution of the cochleagrams to 5ms time bins. This is close to the minimum amplitude modulation period that can be tracked by auditory midbrain neurons (Koumura et al., 2019; Rees and Møller, 1983), which

provide input to A1 via the auditory thalamus. We therefore compare our model unit responses to those of A1 neurons.

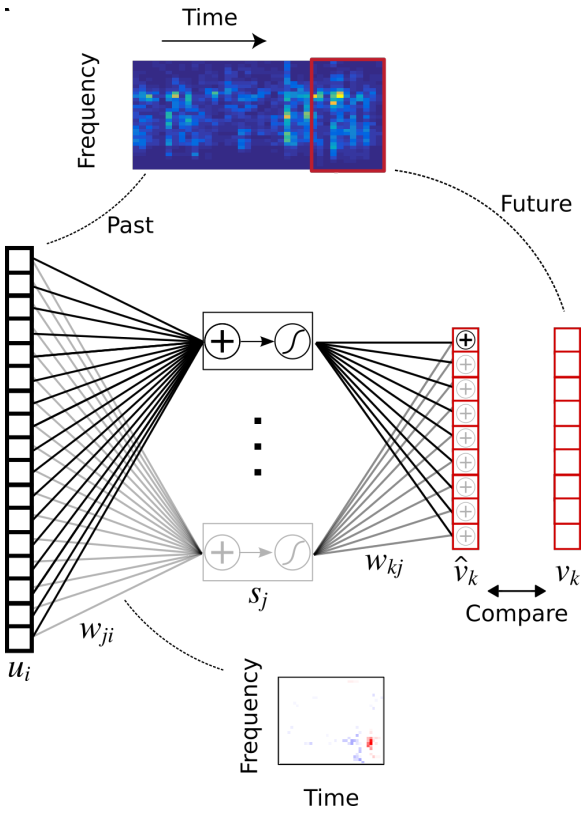


Figure 2.1 | Temporal prediction model implemented using a feedforward artificial neural network trained on natural sound inputs. Network trained on cochleagram clips (spectral content over time) of natural sounds, aims to predict immediate future time steps of each clip from recent past time steps. u_i, input – the past; w_{ji}, input weights; s_j, hidden unit output; w_{kj}, output weights; $\hat{v}_{kn}$, output – the predicted future; v_k, target output – the true future. Hidden unit's RF is the w_{ji} between the input and that unit j.

During training we used sparse, L_1 weight regularization (see Eqn. 2.3 in Methods) to constrain the network to predict future stimuli in a parsimonious fashion, forcing the network to use as few weights as possible while maintaining an accurate prediction. L_1 weight regularization involves adding an extra term to the model cost function (Eqn. 2.3 in Methods) that takes a weighted sum of the absolute values of all input and output weights in the network. With the regularization, the cost function consists of two terms: a mean squared

prediction error term and the regularization term. During training the L_1 regularization penalizes the network for having non-zero weights and therefore encourages convergence to a solution where only weights that contribute strongly towards minimizing the mean squared prediction error are maintained, while small or useless weights are reduced to values very close to zero. This constraint can be viewed as an assumption about the sparse nature of causal dependencies underlying the sensory input, or alternatively as analogous to the energy and space restrictions of neural connectivity. It also prevents our network model from overfitting to its inputs. Note that this sparsity constraint differs from that used in sparse coding models, in that it is applied to the weights rather than the activity of the units, being more like a constraint on the wiring between neurons than a constraint on their firing rates.

For comparison, we trained a sparse coding model (Carlson et al., 2012; Olshausen and Field, 1997, 1996) (https://github.com/zayd/sparsenet) using our dataset. We would expect such a model to perform less well in the temporal domain, because unlike the temporal prediction model, the direction of time is not explicitly accounted for. The sparse coding model was chosen because it has set the standard for normative models of visual RFs (Olshausen and Field, 1997, 1996; van Hateren and Ruderman, 1998; van Hateren and van der Schaaf, 1998), and the same model has also been applied for auditory RFs (Blättler et al., 2011; Brito and Gerstner, 2016; Carlson et al., 2012; Młynarski and McDermott, 2017). Past studies (Carlson et al., 2012; Olshausen and Field, 1997, 1996) have largely analysed the basis functions produced by the sparse coding model and compared their properties to neuronal RFs. To be consistent with these studies we have done the same, and to have a common term, refer to the basis functions as RFs (although strictly, they are projective fields).

There are a few key differences between our temporal prediction model and the sparse coding model we use as a control. First, the goal of our model is to represent only

those features of the input that allow for accurate prediction of the future input, while sparse coding tries to preserve all information about the past or current input. Second, we use a nonlinear encoding model and a simple linear generative model of the future. Sparse coding uses a linear generative model which has a recurrent settling process to arrive at a set of sparse activations that allows it to accurately represent its current input. Third, sparse coding places a sparse prior on the activities of the hidden units. This is equivalent to making an assumption about the sparse nature of the underlying features in natural visual inputs. It is also equivalent to placing a constraint on the metabolic cost of representing a given input by reducing the number of active units. We place no explicit restriction on the nature of the firing of the units in the temporal prediction model. Instead, we use an L_1 (sparse) prior on the weights of the hidden units. This is equivalent to placing a limitation on neuronal wiring and encourages a code that utilises sparse connectivity.

To compare with the model units, we used a set of recorded responses of 114 auditory neurons (including 76 single units) in A1 and the anterior auditory field of 5 anaesthetised ferrets (Willmore et al., 2016) and measured their spectrotemporal RFs (see Methods). Ferrets are commonly used for auditory research, because they are readily trained in a range of sound detection, discrimination or localization tasks (Nodal and King, 2014), the frequency range of their hearing (approximately 40 Hz – 40 kHz (Kavanagh and Kelly, 1988)) overlaps well with (and extends beyond) the human range, and most of their auditory cortex is not buried in a sulcus and hence easily accessible for electrophysiological or optical measurements.

2.2 Qualitative assessment of auditory receptive fields

The A1 RFs we compare to are diverse (Fig. 2.2); their frequency tuning can be narrowband, broadband or more complex, sometimes showing flanking inhibition. They may also lack

clear order or be selective for the direction of frequency modulation (Carlin and Elhilali, 2013). In their temporal tuning, A1 RFs tend to weight recent inputs more heavily, with a temporally asymmetric power profile, involving excitation near the present followed by lagging inhibition of a longer duration (deCharms et al., 1998).

The temporal prediction model RFs (Fig. 2.3) are similarly diverse, showing all of the RF types seen *in vivo* (including examples of localized, narrowband, broadband, complex, disordered and directional RFs) and are well matched in scale and form to those measured in A1. This includes having greater power (root mean square) near the present, with brief excitation followed by longer lagging inhibition, producing an asymmetric power profile. This stands in contrast to previous attempts to model RFs based on efficient and sparse coding hypotheses, which either did not capture the diversity of RFs (Zhao and Zhaoping, 2011), or lacked temporal asymmetry, punctate structure, or appropriate time scale (Brito and Gerstner, 2016; Carlin and Elhilali, 2013; Carlson et al., 2012; Cusack and Carlyon, 2004; Klein et al., 2003; Kozlov and Gentner, 2016).

2.3 Qualitative comparison to other models

We can visually compare the large set of RFs recorded from A1 neurons (Fig. 2.2) to the full set of RFs obtained from the temporal prediction model when trained on auditory inputs (Fig. 2.3) and those of the sparse coding model (Fig. 2.4) when trained on the same auditory inputs. A range of RFs were produced by the sparse coding model, some of which show characteristics reminiscent of A1 RFs, particularly in the frequency domain. However, the temporal properties of A1 neurons are not well captured by these RFs. While some RFs display excitation followed by lagging inhibition, very few, if any, show distinct brief excitation followed by extended inhibition. Instead, neurons that show both excitation and inhibition tend to have a symmetric envelope and these features are randomly localized in

time, and many neurons display temporally elongated structures that are not found in A1 neurons.

Here we have trained a well-established sparse coding model on our dataset of natural sounds. The results presented are similar to those presented by Carlson et. al. (2012), who also trained a sparse coding model on spectrograms of natural sounds. Both our results and those of Carlson et. al. (2012) contain some units whose spectrotemporal RFs are constant over time, some which are roughly gaussian in time, and some which have a positive and negative lobe over time. There are some differences between the RFs produced by the two studies which may be due to the use of different stimulus sets or due to differences in the preprocessing of the auditory inputs.

It should be noted that while some of the model units' RF structures may not appear to be sparse, in the sparse coding model it is the unit activities that are constrained to be sparse not the model unit weights (RFs), unlike in the temporal prediction model. The sparse coding model (through a recurrent settling process) selects the sparsest set of unit activities to represent any given input given the other constraints (such as the reconstruction error). Thus, it is likely that for any given input, the unit activities of the sparse coding model will tend towards being sparse. However, the RFs will not necessarily be sparse.

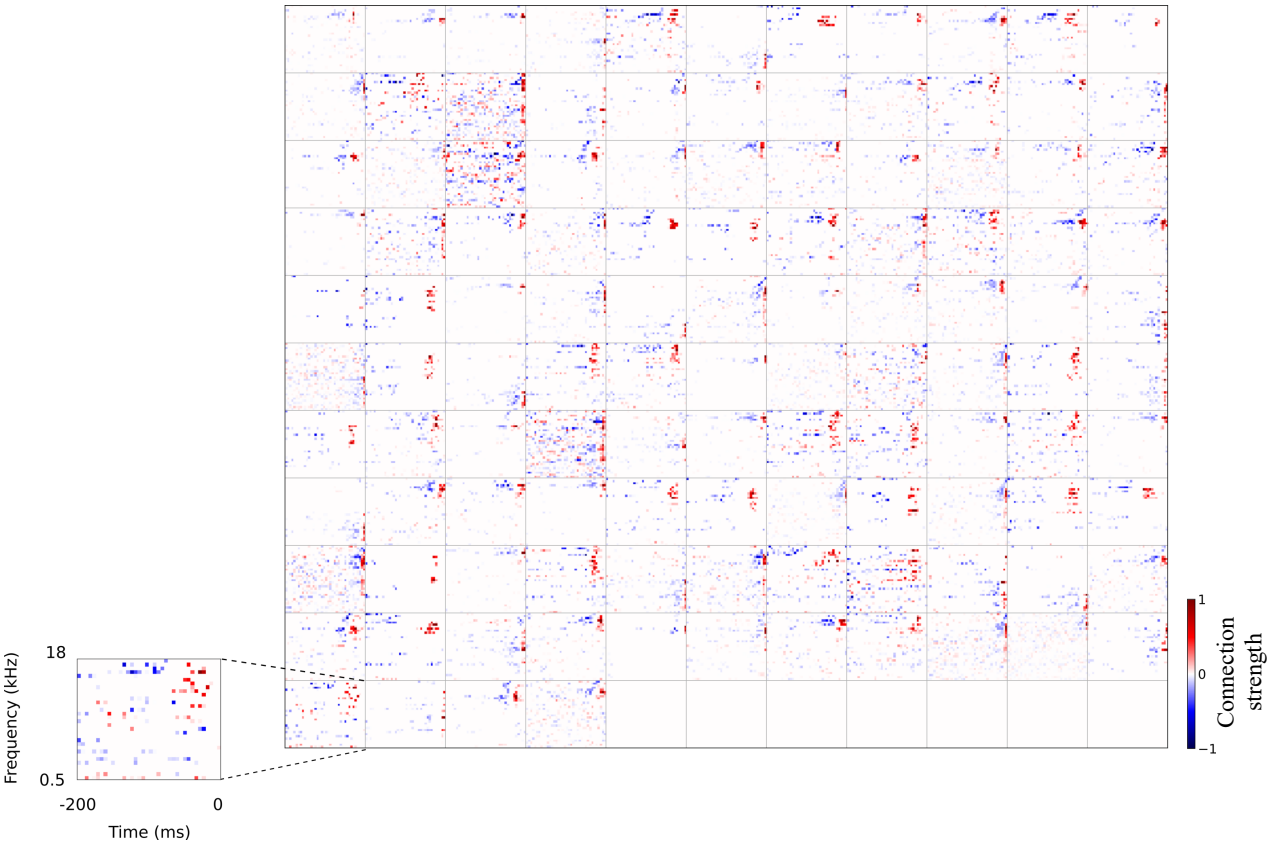


Figure 2.2 | Full dataset of real auditory RFs. 114 neuronal RFs in A1 and AAF of 5 ferrets (Willmore et al., 2016). Most recent two time steps (10ms) were removed to account for conduction delay. Red – excitation, blue - inhibition. Inset shows axes.

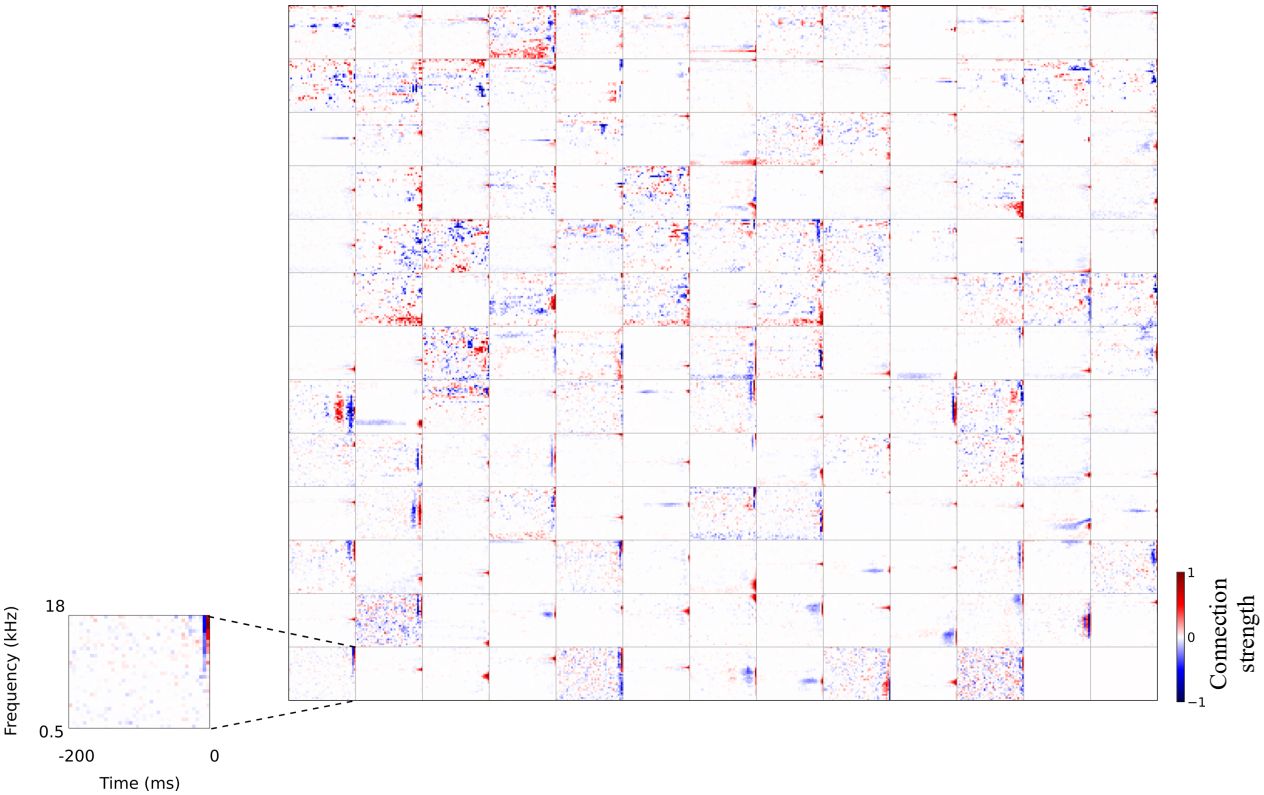


Figure 2.3 | Full set of auditory RFs of the temporal prediction model units. Units were obtained by training the model with 1600 hidden units on auditory inputs. The hidden unit number and L_1 weight regularization strength ($10^{-3.5}$) was chosen because this results in the lowest MSE on the prediction task, as measured using a cross validation set. Many hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving 167 active units. Inactive units were excluded from analysis and are not shown (see Methods). Red – excitation, blue - inhibition. Inset shows axes. Supp. Fig. 2.1 shows the same RFs on a finer timescale. Note that the overall sign of a receptive field learned by the model is arbitrary. Hence, in all figures and analyses, we multiplied each model receptive field by -1 where appropriate to obtain receptive fields which all have positive leading excitation (see Methods). Inset shows axes.

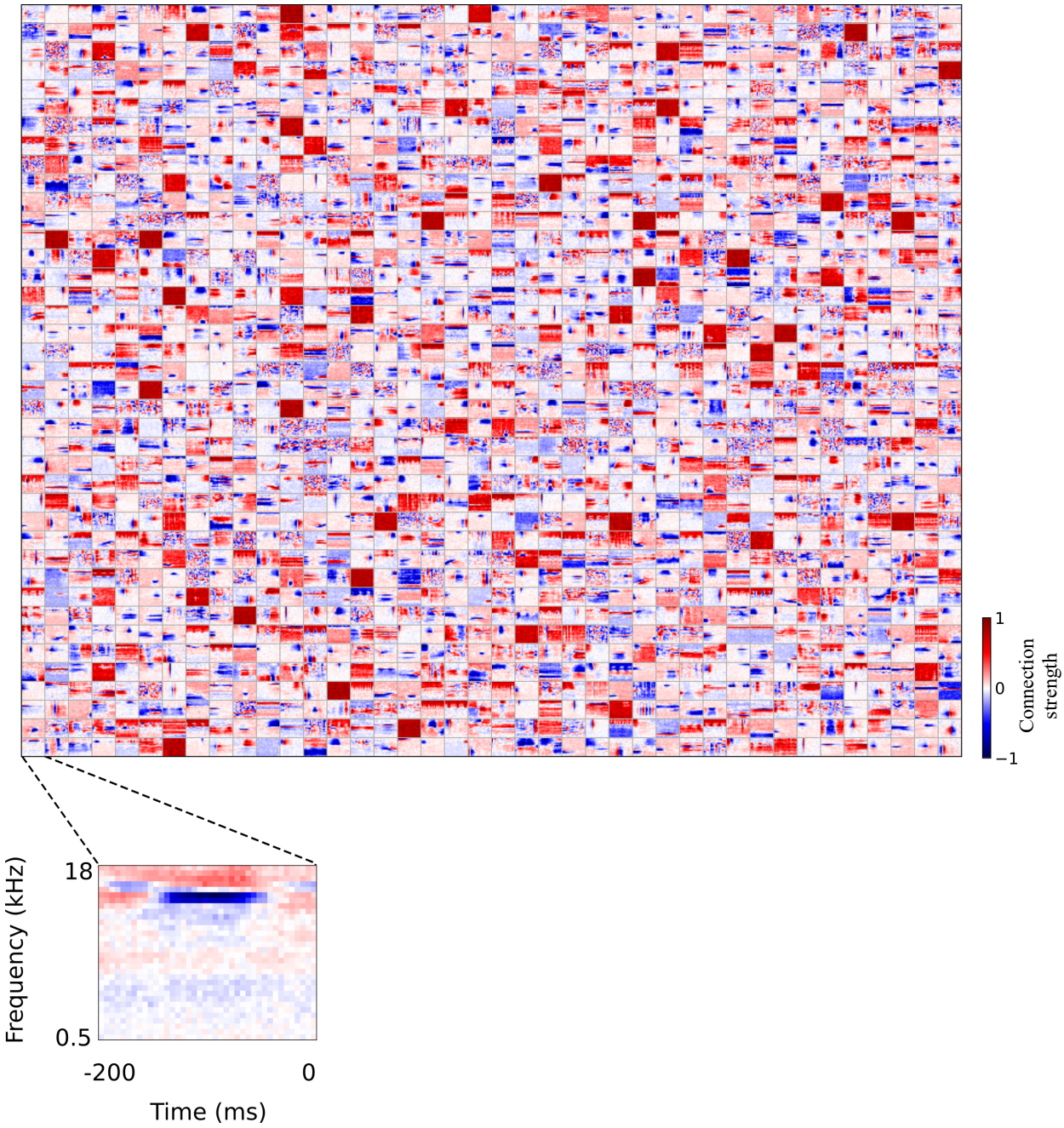


Figure 2.4 | Full set of auditory 'RFs' (basis functions) of sparse coding model used as a control. Units were obtained by training the sparse coding model with 1600 units on the identical auditory inputs used to train the network shown in Fig. 2.3. L_1 regularization of $10^{0.5}$ was applied to the units' activities. This network configuration was selected as it produced unit RFs that most closely resembled those recorded in A1, as determined by visual inspection. Although the basis functions of the sparse coding model are not receptive fields, but projective fields, they tend to be similar in structure (Olshausen and Field, 1997, 1996). In this thesis, to have a common term between models and the data, we refer to sparse coding basis functions as RFs. Red – excitation, blue - inhibition. Inset shows axes.

2.4 Quantitative analysis of auditory receptive fields

We compared the RFs generated by both models to the RFs of the population of real A1 neurons we recorded. We first compared the RFs in a non-parametric manner by measuring the Euclidean distances between the coefficient values of the RFs, and then used multi-dimensional scaling to embed these distances in a two-dimensional space (Fig. 2.5a). The RFs of the sparse coding model span a much larger region than the real A1 and temporal prediction model RFs. Furthermore, the A1 and temporal prediction model RFs occupy a similar region of the space, indicating their greater similarity to each other relative to those of the sparse coding model.

We then examined specific attributes of the RFs to determine points of similarity and difference between each of the models and the recorded data. We first considered the temporal properties of the RFs and found that for the data and the temporal prediction model, most of the power is contained in the most recent time steps (Figs. 2.2-2.4, 2.5b). Given that the direction of time is not explicitly accounted for in the sparse coding model, as expected, it does not show this feature (Figs. 2.4, 2.5b). In the neural data, there appears to be a subpopulation of neurons with an extended time-lag of 30-60ms (the 'bump' in Fig. 2.5b). In contrast, the temporal prediction model does not produce such a subpopulation of units. It is possible that this neural subpopulation consists of a separate class of neurons, whose properties are not captured by the model or that these neurons are activated by a separate neural pathway, which again is not accounted for by the temporal prediction model.

Next, we examined the tuning widths of the RFs in each population for both time and frequency, looking at excitation and inhibition separately. In the time domain, the real data tend to show leading excitation followed by lagging inhibition of longer duration (Figs. 2.2, 2.5c-e). The temporal prediction model also shows many neurons with this temporal

structure, with lagging inhibition of longer duration than the leading excitation (Figs. 2.3, 2.5c-e, and Supp. Fig. 2.1). This is not the case with the sparse coding model, where units tend to show either excitation and inhibition having the same duration or an elongated temporal structure that does not show such stereotyped polarity changes (Figs. 2.4, 2.5c-e). It is also the case that the absolute timescales of excitation and inhibition match more closely in the case of the temporal prediction model (Fig. 2.5c-e), although a few units display inhibition of a longer duration than is seen in the data (Fig. 2.5c). The sparse coding model shows a wide range of temporal spans of excitation and inhibition, in keeping with previous studies (Carlson et al., 2012; Klein et al., 2003).

Regarding the spectral properties of real neuronal RFs, the spans of inhibition and excitation over sound frequency tend to be similar (Fig. 2.5f-h). This is also seen in the temporal prediction model, albeit with slightly more variation (Fig. 2.5f-h). The sparse coding model shows more extensive variation in frequency spans than either the data or our model (Fig. 2.5f-h).

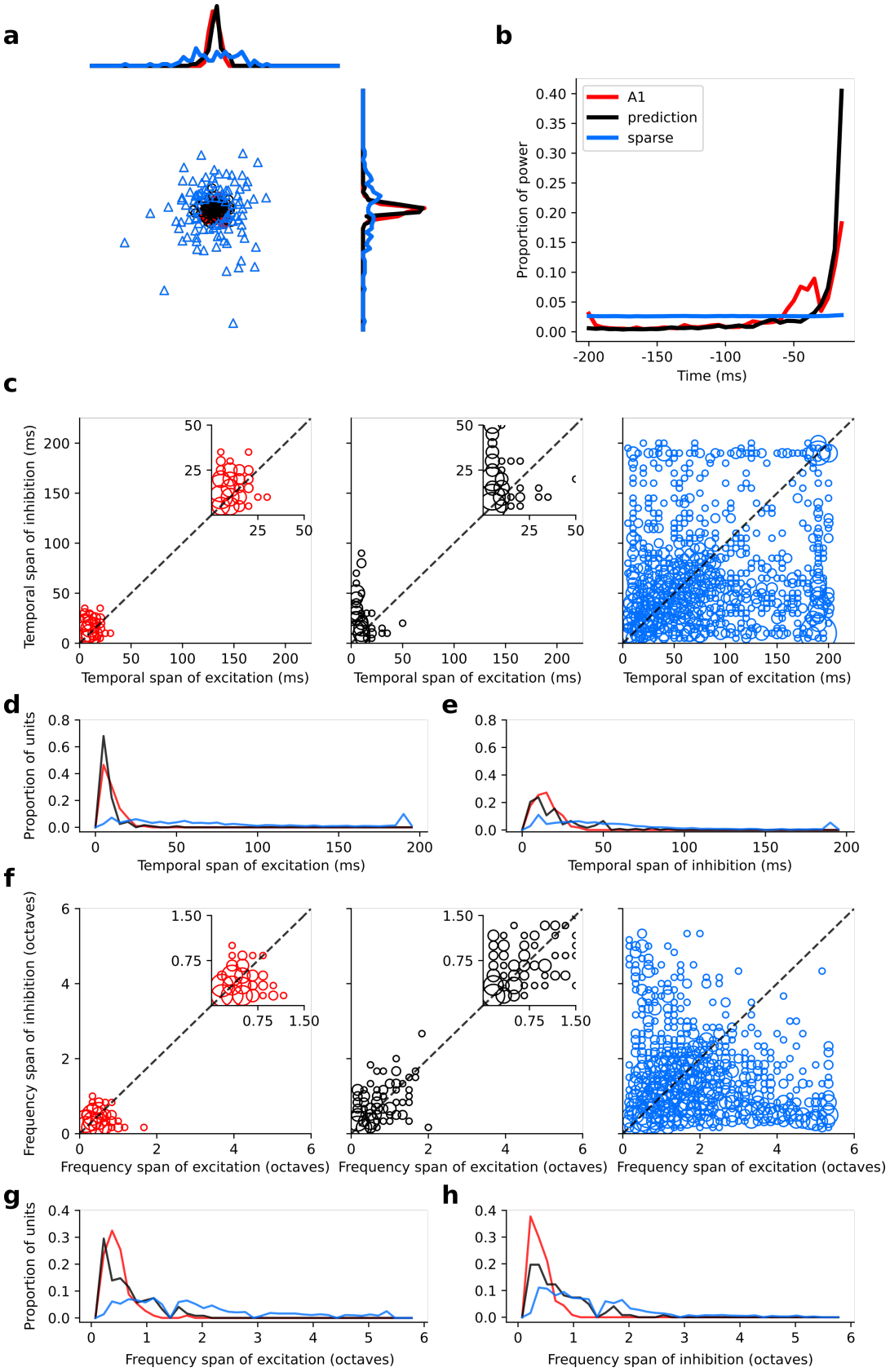


Figure 2.5 | Population measures for real A1, temporal prediction model and sparse coding model auditory spectrotemporal RFs. The population measures are taken from the RFs shown in Figs. 2.2-2.4. **a,** Each point represents a single RF (with 32 frequency and 38 time steps) which has

been embedded in a 2-dimensional space using Multidimensional Scaling (MDS; see Methods). Red circles - real A1 neurons, black circles – temporal prediction model units, blue triangles – sparse coding model units. Colour scheme applies to all subsequent panels. **b,** Proportion of power contained in each time step of the RF, taken as an average across the population of units. **c,** Temporal span of excitatory subfields versus that of inhibitory subfields, for real neurons and temporal prediction and sparse coding model units. The area of each circle is proportional to the number of occurrences at that point. The inset plots, which zoom in on the distribution use a smaller constant of proportionality for the circles to make the distributions clearer. **d,** Distribution of temporal spans of excitatory subfields, taken by summing along the x-axis in **c. e,** Distribution of temporal spans of inhibitory subfields, taken by summing along the y-axis in **c. f,** Frequency span of excitatory subfields versus that of inhibitory subfields, for real neurons and temporal prediction and sparse coding model units. **g,** Distribution of frequency spans of excitatory subfields, taken by summing along the x-axis in **f. h,** Distribution of frequency spans of inhibitory subfields, taken by summing along the y-axis in **f.**

2.5 Optimizing predictive capacity

Under our hypothesis of temporal prediction, we would expect that the better the temporal prediction model network is at predicting the future, the more the RFs of the network should resemble those of real neurons. To examine this hypothesis, we plotted the prediction error of the network as a function of two hyperparameters; the regularization strength and the number of hidden units (Fig. 2.6a). Then, we plotted the similarity between the auditory RFs of real A1 neurons and those of the temporal prediction model (Fig. 2.6b), as measured by the mean KS distances of the temporal and frequency span distributions (Fig. 2.5d-e, g-h, Methods). The set of hyperparameter settings that give good predictions are also those where the temporal prediction model produces RFs that are most similar to those recorded in A1 ($r^2 = 0.8$, $p < 10^{-9}$, $n = 55$). This result argues that A1 neurons are indeed optimized for temporal prediction.

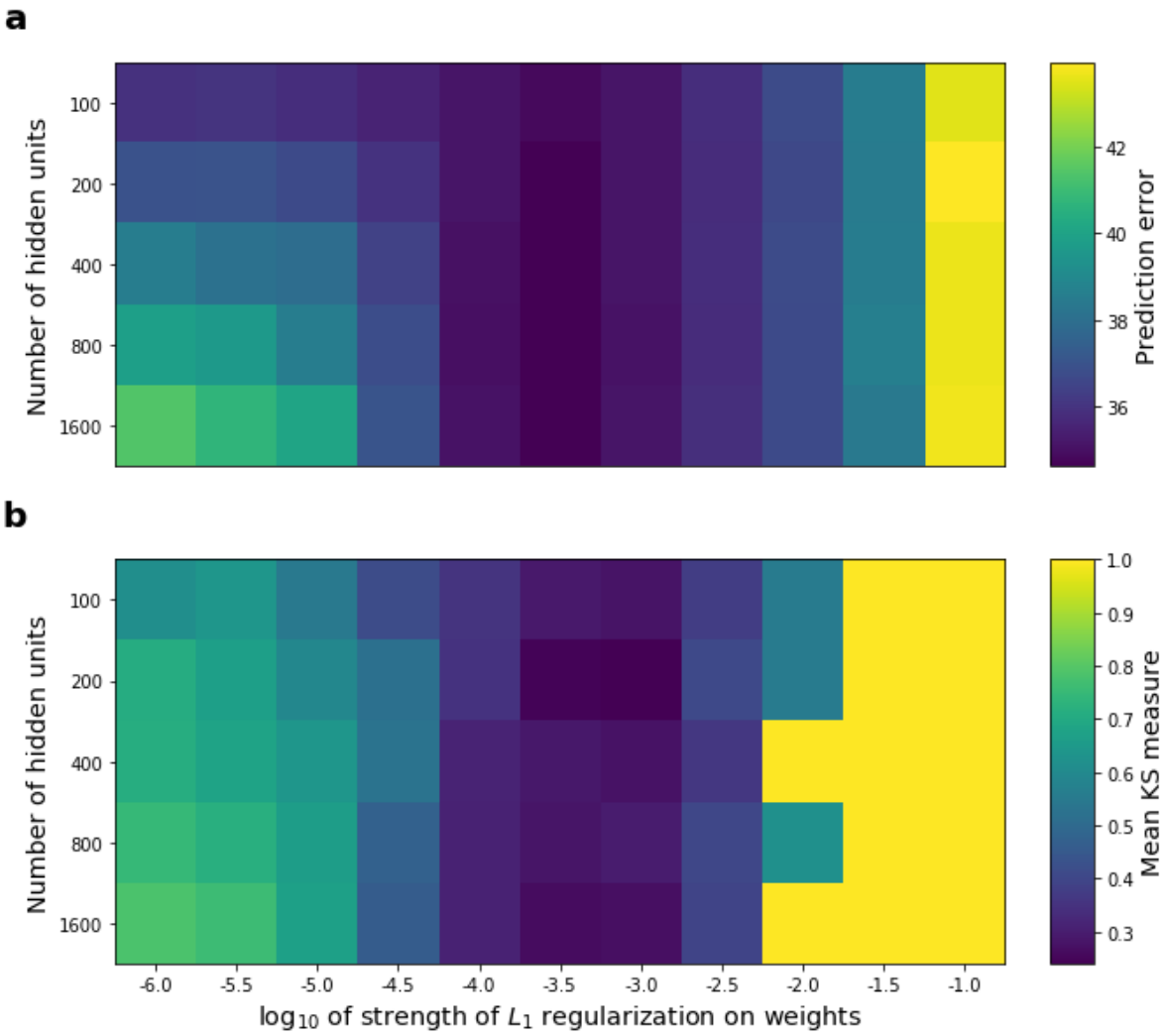


Figure 2.6 | Correspondence between the temporal prediction model's ability to predict future auditory input and the similarity of its units' responses to those of real A1 neurons. Performance of model as a function of number of hidden units and regularization on the weights as measured by **a,** prediction error on validation set at the end of training and **b,** similarity between model units and real A1 neurons. The similarity between the real and model units is measured by averaging the Kolmogorov-Smirnov distance between each of the real and model distributions for the span of temporal and frequency tuning of the excitatory and inhibitory RF subfields (e.g. the distributions in Fig. 2.5d-e and Fig. 2.5g-h). Fig. 2.7 shows the same analysis, performed for the sparse coding model, which does not produce a similar correspondence.

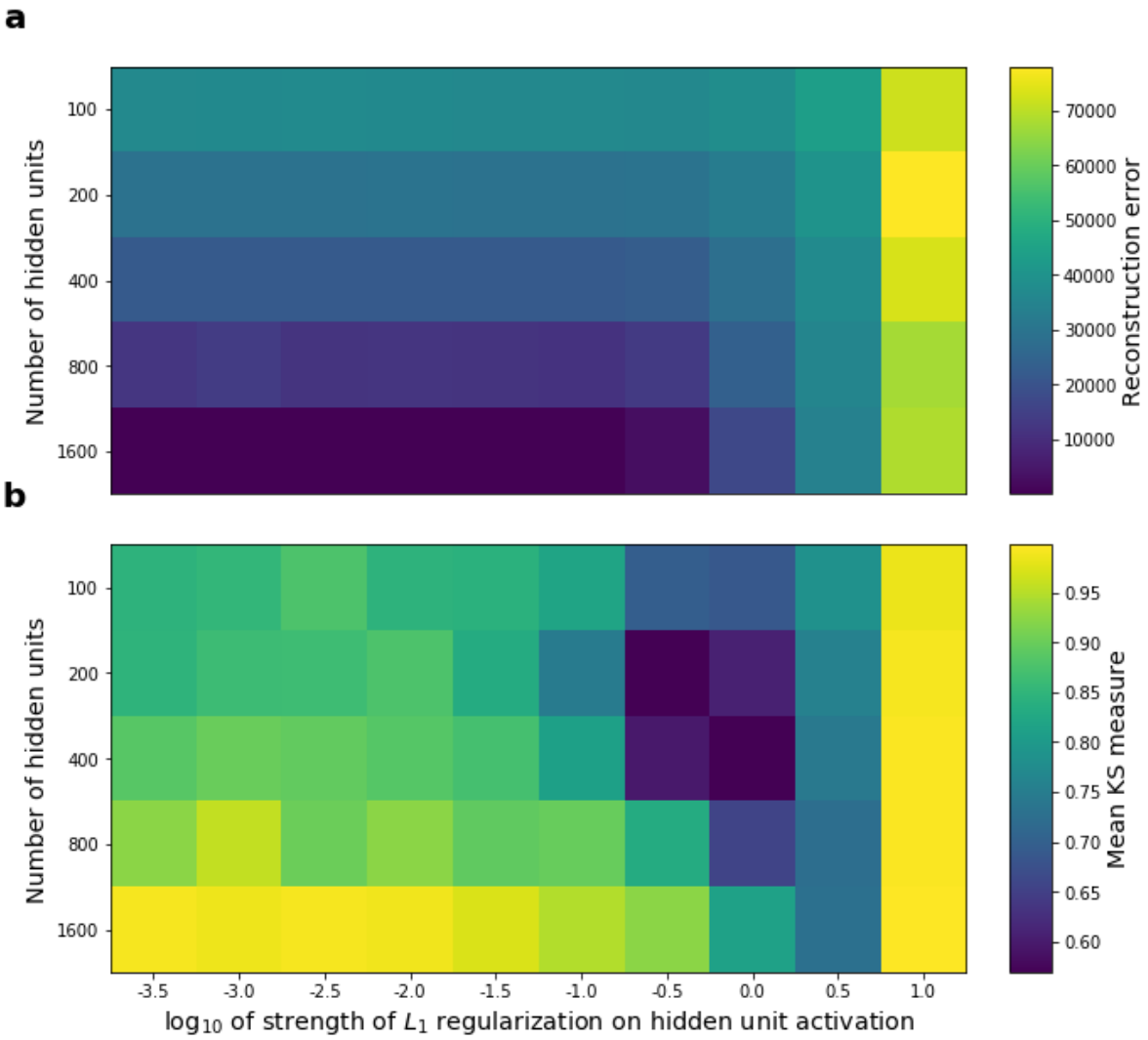


Figure 2.7 | Correspondence between sparse coding model's ability to reproduce its input and the similarity of its units' responses to those of real A1 neurons. Performance of model as a function of number of units and regularization on the weights as measured by **a,** reconstruction error on validation set at the end of training and **b,** similarity between model units and real A1 neurons. The similarity between the real and model units is measured by averaging Kolmogorov-Smirnov distance between each of the real and model distributions for the span of temporal and frequency tuning of the excitatory and inhibitory RF subfields (e.g. the distributions in Fig. 2.5d-e and Fig. 2.5g-h).

When the similarity measure was examined as a function of the same hyperparameters for the sparse coding model (Fig. 2.7), and this was compared to that model's stimulus reconstruction capacity as a function of the same hyperparameters, a monotonic relationship between stimulus reconstruction capacity and similarity of real RFs

was not found (Fig. 2.7; r^2 = -0.05, p = 0.69, n = 50). In previous studies in which comparisons have been made between normative models and real data, the model hyperparameters have been selected to maximize the similarity between the real and model RFs. In contrast, the temporal prediction model provides an independent criterion, the prediction error, to perform hyperparameter selection. To our knowledge, no such effective, measurable, independent criterion for hyperparameter selection has been proposed for other normative models of RFs.

2.6 Variants of the temporal prediction model

The change in the qualitative structure of the RFs as a function of the number of hidden units and L_1 regularization strength can be seen in the interactive supplementary figure (Supp. Fig. 2.2; https://yossing.github.io/temporal_prediction_model/figures/interactive_supplementary_figures.html) and in Supp. Figs. 2.7-2.60. The main effect of the regularization is to restrict the RFs in frequency and time. When the regularization is non-existent or substantially weaker than the optimum for prediction, the auditory RFs become more disordered, losing clear structure in most cases. When the regularization is made stronger than the optimum, the RFs become more punctate. When the regularization strength is at the optimum for prediction, the auditory model RFs qualitatively most closely resemble those of A1 neurons. This is consistent with what we found quantitatively in the previous section.

The temporal prediction model and the sparse coding model both produce RFs with spectral tuning that resembles A1 neurons. To test whether optimization for prediction implicitly optimizes for a sparse response distribution we measured the sparsity of the temporal prediction model's hidden unit activities by the Vinje-Gallant measure (Vinje and Gallant, 2000; Willmore and Tolhurst, 2001) in response to the natural sound validation set.

Examining the relationship between predictive capacity and sparsity, over the range of L_1 weight regularization strength and hidden units explored, we did not find a clear monotonic relationship. Indeed, the hidden unit and L_1 regularization combination with the best prediction had intermediate sparsity. The best-predicting model had sparsity 0.58, and other models had sparsity ranging from 0.42 to 0.69.

We also varied other characteristics of the temporal prediction model to understand their influence. When a different hidden unit nonlinearity (tanh or rectified linear) was used, the networks had similar predictive capacity and produced comparable RFs. However, when the temporal prediction model had linear hidden units, it no longer predicted as well and produced RFs that were less like real neurons in their structure. The linear model RFs generally became more narrowband in frequency with temporally extended excitation, instead of extended lagging inhibition (Supp. Fig. 2.3).

The temporal prediction model and sparse coding model results shown in the main figures of this chapter were trained on inputs with added Gaussian noise (6dB SNR), mimicking inherent noise in the nervous system. To determine the effect of adding this noise, all models were also trained without noise, producing similar results (Supp. Figs. 2.4-2.6). The results were also robust to changes in the duration of the temporal window being predicted. We trained the auditory model to predict a span of either 1, 3, 6, or 9 time steps into the future (Supp. Figs. 2.61-2.63) and found that increasing the number of time steps being predicted had little effect on the RF structure, both qualitatively and by the KS measure of similarity to the real data.

2.7 Discussion

We hypothesized that finding features that can efficiently predict future input from its past is a principle that influences the structure of sensory RFs. We implemented an artificial

neural network model that instantiates a restricted version of this hypothesis. When this model was trained using natural sounds, it produced RFs that are both qualitatively and quantitatively similar to those of A1 neurons. This similarity is particularly notable in the temporal domain; the model RFs have asymmetric envelopes, with a preference for the very recent past, as is seen in A1. Finally, the more accurate a temporal prediction model is at prediction, the more its RFs tend to be like real neuronal RFs by the measures we use for comparison.

2.7.1 Relationship to other models

A number of principles, often acting together, have been proposed to explain the form and diversity of sensory RFs. These include efficient coding (Attneave, 1954; Barlow, 1961; Brito and Gerstner, 2016; Carlson et al., 2012; Olshausen, 2003; Olshausen and Field, 1997, 1996; Srinivasan et al., 1982; Zhao and Zhaoping, 2011), sparseness (Brito and Gerstner, 2016; Carlson et al., 2012; Kozlov and Gentner, 2016; Olshausen, 2003; Olshausen and Field, 1997, 1996; van Hateren and Ruderman, 1998; van Hateren and van der Schaaf, 1998), and slowness (Carlin and Elhilali, 2013; Hurri and Hyvärinen, 2003; Hyvärinen et al., 2003; Wiskott and Sejnowski, 2002). Efficient coding indicates that sensory input should be represented as accurately as possible given certain constraints, such as spike count or energy costs. Sparseness posits that only a small proportion of neurons in the population should be active for a given input. Slowness means that neurons should be sensitive to features that change slowly over time. The temporal prediction principle we describe here provides another unsupervised objective of sensory coding. It has been described in a very general manner by the predictive information bottleneck concept (Bialek et al., 2006, 2001; Palmer et al., 2015; Salisbury and Palmer, 2016). We have instantiated a specific version of this idea, with linear-nonlinear encoding of the input, followed by a linear transform from the

encoding units' output to the prediction. In the following discussion, we describe previous normative models that infer RFs with temporal structure from auditory input and relate them to spectrotemporal RFs in A1.

2.7.2 Auditory normative models

A number of coding objectives have been explored in normative models of A1 spectrotemporal RFs. One approach (Zhao and Zhaoping, 2011) found analytically that the optimal typical spectrotemporal RF for efficient coding was spectrally localized with lagging and flanking inhibition, and showed an asymmetric temporal envelope. However, the resulting RF also showed substantially more flanking inhibition, more ringing over time and frequency, and operated over a much shorter timescale (~10ms) than seen in A1 RFs (Fig. 2.2). Moreover, this approach produced a single generic RF, rather than capturing the diversity of the population.

Other models have produced a diverse range of spectrotemporal RFs. In the sparse coding approach (Blättler et al., 2011; Brito and Gerstner, 2016; Carlson et al., 2012; Młynarski and McDermott, 2017), a spectrogram snippet is reconstructed from a sum of basis functions (a linear generative model), each weighted by its unit's activity, with a constraint to have few active units. This approach is the same as the sparse coding model we used as a control (Fig. 2.4). A challenge with many sparse generative models is that the activity of the units is found by a recurrent iterative process that needs to find a steady state; this is fine for static stimuli such as images, but for dynamic stimuli like sounds it is questionable whether the nervous system would have sufficient time to settle on appropriate activities before the stimulus had changed. Related work also used a sparsity objective, but rather than minimizing stimulus reconstruction error, forced high dispersal (Kozlov and Gentner, 2016) or decorrelation (Carlin and Elhilali, 2013; Klein et al., 2003) of neural

responses. Although lacking some of the useful probabilistic interpretations of sparse generative models, this approach does not require a settling process for inference. An alternative to sparseness is temporal slowness, which can be measured by temporal coherence (Carlin and Elhilali, 2013). Here the linear transform from sequential spectrogram snippets to unit activity is optimized to maximize the correlation of each unit's response over a certain time window, while maintaining decorrelation between the units' activities.

Although the frequency tuning derived with these models can resemble that found in the midbrain or cortex (Blättler et al., 2011; Brito and Gerstner, 2016; Carlin and Elhilali, 2013; Carlson et al., 2012; Klein et al., 2003; Kozlov and Gentner, 2016; Młynarski and McDermott, 2017) (Fig. 2.4), the resulting RFs lack the distinct asymmetric temporal profile and lagging inhibition seen in real midbrain or A1 RFs. Furthermore, they often have envelopes that are too elongated over time, often spanning the full temporal width of the spectrotemporal RF. This is related to the fact that the time window to be encoded by the model is set arbitrarily, and every time point within that window is given equal weighting, i.e. the direction of time is not accounted for. This is in contrast to the temporal prediction model, which naturally gives greater weighting to time-points near the present than the past due to their greater predictive capacity.

2.7.3 Conclusion

In this chapter we introduced the temporal prediction model and showed that when trained on natural auditory inputs it captures many of the properties of A1 neuronal RFs. We have also examined how the model's results compare to those of other normative models of auditory RFs. In the next chapter we apply the same temporal prediction model to visual inputs and compare the model RFs to those of V1 simple cells. Then, in addition to comparing our visual model results to those of other models, we take a broader perspective,

providing a critical evaluation of temporal prediction as we have instantiated it and examining future avenues for development of the model, including incorporating hierarchical structure, which is done in Chapter 4.

2.8 Methods

2.8.1 Data used for model training and testing

Auditory inputs

Auditory stimuli were compiled from databases of human speech (~60%), animal vocalizations (~20%) and sounds from inanimate objects found in natural settings (e.g. running water, rustling leaves; ~20%). Stimuli were recorded using a Zoom H4 or collected from online sources. Natural sounds were obtained from www.freesound.org, contributed by users sedi, higginsdj, jult, kvgarlic, xenognosis, zabuhailo, funnyman374, videog, j-zazvurek, samueljustice00, gfrog, ikbenraar, felix-blume, orbitalchiller, saint-sinner, carlvus, kyster, vflefevre, hitrison, willstepp, timbahrij, xdimebagx, r-nd0mm3m, the-yura, rsilveira-88, stomachache, foongaz, edufigg, yurkobb, sandermotions, darius-kedros, freesoundjon-01, dwightsabeast, borralbi, acclivity, J.Zazvurek, Zabuhailo, soundmary, Darius Kedros, Kyster, urupin, RSilveira and freelibras. Human speech sounds were obtained from http://databases.forensic-voice-comparison.net/(Morrison et al., 2015, 2012).

Each sound was sampled at (or resampled to) 44.1 kHz and converted into a simple 'cochleagram', to make it more analogous to the activity pattern that would be passed to the auditory pathway after processing by the cochlea. To calculate the cochleagram, a power spectrogram was computed using 10ms Hanning windows, overlapping by 5ms (giving time steps of 5ms). The power across neighbouring Fourier frequency components was then aggregated into 32 frequency channels using triangular windows with a base width of 1/3

octave whose center frequencies ranged from 500 to 17,827Hz (1/6th octave spacing, using code adapted from melbank.m, http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html). The cochleagrams were then decomposed into sequential overlapping clips, each of 43 time steps (415 ms) in duration, providing a training set of ~1,000,000 clips (~1.3 hours of audio) and a validation set of ~200,000 clips. To approximately model the intensity compression seen in the auditory nerve (Sachs and Abbas, 1974), each frequency band in the stimulus set was divided by the median value in that frequency band over the training set, and passed through a hill function, defined as $h(x) = cx/(1+cx)$ with c=0.02. Finally, the training and cross-validation sets were normalized by subtracting the mean and dividing by the standard deviation over all time steps, frequency bands and clips in the training set. The first 40 time steps (200 ms) of each clip (the 'past') were used as inputs to the neural network, whose aim was to predict the content of the remaining 3 time steps (15 ms; the 'future').

Addition of Gaussian noise

To replicate the effect of noise found in the nervous system, Gaussian noise was added to the auditory inputs with a signal-to-noise ratio (SNR) of 6dB. This improved the similarity between the model RFs and the data. The results from training the network on inputs without added noise are shown for auditory inputs in Supp. Figs. 2.4, 2.6. The results from the sparse coding model were similar in both cases for inputs with and without noise (Supp. Figs. 2.5, 2.6).

2.8.2 Temporal prediction model

The model and cost function

The temporal prediction model was implemented using a standard fully connected feedforward neural network with one hidden layer. Each hidden unit in the network computed the sum of linearly weighted inputs, and its output was determined by passing this sum through a monotonic nonlinearity. This nonlinearity $s = f(a)$ was either a logistic function $f(a) = 1/(1+\exp(-a))$ or a similar nonlinear function (such as tanh). For results reported here, we used the logistic function, but obtained similar results when we trained the model using $f(a) = \tanh(a)$. For comparison, we also trained the model replacing the nonlinearity with a linear function, where $f(a) = a$. In this case, we found that the RFs tended to be punctate in space or frequency and did not typically show the alternating excitation and inhibition characteristic of real neurons in A1.

Formally, for a network with $i = 1$ to I input variables, $k = 1$ to K output units and a single layer of $j = 1$ to J hidden units, the output s_{jn} of hidden unit j for clip n is given by:

$$s_{jn} = \mathrm{f}\left(b_j + \sum_{i=1}^{I} w_{ji} u_{in}\right) \tag{2.1}$$

The value u_{in} of input variable i for clip n is simply the value for a particular frequency band and time step of the 'past' of cochleagram clip n ($I = 32$ frequencies×40 time steps= 1280). Hence, the index i spans over several frequencies and also over time steps into the past. The subscript n has been dropped for clarity in the figures (Fig. 2.1). The parameters in Equation 2.1 are the connective input weights w_{ji} (between each input variable i and hidden unit j) and the bias b_j (of hidden unit j).

The activity $\hat{v}_{kn}$ of each output unit k, which is the estimate of the true future v_{kn} given the past u_{in}, is given by:

$$\hat{v}_{kn} = b_k + \sum_{j=1}^{J} w_{kj} s_{jn} \tag{2.2}$$

The parameters in Equation 2.2 are the connective output weights w_{kj} (between each hidden unit j and output unit k) and the bias b_k (of output unit k). The activity $\hat{v}_{kn}$ of output unit k for clip n is the estimate for a particular frequency band and time step of the 'future' in the auditory case ($K = 32 \times 3 = 96$).

The parameters w_{ji}, w_{kj}, b_j, and b_k were optimized for the training set by minimizing the cost function given by:

$$E = \frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} (\hat{v}_{kn} - v_{kn})^2 + \lambda \left(\sum_{i=1}^{I} \sum_{j=1}^{J} |w_{ji}| + \sum_{j=1}^{J} \sum_{k=1}^{K} |w_{kj}| \right) \tag{2.3}$$

Thus, E is the sum of the squared error between the prediction $\hat{v}_{kn}$ and the target v_{kn} over all N training examples, plus an L_1 regularization term, which is proportional to the sum of absolute values of all weights in the network and its strength is determined by the hyper-parameter λ. This regularization tends to drive redundant weights to near zero and provides a parsimonious network.

Implementation details

The networks were implemented in Python (https://lasagne.readthedocs.io/en/latest/; http://deeplearning.net/software/theano/). The objective function was minimized using backpropagation as performed by the Adam optimization method (Kingma and Ba, 2014).

An alternative implementation of the model was also made in MATLAB using the Sum-of-Functions Optimizer (Sohl-Dickstein et al., 2014) (https://github.com/Sohl-Dickstein/Sum-of-Functions-Optimizer) to train the network using backpropagation. Training examples were split into minibatches of approximately 7000 training examples each.

During model network training, several hyperparameters were varied, including the regularization strength (λ), the number of units in the hidden layer and the nonlinearity used by each hidden unit. For each hyperparameter setting, the training algorithm was run for 1000 iterations. Running the network for longer (10000 iterations) showed negligible improvement to the prediction error (as measured on the validation set) or change in RF structure.

The effect of varying the number of hidden units and λ on the prediction error for the validation set is shown in Fig. 2.6. The results presented are the networks that predicted best on the validation set after 1000 iterations through the training data. The settings that resulted in the best prediction were 1600 hidden units and $\lambda = 10^{-3.5}$.

Model receptive fields

In the model, the combination of linear weights and nonlinear activation function are similar to the basic linear non-linear (LN) model (Atencio et al., 2008; Chichilnisky, 2001; Rabinowitz et al., 2011; Simoncelli et al., 2004) commonly used to describe neural RFs. Hence, the input weights between the input layer and a hidden unit of the model network are taken directly to represent the unit's RF, indicating the features of the input that are important to that unit.

Because of the symmetric nature of the sigmoid function, f(a) = 1-f(-a), with appropriate modification of the biases, a hidden unit has the same influence on the prediction if its input and output matrices are both multiplied by -1. That is, for unit j, if we convert w_{ij}

to $-w_{ij}$, w_{jk} to $-w_{jk}$, b_j to $-b_j$, and b_k to $-b_k + w_{jk}$, this will have no effect on the prediction or the cost function. This can be done independently for each hidden unit. Hence, the sign of each unit's RF could equally be positive or negative and have the same result on the predictions given by the network. However, we know that auditory units always have leading excitation (Fig. 2.2). Hence, for both the predictive model and for the sparse coding model, we assume leading excitation for each unit. This was done for all analyses.

As more units are added to the model network, the number of inactive units increases. To account for this, we measured the relative strength of all input connections to each hidden unit by summing the square of all input weights for that unit. Units for which the sum of square input weights was <1% of the maximum strength for the population were deemed to be inactive and excluded from all subsequent analyses. The difference in connection strength between active and inactive units was very distinct; a threshold of 0.0001% only marginally increases the number of active units.

2.8.3 Sparse coding model

The sparse coding model was used as a control. The Python implementation of this model (https://github.com/zayd/sparsenet) was trained using the same auditory inputs used to train the predictive model. The training data were divided into mini-batches which were shuffled and the model optimized for one full pass through the data. Inference was performed using the Fast Iterative Shrinkage and Thresholding (FISTA) algorithm. A sparse L_1 prior with strength λ, was applied to the unit activities. A range of λ-values and unit numbers were tried (Fig. 2.7). The learning rate and batch size were also varied until reasonable values were found. As there was no independent criterion by which to determine the 'best' settings, we chose the network that produced basis functions whose receptive fields were most similar to those of real neurons. This was determined using the mean KS measure of similarity (Fig.

2.7). The model configuration was restricted to an overcomplete condition (having more units than the number of input variables) in order to remain consistent with previous instantiations of this model (Carlson et al., 2012; Olshausen and Field, 1997, 1996). In this manner, we selected a sparse coding network with 1600 units, $\lambda=10^{0.5}$, learning rate = 0.01 and 100 mini-batches in the auditory case (Figs. 2.4-2.5). Although these sparse coding basis functions are projective fields, they tend to be similar in structure to receptive fields (Olshausen and Field, 1997), and, for simplicity, are referred to as RFs.

2.8.4 Auditory receptive field analysis

In vivo A1 RF data

We used previously recorded (Willmore et al., 2016) auditory RFs of neurons in the primary auditory cortical fields, A1 and AAF, of 5 anesthetized ferrets as a basis of comparison for the RFs of model units trained on auditory stimuli. Systematic differences in response properties of A1 and AAF neurons are minor and not relevant for this study, and for simplicity here, we refer to neurons from either primary field indiscriminately as "A1 neurons". The neural responses were recorded using multi-electrode arrays (Neuronexus), the methods are detailed in an earlier study (Willmore et al., 2016), and described briefly below. The neuronal recordings used the "BigNat" stimulus set (Willmore et al., 2016), which consists of natural sounds, including animal vocalizations (e.g., ferrets and birds), environmental sounds (e.g., water and wind), and speech. To identify those neural units that were driven by the stimuli, we calculated a "noise ratio" statistic (Rabinowitz et al., 2011; Sahani and Linden, 2003) for each unit and excluded from further analysis any neurons with a noise ratio >40. In total, driven spiking responses of 114 units (75 single unit, 39 multi-unit) were recorded in response to this stimulus set. Then, the auditory (spectrotemporal) RF of each unit was constructed using a previously described method (Willmore et al., 2016).

Briefly, linear regression was performed in order to minimize the squared error between each neuron's spiking response over time and the cochleagram of the stimuli that gave rise to that response. The method used was exactly the same as in our earlier study (Willmore et al., 2016), except that L_1 rather than L_2 regularization was used to constrain the regression. The spectrotemporal RFs of these neurons took the same form as the inputs to the model neural network (i.e., 32 frequencies and 40 time steps over the same range of values) and were therefore comparable to the model units' RFs. In order to account for the latency of auditory cortical responses, the first two time steps (10ms) nearest the present were removed from the real neural RFs, leaving 38 time steps.

Multi-Dimensional Scaling (MDS)

To obtain a non-parametric indication of how similar the model units' RFs were to those of real A1 neurons, each RF was embedded into a 2-dimensional space using classical multidimensional scaling (MDS; https://scikit-learn.org/stable/modules/generated/sklearn.manifold.MDS.html; Fig. 2.5a and Supp. Fig. 2.6a). First, 100 units each from the temporal prediction and sparse coding models and from the real population were chosen at random. To ensure that the model RFs were of the same dimensionality prior to embedding, we removed the two time steps furthest into the past for each model RF. Next, classical MDS was performed. Briefly, classical MDS involves computing a squared proximity matrix, which consists of the squared Euclidian distances between every pair of RFs. This matrix is then centered and the eigenvalues and eigenvectors computed. The two largest eigenvalues and eigenvectors are kept, resulting in a 2-dimensional embedding.

Measuring temporal and frequency spans of RFs

We quantified the span, over time and frequency, of the excitatory and inhibitory subfields of each RF. To do this, each RF was first separated into excitatory and inhibitory subfields, where the excitatory subfield was the RF with negative values set to 0, and the inhibitory subfield the RF with positive values set to 0. In some cases, model units did not exhibit notable inhibitory subfields. To account for this, the power contained in each subfield was calculated (sum of the squares of the subfield). Inhibitory subfields with < 5% of the power of that unit's excitatory subfield were excluded from further analysis. According to this criterion, 44 of 167 active units in the temporal prediction model and 193 of 1600 units in the sparse model did not display inhibition.

Singular value decomposition (SVD) was performed on each subfield separately, and the first pair of singular vectors was taken, one of which is over time, the other over frequency. For the excitatory subfield, the temporal span was measured as the proportion of values in the temporal singular vector that exceeded 50% of the maximum value in the vector. The same analysis provided the temporal span for the inhibitory subfield. Similarly, we measured the frequency spans of the RFs by applying this measure to the frequency singular vectors of the excitatory and the inhibitory subfields.

We also examined, for both real and model RFs, the mean power for each of the 38 time steps in the RFs (Fig. 2.5b), which was calculated as the mean of the squared RF values, over all frequencies and RFs, at each time step.

Mean KS measure

To compare each network's units with those recorded in A1 (Fig. 2.2), the two-sample Kolmogorov-Smirnov (KS) distance between the real and model distribution was measured for both the temporal and spectral span of the excitatory and inhibitory subfields (e.g. the

distributions in Fig. 2.5d-e and Fig. 2.5g-h). These four KS measures were then averaged to give a single mean KS measure for each network, indicating how closely the temporal and frequency characteristics of real and model units matched on average for that network. The KS measure is low for similar distributions and high for distributions that diverge greatly. Thus, networks whose units display temporal and frequency tuning characteristics that match those of real neurons more closely give rise to a lower mean KS measure.

Chapter 3 | Temporal prediction model of natural video inputs reproduces properties of V1 simple cells

We trained a temporal prediction model identical to the one presented in the previous chapter, but this time using natural video clips instead of natural sounds (Fig. 3.1). Networks were trained by optimizing their synaptic weights using backpropagation (see section 1.7.1) to most accurately predict the immediate future of the stimulus from its very recent past. Here, the inputs were patches of videos of animals moving in natural settings, and we trained the network to predict the pixel values for one movie frame (40 ms) into the future, based on the 7 most recent frames (280 ms). After training we examined the model receptive fields and compared their properties to those of V1 simple cells.

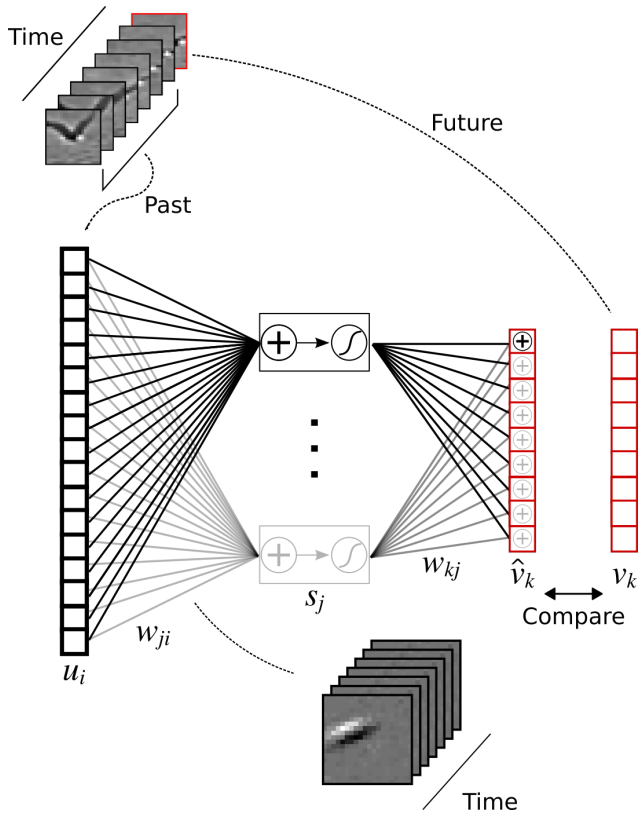


Figure 3.1 | (previous page) Temporal prediction model trained on natural visual inputs. The network was trained on movie clips of natural scenes, with the aim of predicting the immediate future frame of each clip from recent past frames. u_i, input – the past; w_{ji}, input weights; s_j, hidden unit output; w_{kj}, output weights; $\hat{v}_{kn}$, output – the predicted future; v_k, target output – the true future. Hidden unit's RF is the w_{ji} between the input and that unit j.

3.1 Qualitative assessment of visual receptive fields

By eye, substantial similarities were apparent when we compared the temporal prediction model's RFs trained using visual inputs (Fig. 3.2b) with the 3D (space-space-time) and 2D (space-time) spatiotemporal RFs of real V1 simple cells (Fig 3.2a), which were obtained from Ohzawa et al (Ohzawa et al., 1996). Simple cells (Hubel and Wiesel, 1959) have stereotyped RFs containing parallel, spatially localized excitatory and inhibitory regions, with each cell having a particular preferred orientation and spatial frequency (Fig 3.2a) (DeAngelis et al., 1993; Jones and Palmer, 1987b; Ringach, 2002). These features are also clearly apparent in the model RFs (Fig. 3.2b).

Unlike previous models (Hyvärinen et al., 2003; Olshausen, 2003; van Hateren and Ruderman, 1998), the temporal prediction model captures the temporal asymmetry of real RFs. The RF power is highest near the present and decays into the past (Fig. 3.2b), as observed in real neurons (Ohzawa et al., 1996) (Fig. 3.2a). Furthermore, simple cell RFs have two types of spatiotemporal structure: space-time separable RFs (Fig. 3.2aI), whose optimal stimulus resembles a flashing or slowly ramping grating, and space-time inseparable RFs, whose optimal stimulus is a drifting grating (DeAngelis et al., 1993) (Fig. 3.2aII). Our model captures this diversity (Fig. 3.2bI-III separable, Fig. 3.2bIV-VI inseparable).

We also examined linear aspects of the tuning of the output units for the visual temporal prediction model using a response-weighted average to white noise input, and found punctate non-oriented RFs that decay into the past.

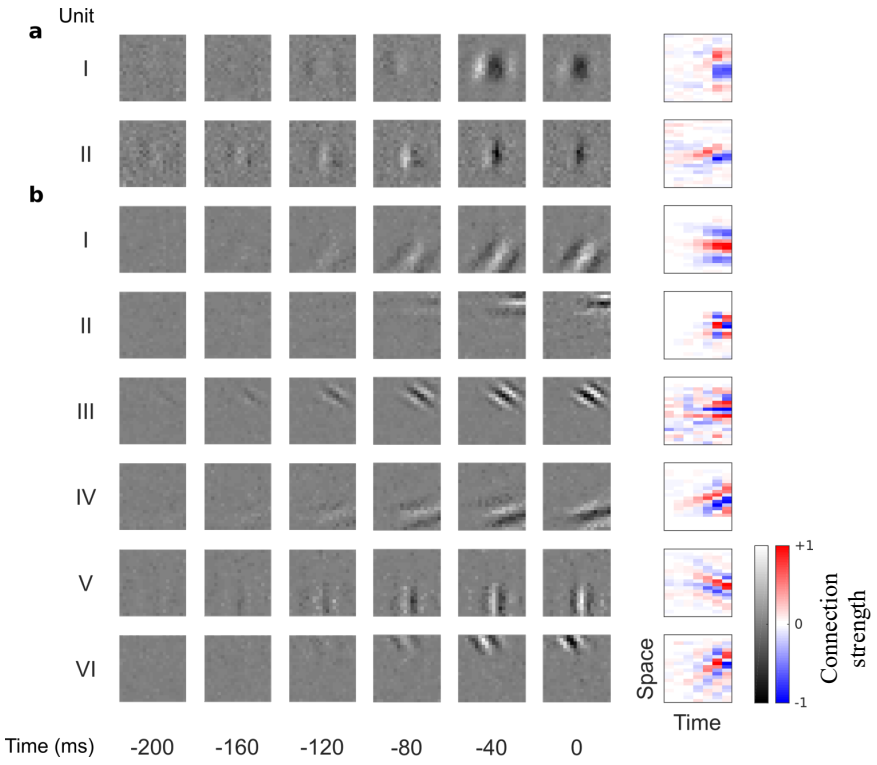


Figure 3.2 | Visual spatiotemporal RFs of real neurons and temporal prediction model units. a, Example spatiotemporal (I, space-time separable, and II, space-time inseparable) RFs of real V1 neurons. Left, grayscale: 3D (space-space-time) spatiotemporal RFs showing the spatial RF at each of the most recent 6 time steps. Most recent time step (40ms) was removed to account for conduction delay. White – excitation, black – inhibition. Right: corresponding 2D (space-time) spatiotemporal RFs obtained by summing along the unit's axis of orientation for each time step. Red – excitation, blue – inhibition. **b,** Example 3D and corresponding 2D spatiotemporal (I-III, space-time separable, and IV-VI, space-time inseparable) RFs of model units when the temporal prediction model is trained to predict the future of natural visual inputs.

3.2 Qualitative comparison to other models

As with the auditory model in the previous chapter, we trained the sparse coding model on the dataset of visual inputs to serve as a control for the temporal prediction model trained on these same inputs. We compared the full population of spatial and 2D spatiotemporal visual RFs of the temporal prediction model (Figs. 3.3, 3.4) and the sparse coding model (Figs. 3.5, 3.6). As shown in previous studies (Bell and Sejnowski, 1997; Olshausen and Field, 1997, 1996; van Hateren and Ruderman, 1998; van Hateren and van der Schaaf, 1998), the sparse

coding model produces RFs whose spatial structure resembles that of V1 simple cells (Fig. 3.5; Supp. Fig. 3.1a-c), but does not capture the asymmetric nature of the temporal tuning of V1 neurons (Fig 3.6; Supp. Fig. 3.1b-c). Furthermore, while it does produce examples of both separable (Supp. Fig. 3.1cI-III) and inseparable (Supp. Fig. 3.1cIV-VI) spatiotemporal RFs, those that are separable tend to be completely stationary over time, resembling immobile rather than flashing gratings (Fig. 3.6; Supp. Fig. 3.1cI-III).

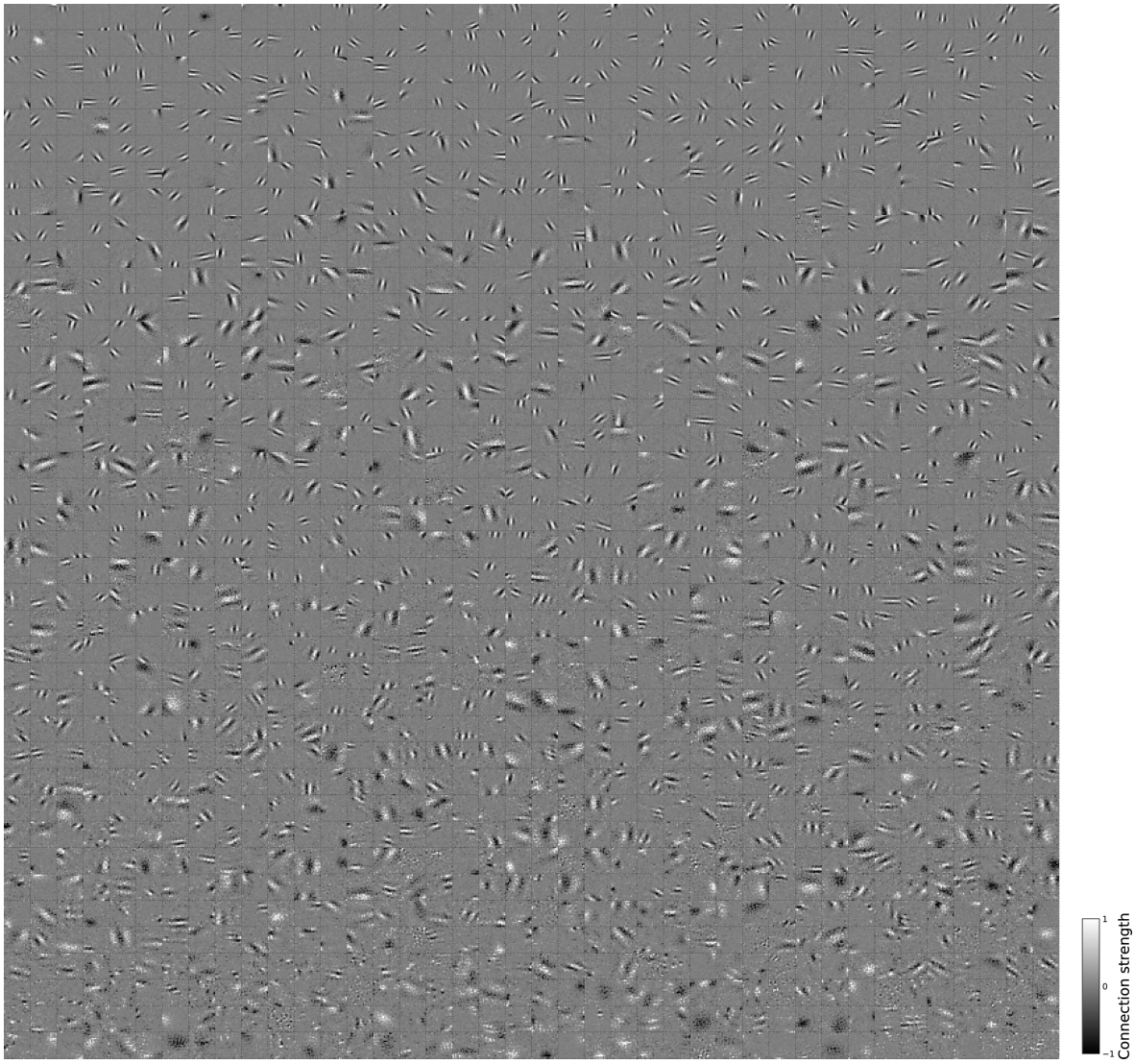


Figure 3.3 | Full set of visual spatial RFs of the temporal prediction model units. Model units were obtained by training the model with 1600 hidden units on visual inputs. The hidden unit number and L_1 weight regularization strength ($10^{-6.25}$) were chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Each square represents the spatial RF

of a single unit, shown at its best time step. The best time step was determined by selecting the time step for which the power (sum of squares) of the RF was greatest. White – excitation, black - inhibition.

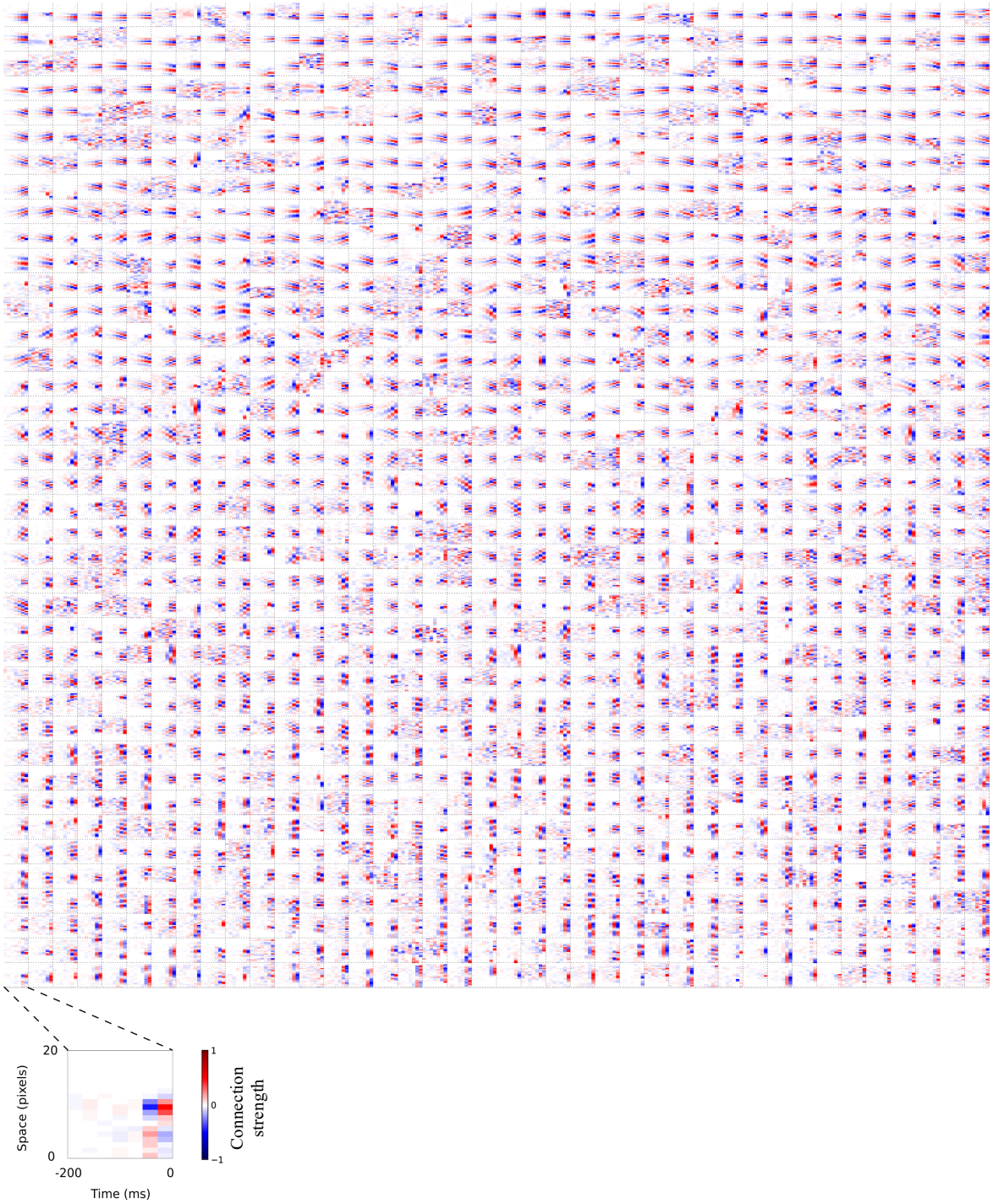


Figure 3.4 | Visual 2D (space-time) spatiotemporal RFs of temporal prediction model units. Each square represents the 2D spatiotemporal RF of a single unit corresponding to the unit in the same position in Fig. 3.3, obtained by summing across space along the axis of the orientation for that unit. Red – excitation, blue - inhibition. Inset shows axes.

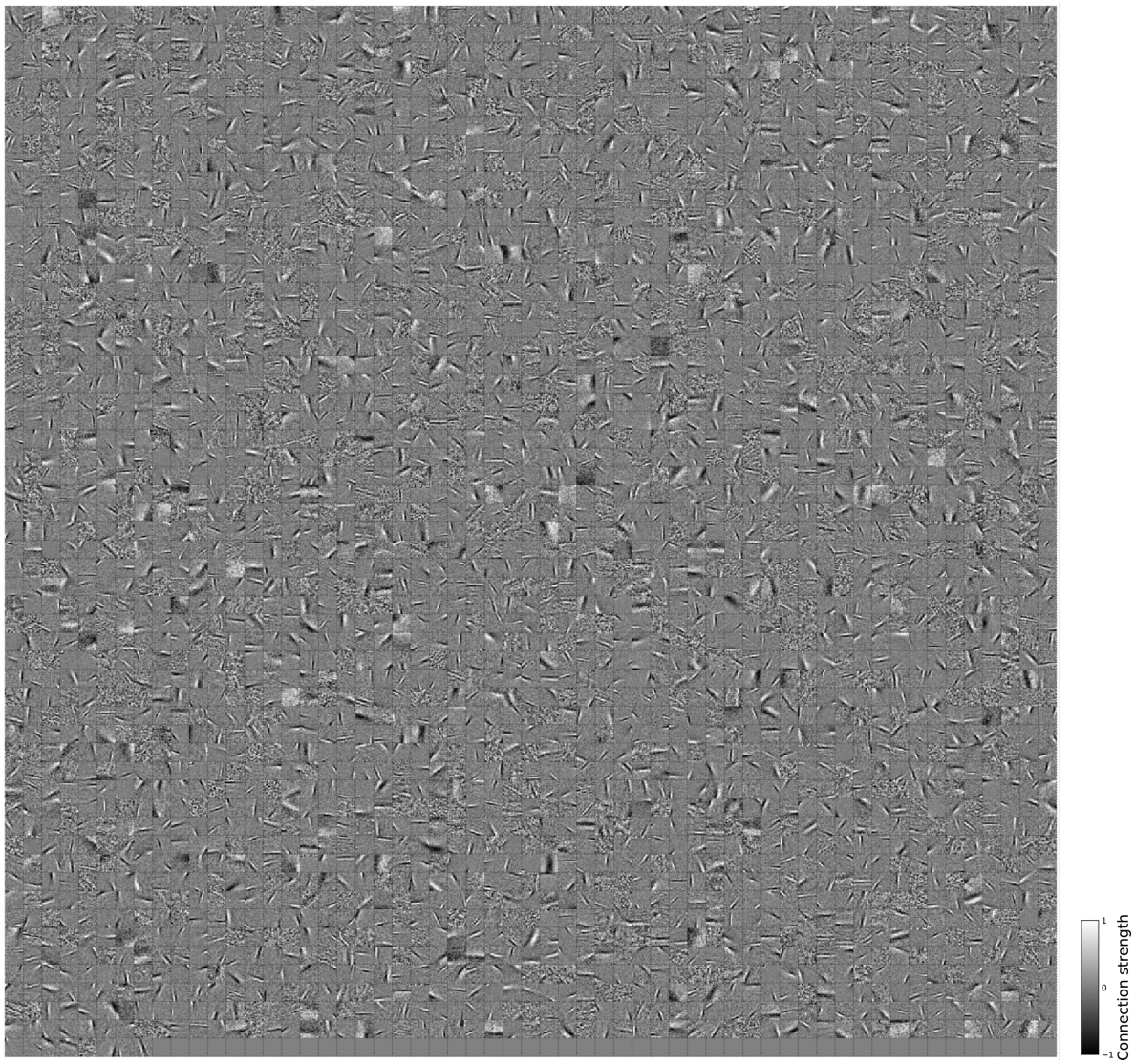


Figure 3.5 | Full set of visual spatial RFs of sparse coding model units. Model units were obtained by training the sparse coding model with 3200 units on identical visual inputs used to train the temporal prediction model. The model configuration (3200 units, L_1 sparsity strength of $10^{0.5}$ on the unit activities) was chosen because this produced the RFs that look most like the RFs of V1 simple cells as determined by visual inspection. We restricted model configurations to the overcomplete case. Each square represents the spatial RF of a single unit, shown at its best time step. The best time step was determined by selecting the time step for which the power (sum of squares) of the RF was greatest. White – excitation, black - inhibition.

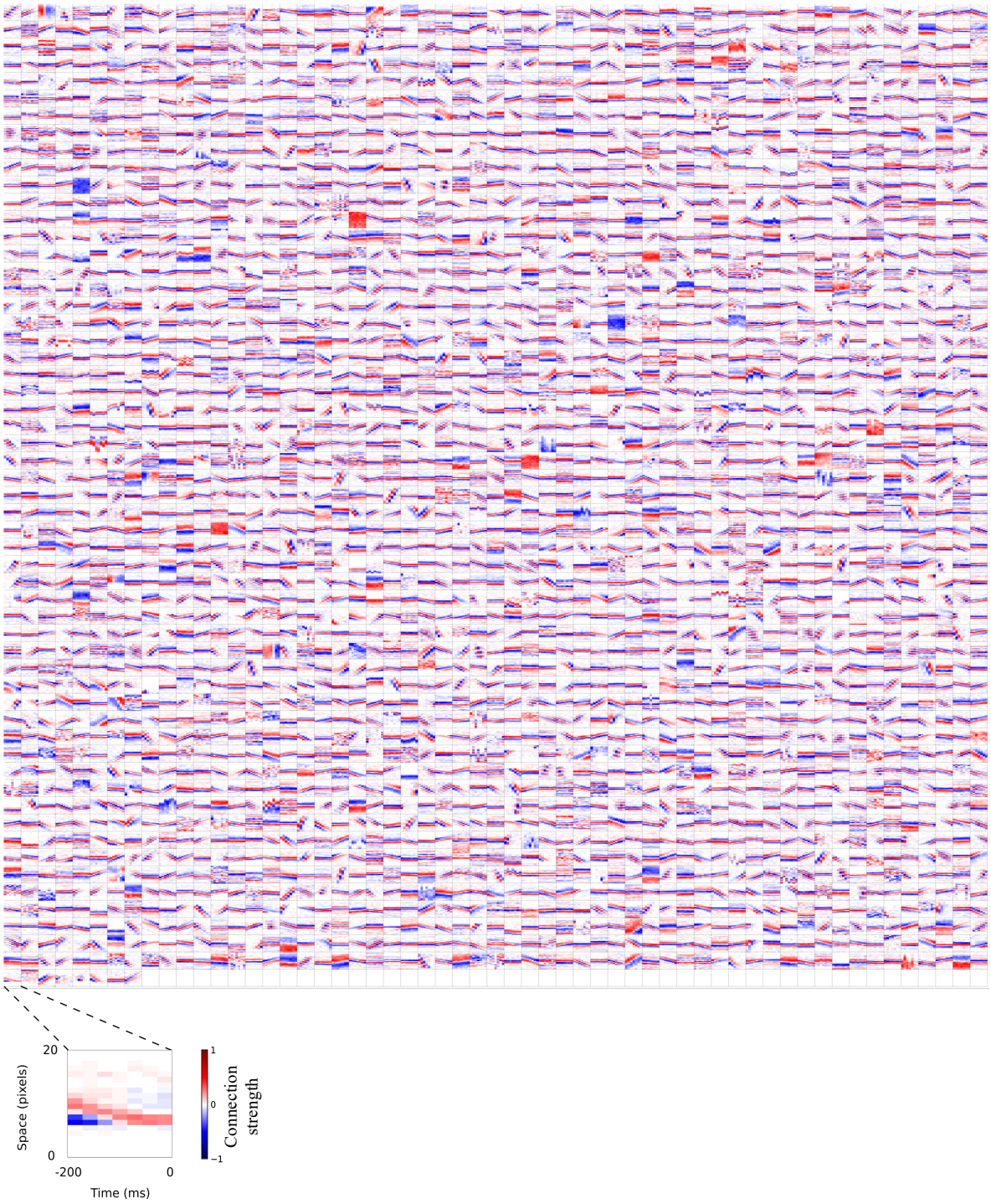


Figure 3.6 | 2D (space-time) visual spatiotemporal RFs of sparse coding model units. Each square represents the 2D spatiotemporal RF of a single unit corresponding to the unit in the same position in Fig. 3.5, obtained by summing across space along the axis of the orientation for that unit. Red – excitation, blue - inhibition. Inset shows axes.

3.3 Quantitative analysis of visual RFs

We also compared the spatiotemporal RFs derived from the temporal prediction and sparse coding models with restricted published datasets summarizing RF characteristics of V1 neurons (Ringach, 2002) and a small number of full spatiotemporal visual RFs acquired from Ohzawa et al (Ohzawa et al., 1996). We assessed the orientation and spatial frequency tuning properties of the models' RFs by fitting Gabor functions to their RFs (see Methods).

We compared temporal properties of the RFs from the neural data and the temporal prediction model. In both cases, most power (sum of square of RFs) is in the most recent time steps (Fig. 3.7a). Previous normative models of spatiotemporal RFs (Hyvärinen et al., 2003; Olshausen, 2003; van Hateren and Ruderman, 1998; Fig. 3.8a) do not show this property, being either invariant over time or localized, but with a symmetric profile that is not restricted to the recent past. While the mean power across the population of model RFs is highest at the most recent time-step and decays into the past, in the neural data, most power is contained within the second most-recent time step, after which it decays into the past (Fig 3.7a). It is possible that real neurons place less weight on the very immediate past time-step in order to prevent simple extrapolation from the most recent input, which might cause overfitting. Placing more emphasis on information further into past could help to produce more accurate predictions over a large set of natural inputs. In this case, introducing additional constraints to the network to encourage this strategy or increasing the size of the training set could help the model to capture this feature of the biology.

We also measured the space-time separability of the RFs of the temporal prediction model (see Methods); substantial numbers of both space-time separable and inseparable units were apparent (631 separable, 969 inseparable; Fig. 3.4). In addition to this, we measured the tilt direction index (TDI) of the model units from their 2D spatiotemporal RFs.

This index indicates spatiotemporal asymmetry in space-time RFs and correlates with direction selectivity (Anzai et al., 2001; Baker, 2001; DeAngelis et al., 1993; Livingstone and Conway, 2007; Pack et al., 2006). The mean TDI for the population was 0.33 with 0.29 SD, comparable with the ranges in the neural data (0.16 with 0.12 SD in cat area 17/18 (Baker, 2001), 0.51 with 0.30 SD in macaque V1 (Livingstone and Conway, 2007)). Finally, we observed an inverse correlation (r^2 = -0.31, $p < 10^{-9}$, n = 1205) between temporal and spatial frequency tuning (See Methods), which is also a property of real V1 RFs (DeAngelis et al., 1993) and is seen in a sparse-coding-related model (van Hateren and Ruderman, 1998).

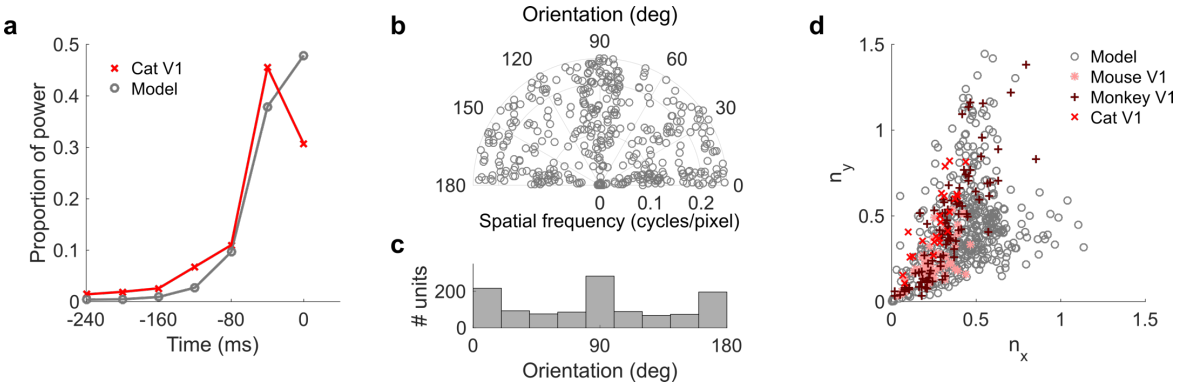


Figure 3.7 | Population measures for real V1 and temporal prediction model spatial and spatiotemporal RFs. a, Proportion of power (sum of squared weights over space and averaged across units) in each time step, for real (Ohzawa et al., 1996) and model populations. **b**, Joint distribution of spatial frequency and orientation tuning for population of model unit RFs at their time step with greatest power. **c,** Distribution of orientation tuning for population of model unit RFs at their time step with greatest power. **d**, Distribution of RF shapes for real neurons (cat (Jones and Palmer, 1987b), mouse (Niell and Stryker, 2008) and monkey (Ringach, 2002)) and model units. n_x and n_y measure RF span parallel and orthogonal to orientation tuning, as a proportion of spatial oscillation period (Ringach, 2002). For **b-d**, only units that could be well approximated by Gabor functions (n = 1205 units; see Methods) were included in the analysis. Of these, only model units that were space-time separable (n = 473) are shown in **d** to be comparable with the neuronal data (Ringach, 2002). A further 4 units with 1.5<n_y<3.1 are not shown in **d**.

Looking at the spatial tuning characteristics of the temporal prediction model's RFs, they display a wide range of orientation and spatial frequency preferences, consistent with the neural data (DeAngelis et al., 1993; Kreile et al., 2011) (Fig. 3.3, Fig. 3.7b-c). Both model

and real RFs (Kreile et al., 2011) show a preference for spatial orientations along the horizontal and vertical axes, although this orientation bias is seen to a greater extent in the temporal prediction model than in the data. The orientation and frequency tuning characteristics are also well captured by sparse coding related models of spatiotemporal RFs (Olshausen, 2003; van Hateren and Ruderman, 1998; Fig. 3.8b-c). The widths and lengths of the RFs of the temporal prediction model, relative to the period of their oscillation, also match the neural data well (Fig. 3.7d). The distribution of units extends along a curve from blob-like RFs, which lie close to the origin in this plot, to stretched RFs with several subfields, which lie further from the origin. A small proportion of the model population have RFs with several short subfields, forming a wing from the main curve in Fig. 3.7d. Although the relative widths and lengths of the RFs are again fairly well captured by previous models (Berkes et al., 2009; Olshausen and Field, 1997, 1996; van Hateren and van der Schaaf, 1998; Fig. 3.8d), only the temporal prediction model seems to be able to capture the blob-like RFs that form a sizeable proportion of the neural data (Ringach, 2002; Fig. 3.7d, 3.8d where n_x and n_y < ~0.25).

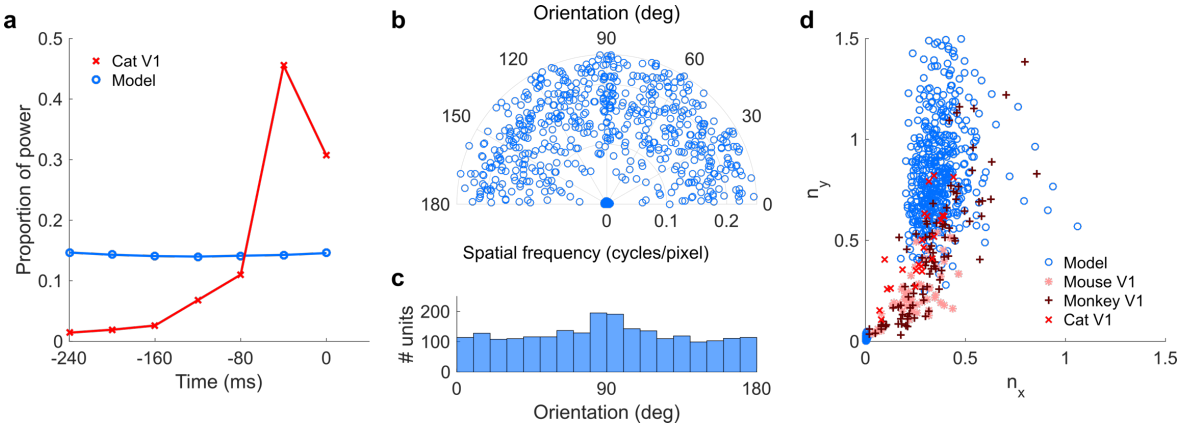


Figure 3.8 | Visual RFs and population measures for real V1 neurons and sparse coding model units. a, Proportion of power (sum of squared weights over space and averaged across units) in each time step, for real (Ohzawa et al., 1996) and model populations. **b**, Joint distribution of spatial frequency and orientation tuning for population of model units at their time step with greatest power. **c,** Distribution of orientation tuning for population of sparse coding model unit RFs at their time step with greatest power. **d**, Distribution of RF shapes for real neurons (cat (Jones and Palmer, 1987b),

mouse (Niell and Stryker, 2008) and monkey (Ringach, 2002)) and model units. For **b-d**, only units that could be well approximated by Gabor functions (n = 2402 units; see Methods) were included in the analysis. Of these, only model units that were space-time separable (n = 881) are shown in **d** to be comparable with the neuronal data (Ringach, 2002).

3.4 Variants of the temporal prediction model

The change in the qualitative structure of the RFs as a function of the number of hidden units and L_1 regularization strength, for the visual model, can be seen in the interactive supplementary figure (Supp. Fig. 3.2; https://yossing.github.io/temporal_prediction_model/figures/interactive_supplementary_figures.html and Supp. Figs. 3.9-3.28). The main effect of the regularization is to restrict the RFs in space. When the regularization is non-existent or substantially weaker than the optimum for prediction, the visual RFs become less localized in space with more elongated bars. When the regularization is made stronger than the optimum, the RFs become more punctate. When the regularization strength is at the optimum for prediction, the visual model RFs qualitatively most closely resemble those of V1 simple cells

The temporal prediction model and the sparse coding model both produce oriented Gabor-like RFs when trained on visual inputs. This raises the possibility that optimization for prediction implicitly optimizes for a sparse response distribution, and hence leads to oriented RFs. To test for this, we measured the sparsity of the temporal prediction model's hidden unit activities by the Vinje-Gallant measure (Vinje and Gallant, 2000; Willmore and Tolhurst, 2001) in response to the natural video validation set. Examining the relationship between predictive capacity and sparsity, over the range of L_1 weight regularization strength and hidden units explored, we did not find a clear monotonic relationship. Indeed, the hidden unit and L_1 regularization combination with the best prediction had intermediate sparsity.

For the visual case, the best-predicting model had sparsity 0.25, and other models within the grid search had sparsity ranging from 0.16 to 0.57.

We also varied other characteristics of the temporal prediction model to understand their influence. When a different hidden unit nonlinearity (tanh or rectified linear) was used, the networks had similar predictive capacity and produced comparable RFs. However, when the temporal prediction model had linear hidden units, it no longer predicted as well and produced RFs that were substantially less similar to real V1 neurons in their structure. At low regularization (the best predicting case), the RFs formed full-field grid-like structures. At higher regularization, they were more punctate, with some units having oriented RFs with short subfields. The RFs also did not change form or polarity over time, but simply decayed into the past.

In order to investigate the effect of the duration of the window being predicted, we trained the temporal prediction model to predict 1, 3 or 6 time steps into the future (Figs 3.2-3.4, Supp Figs. 3.29-3.30). Gabor-like units were present in all cases. Increasing the number of time steps made the RFs more restricted in space and increased the proportion of blob-like RFs.

The temporal prediction model and sparse coding model results shown in Figs. 3.2-3.8 were obtained by training on inputs with added Gaussian noise (6dB SNR), mimicking inherent noise in the nervous system. To determine the effect of adding this noise, all models were also trained without noise, producing similar results (Supp. Figs. 3.3 – 3.8).

3.5 Discussion

We built on the results in the previous chapter by training a single layer temporal prediction model (with the identical model architecture to that used in the previous chapter), but using natural visual inputs. When trained on natural movies the model produced RFs with many

of the properties of V1 simple cells. This similarity is particularly notable in the temporal domain; the model RFs have asymmetric envelopes, with a preference for the very recent past, as is seen in V1. We also quantitatively compared properties of the model RFs to those of V1 simple cells and found extensive similarities between their features.

3.5.1 Relationship to other models

In the following discussion, we describe previous normative models that infer RFs with temporal structure from movie input and relate them to simple cell spatiotemporal RFs in V1. Other normative models of less directly relevant areas, such as spatial receptive fields without a temporal component (Bell and Sejnowski, 1997; Olshausen and Field, 1997, 1996; Rao and Ballard, 1999), complex cells (Berkes and Wiskott, 2005), and retinal receptive fields (Srinivasan et al., 1982) are not focused on in this discussion.

The earliest normative model of spatiotemporal RFs of simple cells used independent component analysis (ICA) (van Hateren and Ruderman, 1998), which is practically equivalent to the critically complete case of the sparse coding model (Olshausen and Field, 1997) we used as a control (Figs. 3.5, 3.6, 3.8 Supp. Figs. 3.1, 3.6-3.8). The RFs produced by this model and our control model reproduced fairly well the spatial aspects of simple cell RFs. However, in contrast to the temporal prediction model, the subset of more 'blob-like' RFs seen in the data are not well captured by our control sparse coding model (Fig. 3.7d and Fig. 3.8d). In the temporal domain, again unlike the temporal prediction model and real V1 simple cells, the RFs of the ICA and sparse coding model are not pressed up against the present with an asymmetrical temporal envelope, but instead show a symmetrical envelope or span the entire range of times examined. A related model (Olshausen, 2003) assumes that a longer sequence of frames is generated by convolving each basis function with a time-varying sparse coefficient and summing the result, so that each basis function is applied at

each point in time. The resulting spatiotemporal RFs are similar to those produced by ICA (van Hateren and Ruderman, 1998), or our control model (Fig. 3.6, Supp. Fig. 3.1c). Although they tend not to span the entire range of times examined, they do show a symmetrical envelope, and require an iterative inference procedure, as described in the previous chapters.

Temporal slowness constraints have also been used to model the spatiotemporal RFs of simple cells. The bubbles (Hyvärinen et al., 2003) approach combines sparse and temporal coherence constraints with reconstruction. The resulting RFs show similar spatial and temporal properties to those found using ICA. A related framework is slow feature analysis (SFA) (Berkes and Wiskott, 2005; Wiskott and Sejnowski, 2002), which enforces temporal smoothness by minimizing the derivative of unit responses over time, while maximizing decorrelation between units. SFA has been used to model complex cell spatiotemporal RFs (over only two time steps (Berkes and Wiskott, 2005)), and a modified version has been used to model spatial (not spatiotemporal) RFs of simple cells (Berkes et al., 2009). These results are not directly comparable with our results or the spatiotemporal RFs of simple cells.

In the slowness framework, the features found are those that persist over time; the presence of such a feature in the recent past predicts that the same feature will be present in the near future. This is also the case for our predictive approach, which, additionally, can capture features in the past that predict features in the future that are subtly or radically different from themselves. The temporal prediction principle will also give different weighting to features, as it values predictive capacity rather than temporal slowness (Creutzig and Sprekeler, 2008). In addition, although slowness models can be extended to model RFs over more than one time step (Berkes and Wiskott, 2005; Carlin and Elhilali, 2013; Hyvärinen et al., 2003), capturing temporal structure, they do not inherently give more

weighting to information in the most recent past and therefore do not give rise to asymmetric temporal profiles in RFs.

There is one study that has directly examined temporal prediction as an objective for visual RFs in a manner similar to ours (Palm, 2012). Here, as in our model, a single hidden layer feedforward neural network was used to predict the immediate future frame of a movie patch from its past frames. However, only two frames of the past were used in this study, so a detailed exploration of the temporal profile of the spatiotemporal RFs was not possible. Nevertheless, some similarities and differences in the spatial RFs between the two frames were noted, and some units had oriented RFs. In contrast to our model, however, many RFs were noisy and did not resemble those of simple cells. Potential reasons for this difference include the use of L_2 rather than L_1 regularization on the weights, an output nonlinearity not present in our model, the optimization algorithm used, network size, or the dataset.

In another recent related study, Chalk and colleagues (2018) developed a more tractable, iterative algorithm that approximates the information bottleneck for non-Gaussian variables (section 1.6.2; Chalk et al., 2016, 2018). They built a model with a linear encoder and linear decoder and trained it to represent natural video inputs under the constraints of the information bottleneck. They show that when a sparse (but not a Gaussian) prior is placed on the distribution of model unit responses, their spatial receptive fields resemble V1 simple cells. Furthermore, by changing the time lag of the decoding filters, they could alternate between a code that was predictive of future input and one that was optimized to decode past inputs. They found that only in the case of the predictive model did the receptive fields show the temporal asymmetry of real RFs, decaying into the past. However, unlike in our model, the predictive information bottleneck produced only space-time separable (non-direction-selective) RFs. One possible reason for this difference is that Chalk et. al. (2018) use a linear

encoding and decoding model, while we use a nonlinear encoder followed by a linear decoder.

3.5.2 Critical evaluation of the temporal prediction model

Although the temporal prediction model's ability to describe neuronal RFs is high, the match with real neurons is not perfect. For example, we found an overrepresentation of vertical and horizontal orientations compared to real V1 data (Fig 3.7b-c). Furthermore, the distribution of RF shapes seen in the model did not perfectly match the distribution seen for V1 simple cells (Fig 3.7d). Some of these differences could be a consequence of the data used to train the model. Although the statistics of natural stimuli are broadly conserved (Field, 1987), there is still variation (Torralba and Oliva, 2003), and the dataset used to train the network may not match the sensory world of the animal experienced during development and over the course of evolution. In future work, it would be valuable to explore the influence of natural datasets with different statistics, and also to match those datasets more precisely to the evolutionary context and individual experience of the animals examined. Furthermore, a comparison of the model with neural data from different species, at different ages, and reared in different environments would be useful.

Another cause of differences between the model and neural RFs may be the recording location of the RFs and how they are characterized. We used the primary sensory cortices as regions for comparison, because we performed transformations on the input data that are similar to the preprocessing that takes place in afferent subcortical structures. We spatially filtered the visual data in a similar way to the retina (Olshausen and Field, 1997, 1996), and spectrally decomposed the auditory data as in the inner ear, and then used time bins (5ms) which are coarser than, but close to, the minimum amplitude modulation period that can be

tracked by auditory midbrain neurons (Rees and Møller, 1983). However, primary cortex is not a homogenous structure, with neurons in different layers displaying certain differences in their response properties (Harris and Mrsic-Flogel, 2013). Furthermore, the methods by which neurons are sampled from the cortex may not provide a representative sample. For example, multi-electrode arrays tend to favour larger and more active neurons. In addition, the method and stimuli used to construct RFs from the data can bias their structure somewhat (Willmore et al., 2016).

Due to the difficulty we encountered in procuring spatiotemporal RF data from more than a few V1 neurons, we were unable to perform a quantitative comparison of the visual model's RF properties with a large dataset of V1 spatiotemporal RFs. Were we to acquire such a dataset, it would be interesting to analyse the visual models in the same way that we analysed the auditory models in the previous chapter. This would allow us to more quantitatively compare how well features of the V1 RFs are captured by the temporal prediction model compared to the sparse coding control. Furthermore, it would allow us to directly compare the performance of the temporal prediction model in the visual domain to its performance in the auditory domain. Without this quantitative analysis, directly comparing the temporal prediction model's performance across these domains would be difficult.

The model presented here is based on a simple feedforward network with one layer of hidden units. This limits its ability to predict features of the future input, and to account for RFs with nonlinear tuning. More complex networks, with additional layers or recurrency may allow the model to account for more complex tuning properties, including those found beyond the primary sensory cortices. Careful, principled adjustment of the preprocessing, or different regularization methods (such as sparseness or slowness applied to the units' activities), may also help. There is an open question as to whether the current model may

eliminate some information that is useful for reconstruction of the past input or for prediction of higher order statistical properties of the future input, which might bring it into conflict with the principle of least commitment (Marr, 1976). It is an empirical question how much organisms preserve information that is not predictive of the future, although there are theoretical arguments against such preservation (Bialek et al., 2001). Such conflict might be remedied, and the model improved, by adding feedback from higher areas or by adding an objective to reconstruct the past or present (Olshausen and Field, 1997, 1996) in addition to predicting the future.

To determine whether the model could help explain neuronal responses in higher areas, it would be useful to develop a hierarchical version of the temporal prediction model, applying the same model again to the activity of the hidden units rather than to the input. This is done in the next chapter. Another useful extension would be to see if the features learnt by the temporal prediction model could be used to accelerate learning of useful tasks such as speech or object recognition, by providing input or initialization for a supervised or reinforcement learning network. Indeed, temporal predictive principles have been shown to be useful for unsupervised training of networks used in visual object recognition (Lotter et al., 2016; Oh et al., 2015; Ranzato et al., 2016; Srivastava et al., 2015).

Finally, it is interesting to consider more explicit possible biological bases for our model. We envisage the input units of the model as thalamic input, and the hidden units as primary cortical neurons. Although the output units could be seen as just a method to optimize the hidden units to find the most predictive code given sensory input statistics, they may also have a physiological analogue. Current evidence suggests that while primary cortical RFs are to an extent hard-wired in form by natural selection, their tuning is also refined by individual sensory experience (Dahmen and King, 2007; Huberman et al., 2008; Kiorpes, 2015). This refinement process may require a predictive learning mechanism in the

animal's brain, at least at some stage of development and perhaps also into adulthood. Hence, one might expect to find a subpopulation of neurons that represent the prediction (analogous to the output units of the model) or the prediction error (analogous to the difference between the output unit activity and the target). Indeed, signals relating to sensory prediction error have been found in A1 (Rubin et al., 2016), though they may also be located in other regions of the brain.

3.6 Methods

3.6.1 Data used for model training and testing

Visual inputs

Videos (without sound, sampled at 25 fps) of wildlife in natural settings were used to create visual stimuli for training the artificial neural network. The videos were obtained from http://www.arkive.org/species, contributed by: BBC Natural History Unit, http://www.gettyimages.co.uk/footage/bbcmotiongallery; BBC Natural History Unit & Discovery Communications Inc., http://www.bbcmotiongallery.com; Granada Wild, http://www.itnsource.com; Mark Deeble & Victoria Stone Flat Dog Productions Ltd., http://www.deeblestone.com; Getty Images, http://www.gettyimages.com; National Geographic Digital Motion, http://www.ngdigitalmotion.com. The longest dimension of each video frame was clipped to form a square image. Each frame was then band-pass filtered (Olshausen and Field, 1997) and downsampled (using bilinear interpolation) over space, to provide 180x180 pixel frames. Non-overlapping patches of 20×20 pixels were selected from a fixed region in the center of the frames, where there tended to be visual motion. The video patches were cut into sequential overlapping clips each of 8 frames duration. Thus, each training example (clip) was made up of a 20×20 pixel section of the

video with a duration of 8 frames (320ms), providing a training set of N = ~500,000 clips from around 5.5 h of video, and a validation set of N = ~100,000 clips. Finally, the training and validation sets were normalized by subtracting the mean and dividing by the standard deviation (over all pixels, frames and clips in the training set). The goal of the neural network was to predict the final frame (the 'future') of each clip from the preceding 7 frames (the 'past').

Addition of Gaussian noise

To replicate the effect of noise found in the nervous system, Gaussian noise was added to the visual inputs with a signal-to-noise ratio (SNR) of 6dB. The addition of noise did not make substantial differences to the RFs of units in the temporal prediction model when trained using visual inputs (Supp. Figs. 3.3 – 3.5). The results from the sparse coding model were also similar for inputs with and without noise (Supp. Figs. 3.6 – 3.8).

3.6.2 Temporal prediction model

The model and cost function

The identical temporal prediction model to the one presented in the previous chapter (section 2.8.2) was used. The effect of varying the number of hidden units and λ on the prediction error for the validation set is shown in the interactive version of Supp. Fig. 3.2. The results presented are the networks that predicted best on the validation set after 1000 iterations through the training data. In the visual case, the optimal settings were 1600 hidden units and $\lambda = 10^{-6.25}$.

3.6.3 Sparse coding model

As a control, we used the same sparse coding model as in the previous chapter (section 2.8.3). As there was no independent criterion by which to determine the 'best' settings, we

chose the network that produced basis functions whose receptive fields were most similar to those of real neurons. As a similarity measure was not performed, this was done by inspection. We limited our choice of network configurations to be overcomplete, in keeping with previous studies (Olshausen and Field, 1997). The network selected was trained with 3200 units, $\lambda=10^{0.5}$, learning rate = 0.05 and 100 mini-batches (Figs. 3.4, 3.5). Although these sparse coding basis functions are projective fields, they tend to be similar in structure to receptive fields (Olshausen and Field, 1997, 1996), and, for simplicity, are referred to as RFs.

3.6.4 Visual receptive field analysis

In vivo V1 RF data

Visual RFs measured using recordings from V1 simple cells were compared against the model (Fig. 3.2a, and Fig. 3.7a, 3.8a, cat (Ohzawa et al., 1996) and Fig. 3.7d, 3.8d, cat (Jones and Palmer, 1987b), mouse (Niell and Stryker, 2008) and monkey (Ringach, 2002)). The in vivo data were taken from the authors' website (Ringach, 2002) or extracted from relevant papers (Jones and Palmer, 1987b) or provided by the authors (Niell and Stryker, 2008; Ohzawa et al., 1996).

Fitting Gabors

In order to quantify tuning properties of the model's visual RFs, two-dimensional (2D) Gabors were fitted to the optimal time step of each unit's response (Jones and Palmer, 1987b; Ringach, 2002). This allowed comparison to previous experimental studies which parameterised real RFs by the same method (Ringach, 2002). The optimal time step was defined (Ringach, 2002) as the time step of the unit's response which contained the most

power (sum of square values). The Gabor function has been shown to provide a good approximation for most spatial aspects of visual simple cell RFs (Jones and Palmer, 1987b; Ringach, 2002). The 2D Gabor is given as:

$$G(x', y') = A\exp\left(-\left(\frac{x'}{\sqrt{2}\sigma_x}\right)^2 - \left(\frac{y'}{\sqrt{2}\sigma_y}\right)^2\right)\cos(2\pi f x' + \phi) \tag{3.1}$$

where, the spatial coordinates (x', y') are acquired by translating the center of the RF (x_0, y_0) to the origin and rotating the RF by its spatial orientation θ:

$$x' = (x - x_0)\cos\theta + (y - y_0)\sin\theta \tag{3.2}$$

$$y' = -(x - x_0)\sin\theta + (y - y_0)\cos\theta \tag{3.3}$$

σ_x and σ_y provide the width of the Gaussian envelope in the x' and y' directions, while f and ϕ parameterize the spatial frequency and phase of the sinusoid along the x' axis. A parameterizes the height of the Gaussian envelope.

For each RF, the parameters (A, x_0, y_0, σ_x, σ_y, θ, f, ϕ) of the Gabor were fitted by minimizing the mean squared error between the Gabor model and the RF using the minFunc minimization package (http://www.cs.ubc.ca/~schmidtm/Software/minFunc.html). In order to avoid local minima, the fitting was performed in two steps. First, the spatial RF was converted to the spectral domain using a 2D Fourier transform. Since the amplitude spectrum of the Fourier transform of a 2D Gabor is a 2D Gaussian (Jones and Palmer, 1987b), which is easier to fit, an estimate of many of the parameters was obtained by first fitting a 2D Gaussian in the spectral domain (using the amplitude spectrum). Using the parameters

obtained from the spectral fitting as initial estimates, a 2D Gabor was then fitted to the original RF in the spatial domain.

The fitted parameters provided a good estimate of the units' responses, with residual errors between the spatial responses and the corresponding Gabor fits being small and lacking spatial structure. The median pixel-wise correlation coefficient of the Gabor fits for the temporal prediction model units was 0.88. Units whose fitted Gabors had a poor fit (those with a correlation coefficient <0.7; 214 units) were excluded from further analysis. We also excluded units with a high correlation coefficient (>0.7) if the center position of the Gabor was estimated to be outside the RF, and hence only the Gabor's tail was being fitted to the response (39 units), and those for which the estimated standard deviation of the Gaussian envelope in either x or y was <0.5 pixels, which meant very few non-negligible pixel values were used to constrain the parameters (146 units). Together, these exclusion criteria (which sometimes overlapped), led to 395 of the 1600 units being excluded for the temporal prediction model.

2D spatiotemporal receptive fields

In order to better view their temporal characteristics we collapsed the three dimensional (3D) spatiotemporal real and model RFs (space-space-time) along a single spatial direction to create 2D spatiotemporal (space-time) representations (DeAngelis et al., 1993). First, we determined the 3D RFs' optimal time step (the time step with the largest sum of squared values). We then acquired the rotation and translation that centers the RF on zero and places the oriented bars parallel to the y-axis at the optimal time step, using the Gabor parameterization of each unit at its optimal time step. We applied this fixed transformation to each time step and collapsed the RF by summing the activity along the newly defined y-axis. The resulting 2D (space-time) RFs provide intuitive visualization of the RF across time,

while losing minimal information. For the RFs of real neurons (Ohzawa et al., 1996), the most recent time step (40ms) of the 3D and 2D spatiotemporal RFs were removed to account for the latency of V1 neurons (Fig 3.2a; 3.7a, 3.8a).

Estimating space-time separability

The population of model units contained both space-time (ST) separable and inseparable units. First the two spatial dimensions of the 20x20x7 3D RF were collapsed to a single vector to yield a single 400x7 matrix. The singular value decomposition (SVD) of this matrix was then taken and the singular values examined. If the ratio between the second and first singular value was ≥0.5, the unit was deemed to be inseparable. Otherwise, the unit was deemed to be separable. Examining the 20x7 2D spatiotemporal RFs (obtained as outlined in the preceding section; Fig. 3.4) showed this to be an accurate way of separating space-time separable and inseparable units.

Spatial RF structure

For comparison with the real V1 RF and previous theoretical studies, the width and length of our model's RFs were measured relative to the period of their spatial frequency (Ringach, 2002). Here, $n_x = \sigma_x f$ gives a measure of the length of the bars in the RF, while $n_y = \sigma_y f$ gives a measure of the number of oscillations of its sinusoidal component. Thus, in the n_y, n_x plane, blob-like RFs with few cycles lie close to the origin, while stretched RFs with many subfields lie away from the origin. RFs with values high along the n_y axis, have many bars, while those far along the n_x axis have long bars. As in Ringach (Ringach, 2002) only space-time separable units were included in this analysis.

Temporal weighting profile of the population

The mean power for each of the 7 time steps of the RFs was examined for both real and model populations (Fig. 3.7a, 3.8a). The temporal weighting profile was calculated as the mean, over space and the population, of the squared values of the 2D spatiotemporal RFs at each time step.

Tilt direction index

The tilt direction index (TDI) (Anzai et al., 2001; Baker, 2001; DeAngelis et al., 1993; Livingstone and Conway, 2007; Pack et al., 2006) of an RF is given by $(R_p - R_q)/(R_p + \mathrm{R_q})$, where R_p is the amplitude at the peak of the 2D Fourier transform of the 2D spatiotemporal RF, found at spatial frequency F_{space} and temporal frequency F_{time}. R_q is the amplitude at $(F_{\mathrm{space}}, -F_{\mathrm{time}})$ in the 2D Fourier transform. The mean and standard deviations of TDI for experimental data for the cat (Baker, 2001) and macaque (Livingstone and Conway, 2007) were measured from data extracted from figures in the relevant references (Figure 11A and the low-contrast axis of Figure 3A respectively).

Peak temporal frequency

The 2D spatiotemporal RFs were also useful for calculating further temporal response properties of the model. The temporal frequency was calculated as the peak temporal frequency of each spatiotemporal RF as measured from its 2D Fourier transform.

Chapter 4 | Hierarchical temporal prediction captures motion processing from retina to higher visual cortex

4.1 Introduction

In the previous chapters we introduced the temporal prediction framework, which posits that sensory systems are optimized to represent features in natural stimuli that enable prediction of future sensory input. This would be useful for guiding future action, uncovering underlying variables, and discarding irrelevant information (Bialek et al., 2001).Temporal prediction relates to a class of principles, such as the predictive information bottleneck (Bialek et al., 2001; Chalk et al., 2018; Salisbury and Palmer, 2016) and slow feature analysis (Berkes and Wiskott, 2005), that similarly involve selectively encoding only features that are of behavioral relevance. This class of principles differs from others that are more typically used to explain sensory coding – efficient coding (Barlow, 1961; Srinivasan et al., 1982), sparse coding (Olshausen and Field, 1997, 1996) and predictive coding (Huang and Rao, 2011; Rao and Ballard, 1999) – that aim instead to efficiently represent all current and perhaps past input. Although these principles have been successful in accounting for various visual receptive field (RF) properties in V1 (Bell and Sejnowski, 1997; Berkes and Wiskott, 2005; Chalk et al., 2018; Olshausen and Field, 1996; Rao and Ballard, 1999; Singer et al., 2018), no single principle has so far been able to explain the diverse spatiotemporal tuning that emerges along the dorsal visual stream, which is responsible for the processing of object motion.

Any general principle of visual encoding needs to explain temporal aspects of neural tuning – the encoding of visual scenes in motion rather than static images. It is also important that any general principle is largely unsupervised. Some features of the visual system have been reproduced by deep supervised network models optimized for image classification using large labelled datasets (e.g. images labelled as cat, dog, car; Yamins and DiCarlo, 2016). While these models can help to explain the RF properties of the likely hard-wired retina (Lindsey et al., 2019), they are less informative if neuronal tuning is influenced by experience, as in cortex, since most sensory input is unlabeled except for sporadic reinforcement signals. The temporal prediction approach is unsupervised (i.e., it requires no labelled data), and inherently applies to the temporal domain.

The non-hierarchical temporal prediction model we presented in the previous chapter finds features in its inputs that are predictive of future input. However, the pattern of occurrence of these particular predictive features over time may themselves have a predictable structure. Hence a second temporal prediction model could be stacked atop the first to find this structure. This could be done repeatedly so as to find the predictable structure of increasingly high-level statistics of the input.

In this chapter we develop such a hierarchical form of the temporal prediction model that predicts increasingly high-level statistics of natural dynamic visual input. It accounts not only for linear tuning properties of V1 simple cells, as in our previous non-hierarchical temporal prediction model (Chapter 3), but also for the diversity of linear and non-linear spatiotemporal tuning properties that emerge along the visual pathway. In particular, the temporal tuning properties in successive hierarchical stages of the model progress from resembling the magnocellular and parvocellular neurons of the retina and visual thalamus, to the direction-selective simple and complex cells of V1, and finally to units that are sensitive to two-dimensional features of motion, as seen in end-stopping and pattern-

selective cells in the cortex. The capacity of this model to explain tuning properties at multiple levels of the visual motion pathway using iterated application of a single process, suggests that optimization for temporal prediction may be a fundamental principle of sensory neural processing.

4.2 Results

4.2.1 The hierarchical temporal prediction model

We instantiated temporal prediction as a hierarchical model consisting of stacked feedforward single-hidden-layer convolutional neural networks (Fig. 4.1). Image statistics tend to be similar across space, so convolution allows the networks to be efficiently trained and implemented while providing the deeper stacks the capacity to receive input from a wide span of the image. The first stack was trained to predict the immediate future frame (40 ms) of unfiltered natural video inputs from the previous 5 frames (200 ms). Each subsequent stack was then trained to predict the future hidden-unit activity of the stack below it from the past activity in response to the natural video inputs. L_1 regularization was applied to the weights of each stack, akin to a constraint on neural wiring. All stacks were trained using backpropagation (see section 1.7.1).

The video inputs consisted of animals moving in natural environments, filmed with panning, still and occasionally zooming cameras. The four stacks respectively used 50, 100, 200 and 400 kernels (hidden units per spatial position), however as we only use valid convolutions the number of spatial positions decreases with each stack (see Table 1, Methods). This results in 14,450, 22,500, 33,800 and 48,400 hidden units over all spatial positions in each stack respectively.

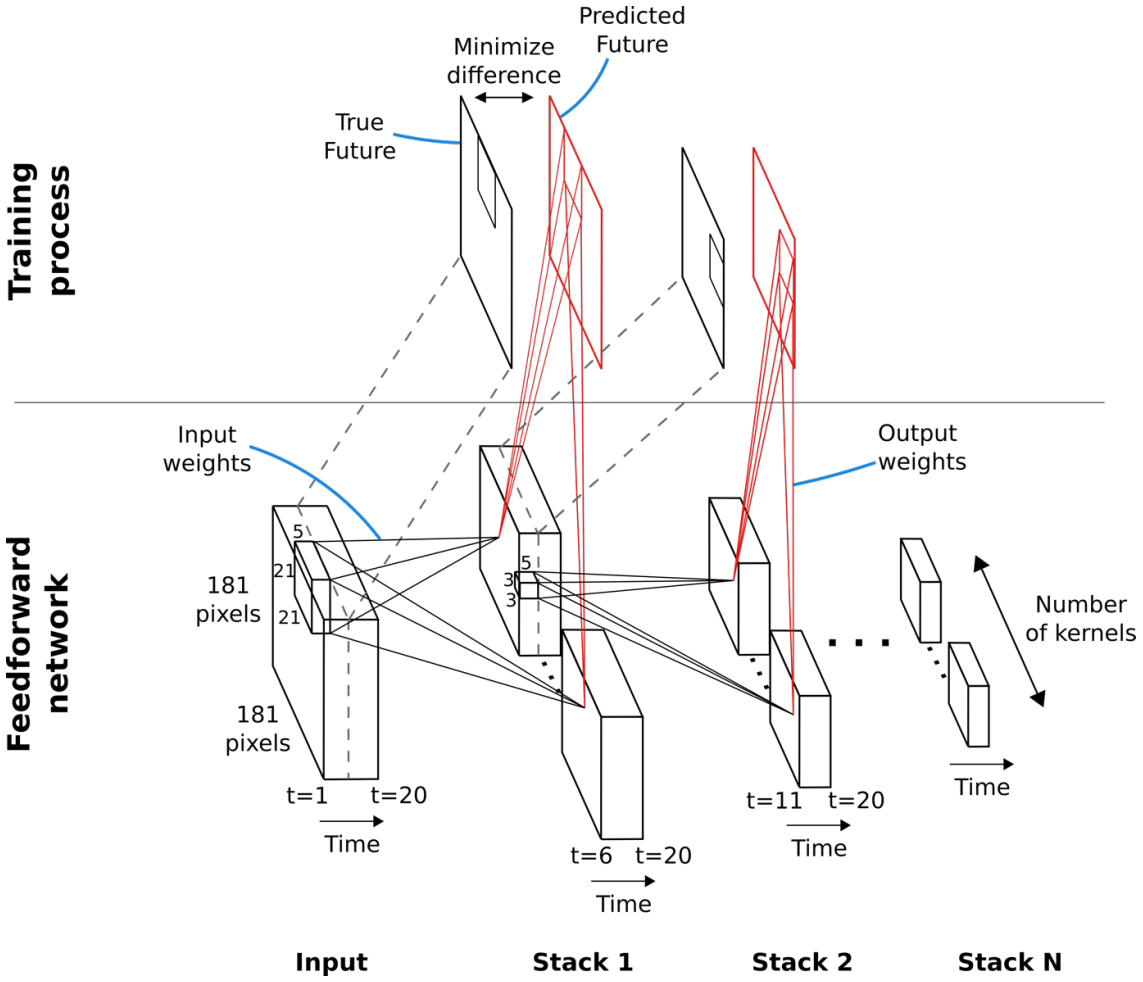


Figure 4.1 | Hierarchical temporal prediction model. Schematic of model architecture. Each stack is a single hidden-layer feedforward convolutional network, which is trained to predict the future time step of its input from the previous 5 time steps. The first stack is trained to predict future pixels of natural video inputs from their past. Subsequent stacks are trained to predict future time steps of the hidden-layer activity in the stack below, based on their past responses to the same natural video inputs. That is, for each stack, its input is transformed through input weight kernels (small boxes within large boxes) by a convolution, then rectification is applied to provide the hidden unit activity (large boxes), this activity is then transformed through an output weight kernel (red square within large red square) by another convolution, to provide a prediction of the immediate future input (large red square). This is then compared to the true future (large black square), and the difference minimized. Hence each stack is trained separately, but input and hidden stages of each stack are stacked together to form a deep feedforward network.

4.2.2 Temporal prediction of natural inputs produces retinal-like units

After training, we examined the input weights of the units in the first stack. Each hidden unit can be viewed as a linear non-linear (LN) model (Chichilnisky, 2001; Simoncelli et al., 2004), as commonly used to describe neuronal RFs. With L_1 weight regularization slightly above the optimum for prediction, the RFs of the units showed spatially-localized center-surround tuning with a decaying temporal envelope, characteristic of retinal and lateral geniculate nucleus (LGN) neurons (Barlow, 1953; Kuffler, 1953; Shapley and Perry, 1986). The model units' RFs have either an excitatory (ON) or inhibitory (OFF) blob-like structure at the 0ms time step, often with a surround of opposing sign in the same or previous time step (Fig. 4.2a). Both ON and OFF units can have either small RFs that do not change polarity (Fig. 4.2a, units 1-4; Fig. 4.2b, bottom left) over time, or large RFs that switch polarity over time (Fig. 4.2a, units 5-6; Fig. 4.2b, top right). This is reminiscent of the four main cell types in the primate retina and LGN: the parvocellular-pathway ON/OFF neurons and the more change-sensitive magnocellular-pathway ON/OFF neurons, respectively (Shapley and Perry, 1986).

Interestingly, simply decreasing L_1-regularization strength causes the model RFs to change from center-surround tuning to Gabor-like tuning, resembling localized oriented bars that shift over time (Fig. 4.2c). It is possible that this balance, between a code that is optimal for prediction and one that prioritizes efficient wiring, might underlie differences in the retina and LGN of different species. The retinas of mice and rabbits contain many neurons with oriented and direction-tuned RFs, whereas cats and macaques mostly have center-surround RFs (Scholl et al., 2013). Efficient retinal wiring may be more important in some species, due, for example, to different constraints on the width of the optic nerve or different impacts of light scattering by superficial retinal cell layers.

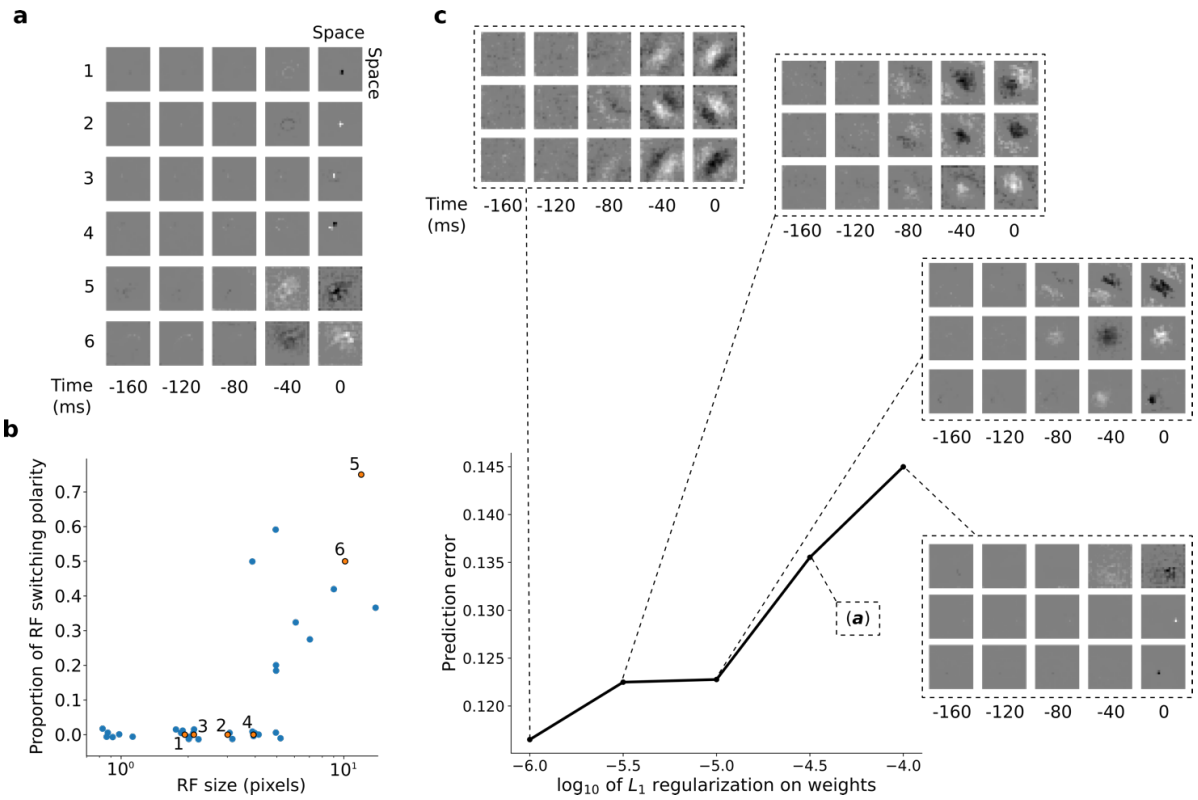


Figure 4.2 | RFs of trained first stack of the model show retina-like tuning. **a**, Example RFs with center-surround tuning characteristic of neurons in retina and LGN. RFs are small and do not switch polarity over time (units 1-4) or large and switch polarity (units 5-6), resembling cells along the parvocellular and magnocellular pathways, respectively. **b**, RF size plotted against proportion of the pixels in the RF that switch polarity over the course of the most recent two time steps. Units in **a** labelled and shown in orange. **c**, Effect of changing regularization strength on the qualitative properties of RFs.

4.2.3 Hierarchical temporal prediction produces simple and complex cell tuning

Using the trained center-surround-tuned network as the first stack, a second stack was added to the model and trained. The output of each second stack unit results from a linear-nonlinear-linear-nonlinear transformation of the visual input, and hence we estimated their RFs by reverse correlation with binary noise input. The resulting RFs were Gabor-like over space, resembling those of V1 simple cells (Hubel and Wiesel, 1968, 1965, 1959). The RF

envelopes decayed into the past, and often showed spatial shifts or polarity changes over time, indicating direction or flicker sensitivity, as is also seen in V1 (DeAngelis et al., 1993) (Fig. 4.3a,b, I; Supp. Fig. 4.1). Using full-field drifting sinusoidal gratings (Fig. 4.3a,b II), we found that most units were selective for stimulus orientation, spatial and temporal frequency (Fig. 4.3a,b, IV-VI), and some were also direction selective (Fig. 4.3b). Responses to the optimum grating typically oscillate over time between a maximum when the grating is in phase with the RF and 0 when the grating is out of phase (Fig. 4.3a,b, III). These response characteristics are typical of V1 simple cells (Movshon et al., 1978a).

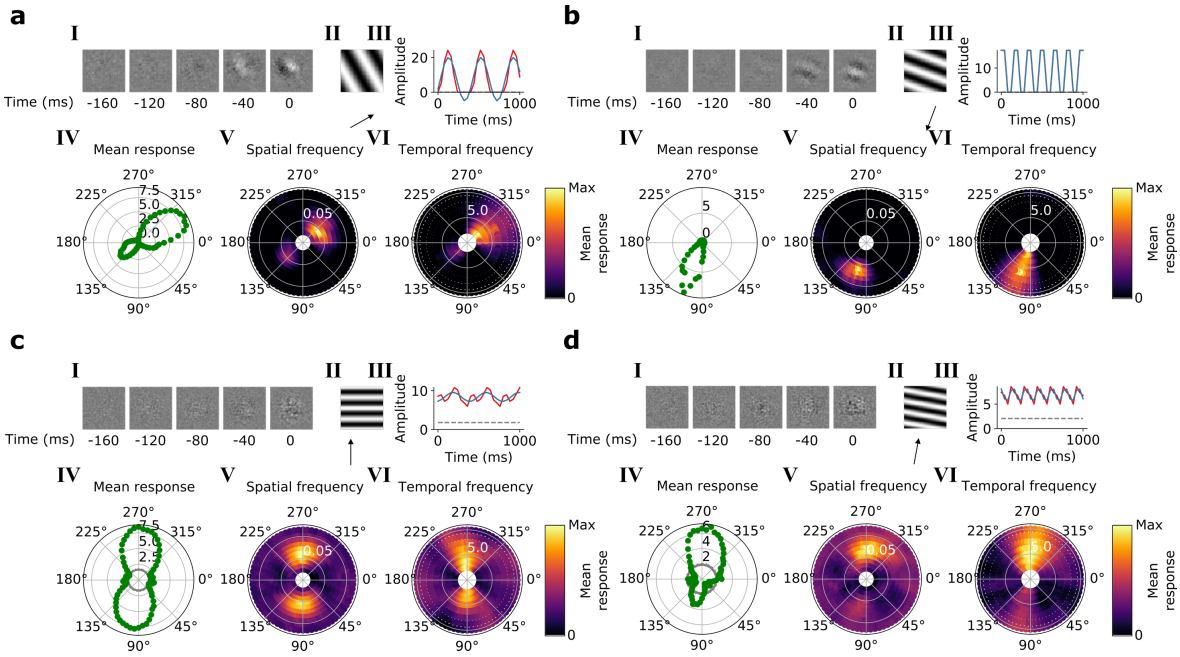


Figure 4.3 | Qualitative tuning properties of model units in stacks 2 and 3. **a**,**b**, Tuning properties of two example units from the 2nd stack of the model, including (I) the linear RF, (II) the drifting grating that best stimulates this unit and (III) the amplitude of the unit's response to this grating over time. Red line: unit response; blue line: best-fitting sinusoid; gray dashed line: response to blank stimulus (sometimes overlapping with x-axis). (IV) The unit's mean response over time plotted against orientation (in degrees) for gratings presented at its optimal spatial and temporal frequency. (V, VI) Tuning curves showing the joint distribution of responses to (V) orientation (in degrees) and spatial frequency (in cycles/pixel) at the preferred temporal frequency and to (VI) orientation and temporal frequency (in Hz) at the preferred spatial frequency. In V and VI the color represents the

mean response over time to the grating presented. **c**,**d**, As in **a**,**b** for example units in the 3rd stack. See also Supp. Figs. 4.1-4.3.

In the third and fourth stack, we followed the same procedures as in the second stack. Most of these units are also tuned for orientation, spatial frequency, temporal frequency and in some cases for direction (Fig. 4.3c,d, IV-VI; Supp. Figs. 4.2,4.3). However, while some units resembled simple cells, most resembled the complex cells of V1 and secondary visual areas (V2/V3) (Hubel and Wiesel, 1965). Complex cells are tuned for orientation and spatial and temporal frequency, but are relatively invariant to the phase of the optimum grating (Movshon et al., 1978b); each cell's response to its optimum grating has a high mean value and changes little with the grating's phase (Fig. 4.3c,d, II,III). Whether a neuron is assigned as simple or complex is typically based on the modulation ratio in such plots (<1 indicates complex; see methods; Skottun et al., 1991). Model units with low modulation ratios had little discernible structure in their RFs (Fig. 4.3c,d, I), another characteristic feature of V1 complex cells (Rust et al., 2005; Schwartz et al., 2006; Simoncelli et al., 2004).

We do not claim that any stack in the model corresponds directly to any specific stage along the visual pathway, rather that the order that tuning properties emerge as one moves up the stacks resembles the order that they emerge as one moves up the visual pathway. For example, it can be argued that at least some of the units in stack 4 may correspond to units seen in higher visual areas. With this caveat in mind, we quantified the tuning characteristics of units in stacks 2-4 and compared them to published V1 data (Ringach et al., 2002) (Fig. 4.4a-j). The distribution of modulation ratios is bimodal in both V1 (Ringach et al., 2002; Skottun et al., 1991) and our model (Fig. 4.4a). Both model and real neurons were typically orientation-selective, but with the model units having weaker tuning as measured by orientation bandwidth (median data (Ringach et al., 2002): 23.5°, model: 37.5°; Fig 4.4b) and circular variance (median data (Ringach et al., 2002) : simple cells 0.45, complex cells

0.66 ; median model : simple cells 0.45, complex cells 0.85 ; Fig 4.4c,d). Orientation-tuned units (circular variance < 0.9) in the second stack were exclusively simple (modulation ratios > 1), whereas those in subsequent stacks became increasingly complex (Fig. 4.4a,c-f). In both model and data, circular variance was inversely correlated with the modulation ratio (Fig. 4.4c-e,h). Similarly, orientation bandwidth varied with modulation ratio in a similar way in both model and data (Fig. 4.4f,i). Model units showed a range of direction selectivity preferences (Fig. 4.4g), with simple cell-like units (69% with direction selectivity index, DSI ≥ 0.5; n=107/155; Fig. 4.4j) tending to be more direction tuned than complex cell-like units (20% with DSI ≥ 0.5, n=42/205; Fig. 4.4j) as is seen in V1 (Kim and Freeman, 2016).

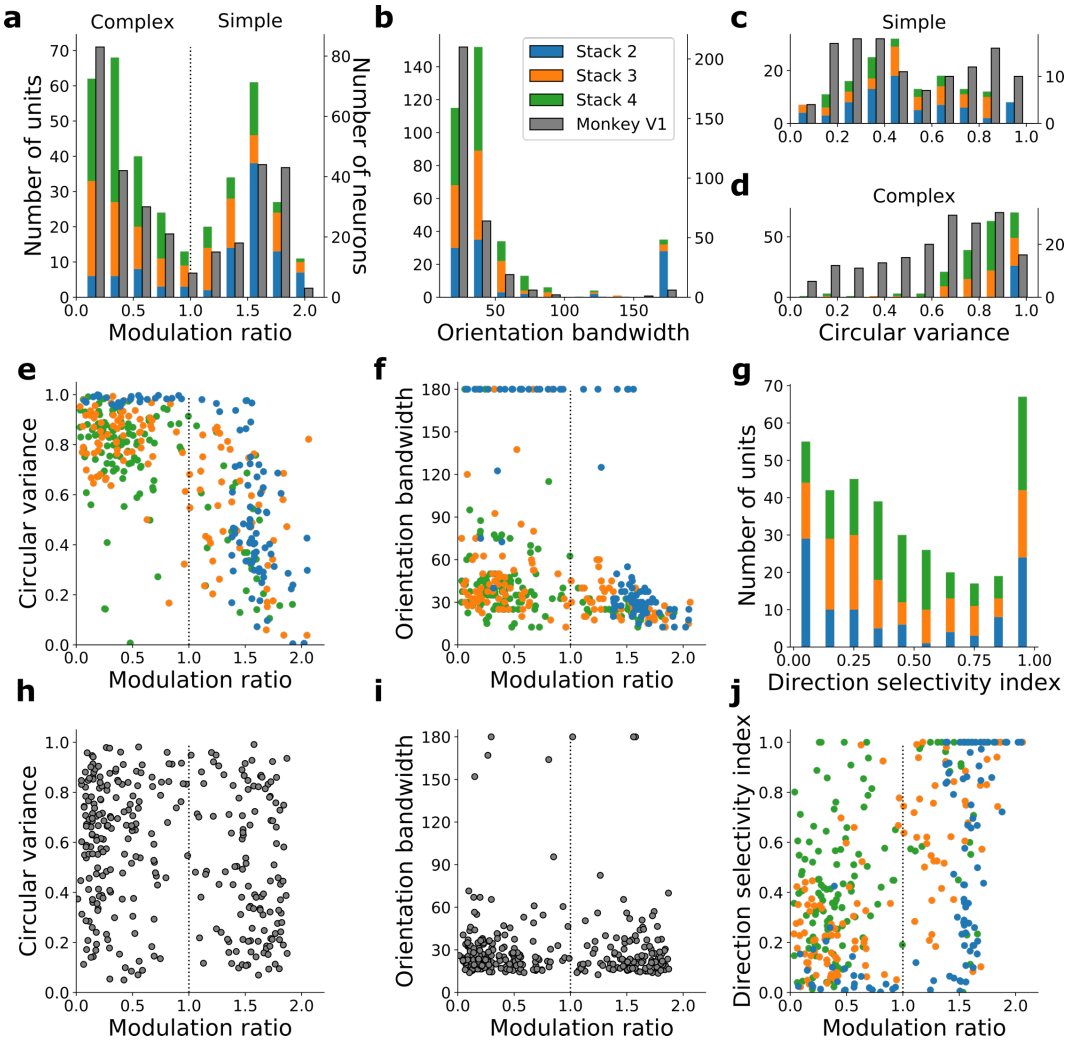


Figure 4.4 | (previous page) Quantitative tuning properties of model units in stacks 2-4 in response to drifting sinusoidal gratings and corresponding measures of macaque V1 neurons. **a-d,** Histograms showing tuning properties of model and macaque V1 neurons as measured using drifting gratings. Units from stacks 2-4 are plotted on top of each other to also show the distribution over all stacks; this is because no stack can be uniquely assigned to V1. Modulation ratio measures how simple or complex a cell is; orientation bandwidth and circular variance are measures of orientation selectivity. **e, f,** Joint distributions of tuning measures for model units. Each dot is a model unit. **g,** Distribution of direct selectivity for model units. **h, i,** Joint distributions of tuning measures for V1 data. **j,** Joint distribution of direction selectivity and modulation ratio for model units.

It is interesting to ask whether these results are a consequence of the convolutional hierarchical structure of the model, the temporal prediction objective, or if they require both together. At least some hierarchy or recurrency in model structure is required to obtain complex cells, the simple linear-nonlinear structure of hidden units in our previous non-hierarchical temporal prediction model is incapable of producing units resembling complex cells, although it can produce simple-cell-like units (as presented in Chapter 3). To investigate whether a model with the same convolutional hierarchical structure as our model, but with a different objective function than temporal prediction could produce complex cells, we instead trained the model at each stack to efficiently estimate the present input given present and past input, rather than to predict its future input. This present-estimating model is very similar to a stacked autoencoder (Bengio et al., 2006) - a powerful method to initialize deep learning models using unsupervised learning. Typically, a stacked autoencoder only receives input from a single image, which it attempts to reproduce at the output. However, in order to maintain similarity to the temporal prediction model, we trained the control model to reproduce the most recent time step of the video input from the same time step and the preceding 4 time steps. This is a simpler task than predicting the future frame, since the target being predicted is identical to the most recent time-step of the input. Thus, we would expect such a model to have very limited capacity to represent the temporal structure of the movies.

When we trained this model, with the same form and dataset as the temporal prediction model, we indeed found that while the first stack could reproduce the center-surround spatial tuning seen in the retina, their RFs had no temporal structure – they only depended on the present time step. Furthermore, this stacked autoencoder could not produce magnocellular-like units that switch polarity, nor at any stack could it produce units that resembled simple cells or complex cells, either spatially or temporally (Supplementary Fig. 4.4). This suggests that both hierarchy and temporal prediction acting together produce the diverse tuning properties we observe in our model.

As a further control model, we shuffled the input weights across space, time and units for each stack of the trained hierarchical temporal prediction model. In this case, the RFs in all stacks lacked discernable structure, being neither simple-cell-like nor center-surround, with only patchy spatial frequency and orientation tuning in response to gratings. Because the RFs of complex cells are typically unstructured, we could not rule out the existence of putative complex cells by the lack of structure in their RFs and so measured the responses to drifting gratings of the units in the shuffled network.

Cells responding like complex cells were substantially rarer in the shuffled network, with there being fewer than 25% as many orientation-tuned (circular variance < 0.9) units with modulation ratios <1 across all stacks. The fact that we do not see any center-surround or simple cells and that there are far fewer units meeting our definition of complex cells shows that the precise ordering of the weights learned through temporal prediction (as opposed to these weights randomly shuffled) is important for capturing the tuning phenomena observed in the brain.

4.2.4 Model units are tuned to two-dimensional features of visual motion

Simple and complex cells extract many dynamic features from natural scenes. However, their small RFs prevent individual neurons from tracking the motion of objects because of the aperture problem; when viewed through a small aperture, the direction of motion of an extended edge is ambiguous, with only the component of motion perpendicular to the cell's preferred orientation being represented. Two classes of neurons exist that can recover 2-dimensional motion information and overcome the aperture problem. End-stopped neurons, found in primary and secondary visual areas, respond unambiguously to the direction of motion of endpoints of restricted moving contours (Pack et al., 2003). Pattern-selective neurons, in MT of primates, solve the problem for the more general case, likely by integrating over input from many direction-selective V1 complex cells (Movshon et al., 1985; Movshon and Newsome, 1996; Rust et al., 2006; Simoncelli and Heeger, 1998; Zeki, 1974), and hence respond selectively to the motion of patterns as a whole.

To investigate end-stopping in our model units, we applied circular masks of different sizes to the grating stimuli. Some units displayed end-stopping, responding most strongly to gratings with an intermediate mask radius, with the response decreasing as the radius increased beyond this (Fig. 4.5a,b). To determine whether these end-stopped units unambiguously represent the direction of motion of end-points, we measured two-bar response maps (Pack et al., 2003), which determine response dependence on the horizontal and vertical components of motion (see Methods).

Examining the two-bar maps, the position (0,0) indicates that the bar was in the same position in two successive frames, while the vertical and horizontal axes represent movement in these directions. Positive activity means that the unit was excited by movement in that direction, while negative activity indicates inhibition of the unit to movement in the given direction. A non-end-stopped unit will respond to any movement with a component in the

preferred direction of the cell. This results in an elongated response profile on the two-bar map (Fig. 4.5c, right). An end-stopped unit will only respond to movement in the cell's preferred direction, resulting in a two-bar map whose excitatory activity is limited to a more circumscribed region (Pack et al., 2003; Fig. 4.5c, left).

Recordings from V1 indicate that more strongly end-stopped neurons have a weak tendency for less ambiguous motion tuning in these maps (i.e. have less elongated excitatory regions; Pack et al., 2003). Consistent with this, our model produces examples of end-stopped units with less ambiguous motion tuning (Fig. 4.5b,c, first two panels) and non-end-stopped units with more ambiguous motion tuning (Fig. 4.5b,c, last panel).

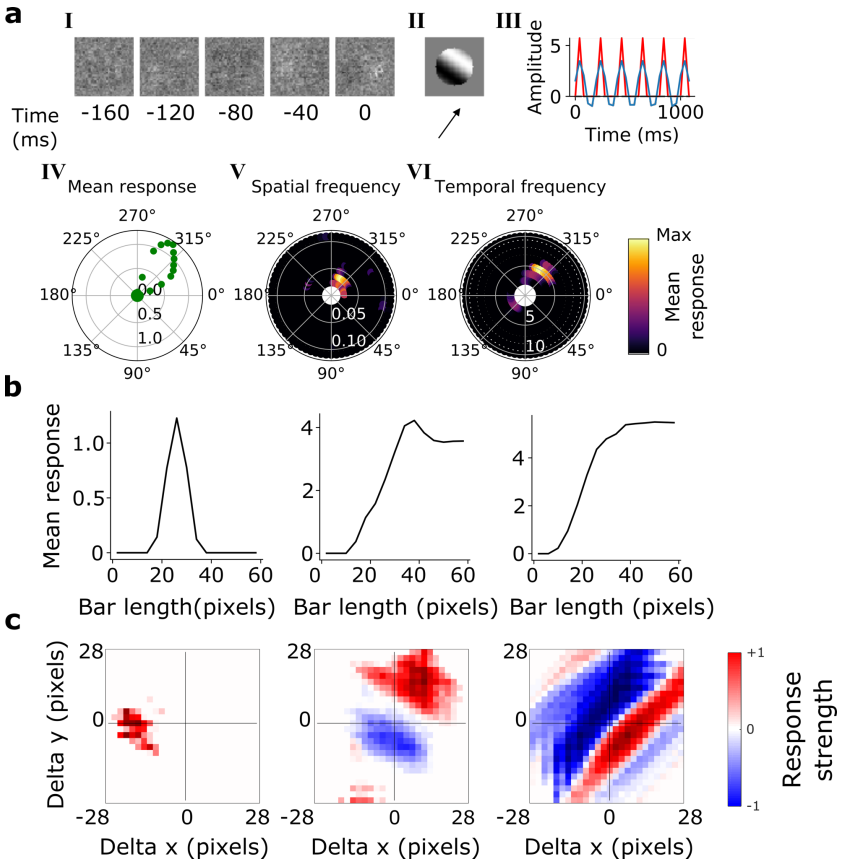


Figure 4.5 | End-stopping. a, Example end-stopped model unit. I-VI as in Fig. 4.3. **b**, Response as a function of bar length for the unit in **a** (left) and two other example units (middle, right). **c**, 2-bar

maps of units with corresponding bar-length tuning plots shown in **b**. All units shown are taken from stack 2 of the model.

To investigate pattern selectivity in our model units, we measured their responses to drifting plaids, comprising two superimposed drifting sinusoidal gratings with different orientations. The net direction of the plaid movement lies midway between these two orientations (Fig. 4.6a). In V1 and MT, component-selective cells respond maximally when the plaid is oriented such that either one of its component gratings moves in the preferred direction of the cell (as measured by a drifting grating). This results in two peaks in plaid-direction tuning curves (Movshon et al., 1985; Rust et al., 2006; Smith et al., 2005). Conversely, pattern-selective cells (in macaque typically seen in MT and not V1) have a single peak in their direction tuning curves, when the plaid's direction of movement aligns with the preferred direction of the cell (Movshon et al., 1985; Rust et al., 2006; Smith et al., 2005). We see examples of both component-selective units (in stacks 2-4) and pattern-selective units (only in stack 4) in our model, as indicated by plaid-direction tuning curves (Fig. 4.6b) and plots of response as a function of the directions of each component (Fig. 4.6c). The (present-estimating) stacked-autoencoder and shuffled network controls did not show pattern-motion-selective units.

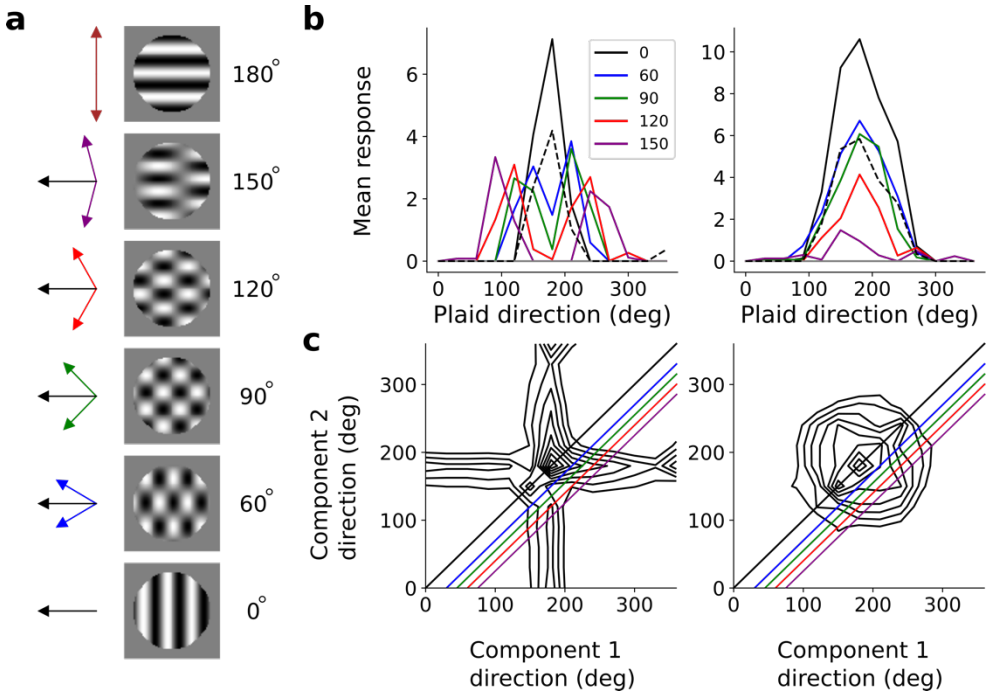


Figure 4.6 | Pattern sensitivity a, Example plaid stimuli used to measure pattern selectivity. Black arrow, direction of pattern motion. Colored arrows, directions of component motion. **b,** Direction tuning curves showing the response of an example component-selective (left) and pattern-selective (right) unit to grating and plaid stimuli. Colored lines, response to plaid stimuli composed of gratings with the indicated angle between them. Black solid line, unit's response to double intensity grating moving in the same direction as plaids. Dotted line, response to single intensity grating moving in the same direction. Gray line, response to blank stimulus. **c,** Surface contour plots showing response of units in **b** to plaids as a function of the direction of the grating components. Colored lines denote loci of plaids whose responses are shown in the same colors in **b**. Contour lines range from 20% of the maximum response to the maximum in steps of 10%. For clarity, all direction tuning curves are rotated so that the preferred direction of the response to the optimal grating is at 180˚. Responses are mean amplitudes over time.

The hierarchical temporal prediction model is not explicitly designed to model the pathway upon which MT lies (i.e. the dorsal pathway) over the many other higher visual areas (such as those on the ventral pathway). Some units in the 4th stack of the model could also be representing features found in V1 or in some of these other higher areas; however, in this chapter we have chosen to focus on motion sensitivity such as is found in the dorsal pathway. One feature of MT neurons is that the vast majority (>90%, (Wang and Movshon, 2016)) are highly direction sensitive (DSI>0.5), hence we took all direction-selective units

in stack 4 (n=60, DSI>0.5) as putative MT-like neurons. Only a few (5%, n=3/60) of these direction-selective units are pattern-motion selective, a much lower proportion than is seen in MT (~23%, n=179/792) (Wang and Movshon, 2016). It is possible that in order to see a similar proportion of pattern-selective units to MT we would need a model incorporating recurrent processing, as is suggested by some (Pack et al., 2001), but not all (Rust et al., 2006) physiological studies of pattern-selectivity in MT. A further possible reason for the low proportion of pattern-selective cells is that the videos used to train our model typically involve animals moving in natural environments filmed with still or slowly panning cameras. Incorporating other types of motion into the training videos, such as self-motion or eye movements, might increase the proportion of pattern-selective units.

4.3 Discussion

We have presented a simple model that hierarchically instantiates temporal prediction. This model has several advantages: it is unsupervised; it operates over spatiotemporal inputs and its hierarchical implementation is general, allowing the same network model to learn features resembling each level of the visual processing hierarchy with little fine-tuning or modification to the model structure at each stage. This simple model accounts for many spatial and temporal tuning properties of cells in the dorsal visual pathway, from the center-surround tuning of cells in the retina and LGN, to the spatiotemporal tuning and direction selectivity of V1 simple and complex cells, and the motion processing of end-stopped and pattern selective cells.

There are other normative models of visual processing, based on a range of principles, which can account for a number of properties of visual neurons. Prominent theories include predictive coding, sparse coding, independent component analysis (ICA) and temporal coherence. The predictive coding framework postulates that sensory systems

learn the statistical regularities present in natural inputs, feeding forward the errors caused by deviations from these regularities to higher areas (Huang and Rao, 2011; Rao and Ballard, 1999; Srinivasan et al., 1982). In this process, the predictable components of the input signal are removed and only unexpected inputs are fed forward through the hierarchy. This should be distinguished from temporal prediction, which performs selective coding, where predictive elements of the input are explicitly represented in neuronal responses and non-predictable elements are discarded (Bialek et al., 2001; Chalk et al., 2018; Salisbury and Palmer, 2016). Sparse coding (Olshausen and Field, 1997, 1996), which shares similarities with predictive coding, is built on the idea that an overcomplete set of neurons is optimized to represent inputs as accurately as possible using only few active neurons for a given input. ICA (Bell and Sejnowski, 1997; van Hateren and Ruderman, 1998; van Hateren and van der Schaaf, 1998) is a related framework that finds maximally independent features of the inputs. Sparse coding and ICA are practically identical in cases where a critically complete code is used (Huang and Rao, 2011; Olshausen and Field, 1997). In these frameworks, as with predictive coding, the aim is to encode all current or past input, whereas in temporal prediction, only features that are predictive of the future are encoded and other features are discarded.

Another set of approaches, slow feature analysis (Berkes and Wiskott, 2005; Wiskott and Sejnowski, 2002) (SFA) and slow subspace analysis (Kayser et al., 2001) (SSA), stem from the idea of temporal coherence (Földiák, 1991), which suggests that a key goal of sensory processing is to identify slowly varying features of natural inputs. SFA is related to temporal prediction because features that vary slowly are likely to be predictive of the future. However, SFA and temporal prediction may give different weighting to the features that they find (Creutzig and Sprekeler, 2008), and SFA could also fail to capture features that do not vary slowly, but are predictive of the future.

We will focus on unsupervised normative models (i.e. those trained on natural inputs) because they are the most relevant to our model. Broadly, these models can be divided into several categories: local models, trained to represent features of a specific subset of neurons, such as simple cells in V1, and hierarchical models, which attempt to explain features of more than one cell type (such as simple and complex cells) in a single model. These two categories can be further divided into models that are trained on natural spatial inputs (images) and those that are trained on natural spatiotemporal inputs (movies).

Among local models, sparse coding and ICA are the standard normative models of V1 simple cell RFs (Bell and Sejnowski, 1997; Olshausen, 2003; Olshausen and Field, 1997, 1996; van Hateren and Ruderman, 1998; van Hateren and van der Schaaf, 1998). Typically, these models are trained using still natural images and have shown remarkable success in accounting for the spatial features of V1 simple cell RFs (Bell and Sejnowski, 1997; Olshausen and Field, 1997, 1996; van Hateren and van der Schaaf, 1998). However, models trained on static images are unable to account for temporal aspects of neuronal RFs, such as direction selectivity or two-dimensional motion processing. The ICA and sparse coding frameworks have been extended to model features of spatiotemporal inputs (Chalk et al., 2018; Olshausen, 2003; van Hateren and Ruderman, 1998). While these models capture many of the spatial tuning properties of simple cells, as discussed in the previous chapter, they tend to produce symmetric temporal envelopes that do not match the asymmetric envelopes of real neurons (Singer et al., 2018). Capturing temporal features is especially important when building a normative model of the dorsal visual stream, which is responsible for processing cues related to visual motion. When trained to find slowly varying features in natural video inputs, SFA models (Berkes and Wiskott, 2005) find features with tuning properties that closely resemble those of V1 complex cells, including phase invariance to drifting sinusoidal gratings and end- and side-inhibition. A sparse prior must be applied to

the activities of the model units in order to produce spatial localization – a key feature of V1 complex cells (Lies et al., 2014). Although SFA can account for complex cell tuning, on its own this framework does not provide a normative explanation for simple cells.

A notable hierarchical model, predictive coding (Rao and Ballard, 1999) provides a powerful framework for learning hierarchical structure from visual inputs in an unsupervised learning paradigm. When applied to natural images, predictive coding has been used successfully to explain the orientation tuning of simple cells in V1 and some nonlinear tuning properties of neurons in this area, such as end-stopping (Rao and Ballard, 1999). However, it is not clear whether this framework can reproduce the phase-invariant tuning of complex cells. Nor has it been shown to account for direction selectivity, end-stopped tuning for motion (Pack et al., 2003), or pattern motion sensitivity (Movshon et al., 1985).

Hierarchical ICA models (and related models) provide another approach. These consist of two-layer networks that are trained on natural images with an independence prior placed on the unit activities (Hyvärinen and Hoyer, 2000, 2001; Karklin and Lewicki, 2009; Osindero et al., 2006). They have been shown to produce simple cell subunits in the first layer of the network and phase-invariant tuning reminiscent of complex cells in the second layer. However, these models typically incorporate aspects to specifically encourage complex cell-like characteristics in the form of a quadratic nonlinearity resembling the complex cell energy model (Adelson and Bergen, 1985). In some models (Hyvärinen and Hoyer, 2000), phase invariance is enforced by requiring the outputs of individual subunits to be uncorrelated (Kording et al., 2003). This is in contrast to our model where the phase invariance is learned as a consequence of the general principle of finding features that can efficiently predict future input. An advantage to learning the invariance rather than hand-crafting it is that the identical model architecture can then be applied hierarchically to explain features in higher visual areas without changing the form of the model.

Some hierarchical models based on temporal coherence have been trained on spatiotemporal inputs (videos of natural scenes). These models have been shown to capture properties of both simple and complex cells (Hurri and Hyvärinen, 2003; Kayser et al., 2001; Kording et al., 2003). Typically, they share a similar structure and assumptions with the hierarchical ICA models outlined above, consisting of a two-layer model where the outputs of one or more simple cell subunits are squared and passed forward to a complex cell layer. Other models have combined sparsity/independence priors with a temporal slowness constraint in a hierarchical model (Berkes et al., 2009; Cadieu and Olshausen, 2012; Chen et al., 2018; Hyvärinen et al., 2003). Since sparsity constraints tend to produce simple cell tuning and slowness constraints result in complex cell tuning, these models produce units with both types of selectivity. This contrasts with our model which produces both types of selectivity with a single objective and also accounts for some tuning properties of higher visual areas.

Previous studies using principles related to temporal prediction have produced non-hierarchical models of retina or primary cortex alone and demonstrated retina-like RFs (Ocko et al., 2018), simple-cell like RFs (Chalk et al., 2018; Palm, 2012; Singer et al., 2018), or RFs resembling those found in primary auditory cortex (Singer et al., 2018). However, any general principle of neural representation should be extendable to a hierarchical form and not tailored to the region it is attempting to explain. Here we show that temporal prediction can indeed be made hierarchical and so reproduce the major motion-tuning properties that emerge along the dorsal visual pathway from retina to MT. This model captures not only linear tuning features such as those seen in center-surround retinal neurons and direction-selective simple cells, but also nonlinear features seen in complex cells, end-stopped cells and pattern-sensitive neurons. The capacity of our hierarchical temporal prediction model to account for so many tuning features at multiple levels of the visual

system suggests that the same framework may well explain many more features of the brain than we have investigated. Furthermore, iterated application of temporal prediction, as performed by our model, could be used to make predictions about the tuning properties of neurons in brain pathways that are much less well understood than those of the visual system. Our results suggest that, by learning behaviorally-useful features from dynamic unlabeled data, temporal prediction may represent a fundamental coding principle in the brain.

4.4 Methods

4.4.1 Data used for model training and testing

Visual inputs

Videos (grayscale, without sound, sampled at 25 fps) of wildlife in natural settings were used to create visual stimuli for training the artificial neural network. The videos were obtained from http://www.arkive.org/species, contributed by: BBC Natural History Unit, http://www.gettyimages.co.uk/footage/bbcmotiongallery; BBC Natural History Unit & Discovery Communications Inc., http://www.bbcmotiongallery.com; Granada Wild, http://www.itnsource.com; Mark Deeble & Victoria Stone Flat Dog Productions Ltd., http://www.deeblestone.com; Getty Images, http://www.gettyimages.com; National Geographic Digital Motion, http://www.ngdigitalmotion.com. The longest dimension of each video frame was clipped to form a square image. Each frame was then down-sampled (using bilinear interpolation) over space, to provide 181 x 181 pixel frames. The video patches were cut into non-overlapping clips, each of 20 frames duration (800 ms). We used a training set of N = ~1305 clips from around 17 min of video, and a validation set of N = ~145 clips. Finally, each clip was normalized by subtracting the mean and dividing by the standard deviation of that clip.

4.4.2 Hierarchical temporal prediction model

The model and cost function

The hierarchical temporal prediction model consisted of stacked feedforward single-hidden-layer 3D convolutional neural networks. Each stack consisted of an input layer, a convolutional hidden layer and a 'transposed convolutional' output layer. Each unit (convolutional kernel) in the hidden layer performed 3D convolution over its inputs (over time and 2D space; Fig. 4.1) and its output was determined by passing the result of this operation through a rectified linear function. Following the hidden layer there was a 'transposed convolutional' output layer, which again performed convolution (and dilation for stride >1). Each stack was trained to minimize the difference between its output and its target. The target was the input at the immediate future time step.

The first stack of the model was trained to predict the immediate future frame (40 ms) of unfiltered natural video inputs from the previous 5 frames (200 ms). Each subsequent stack was then trained to predict the immediate future hidden-unit activity of the stack below it from the past hidden-unit activity in response to the natural video inputs. This process was repeated until 4 stacks had been trained. The first stack used 50 hidden units and this number was doubled with each added stack, until we had 400 units in the 4th stack.

More formally, each stack of model can be described by a network of the same form. The input to the network has $i = 1$ to to I input channels. For channel i, for clip n, the input $\underline{\mathbf{U}}_{in}$ is a rank-3 tensor spanning time and 2D-space with $x = 1$ to X and $y = 1$ to Y spatial positions, and $t = 1$ to T time steps. Throughout the Methods, capital, bold and underlined variables are rank-3 tensors over time and 2D-space, otherwise variables are scalars. The first stack has only a single input channel (the grayscale input frames, $I = 1$). Each subsequent

stack had as many input channels (I) as the number of hidden units (feature maps) in the previous stack.

The network has a single hidden layer of $j = 1$ to J convolutional kernels. For clip n and kernel j, the output of each kernel is a rank-3 tensor over time and 2D-space, $\underline{\mathbf{H}}_{jn}$:

$$\underline{\mathbf{H}}_{jn} = \mathrm{f}\left(b_j + \sum_{i=1}^{I} \underline{\mathbf{U}}_{in} * \underline{\mathbf{W}}_{ji}\right) \tag{4.1}$$

The parameters in Equation 1 are the connective input weights of kernels $\underline{\mathbf{W}}_{ji}$ (between each input channel i and hidden unit j) and the bias b_j (of hidden unit j). f( ) is the rectified linear function and * is the 3D convolutional operator over the two spatial and one temporal dimensions of the input, with stride (s_1, s_2, s_3). Each hidden layer kernel $\underline{\mathbf{W}}_{ji}$ is 3D with size (X', Y', T'). No zero padding is applied to the input.

The output of the network predicts the future activity of the input. Hence, the number of input channels (I) always equals the number of output channels (K) for each stack. To ensure that the predicted output has the same size as the input when a stride of >1 is used, the hidden layer representation is dilated by adding s-1 zeros between adjacent elements, where $s = (s_1, s_2, s_3)$ is the stride of the convolutional operator in the hidden layer. The dilated hidden-unit output is $\underline{\mathbf{H}}_{jn}^{\mathrm{dil}}$. When stride=1, $\underline{\mathbf{H}}_{jn}^{\mathrm{dil}} = \underline{\mathbf{H}}_{jn}$.

The activity $\underline{\widehat{\mathbf{V}}}_{kn}$ of each output channel k is the estimate of the true future $\underline{\mathbf{V}}_{kn}$ given the past $\underline{\mathbf{U}}_{in}$. $\underline{\mathbf{V}}_{kn}$ is simply $\underline{\mathbf{U}}_{in}$ shifted one time step into the future, for every element $k = i$. This prediction $\underline{\widehat{\mathbf{V}}}_{kn}$ is estimated from the hidden unit output $\underline{\mathbf{H}}_{jn}^{\mathrm{dil}}$ by:

$$\underline{\widehat{\mathbf{V}}}_{kn} = b_k + \sum_{j=1}^{J} \underline{\mathbf{H}}_{jn}^{\mathrm{dil}} * \underline{\mathbf{W}}_{kj} \tag{4.2}$$

The parameters in Equation 2 are the connective output kernels $\underline{\mathbf{W}}_{kj}$ (the weights between each hidden unit j and output channel k) and the bias b_k. Each output kernel $\underline{\mathbf{W}}_{kj}$ is 3D with size (X',Y',1), predicting a single time step into the future based on hidden layer activity in that portion of space.

The parameters $\underline{\mathbf{W}}_{ji}$, $\underline{\mathbf{W}}_{kj}$, b_j, and b_k were optimized for the training set by minimizing the cost function given by:

$$E = \frac{1}{NKXYT}\sum_{n=1}^{N}\sum_{k=1}^{K}\left\|\underline{\hat{\mathbf{V}}}_{kn}-\underline{\mathbf{V}}_{kn}\right\|_2^2 + \lambda\left(\sum_{i=1}^{I}\sum_{j=1}^{J}\left\|\underline{\mathbf{W}}_{\mathrm{ji}}\right\|_1 + \sum_{j=1}^{J}\sum_{k=1}^{K}\left\|\underline{\mathbf{W}}_{kj}\right\|_1\right) \tag{4.3}$$

Where $\|\ \|_p$ is the entrywise p-norm of the tensor over time and 2D-space, p = 2 is the sqrt of the sum of squares of all values in the tensor, and p = 1 is the sum of absolute values. Thus, E is the sum of the squared error between the prediction $\underline{\hat{\mathbf{V}}}_{kn}$ and the target $\underline{\mathbf{V}}_{kn}$, plus an L_1-norm regularization term, which is proportional to the sum of absolute values of all weights in the network and its strength is determined by the hyper-parameter λ. This regularization tends to drive redundant weights to near zero and provides a parsimonious network.

Implementation details

The networks were implemented in Python (https://lasagne.readthedocs.io/en/latest/; http://deeplearning.net/software/theano/). The objective function was minimized using backpropagation as performed by the Adam optimization method (Kingma and Ba, 2014) with hyperparameters β1 and β2 kept at their default settings of 0.9 and 0.999, respectively, and the learning rate (α) varied as detailed below. Training examples were split into minibatches of 32 training examples each.

During model network training, several hyperparameters were varied, including the regularization strength (λ) and the learning rate (α). For each hyperparameter setting, the training algorithm was run for 1000 iterations. The effect of varying λ on the prediction error (the first term of Equation 3) and receptive field structure of the first stack is shown in Fig. 4.2. For all subsequent stacks, we varied λ between 10^{-5} and 10^{-7} and picked the network with the lowest prediction error (mean squared error) on a held-out validation set. We measured the predictive capacity of each network by taking the average prediction error on the validation set over the final 50 iterations. The settings for each stack are presented in Table 1:

Table 1 | Model parameter settings for each stack.

Stack	Input size (X,Y,T,I)	Hidden layer Size	Kernel size (X',Y',T')	Spatial and temporal extent	Stride (s_1,s_2,s_3)	Number of kernels	Learning rate (α)	L_1 Regularization strength (λ)
1	181x181x20x1	17x17x16x50	21x21x5	21x21x5	10x10x1	50	10^{-2}	$10^{-4.5}$
2	17x17x16x50	15x15x12x100	3x3x5	41x41x9	1x1x1	100	10^{-4}	10^{-6}
3	15x15x12x100	13x13x8x200	3x3x5	61x61x13	1x1x1	200	10^{-4}	10^{-6}
4	13x13x8x200	11x11x4x400	3x3x5	81x81x17	1x1x1	400	10^{-4}	10^{-6}

4.4.3 Model unit spatiotemporal extent and receptive fields

Due to the convolutional form of the hidden layer, each hidden unit can potentially receive from a certain span over space and time. We call this the unit's spatial and temporal extent. For stack 1, this extent is given by the kernel size (21 x 21 x 5, space x space x time). For stack 2, the extent of each hidden unit is a function of its kernel size and the kernel size and stride of the hidden units in the previous stack, resulting in an extent of 41 x 41 x 9. Similarly,

the extent of each hidden unit in stack 3 is 61 x 61 x 13 and in stack 4 is 81 x 81 x 17. The RF size of a unit can be considerably smaller than the hidden unit's extent.

In the first stack of the model, the combination of linear weights and nonlinear activation function are similar to the basic linear non-linear (LN) model (Chichilnisky, 2001; Simoncelli et al., 2004) commonly used to describe neuronal RFs. Hence, the input weights between the input layer and a hidden unit of the model network are taken directly to represent the unit's RF, indicating the features of the input that are important to that unit. The output activities of hidden units in stacks 2-4 are transformations with multiple linear and nonlinear stages, and hence we estimated their RFs by applying reverse correlation. To do this, we measured the responses of each unit to 100,000 binary noise inputs with amplitude ±3, which were presented at the input layer (of stack 1), and performed reverse correlation.

4.4.4 In vivo V1 receptive field data

Responses to drifting gratings measured using recordings from V1 simple and complex cells were compared against the model (Fig. 4.4). The *in vivo* data were taken from http://www.ringachlab.net/lab/Data.html (Ringach et al., 2002).

4.4.5 Receptive field size and polarity

We measured the size of the RFs of the units in the first stack and examined the relationship between the RF size and the proportion of the RFs switching polarity. For each unit, all pixels in the most recent time step with intensities ≥50% of the maximum pixel intensity in that time step are included in the RF. The RF size was determined by counting the number of pixels fitting this criterion. We then counted the proportion of pixels included in the RF that changed sign (either positive to negative or vice versa) between the two most recent

time steps. The relationship between these two properties for the units in the first stack is shown in Fig. 4.2b.

4.4.6 Drifting sinusoidal gratings

In order to characterize the tuning properties of the model's visual RFs, we measured the responses of each unit to full-field drifting sinusoidal gratings. For each unit, we measured the response to gratings with a wide range of orientations, spatial frequencies and temporal frequencies until we found the parameters that maximally stimulated that unit (giving rise to the highest mean response over time). We define this as the optimal grating for that unit. In cases where orientation or tuning curves were measured, the gratings with optimal spatial and temporal frequency for that unit were used and were varied over orientation. Each grating alternated between an amplitude of ±3 on a gray (0) background. Some units, especially in higher stacks, had weak or no responses to drifting sinusoidal gratings. To account for this, we excluded any units with a mean response (over time) of <1% of the maximum mean response of all the units in that stack. As a result of this, 0/100, 88/200 and 261/400 units were excluded from the 2nd, 3rd and 4th stacks, respectively.

We measured several aspects of the V1 neuron and model unit responses to the drifting gratings. For each unit, we measured the circular variance, orientation bandwidth, modulation ratio and direction selectivity.

Circular variance

Circular variance (CV) is a global measure of orientation selectivity. For a unit with mean response over time r_q to a grating with angle θ_q, with angles θ spanning the range of 0 to 360˚

in equally spaced intervals of 5˚ and measured in radians, the circular variance is defined as (Ringach et al., 2002):

$$CV = 1 - \frac{\sqrt{(\sum_q r_q \sin(2\theta_q))^2 + (\sum_q r_q \cos(2\theta_q))^2}}{\sum_q r_q} \quad (4.4)$$

Orientation bandwidth

We also measured the orientation bandwidth (Ringach et al., 2002), which provides a more local measure of orientation selectivity. First, we smoothed the direction tuning curve with a Hanning window filter with a half-width at half-height of 13.5°. We then determined the peak of the orientation tuning curve. The orientation angles closest to the peak for which the response was $1/\sqrt{2}$ (or 70.7%) of the peak response were measured. The orientation bandwidth was defined as half of the difference between these two angles. We limited the maximum orientation bandwidth to 180°.

Modulation ratio

We measured the modulation ratio of each unit's response to its optimal sinusoidal grating. The modulation ratio is defined as:

$$M = F1/F0 \quad (4.5)$$

where F1 is the amplitude of the best-fitting sinusoid to the unit's response over time to the drifting grating. F0 is the mean response to the grating over time.

Direction selectivity index

To measure the direction selectivity index, we obtained each unit's direction tuning curve at its optimal spatial and temporal frequency. We measured the peak of the direction tuning curve, indicating the unit's response to gratings presented in the preferred direction (r_p) as well as the response to the grating presented in the opposite (non-preferred) direction (r_{np}). The direction selectivity index is then defined as:

$$DSI = (r_p - r_{np})/(r_p + r_{np}) \tag{4.6}$$

4.4.7 Measuring end-stopping

In order to investigate end-stopping, we measured the responses of the hidden units to the same set of drifting gratings but with a circular mask applied to the inputs (e.g. Fig 4.5a, ii). Masks with a range of spatial extents were tested and the response of the units as a function of this spatial extent was measured (Fig. 4.5b).

4.4.8 Sparse noise stimuli and two-bar maps

We measured the responses of the hidden units to 'sparse noise' (moving two-bar) stimuli (Pack et al., 2003). Each stimulus contained a single oriented bar over the two most recent time steps and a blank stimulus in the preceding time steps. For each unit, the bar was oriented in the preferred orientation (as measured using drifting sinusoidal gratings) of the unit being probed. The length and width of the bar were limited to 50% and 10% of the unit's spatial extent, respectively. This was typically enough for the bar to be longer than the unit's spatial RF. In the first time step with a bar, the center position of the bar (its x and y coordinate) was selected from a dense grid of spatial positions starting from the center of

visual space and covering 1/3 of the unit's spatial extent in each direction. In the second time step, another bar position was selected from the same grid. This way, displacement of the bar from each grid position to each other grid position was used to stimulate the unit. To generate two-bar response maps, we measured the response of the unit as a function of the vertical and horizontal displacement (the starting position minus the end position) and then averaged over starting position. This gives a map of the unit's response as a function of the displacement of the stimulus regardless of the starting position. We performed this procedure using all combinations of pairs of white (amplitude +3) and black (amplitude -3) bars on a gray (amplitude 0) background. This yielded four maps (white-to-white, black-to-black, white-to-black and black-to-white). We then summed the same contrast maps (white-to-white and black-to-black) and subtracted the opposite contrast maps (white-to-black and black-to-white) to yield the final two-bar map for each unit (Fig 4.5c). This preserves directional responses while eliminating the responses that depend only on the spatial position of the bars in each frame (Pack et al., 2003).

4.4.9 Drifting plaid stimuli

In order to test whether units were pattern selective, we measured their responses to drifting plaid stimuli. Each plaid stimulus was composed of two superimposed half-intensity (amplitude 1.5) sinusoidal gratings with different orientations. The net direction of the plaid movement lies midway between these two orientations (Fig. 4.6a). As with the sinusoidal inputs, plaids with a variety of orientations, spatial frequencies, temporal frequencies and spatial extents (as defined by the extent of a circular mask) were tested. For each unit, the direction tuning curves of the optimal plaid stimulus (that giving rise to the largest mean response over time) were measured (Fig. 4.6b,c).

Chapter 5 | Discussion

5.1 Summary of results

In this thesis I have explored temporal prediction as a principle underlying the representations in visual and auditory areas. In Chapter 2 I showed that when trained on auditory inputs, a single layer temporal prediction model produces receptive fields with spectrotemporal tuning properties resembling those of A1 neurons. In Chapter 3 I showed that the same model, when trained on natural video inputs, produces receptive fields with spatial and temporal characteristics resembling those of V1 simple cells. Finally, in Chapter 4 I demonstrated that a hierarchical temporal prediction model produces units with tuning properties resembling those seen along several stages of the visual hierarchy, from the center-surround tuning of retinal ganglion cells and LGN neurons to the tuning of V1 simple and complex cells to the pattern-motion sensitivity of MT neurons.

5.2 Relating normative frameworks

Ultimately, a good normative model must account for how sensory representations are shaped by the need to represent features of inputs from the natural environment that are behaviourally relevant given anatomical and physiological constraints (Simoncelli and Olshausen, 2001). Various normative frameworks place different emphasis on these three considerations (statistics of natural inputs, finding behaviourally relevant features and accounting for the constraints of neural systems). Efficient coding, predictive coding, sparse coding and ICA focus on the first and third considerations: finding a code that is optimal for representing features that occur in natural scenes while minimizing the metabolic cost of

such a representation (such as by using few neurons to represent any given input). They place less emphasis on the second consideration (finding behaviourally relevant features) and instead assume that maximal information about all input signals should be encoded. Slow feature analysis builds on this framework and posits that only features that vary slowly in time have behavioural relevance for an animal and should therefore be encoded. Temporal prediction also accounts for temporal features of natural inputs but places a different assumption on which features are likely to be behaviourally relevant, suggesting that only features that are predictive of future input should be encoded.

Temporal prediction provides a first step in extracting behaviourally relevant features from natural inputs. In addition to temporal prediction, there are many more goals that have behavioural relevance to an animal and are likely to determine which features of sensory inputs are explicitly encoded. These include both general goals such as efficient representation and more proximal goals such as the categorization, localization or tracking of objects in the surrounding environment. We explore this idea in more detail below in section 5.5.7.

5.3 Strengths of the temporal prediction model

Temporal prediction has several strengths as an objective function for sensory processing. First, it can capture underlying features in the world (Bialek et al., 2001); this is also the case with sparseness (Olshausen and Field, 1997, 1996) and slowness (Wiskott and Sejnowski, 2002), but temporal prediction will prioritize different features (Creutzig and Sprekeler, 2008). Second, it can predict future inputs, which is very important for guiding action, especially given internal processing delays (discussed in more detail in sections 1.6.1 and 5.5.5). Third, objectives such as efficient or sparse reconstruction retain everything about the stimulus, whereas an important part of neural information processing is the selective

elimination of irrelevant information (Marzen and DeDeo, 2017). Prediction provides a good initial criterion for eliminating potentially unwanted information. Fourth, prediction provides a natural method to determine the hyperparameters of the model (such as regularization strength, number of hidden units, activation function and temporal window size). Many other models select their hyperparameters depending on what best reproduces the neural data, whereas we have an independent criterion – the capacity of the network to predict the future. One notable hyperparameter is how many time steps of past input to encode. This is naturally decided by our model because only time steps that help predict the future have significant weighting. Fifth, the temporal prediction model computes neuronal activity without needing to settle to a steady state, unlike some other models (Blättler et al., 2011; Brito and Gerstner, 2016; Carlson et al., 2012; Młynarski and McDermott, 2017; Olshausen and Field, 1997, 1996). For dynamic stimuli, a model that requires settling may not reach equilibrium in time to be useful. Sixth, and most importantly, temporal prediction successfully models many aspects of the RFs of cortical neurons. In addition to accounting for spatial and spectral tuning in V1 and A1, respectively, at least as well as other normative models, it reproduces the temporal properties of RFs, particularly the asymmetry of their temporal envelopes - something few previous models have attempted to explain. Furthermore, it is able to account for the tuning properties of a diverse range of neurons from retina to V1 simple and complex cells and pattern selective cells in higher visual areas.

5.4 Comparing computational models and making experimental predictions

There are a number of ways to assess the effectiveness of different normative models of sensory processing (such as predictive coding, sparse coding and temporal prediction) in

explaining how computations are performed in the brain. In this thesis I have compared the RFs and other forms of feature selectivity for the temporal prediction model units, control models and populations of real neurons. One could also compare the moment-by-moment responses of different populations of real and model neurons to a set of natural inputs.

One technique, developed by Kriegeskorte and colleagues (Kriegeskorte, 2009; Kriegeskorte et al., 2008), involves computing a response dissimilarity matrix (RDM), which indicates the dissimilarity of the responses across all neurons in the population to every pair of stimuli. By comparing the RDMs of populations of real and model neurons to the same set of natural inputs, the extent to which a particular subset of neurons in a particular model resembles the activity of a corresponding subset of neurons in a region of the brain can be quantified. Another approach is to see how well one can linearly predict neuronal responses from model unit responses. This works under the assumption that if the model and the brain are both doing the same nonlinear transformation of the input and therefore using equivalent basis functions, it should be possible to linearly map between these two sets of these basis functions (Yamins and DiCarlo, 2016). These techniques have been used to evaluate the similarity between a range of deep supervised models and corresponding visual and auditory areas (Cadieu et al., 2014; Kell and McDermott, 2019; Kell et al., 2018; Khaligh-Razavi and Kriegeskorte, 2014; Yamins and DiCarlo, 2016). We could perform similar analyses for different stacks of the temporal prediction model and corresponding sensory areas and compare the results to other supervised and unsupervised models. This analysis could also be used to make a direct quantitative comparison between the performance of the temporal prediction model across auditory and visual modalities.

Although the results presented in this thesis suggest that temporal prediction may explain why many features of sensory neurons take the forms that they do, we are agnostic as to whether the features are hard-wired by evolution or learned over the course of

development. This is likely to depend on the region in question, with retina more hard wired and cortex a mixture of innate tuning and learning from sensory input (Huberman et al., 2008; Kiorpes, 2015). If they are learned, this suggests that while some neurons represent the predictive features, a subset of neurons in the cortex (or elsewhere) might represent the prediction while still another fraction of neurons could represent the prediction error used to train the network. It is also possible that predictive features could be learned without the need for explicit error neurons (Whittington and Bogacz, 2019).

In contrast to this, predictive coding would posit that many neurons, especially those projecting to higher areas, should encode prediction error. One way of testing which, if either, of these of these theories better explains cortical representations would be to measure the proportion of error-neurons in cortex and identify where they occur in cortical microcircuits. For example, several studies have shown the presence of a fraction of neurons along the auditory pathway that increase their firing rate when presented with stimuli that deviate from the statistical structure of other recent inputs (Rubin et al., 2016; Ulanovsky et al., 2003). This ties in to a long history of research into neuronal responses to deviant or surprising stimuli (Nelken, 2014). The extent to which the presence of these neurons supports the predictive coding or temporal prediction hypotheses is an active area of investigation (Carbajal and Malmierca, 2018; Rubin et al., 2016).

Under the hypothesis of temporal prediction, many neurons in sensory areas should represent predictive features of their inputs. Measuring the predictive capacity of populations of neurons is challenging. To tackle this, a few recent studies have used the predictive information bottleneck (outlined in section 1.6.2) to assess how much predictive information is carried by retinal neurons about simple, low dimensional stimuli (Palmer et al., 2015; Salisbury and Palmer, 2016). They show that populations of retinal ganglion cells encode the maximal amount of predictive information for a given coding capacity. However, in

order to elucidate whether and how temporal prediction of high dimensional naturalistic stimuli is performed by sensory neurons, further experimental work needs to be done.

5.5 Limitations of the temporal prediction model and future developments

Our models are simple feedforward neural networks, each with one or more layers of hidden units with sigmoidal or rectifying nonlinearities and are trained using backpropagation. Thus, the models make many simplifying assumptions that do not account for many of the known mechanisms and motifs present in networks of neurons in the brain. Our models could be improved by adding more biologically realistic constraints. This could allow the models to learn new forms of feature selectivity or improve the match between the existing features and those seen in corresponding sensory areas. Here I list a few ways in which our models differ from the biology and how these models could be improved to reduce these differences.

5.5.1 Recurrent connectivity

Neurons in biological neural networks are recurrent, with memories of recent past inputs being stored in their membrane potentials and synaptic weights. They also have many connections to other neurons in the same cortical areas (Barbour and Callaway, 2008; Cossell et al., 2015; Iacaruso et al., 2017; Ko et al., 2011; Kratz and Manis, 2015; Rossi et al., 2019) and receive top-down inputs from neurons in higher areas (Felleman and Van Essen, 1991). In contrast, the temporal prediction models presented in the previous chapters are purely feedforward, with no self, lateral or top-down recurrent connections. Incorporating recurrent connections into our model networks would provide several distinct advantages:

Currently each hidden unit in the model receives weighted inputs from several past time steps simultaneously. This is equivalent to making an assumption that the input is passed to each neuron via multiple synaptic pathways with different conduction delays. With self-recurrent connections, each unit would have an internal memory of its past activity and therefore could receive synaptic input from only the immediate past time step.

Having recurrent connections to other neurons in the same layer would allow the model neurons to perform more complex computations in a single layer. It would also provide a way for us to compare the connectivity between neurons within a given layer of the model to the connectivity seen between neurons in a corresponding cortical area. The connectivity between neurons in V1 is functionally specific. For example, there is evidence that in V1, neurons are more likely to receive excitatory inputs from neurons whose RFs share the same orientation preference and are co-axially aligned in visual space (Cossell et al., 2015; Iacaruso et al., 2017; Ko et al., 2011). Furthermore, direction-tuned neurons are more likely to receive excitatory inputs from other neurons tuned for inputs upstream in visual space (i.e. in the co-orthogonal space opposite to its direction of preference) and from inhibitory neurons whose RFs are downstream in visual space (Rossi et al., 2019).

In a recent collaboration (in preparation; Fristed et al., 2019), we have implemented a single hidden-layer temporal prediction model where units receive recurrent connections both from themselves at previous time steps and from other units in the same layer. When trained on natural video inputs, this recurrent network learned the prediction task better (having a lower mean squared prediction error) than feedforward versions of the model with equivalent dimensionality, while receiving input from only one video frame at a time. Furthermore, with a single hidden layer, these networks produced neurons with tuning properties resembling both simple and complex cells – a feature that is seen in some mechanistic models of V1 (Tao et al., 2004). Finally, the neurons in this network showed

similar patterns of connectivity to those seen in V1, with units preferentially receiving excitatory inputs from other units with the same orientation tuning and whose RFs were co-axially displaced in visual space (Cossell et al., 2015; Iacaruso et al., 2017; Ko et al., 2011). Furthermore, direction-tuned units received most excitatory inputs from upstream units and inhibitory inputs from downstream units, as is seen in V1 (Rossi et al., 2019).

The next step would be to extend the recurrent network to a hierarchical form, by stacking recurrent temporal prediction networks, and see whether this model replicates features of the biology better than is possible with the stacked feedforward network presented in Chapter 5. Another interesting avenue would be to perform a detailed comparison between the simple and complex cells produced by the hierarchical feedforward model and the recurrent model. This could offer new insight into the long-held debate (Martinez and Alonso, 2003) between hierarchical and recurrent models of complex cells. The former suggest that complex cells are a separate class of neurons to simple cells and that their tuning is formed by receiving input hierarchically from simple cells (Adelson and Bergen, 1985; Hubel and Wiesel, 1962). The latter hypothesis suggests that simple and complex cells are both from the same population of neurons but receive different levels of recurrent input (Chance and Abbott, 1999; Debanne et al., 1998; Mechler and Ringach, 2002; Tao et al., 2004).

It would be interesting to see what would happen in a recurrent model trained on auditory inputs. A further improvement to the model could be made by incorporating feedback connections from higher to lower layers in the network either in the recurrent or feedforward architectures. These connections could provide the basis for changes in neuronal selectivity with context, as is seen in V1, A1 and other low level sensory areas (Angeloni and Geffen, 2018; Crist et al., 2001; Fritz et al., 2003; Gilbert et al., 2001). They

could also help to refine the features in lower layers during training so that they are more useful to neurons in higher layers.

5.5.2 Adding linear or non-linear processing stages to model layers

The linear-nonlinear neurons used in our model provide a good first approximation of biological neural computation (section 1.3). However, real neurons typically perform other linear and nonlinear transformations of their inputs, for example, divisive normalization. Divisive normalization is thought to be a canonical computation that is performed across many stages of sensory hierarchies (Carandini and Heeger, 2012). In a recent study, we have included a divisive normalization step to each stack in the hierarchical model presented in Chapter 4. Results are still preliminary, but the addition of this term seems to produce several improvements to the model unit responses, including response saturation and pattern-selectivity that corresponds better to the biology. Notably, when divisive normalization is included, pattern-selective units respond more strongly to plaid inputs than to drifting gratings moving in their preferred direction of motion. This is a characteristic feature of pattern-selective MT neurons (Rust et al., 2006) and is not captured by the model presented in Chapter 4 (Fig. 4.6). In future work, divisive normalization or other linear or nonlinear computations could be incorporated as standard steps in the model.

5.5.3 Using spike codes instead of rate codes

Real neurons transmit information using discrete spikes, while our model units have continuous outputs, approximating neuronal firing rates. In an ongoing collaboration, we are exploring forms of the model that use units with stochastically or deterministically generated all-or-none spiking outputs. With this model we hope to identify what computational advantages are gained by incorporating these more realistic aspects of biological neurons. A

possible advantage is that using stochastic neurons might allow the network to represent a distribution of possible futures by sampling rather than deterministically predicting the average future for a given past, as is done in the models presented in the Chapters 2-4. This idea is explored in more detail in the following section.

5.5.4 Modifying assumptions about the conditional distribution of futures

In the models presented, we have used a mean squared error (L_2) loss term. This makes the implicit assumption that there are a number of possible futures for a given past and that these futures are Gaussian distributed about the future value that we predict. We predict the mean of this distribution under the assumption that this distribution is an isotropic multivariate Gaussian. In natural scenes, for a given past there are many possible futures, the distribution of which is likely to be multi-modal and non-Gaussian (Mathieu et al., 2015).

We could change our assumptions about the distribution, trying for instance an L_1 loss term, which assumes that possible futures are drawn from a Laplacian rather than a Gaussian distribution. Mathieu et al (Mathieu et al., 2015) have previously shown that using an L_1 loss during network training gives rise to better predictions of future movie frames than using an L_2 loss, as quantified by similarity and sharpness measures. This is especially true for frames that are further in the future than the immediate next frame. It would be interesting to see how the representations learned by our temporal prediction model change when using such a loss term.

We could also try to model the full distribution of possible futures instead of just predicting the mean future value. One way to accomplish this would be to add variance and covariance terms to the model output and learn those parameters from the data. Another approach would be to use a Generative Adversarial Network (GAN) (Goodfellow et al., 2014), which provides a very flexible way of estimating likely futures from multimodal

distributions (Mathieu et al., 2015; Vondrick and Torralba, 2017; Zhao et al., 2016). In brief, a GAN consists of two parts: a generator and a discriminator, which are trained simultaneously. In this context, the generator predicts a future frame of the input, given one or more past frames. The role of the discriminator is then to distinguish between the true future frame and the generator's predicted frame. During training, the discriminator estimates the probability of the generated prediction being from the true distribution of futures and assigns a reward to the generator proportionally. In this way, many assumptions about the exact form of the output distribution are removed. A third way of modelling a distribution of possible futures would be to use stochastic spiking neurons in the model network. The stochastic nature of the neuronal firing has been suggested to represent probability distributions through sampling (Hoyer and Hyvärinen, 2003; Orbán et al., 2016). Using stochastic neurons in the model may allow predictions to be drawn as samples from a distribution which is learned rather than deterministically predicting a single future for a given set of past inputs.

5.5.5 Time-scales of prediction

In the models presented in this thesis, prediction was performed over short time-scales, predicting between 1 and 9 time steps (5-45ms) into the future in the auditory model (section 2.6), between 1 and 6 time steps (40-240ms) ahead in the single layer visual model (section 3.4) and 1 time step (40ms) ahead in each stack of the hierarchical visual model (section 4.2.1). These time-scales may be enough to account for short transduction, conduction and transmission delays (Camalier et al., 2012; Maunsell and Gibson, 1992; Schmolesky et al., 1998), as well as for very rapid motor responses (Bixler et al., 1967; Yeomans and Frankland, 1996). However, for an animal, prediction is likely to have relevance over a much wider range of time-scales - from milliseconds to years (Bialek et al., 2001; Hawkins and Blakeslee, 2004; Palmer et al., 2015). Furthermore, prediction on longer time-scales may be

more feasible for some higher-level statistics. For example, one may be able to predict the variance of the input further into the future than its exact value.

It would be interesting to investigate how the model-unit representations would change given different time-scales of prediction, especially in higher stacks of the hierarchical model. Because predictions typically become more uncertain the further you predict into the future (see for example, Mathieu et al. (2015)) incorporating techniques that account for this uncertainty (such as those listed in the preceding section) might be necessary in order to predict successfully on longer time-scales.

5.5.6 Using more biologically realistic learning rules

Our model is trained using backpropagation over a single hidden layer for each stack. Although the biological plausibility of backpropagation has long been questioned, increasingly biologically realistic methods for training networks are being developed (Whittington and Bogacz, 2019). One especially interesting line of research examines how learning akin to backpropagation could be approximated by local sets of neurons using spike time-dependent plasticity (Bengio et al., 2017, 2015). For instance, a recent study has observed that a local circuit implementing predictive coding for supervised learning, where weights between neurons are updated using simple Hebbian plasticity, leads to learning that approximates backpropagation (Whittington and Bogacz, 2017). It would be interesting to see what types of biologically plausible circuit motifs and weight-update rules would facilitate learning in the context of temporal prediction.

5.5.7 Incorporating other goals into the model

There are two main approaches taken when modelling sensory representations: bottom-up (unsupervised) learning and top-down (supervised) learning. The bottom-up approach typically starts with a close examination of the statistics of natural inputs. These models then

attempt to explain how these statistics may be used by sensory systems to carry out very general goals of the system. Examples of these goals include efficient representation (Attneave, 1954; Barlow, 1961; Bell and Sejnowski, 1997; Olshausen and Field, 1996; Srinivasan et al., 1982) or, as in our case, efficient prediction of future input. In contrast, the top-down approach examines specific (task-relevant) goals of a system and trains models to perform these goals using large labelled datasets. Examples of these goals include object classification, object tracking, sound source localization, speech recognition and many others. Deep learning models utilizing large labelled datasets perform particularly well on these tasks (LeCun et al., 2015). Top-down approaches are useful because they provide models that can be used in many practical applications. However, they typically require large datasets with many thousands or millions of labelled training examples. The brain does not have access to labelled data save for the occasional reward or punishment signal, which is sparse and hard to relate to raw sensory inputs. Thus, as tools for understanding sensory processing, supervised models can provide only limited insight into how the brain learns to represent its inputs. Bottom-up, unsupervised approaches extract features directly from the statistics of inputs and therefore require little-to-no labelled data. Furthermore, while some studies examine the links between deep supervised models and corresponding areas of the brain (Cadieu et al., 2014; Kell and McDermott, 2019; Kell et al., 2018; Khaligh-Razavi and Kriegeskorte, 2014; Yamins and DiCarlo, 2016), in most cases deep learning models are engineered for task performance and have only weak links to biological neural systems.

Combining top-down and bottom-up approaches for machine learning purposes has been an area of intense investigation and encompasses the fields of semi-supervised and transfer learning (Bengio, 2012; Bengio et al., 2012, 2006; Hinton et al., 2006; Lee et al., 2009; Raina et al., 2007). Here, models are trained using unlabelled data to produce useful representations that help in the performance of specific tasks (such as classification etc.) and

then fine-tuned using few labelled training examples. Sparse coding models, ICA and denoising autoencoders have been widely used in these paradigms (Bengio, 2012; Bengio et al., 2012). More recently, models performing next-frame video prediction have been used in an attempt to find representations that are useful for goals such as action recognition (Srivastava et al., 2015; Vondrick et al., 2016), object classification (Canziani and Culurciello, 2017; Vondrick and Torralba, 2017) and action generation in video games (Oh et al., 2015). Very few studies (Lotter et al., 2018; Palm, 2012), however, have made any attempt to compare the representations learned by these models to those of visual neurons.

We have taken a bottom-up approach in the design of our temporal prediction model and have found extensive similarities between the representations learned by the model and those used by sensory neurons. It would be interesting to see what would happen if additional, proximal goals (such as action recognition or object classification) were added to our temporal prediction model. We could then explore whether the neural-like representations learned by our model are useful for the performance of these tasks. We could also examine how the representations of the model units change with the addition of these auxiliary goals.

Furthermore, we could explore the effects of including additional unsupervised goals in the model. For instance, temporal prediction and predictive coding related frameworks are not necessarily exclusive of each other. Indeed, the brain could be constrained by both principles to a degree. Or, perhaps different systems could be governed more by one principle than another.

5.5.8 Modelling other sensory areas

In this thesis, I have shown how the temporal prediction model can account for many of the known tuning properties of neurons along successive stages of the visual pathway and in A1. However, the real power of modelling lies in the ability to make novel predictions that can

then be tested experimentally. The auditory system is hierarchical (section 1.2). However, the way in which tuning progresses from IC to A1 and from A1 to higher cortical areas is far less well understood than in equivalent pathways of visual system (King et al., 2018; Sharpee et al., 2011). A few studies have suggested ways in which neurons might become selective for increasingly abstract features as one moves up the auditory hierarchy (Carruthers et al., 2015; Kell et al., 2018; Sharpee et al., 2011; Tian et al., 2013). For instance, some authors suggest that there may be auditory analogs to simple and complex cells in A1 (Tian et al., 2013). Others have found evidence for representations that are selective for different types of vocalizations but invariant to acoustic transformations of sounds as signals pass from A1 to higher auditory areas (Carruthers et al., 2015). Whether the auditory system contains the equivalent of 'what' and 'where' streams governing the separate processing of semantic and spatial content of sounds is a topic of contention (King et al., 2018; King and Nelken, 2009; Rauschecker, 2018), but there is evidence that these signals are highly mixed both across and within auditory cortical regions (Bizley et al., 2009). This is further complicated by the growing evidence for the importance of corticothalamocortical loops, as well as other descending pathways, in sensory processing and cognitive functions (Lohse et al., 2019). Extending the auditory temporal prediction model to a hierarchical form provides an opportunity to offer new insights into some of these controversies and importantly, to make completely novel predictions about the types of feature selectivity that might be expected as one moves up the auditory hierarchy. The forms of feature selectivity predicted by the model neurons could then be looked for experimentally through the use of electrophysiological, imaging or other techniques.

5.6 Conclusion

In this thesis I have explored the idea that efficient prediction of future input may be a key principle underlying the forms of stimulus selectivity exhibited by sensory neurons. I have shown that when trained to predict the immediate future, simple feedforward network models reproduce many of the key spectrotemporal tuning properties of A1 neurons and many spatial and temporal tuning features of neurons along the dorsal visual stream, from the retina, to V1 to MT. The capacity of the temporal prediction model to account for so many known tuning features in different visual and auditory regions suggests that the same framework may well explain many more features of the brain than we have investigated. It could also be used to make predictions about tuning properties of neurons in brain pathways that are much less well understood than those of the visual system. Indeed, temporal prediction is likely to play an important role not only in visual and auditory sensation, but in many other aspects of neural processing.

References

Adelson EH, Bergen JR. 1985. Spatiotemporal energy models for the perception of motion. *J Opt Soc Am A* **2**:284.

Angeloni C, Geffen MN. 2018. Contextual modulation of sound processing in the auditory cortex. *Curr Opin Neurobiol* **49**:8–15.

Anzai A, Ohzawa I, Freeman RD. 2001. Joint-encoding of motion and depth by visual cortical neurons: neural basis of the Pulfrich effect. *Nat Neurosci* **4**:513–518.

Atencio CA, Sharpee TO, Schreiner CE. 2008. Cooperative nonlinearities in auditory cortical neurons. *Neuron* **58**:956–966.

Attneave F. 1954. Some informational aspects of visual perception. *Psychol Rev* **61**:183–93.

Baker CL. 2001. Linear filtering and nonlinear interactions in direction-selective visual cortex neurons: a noise correlation analysis. *Vis Neurosci* **18**:465–85.

Barbour DL, Callaway EM. 2008. Excitatory local connections of superficial neurons in rat auditory cortex. *J Neurosci* **28**:11174–85.

Barlow HB. 1961. Possible principles underlying the transformations of sensory messages In: Rosenblith WA, editor. Sensory Communication. Cambridge, MA: MIT Press. pp. 217–234.

Barlow HB. 1953. Summation and inhibition in the frog's retina. *J Physiol* 69–88.

Bell AJ, Sejnowski TJ. 1997. The "independent components" of natural scenes are edge filters. *Vision Res* **37**:3327–3338.

Bengio Y. 2012. Deep learning of representations for unsupervised and transfer learning. *JMLR Work Conf Proc* **27**:17–37.

Bengio Y, Courville A, Vincent P. 2012. Representation learning: a review and new

perspectives. *IEEE Trans Pattern Anal Mach Intell* **35**:1798–1828.

Bengio Y, Lamblin P, Popovici D, Larochelle H. 2006. Greedy layer-wise training of deep networksProc. Advances in Neural Information Processing Systems. pp. 153–160.

Bengio Y, Lee D-H, Bornschein J, Mesnard T, Lin Z. 2015. Towards biologically plausible deep learning. *arXiv:15020415*.

Bengio Y, Mesnard T, Fischer A, Zhang S, Wu Y. 2017. STDP-Compatible approximation of backpropagation in an energy-based model. *Neural Comput* **29**:555–577.

Berkes P, Turner RE, Sahani M. 2009. A structured model of video reproduces primary visual cortical organisation. *PLoS Comput Biol* **5**.

Berkes P, Wiskott L. 2005. Slow feature analysis yields a rich repertoire of complex cell properties. *J Vis* **5**:579–602. doi:10.1167/5.6.9

Bialek W, De Ruyter Van Steveninck RR, Tishby N. 2006. Efficient representation as a design principle for neural coding and computation. *2006 IEEE Int Symp Inf Theory* 659–663.

Bialek W, Nemenman I, Tishby N. 2001. Predictability, complexity, and learning. *Neural Comput* **13**:2409–63.

Bixler EO, Bartlett NR, Lansing RW. 1967. Latency of the blink reflex and stimulus intensity. *Percept Psychophys* **2**:559–560.

Bizley JK, Walker KMM, Silverman BW, King AJ, Schnupp JWH. 2009. Interdependent encoding of pitch, timbre, and spatial location in auditory cortex. *J Neurosci* **29**:2064–75.

Blättler F, Hahnloser RHR, Doupe A, Hahnloser R, Wilson R. 2011. An efficient coding hypothesis links sparsity and selectivity of neural responses. *PLoS One* **6**:e25506.

Brito CSN, Gerstner W. 2016. Nonlinear Hebbian learning as a unifying principle in receptive field formation. *PLoS Comput Biol* **12**.

Cadieu CF, Hong H, Yamins DLK, Pinto N, Ardila D, Solomon EA, Majaj NJ, DiCarlo JJ. 2014. Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Comput Biol* **10**:e1003963.

Cadieu CF, Olshausen BA. 2012. Learning intermediate-level representations of form and motion from natural movies. *Neural Comput* **24**:827–866.

Camalier CR, D'Angelo WR, Sterbing-D'Angelo SJ, de la Mothe LA, Hackett TA. 2012. Neural latencies across auditory cortex of macaque support a dorsal stream supramodal timing advantage in primates. *Proc Natl Acad Sci* **109**:18168–18173.

Canziani A, Culurciello E. 2017. CortexNet: a generic network family for robust visual temporal representations. *arXiv:170602735*.

Carandini M, Demb JB, Mante V, Tolhurst DJ, Dan Y, Olshausen BA, Gallant JL, Rust NC. 2005. Do we know what the early visual system does? *J Neurosci* **25**:10577–10597.

Carandini M, Heeger DJ. 2012. Normalization as a canonical neural computation. *Nat Rev Neurosci* **13**:51–62.

Carbajal G V, Malmierca MS. 2018. The neuronal basis of predictive coding along the auditory pathway: from the subcortical roots to cortical deviance detection. *Trends Hear* **22**:2331216518784822.

Carlin MA, Elhilali M. 2013. Sustained firing of model central auditory neurons yields a discriminative spectro-temporal representation for natural sounds. *PLoS Comput Biol* **9**.

Carlson NL, Ming VL, DeWeese MR. 2012. Sparse codes for speech predict spectrotemporal receptive fields in the inferior colliculus. *PLoS Comput Biol* **8**.

Carruthers IM, Laplagne DA, Jaegle A, Briguglio JJ, Mwilambwe-Tshilobo L, Natan RG, Geffen MN. 2015. Emergence of invariant representation of vocalizations in the

auditory cortex. *J Neurophysiol* **114**:2726–2740.

Casagrande VA. 1999. The mystery of the visual system K pathway. *J Physiol* **517 (Pt 3)**:630.

Chalk M, Marre O, Tkacik G. 2016. Relevant sparse codes with variational information bottleneck. *Adv Neural Inf Process Syst* 1957–1965.

Chalk M, Marre O, Tkačik G. 2018. Toward a unified theory of efficient, predictive, and sparse coding. *Proc Natl Acad Sci U S A* **115**:186–191.

Chance F, Abbott LF. 1999. Complex cells as cortically amplified simple cells. *Nat Neurosci* **2**:277–282.

Chen Y, Paiton DM, Olshausen BA. 2018. The sparse manifold transform. *arXiv*.

Chichilnisky EJ. 2001. A simple white noise analysis of neuronal light responses. *Network* **12**:199–213.

Cossell L, Iacaruso MF, Muir DR, Houlton R, Sader EN, Ko H, Hofer SB, Mrsic-Flogel TD. 2015. Functional organization of excitatory synaptic strength in primary visual cortex. *Nature* **518**:399–403.

Creutzig F, Sprekeler H. 2008. Predictive coding and the slowness principle: an information-theoretic approach. *Neural Comput* **20**:1026–1041.

Crist RE, Li W, Gilbert CD. 2001. Learning to see: experience and attention in primary visual cortex. *Nat Neurosci* **4**:519–525.

Cusack R, Carlyon R. 2004. Auditory perceptual organization inside and outside the laboratory In: Neuhoff JG, editor. Echological Pyschoacoustics. San Diego: Elsevier. pp. 15–48.

Dahmen JC, King AJ. 2007. Learning to hear: plasticity of auditory cortical processing. *Curr Opin Neurobiol* **17**:456–464.

Dayan P, Abbott L. 2001. Theoretical neuroscience: computational and mathematical

modeling of neural systems. Cambridge, MA, USA: MIT Press.

DeAngelis GC, Ohzawa I, Freeman RD. 1995. Receptive-field dynamics in the central visual pathways. *Trends Neurosci* **18**:451–458. doi:10.1016/0166-2236(95)94496-R

DeAngelis GC, Ohzawa I, Freeman RD. 1993. Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex. I. General characteristics and postnatal development. *J Neurophysiol* **69**:1091–1117.

Debanne D, Shulz DE, Fregnac Y. 1998. Activity-dependent regulation of "on" and "off" responses in cat visual cortical receptive fields. *J Physiol* **508**:523–48.

deCharms RC, Blake DT, Merzenich MM. 1998. Optimizing sound features for cortical neurons. *Science (80-)* **280**:1439–43.

Deng J, Dong W, Socher R, Li L, Li K, Fei-fei L. 2009. Imagenet: A large-scale hierarchical image database. *CVPR*.

DiCarlo JJ, Maunsell JHR. 2005. Using neuronal latency to determine sensory–motor processing rathways in reaction time tasks. *J Neurophysiol* **93**:2974–2986.

Eliasmith C, Anderson CH. 2003. Neural engineering : computation, representation, and dynamics in neurobiological systems. Cambridge, Mass: MIT Press.

Farabet C, Couprie C, Najman L, LeCun Y. 2013. Learning hierarchical features for scene Labeling. *IEEE Trans Pattern Anal Mach Intell* **35**:1915–1929.

Felleman DJ, Van Essen DC. 1991. Distributed hierarchical processing in the primate cerebral cortex. *Cereb Cortex* **1**:1–47.

Field DJ. 1994. What Is the Goal of Sensory Coding? *Neural Comput* **6**:559–601.

Field DJ. 1987. Relations between the statistics of natural images and the response properties of cortical cells. *J Opt Soc Am A* **4**:2379.

Földiák P. 1991. Learning invariance from transformation sequences. *Neural Comput* **3**:194–200.

Fristed E, Singer Y, King AJ, Iacaruso MF, Harper NS. 2019. Prediction of future input explains lateral connectivity in primary visual cortex. *Cosyne Abstr*.

Friston K. 2003. Learning and inference in the brain. *Neural Networks* **16**:1325–1352.

Fritz J, Shamma S, Elhilali M, Klein D. 2003. Rapid task-related plasticity of spectrotemporal receptive fields in primary auditory cortex. *Nat Neurosci* **6**:1216–1223.

Gilbert CD, Sigman M, Crist RE. 2001. The neural basis of perceptual learning. *Neuron* **31**:681–697.

Goodale MA, Milner AD. 1992. Separate visual pathways for perception and action. *Trends Neurosci* **15**:20–5.

Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. 2014. Generetative adversarial networks. *Adv Neural Inf Process Syst*.

Harris KD, Mrsic-Flogel TD. 2013. Cortical connectivity and sensory coding. *Nature* **503**:51–58.

Hawkins J, Blakeslee S. 2004. On intelligence: how a new understanding of the brain will lead to the creation of truly intelligent machines. New York, NY, USA: Times Books.

Helmholtz H. 1867. Concerning the perceptions in general In: Southal (trans.) J, editor. Treatise on Physiological Optics (3rd Ed). Dover, New York. pp. 1–37.

Hinton G, Deng L, Yu D, Dahl G, Abdel-rahman M, Jaitly N, Senior A, Vanhoucke V, Nguyen P, Sainath T, Kingsbury B. 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal Process Mag* **29**:82–97.

Hinton GE. 1989. Connectionist learning procedures. *Artif Intell* **40**:185–234.

Hinton GE, Osindero S, Teh Y-W. 2006. A fast learning algorithm for deep belief nets. *Neural Comput* **18**:1527–1554.

Hoyer PO, Hyvärinen A. 2003. Interpreting neural response variability as Monte Carlo sampling of the posterior. *Adv Neural Inf Process Syst* 293–300.

Huang Y, Rao RPN. 2011. Predictive coding. *Wiley Interdiscip Rev Cogn Sci* **2**:580–593.

Hubel DH, Wiesel TN. 1968. Receptive fields and functional architecture of monkey striate cortex. *J Physiol* **195**:215–243.

Hubel DH, Wiesel TN. 1965. Receptive fields and the functional architecture in two striate visual areas (18 and 19) of the cat. *J Neurophysiol* **28**:229–89.

Hubel DH, Wiesel TN. 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J Physiol* **160**:106–154.

Hubel DH, Wiesel TN. 1959. Receptive fields of single neurones in the cat's striate cortex. *J Physiol* **148**:574–91.

Huberman AD, Feller MB, Chapman B. 2008. Mechanisms underlying development of visual maps and receptive fields. *Annu Rev Neurosci* **31**:479–509.

Hurri J, Hyvärinen A. 2003. Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Comput* **15**:663–691.

Hyvärinen a, Hoyer P. 2000. Emergence of phase- and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural Comput* **12**:1705–1720.

Hyvärinen A, Hoyer PO. 2001. A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Res* **41**:2413–2423.

Hyvärinen A, Hurri J, Väyrynen J. 2003. Bubbles: a unifying framework for low-level statistical properties of natural image sequences. *J Opt Soc Am A* **20**:1237–1252.

Iacaruso MF, Gasler IT, Hofer SB. 2017. Synaptic organization of visual space in primary visual cortex. *Nature* **547**:449–452.

Jones JP, Palmer LA. 1987a. The two-dimensional spatial structure of simple receptive fields in cat striate cortex. *J Neurophysiol* **58**:1187–1211.

Jones JP, Palmer LA. 1987b. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *J Neurophysiol* **58**:1233–1258.

Karklin Y, Lewicki MS. 2009. Emergence of complex cell properties by learning to generalize in natural scenes. *Nature* **457**:83–86.

Kavanagh GL, Kelly JB. 1988. Hearing in the ferret (Mustela putorius): effects of primary auditory cortical lesions on thresholds for pure tone detection. *J Neurophysiol* **60**:879–888.

Kayser C, Einhäuser W, Dümmer O, König P, Körding K. 2001. Extracting slow subspaces from natural videos leads to complex cells In: Dorffner G, Bischof H, Hornik K, editors. ICANN 2001. LNCS. Springer, Berlin, Heidelberg. pp. 1075–1080.

Kell AJ, McDermott JH. 2019. Deep neural network models of sensory systems: windows onto the role of task constraints. *Curr Opin Neurobiol* **55**:121–132.

Kell AJE, Yamins DLK, Shook EN, Norman-Haignere S V, McDermott JH. 2018. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron* **98**:630-644.e16.

Khaligh-Razavi S-M, Kriegeskorte N. 2014. Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Comput Biol* **10**:e1003915.

Kim T, Freeman RD. 2016. Direction selectivity of neurons in the visual cortex is non-linear and lamina-dependent. *Eur J Neurosci* **43**:1389–99.

King AJ, Nelken I. 2009. Unraveling the principles of auditory cortical processing: can we learn from the visual system? *Nat Neurosci* **12**:698–701. doi:10.1038/nn.2308

King AJ, Teki S, Willmore BDB. 2018. Recent advances in understanding the auditory

cortex. *F1000Research* **7**.

Kingma DP, Ba J. 2014. Adam: a method for stochastic optimization. *arXiv:14126980*.

Kiorpes L. 2015. Visual development in primates: Neural mechanisms and critical periods. *Dev Neurobiol* **75**:1080–90.

Klein DJ, König P, Körding KP. 2003. Sparse spectrotemporal coding of sounds. *EURASIP J Appl Signal Processing* **2003**:1–9.

Ko H, Hofer SB, Pichler B, Buchanan KA, Sjöström PJ, Mrsic-Flogel TD. 2011. Functional specificity of local synaptic connections in neocortical networks. *Nature* **473**:87–91.

Kording KP, C K, W E, P. K. 2003. How are complex cell properties adapted to the statistics of natural stimuli? *J Neurophysiol* **91**:206–212.

Koumura T, Terashima H, Furukawa S. 2019. Cascaded Tuning to Amplitude Modulation for Natural Sound Recognition. *J Neurosci* **39**:5517–5533.

Kozlov AS, Gentner TQ. 2016. Central auditory neurons have composite receptive fields. *Proc Natl Acad Sci* **113**:1441–1446.

Kratz MB, Manis PB. 2015. Spatial organization of excitatory synaptic inputs to layer 4 neurons in mouse primary auditory cortex. *Front Neural Circuits* **9**:17.

Kreile AK, Bonhoeffer T, Hübener M. 2011. Altered visual experience induces instructive changes of orientation preference in mouse visual cortex. *J Neurosci* **31**:13911–13920.

Kriegeskorte N. 2009. Relating population-code representations between man, monkey, and computational models. *Front Neurosci* **3**:363–373.

Kriegeskorte N, Mur M, Ruff DA, Kiani R, Bodurka J, Esteky H, Tanaka K, Bandettini PA. 2008. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron* **60**:1126–41.

Krizhevsky A, Sutskever I, Hinton GE. 2012. ImageNet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* **25**:1090–1098.

Kuffler SW. 1953. Discharge patterns and functional organization of mammalian retina. *J Neurophysiol* **16**:37–68.

LeCun Y, Bengio Y, Hinton G. 2015. Deep learning. *Nature* **521**:436–444.

LeCun Y, Boser BE, Denker JS, Henderson D, Howard RE, Hubbard WE, Jackel LD. 1990. Handwritten Digit Recognition with a Back-Propagation Network. *Adv Neural Inf Process Syst* 396–404.

LeCun Y, Bottou L, Bengio Y, Haffner P. 1998. Gradient-based learning applied to document recognition. *Proc IEEE* **86**:2278–2324.

Lee H, Grosse R, Ranganath R, Ng AY. 2009. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *Proc 26th Annu Int Conf Mach Learn ICML 09* **2008**:1–8.

Lies JP, Häfner RM, Bethge M. 2014. Slowness and Sparseness Have Diverging Effects on Complex Cell Learning. *PLoS Comput Biol* **10**:9–12.

Lindsey J, Ocko SA, Ganguli S, Deny S. 2019. A unified theory of early visual representations from retina to cortex through anatomically constrained deep CNNs. *arXiv:190100945*.

Livingstone MS, Conway BR. 2007. Contrast affects speed tuning, space-time slant, and receptive-rield organization of simple cells in macaque V1. *J Neurophysiol* **97**:849–857.

Lohse M, Bajo VM, King AJ. 2019. Development, organization and plasticity of auditory circuits: Lessons from a cherished colleague. *Eur J Neurosci* **49**:990–1004.

Lotter W, Kreiman G, Cox D. 2018. A neural network trained to predict future video frames mimics critical properties of biological neuronal responses and perception.

arXiv:180510734.

Lotter W, Kreiman G, Cox D. 2016. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv:160508104*.

Marr D. 1976. Early processing of visual information. *Phil Trans R Soc Lond B* **275**:441–475.

Marr D, Hildreth E. 1980. Theory of edge detection. *Proc R Soc London Ser B Biol Sci* **207**:187–217.

Martinez LM, Alonso J-M. 2003. Complex receptive fields in primary visual cortex. *Neurosci* **9**:317–331.

Marzen SE, DeDeo S. 2017. The evolution of lossy compression. *J R Soc Interface* **14**.

Mathieu M, Couprie C, LeCun Y. 2015. Deep multi-scale video prediction beyond mean square error. *ICLR*.

Maunsell JH, Gibson JR. 1992. Visual response latencies in striate cortex of the macaque monkey. *J Neurophysiol* **68**:1332–1344.

Mechler F, Ringach DL. 2002. On the classification of simple and complex cells. *Vision Res* **42**:1017–1033. doi:10.1016/S0042-6989(02)00025-1

Młynarski W, McDermott JH. 2017. Learning Mid-Level Auditory Codes from Natural Sound Statistics. *arXiv:170107138*.

Morrison GS, Rose P, Zhang C. 2012. Protocol for the collection of databases of recordings for forensic-voice-comparison research and practice. *Aust J Forensic Sci* **44**:155–167. doi:10.1080/00450618.2011.630412

Morrison GS, Zhang C, Enzinger E, Ochoa F, Bleach D, Johnson M, Folkes BK, De Souza S, Cummins N, Chow D. 2015. Forensic database of voice recordings of 500+ Australian English speakers.

Movshon JA, Adelson EH, Gizzi MS, Newsome WT. 1985. The analysis of moving visual

patterns. *Pattern Recognit Mech* 117–151.

Movshon JA, Newsome WT. 1996. Visual response properties of striate cortical neurons projecting to area MT in macaque monkeys. *J Neurosci* **16**:7733–7741.

Movshon JA, Thompson ID, Tolhurst DJ. 1978a. Spatial summation in the receptive fields of simple cells in the cat's striate cortex. *J Physiol* **283**:53–77.

Movshon JA, Thompson ID, Tolhurst DJ. 1978b. Receptive field organization of complex cells in the cat's striate cortex. *J Physiol* **283**:79–99.

Nassi JJ, Callaway EM. 2009. Parallel processing strategies of the primate visual system. *Nat Rev Neurosci* **10**:360–72.

Nassi JJ, Callaway EM. 2006. Multiple Circuits Relaying Primate Parallel Visual Pathways to the Middle Temporal Area. *J Neurosci* **26**:12789–12798.

Nelken I. 2014. Stimulus-specific adaptation and deviance detection in the auditory system: experiments and models. *Biol Cybern* **108**:655–663.

Niell CM, Stryker MP. 2008. Highly Selective Receptive Fields in Mouse Visual Cortex. *J Neurosci* **28**.

Nijhawan R. 1994. Motion extrapolation in catching. *Nature* **370**:256–257.

Nodal FR, King AJ. 2014. Hearing and Auditory Function in FerretsBiology and Diseases of the Ferret. Ames, USA: John Wiley & Sons, Inc. pp. 685–710.

O'Reilly RC, Wyatte D, Rohrlich J. 2014. Learning Through Time in the Thalamocortical Loops. *arXiv:14073432*.

Ocko SA, Lindsey J, Ganguli S, Deny S. 2018. The emergence of multiple retinal cell types through efficient coding of natural movies. *bioRxiv* 458737.

Oh J, Guo X, Lee H, Lewis R, Singh S. 2015. Action-Conditional Video Prediction using Deep Networks in Atari Games. *Adv Neural Inf Process Syst* **28**.

Ohzawa I, DeAngelis GC, Freeman RD. 1996. Encoding of binocular disparity by simple

cells in the cat's visual cortex. *J Neurophysiol* **75**:1779–805.

Olshausen BA. 2003. Learning sparse, overcomplete representations of time-varying natural images. *IEEE Int Conf Image Process*.

Olshausen BA, Field DJ. 1997. Sparse coding with an overcomplete basis set: a strategy employed by V1? *Vision Res*.

Olshausen BA, Field DJ. 1996. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*.

Orbán G, Berkes P, Fiser J, Lengyel M. 2016. Neural Variability and Sampling-Based Probabilistic Representations in the Visual Cortex. *Neuron* **92**:530–543.

Osindero S, Welling M, Hinton GE. 2006. Topographic Product Models Applied To Natural Scene Statistics. *Neural Comput* **18**:381–414.

Pack CC, Berezovskii VK, Born RT. 2001. Dynamic properties of neurons in cortical area MT in alert and anaesthetized macaque monkeys. *Nature* **414**:905–908.

Pack CC, Conway BR, Born RT, Livingstone MS. 2006. Spatiotemporal structure of nonlinear subunits in macaque visual cortex. *J Neurosci* **26**:893–907.

Pack CC, Livingstone MS, Duffy KR, Born RT. 2003. End-stopping and the aperture problem: two-dimensional motion signals in macaque V1. *Neuron* **39**:671–680.

Palm RB. 2012. Prediction as a candidate for learning deep hierarchical models of data. Technical University of Denmark, (DTU) Informatics.

Palmer SE, Marre O, Berry M, Bialek W. 2015. Predictive information in a sensory population. *Proc Natl Acad Sci* **112**:6908–6913.

Price DJ, Morgan JE. 1987. Spatial properties of neurones in the lateral geniculate nucleus of the pigmented ferret. *Exp Brain Res* **68**:28–36.

Rabinowitz NC, Willmore BDB, Schnupp JWH, King AJ. 2011. Contrast Gain Control in Auditory Cortex. *Neuron* **70**:1178–1191.

Raina R, Battle A, Lee H, Packer B, Ng AY. 2007. Self-taught learning: transfer learning from unlabeled dataProceedings of the 24th International Conference on Machine Learning - ICML '07. New York, New York, USA: ACM Press. pp. 759–766.

Ranzato M, Szlam A, Bruna J, Mathieu M, Collobert R, Chopra S. 2016. Video (language) modeling: a baseline for generative models of natural videos.

Rao RP. 1999. An optimal estimation approach to visual perception and learning. *Vis Res* **39**:1963–89.

Rao RP, Ballard DH. 1999. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nat Neurosci* **2**:79–87.

Rauschecker JP. 2018. Where, when, and how: are they all sensorimotor? Towards a unified view of the dorsal pathway in vision and audition. *Cortex* **98**:262–268.

Rees A, Møller AR. 1983. Responses of neurons in the inferior colliculus of the rat to AM and FM tones. *Hear Res* **10**:301–330.

Ringach DL. 2002. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. *J Neurophysiol* **88**:455–463.

Ringach DL, Shapley RM, Hawken MJ. 2002. Orientation selectivity in macaque V1: diversity and laminar dependence. *J Neurosci* **22**:5639–51.

Rossi LF, Harris K, Carandini M. 2019. Excitatory and inhibitory intracortical circuits for orientation and direction selectivity. *bioRxiv* 556795.

Rubin J, Ulanovsky N, Nelken I, Tishby N. 2016. The representation of prediction error in auditory cortex. *PLoS Comput Biol* **10**:1–28.

Rumelhart DE, Hinton GE, Williams RJ. 1986. Learning representations by back-propagating errors. *Nature* **323**:533–536.

Rust NC, Mante V, Simoncelli EP, Movshon JA. 2006. How MT cells analyze the motion of visual patterns. *Nat Neurosci* **9**:1421–1431.

Rust NC, Schwartz O, Movshon JA, Simoncelli EP. 2005. Spatiotemporal elements of macaque V1 receptive fields. *Neuron* **46**:945–956.

Sachs MB, Abbas PJ. 1974. Rate versus level functions for auditory-nerve fibers in cats: tone-burst stimuli. *J Acoust Soc Am* **56**:1835–47.

Sahani M, Linden JFJF. 2003. How linear are auditory cortical responses? *Adv Neural Inf Process Syst* **15**:109–116.

Salisbury JM, Palmer SE. 2016. Optimal prediction in the retina and natural motion statistics. *J Stat Phys* **162**:1309–1323.

Schmolesky MT, Wang Y, Hanes DP, Thompson KG, Leutgeb S, Schall JD, Leventhal AG. 1998. Signal timing across the macaque visual system. *J Neurophysiol* **79**:3272–3278.

Schnupp JWH, Mrsic-Flogel TD, King AJ. 2001. Linear processing of spatial cues in primary auditory cortex. *Nature* **414**:200–204.

Scholl B, Tan AYY, Corey J, Priebe NJ. 2013. Emergence of orientation selectivity in the mammalian visual pathway. *J Neurosci* **33**:10616–10624.

Schwartz O, Pillow JW, Rust NC, Simoncelli EP. 2006. Spike-triggered neural characterization. *J Vis* **6**:13.

Shapley R, Perry H V. 1986. Cat and monkey retinal ganglion cells and their visual functional roles. *Trends Neurosci* **9**:229–235.

Sharpee TO, Atencio CA, Schreiner CE. 2011. Hierarchical representations in the auditory cortex. *Curr Opin Neurobiol* **21**:761–767.

Simoncelli E, Pillow JW, Paninski L, Schwartz O. 2004. Characterization of neural responses with stochastic stimuli In: Gazzaniga M, editor. The Cognitive Neurosciences, III. MIT Press. pp. 327–338.

Simoncelli EP, Heeger DJ. 1998. A model of neuronal responses in visual area MT. *Vision*

Res **38**:743–761.

Simoncelli EP, Olshausen BA. 2001. Natural image statistics and neural representation. *Annu Rev Neurosci* **24**:1193–1216.

Singer Y, Teramoto Y, Willmore BDB, King AJ, Schnupp JWH, Harper NS. 2018. Sensory cortex is optimized for prediction of future input. *Elife* **7**:e31557.

Singer Y, Willmore BDB, King AJ, Harper NS. 2019. Hierarchical temporal prediction captures motion processing from retina to higher visual cortex. *bioRxiv* 575464.

Skottun BC, De Valois RL, Grosof DH, Movshon JA, Albrecht DG, Bonds AB. 1991. Classifying simple and complex cells on the basis of response modulation. *Vision Res* **31**:1079–1086.

Smith MA, Majaj NJ, Movshon JA. 2005. Dynamics of motion signaling by neurons in macaque area MT. *Nat Neurosci* **8**:220–228.

Softky WR. 1996. Unsupervised pixel-prediction. *Adv Neural Inf Process Syst*.

Sohl-Dickstein J, Poole B, Ganguli S. 2014. Fast large-scale optimization by unifying stochastic gradient and quasi-Newton methods. *Proc 31st Int Conf Mach Learn* **32**:604–612.

Srinivasan M V, Laughlin SB, Dubs A. 1982. Predictive coding: a fresh view of inhibition in the retina. *Proc R Soc London Ser B, Biol Sci* **216**:427–59.

Srivastava N, Mansimov E, Salakhutdinov R. 2015. Unsupervised learning of video representations using LSTMs. *arXiv:1502:04681*.

Sutton RS, Barton AG. 1981. An adaptive network that constructs and uses an internal model of its world. *Cogn Brain Theory* **4**:217–246.

Tao L, Shelley M, McLaughlin D, Shapley R. 2004. An egalitarian network model for the emergence of simple and complex cells in visual cortex. *Proc Natl Acad Sci U S A* **101**:366–71.

Tian B, Kuśmierek P, Rauschecker JP. 2013. Analogues of simple and complex cells in rhesus monkey auditory cortex. *Proc Natl Acad Sci U S A* **110**:7892–7.

Tishby N, Pereira FC, Bialek W. 2000. The information bottleneck method. *arXiv:physics/0004057*.

Torralba A, Oliva A. 2003. Statistics of natural image categories. *Comput Neural Syst* **14**:391–412.

Ulanovsky N, Las L, Nelken I. 2003. Processing of low-probability sounds by cortical neurons. *Nat Neurosci* **6**:391–398.

Ungerleider LG, Mishkin M. 1982. Two cortical visual systems In: Ingle DJ, Goodale MA, Mansfield RJW, editors. Analysis of Visual Behavior. Cambridge, MA: MIT Press. pp. 549–586.

Van Essen DC, Gallant JL. 1994. Neural mechanisms of form and motion processing in the primate visual system. *Neuron* **13**:1–10.

van Hateren JH, Ruderman DL. 1998. Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proc R Soc Lond B* **265**:2315–2320.

van Hateren JH, van der Schaaf A. 1998. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc R Soc London B* **265**:359–366.

Vinje WE, Gallant JL. 2000. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science (80-)* **287**:1273–6.

Vondrick C, Pirsiavash H, Torralba A. 2016. Generating videos with scene dynamics. *Adv Neural Inf Process Syst*.

Vondrick C, Torralba A. 2017. Generating the future with adversarial Transformers2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE. pp.

2992–3000.

Wang HX, Movshon JA. 2016. Properties of pattern and component direction-selective cells in area MT of the macaque. *J Neurophysiol* **115**:2705–2720.

Whittington JCR, Bogacz R. 2019. Theories of error back-propagation in the brain. *Trends Cogn Sci*.

Whittington JCR, Bogacz R. 2017. An approximation of the error backpropagation algorithm in a predictive coding network with local Hebbian synaptic plasticity. *Neural Comput* **29**:1229–1262.

Willmore BDB, Schoppe O, King AJ, Schnupp JWH, Harper NS. 2016. Incorporating midbrain adaptation to mean sound level improves models of auditory cortical processing. *J Neurosci* **36**:280–9.

Willmore BDB, Tolhurst DJ. 2001. Characterizing the sparseness of neural codes. *Netw Comput Neural Syst* **12**:255–70.

Wiskott L, Sejnowski TJ. 2002. Slow feature analysis: unsupervised learning of invariances. *Neural Comput* **14**:715–770.

Yamins DLK, DiCarlo JJ. 2016. Using goal-driven deep learning models to understand sensory cortex. *Nat Neurosci* **19**:356–365.

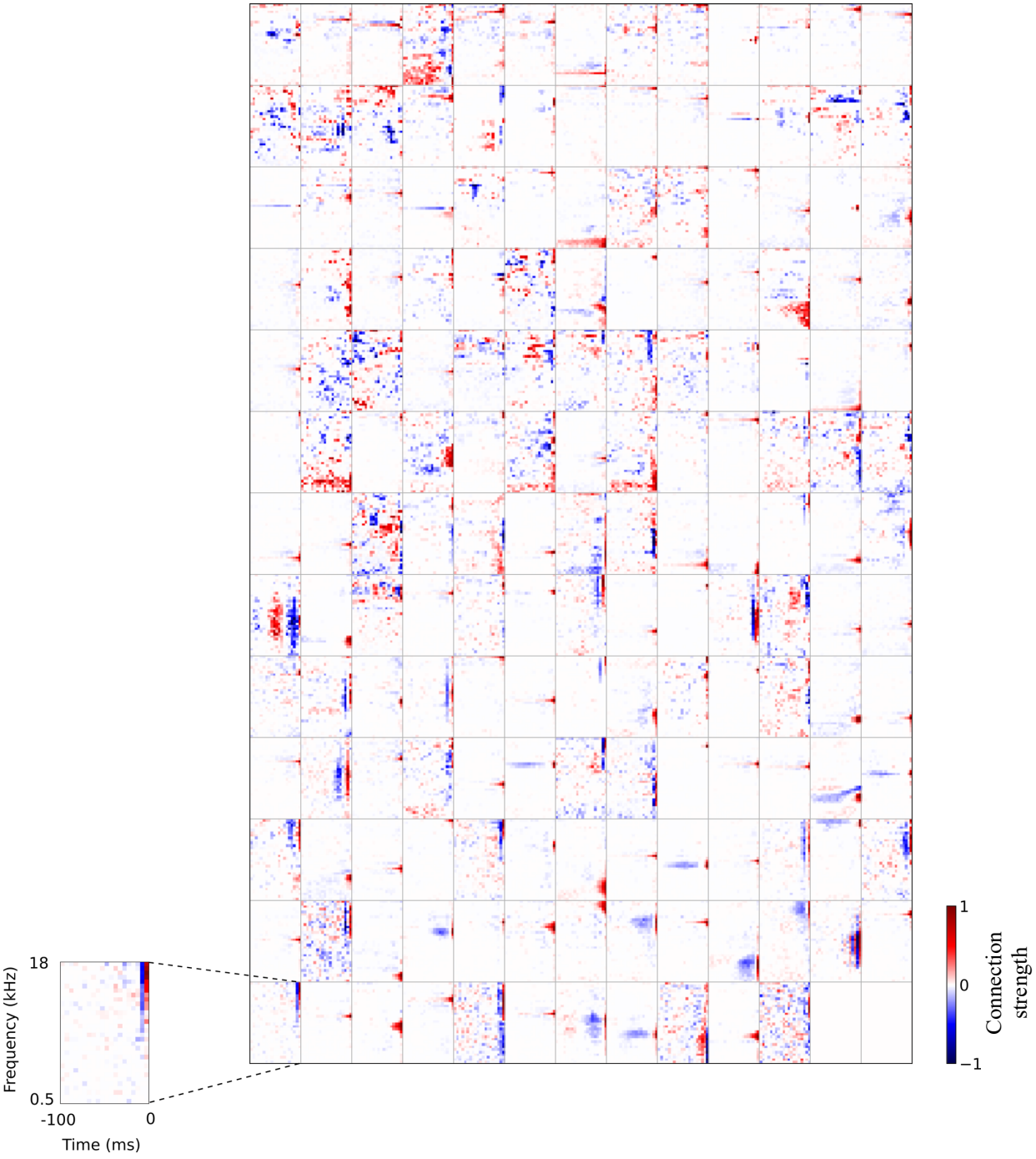
Yeomans JS, Frankland PW. 1996. The acoustic startle reflex: neurons and connections. *Brain Res Rev* **21**:301–314.

Zeki SM. 1974. Functional organization of a visual area in the posterior bank of the superior temporal sulcus of the rhesus monkey. *J Physiol* **236**:549–573.

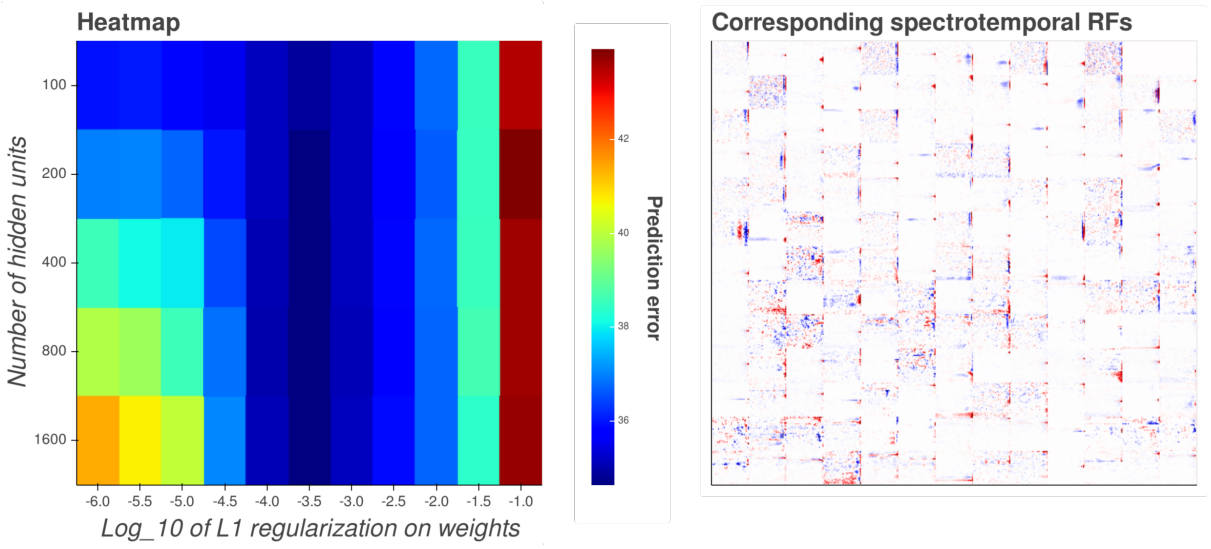
Zhao J, Mathieu M, LeCun Y. 2016. Energy-based generative adversarial network. *arXiv:160903126*.

Zhao L, Zhaoping L. 2011. Understanding auditory spectro-temporal receptive fields and their changes with input statistics by efficient coding principles. *PLoS Comput Biol* **7**.

Supplementary figures



Supplementary Figure 2.1 | Full set of auditory RFs of the temporal prediction model units shown on a finer timescale. All details are as in Fig. 2.2, but only the most recent 100ms of the response profile is shown in order to illustrate details of the RFs. Inset shows axes.



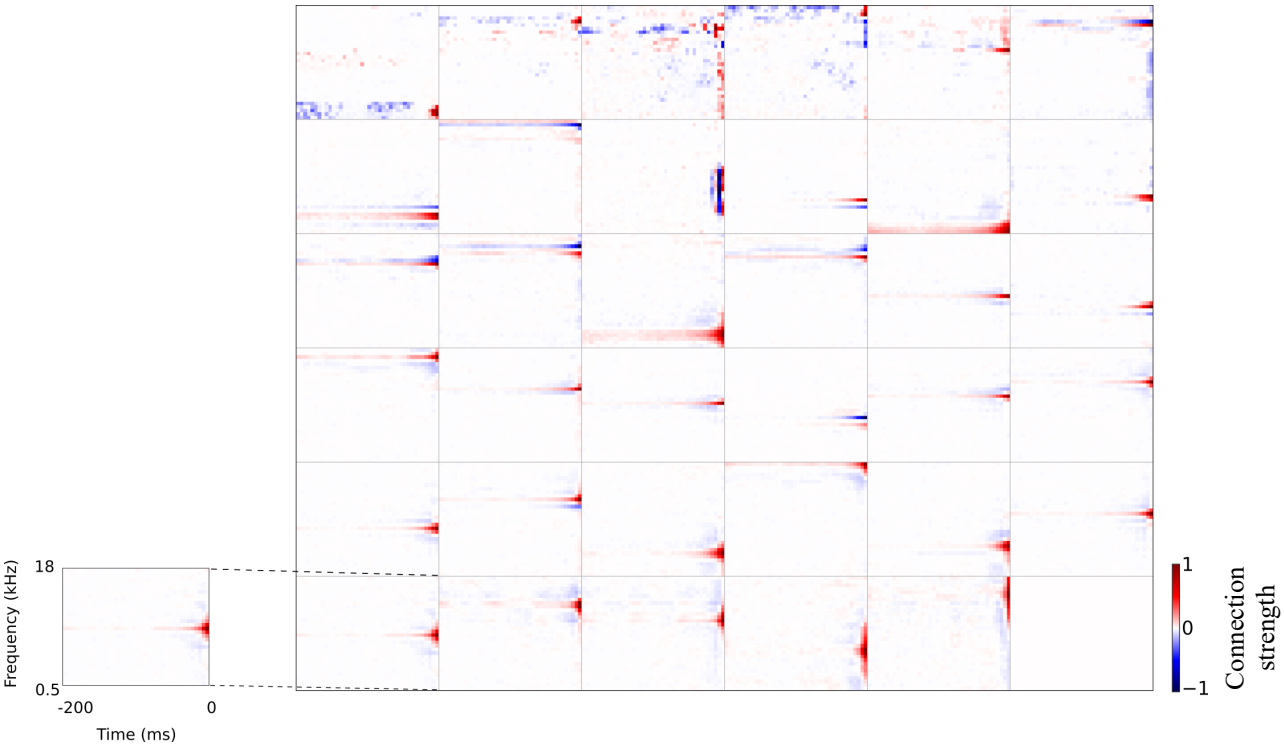
Supplementary Figure 2.2 | Interactive figure exploring the relationship between the strength of L_1 regularization on the network weights and the structure of the RFs the network produces when the network is trained on auditory inputs.

The interactive version of this figure can be found at: https://yossing.github.io/temporal_prediction_model/figures/interactive_supplementary_figures.html.

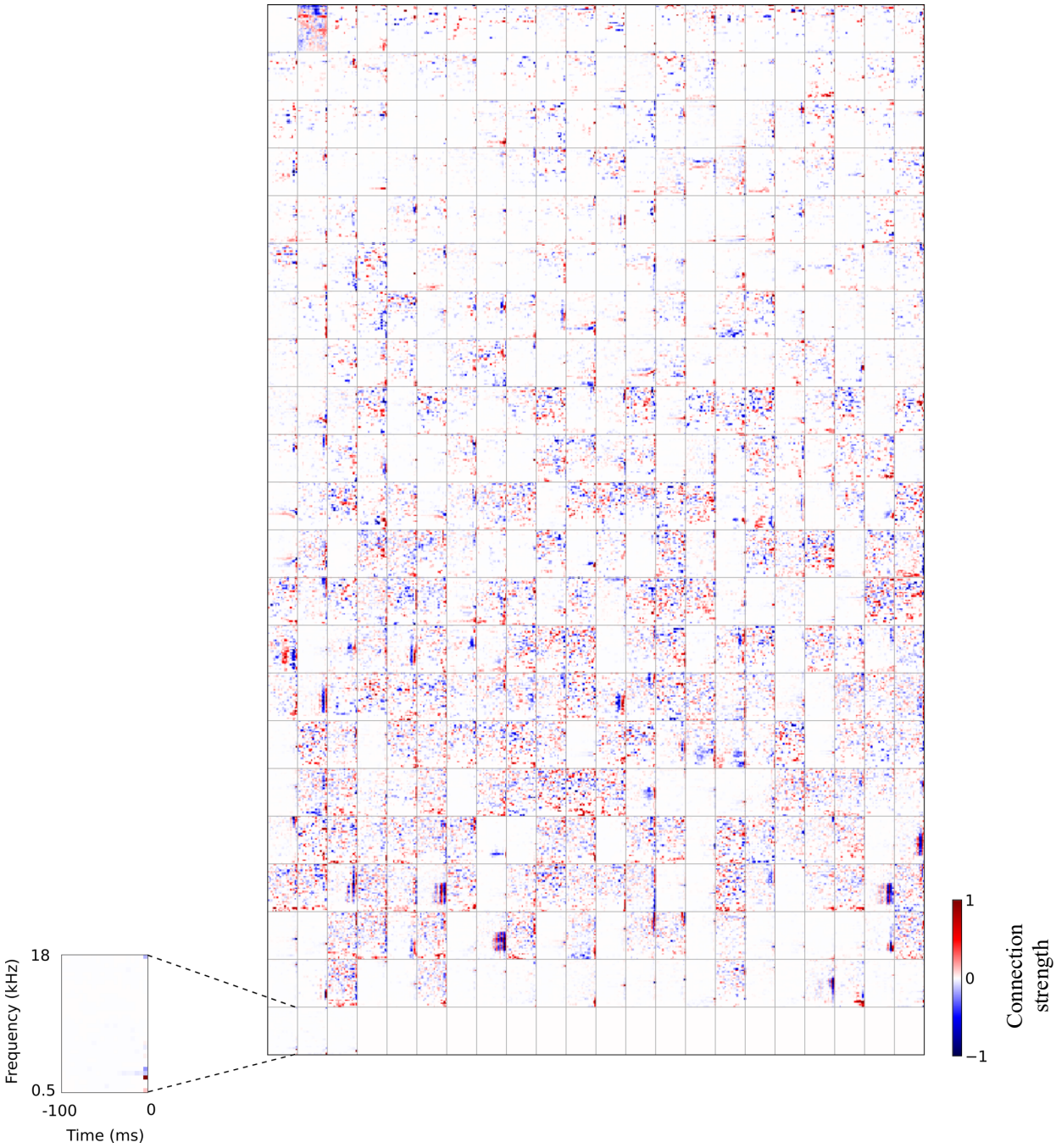
The left hand panel shows the performance of the network with the hyperparameter settings specified on the x and y axes. The x axis signifies the strength of L_1 regularization placed on the weights of the network during training. The y axis signifies the number of hidden units in the network. The colour represents the performance of model measured by the reconstruction error on held out validation set at the end of training.

How to interact with the figure:

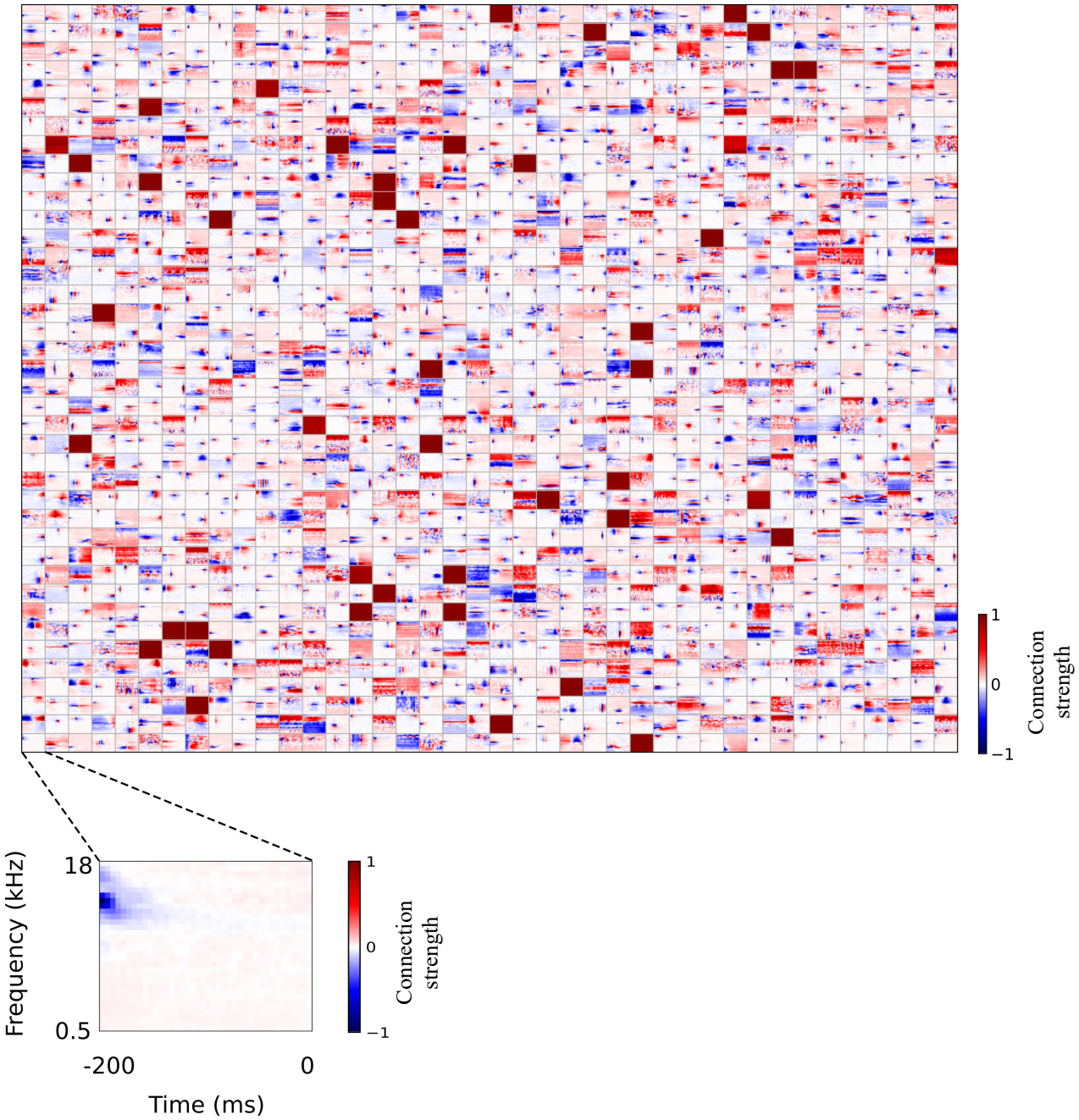
Hover over a point in the left hand panel to show the corresponding spatial receptive fields of the network in the right hand panel. Using the settings near the right hand panel, zoom, pan and reset the image to explore the shapes of the spectrotemporal receptive fields. Many hidden units' weight matrices decayed to near zero during training. Inactive units were excluded from analysis and are not shown.



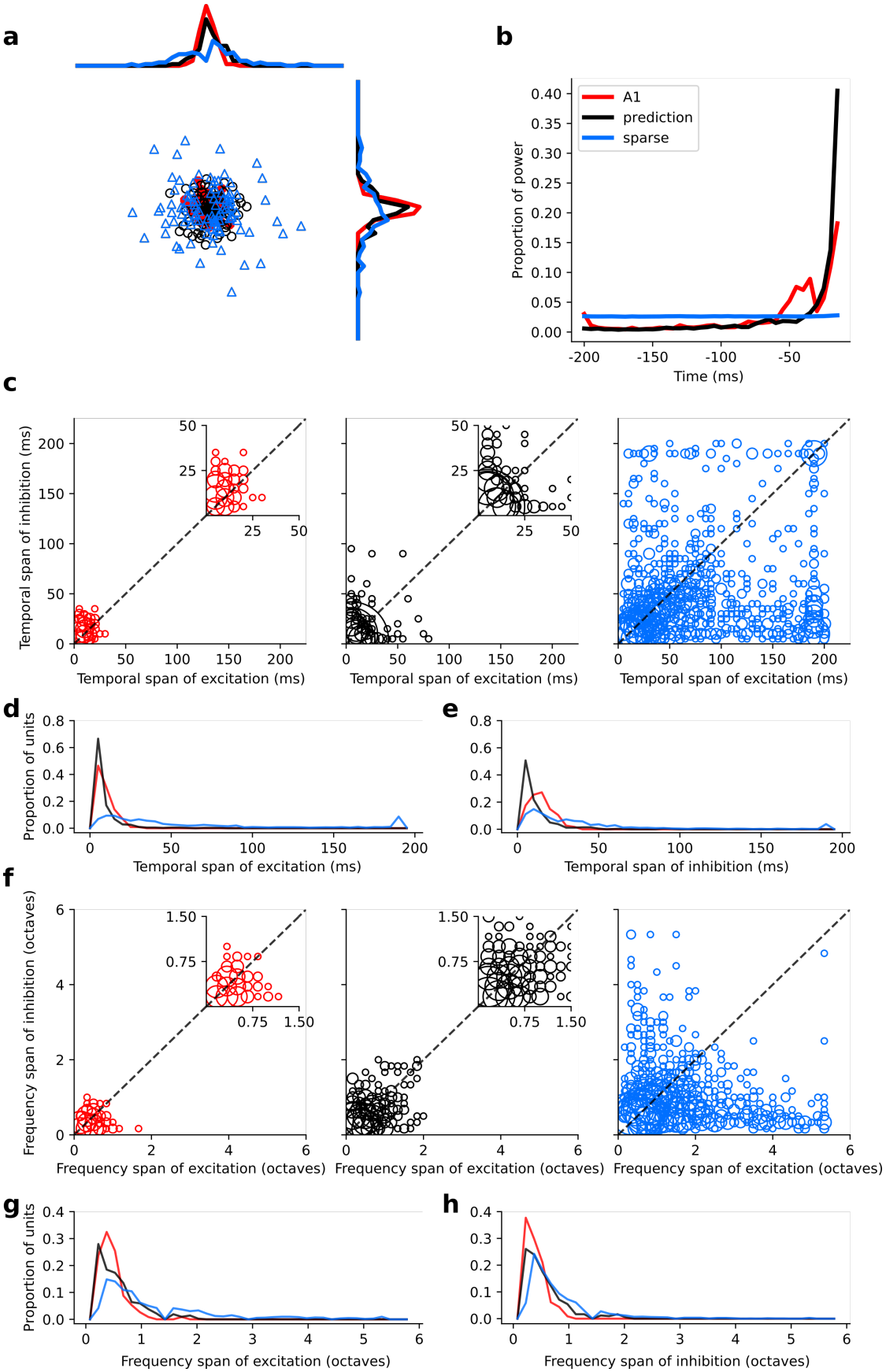
Supplementary Figure 2.3 | Full set of auditory RFs of the temporal prediction model units using a linear activation function. Units were obtained by training the model with 1600 hidden units on auditory inputs. The hidden unit number and L_1weight regularization strength ($10^{-2.25}$) was chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Almost all hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving 35 active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Inset shows axes.



Supplementary Figure 2.4 | Full set of auditory RFs of the temporal prediction model units trained on auditory inputs without added noise. Units were obtained by training the model with 1600 hidden units on auditory inputs. The hidden unit number and L_1weight regularization strength (10^{-4}) was chosen because it results in the lowest MSE on the prediction task, as measured using a cross validation set. Many hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving 465 active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Inset shows axes.

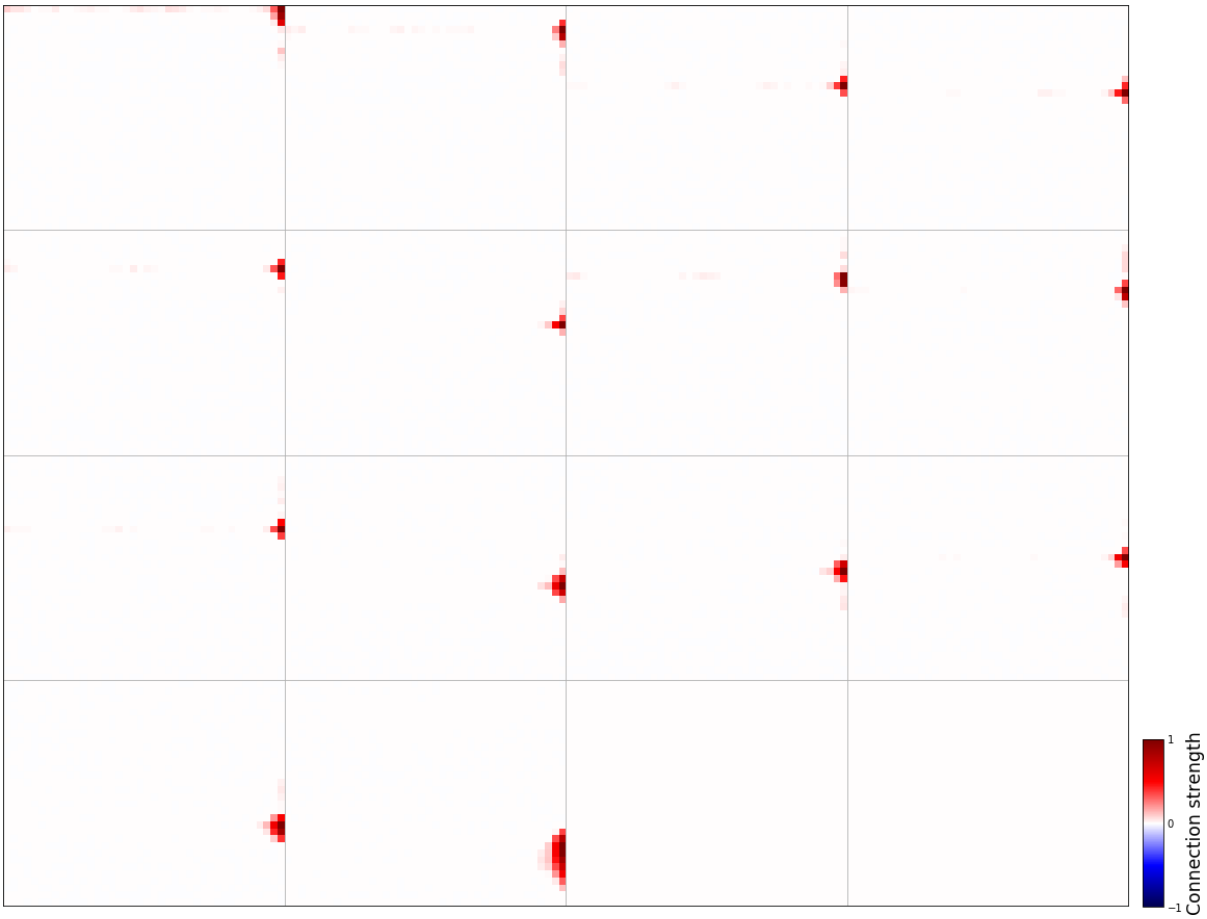


Supplementary Figure 2.5 | Full set of auditory RFs of sparse coding model trained on auditory inputs without added noise. Units were obtained by training the sparse coding model with 1600 units on the identical auditory inputs used to train the network shown in Supp. Fig. 2.4. L_1 regularization of $10^{0.5}$ was applied to the units' activities. This network configuration was selected as it produced unit RFs that most closely resembled those recorded in A1, as determined by visual inspection. Red – excitation, blue - inhibition. Inset shows axes.

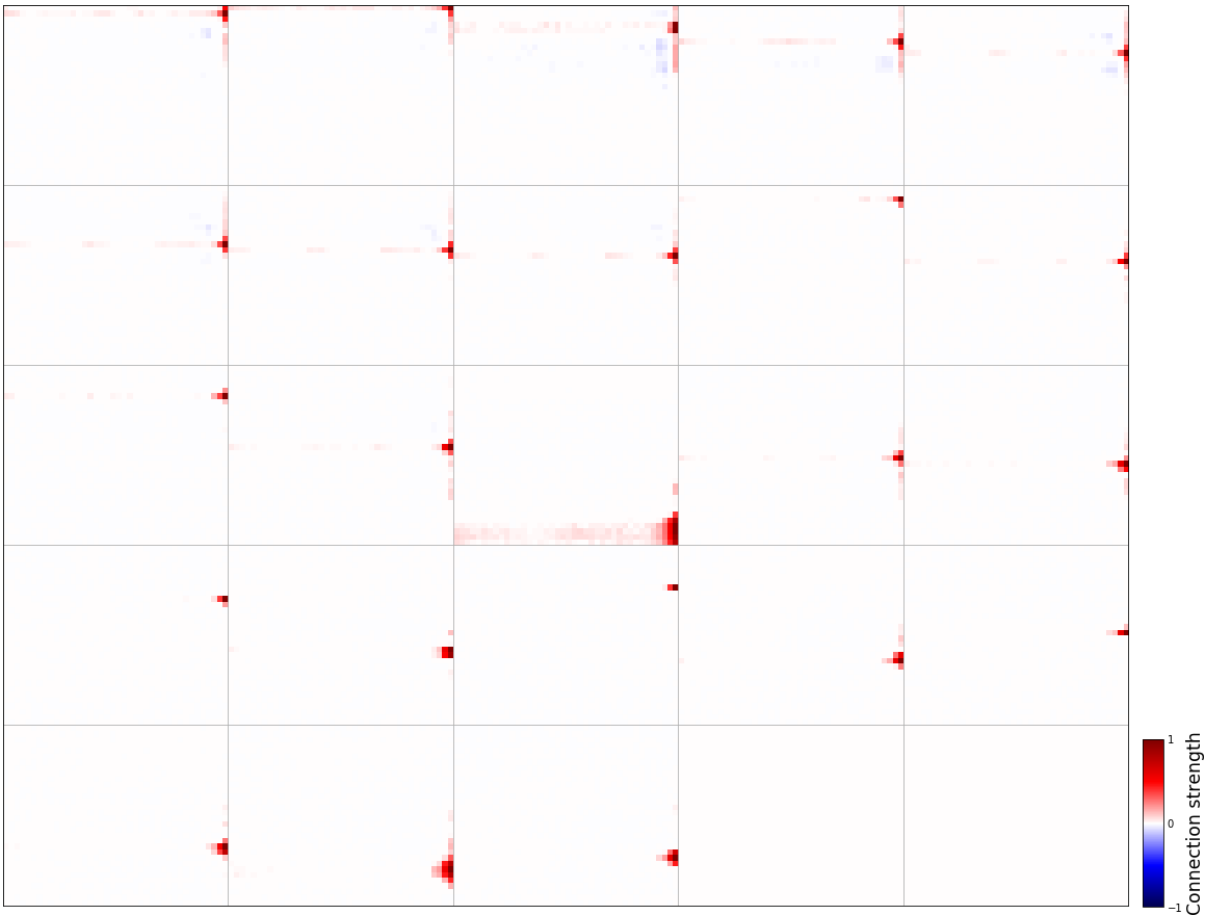


Supplementary Figure 2.6 | Population measures for real A1 spectrotemporal RFs and temporal prediction and sparse coding model auditory RFs when models are trained on

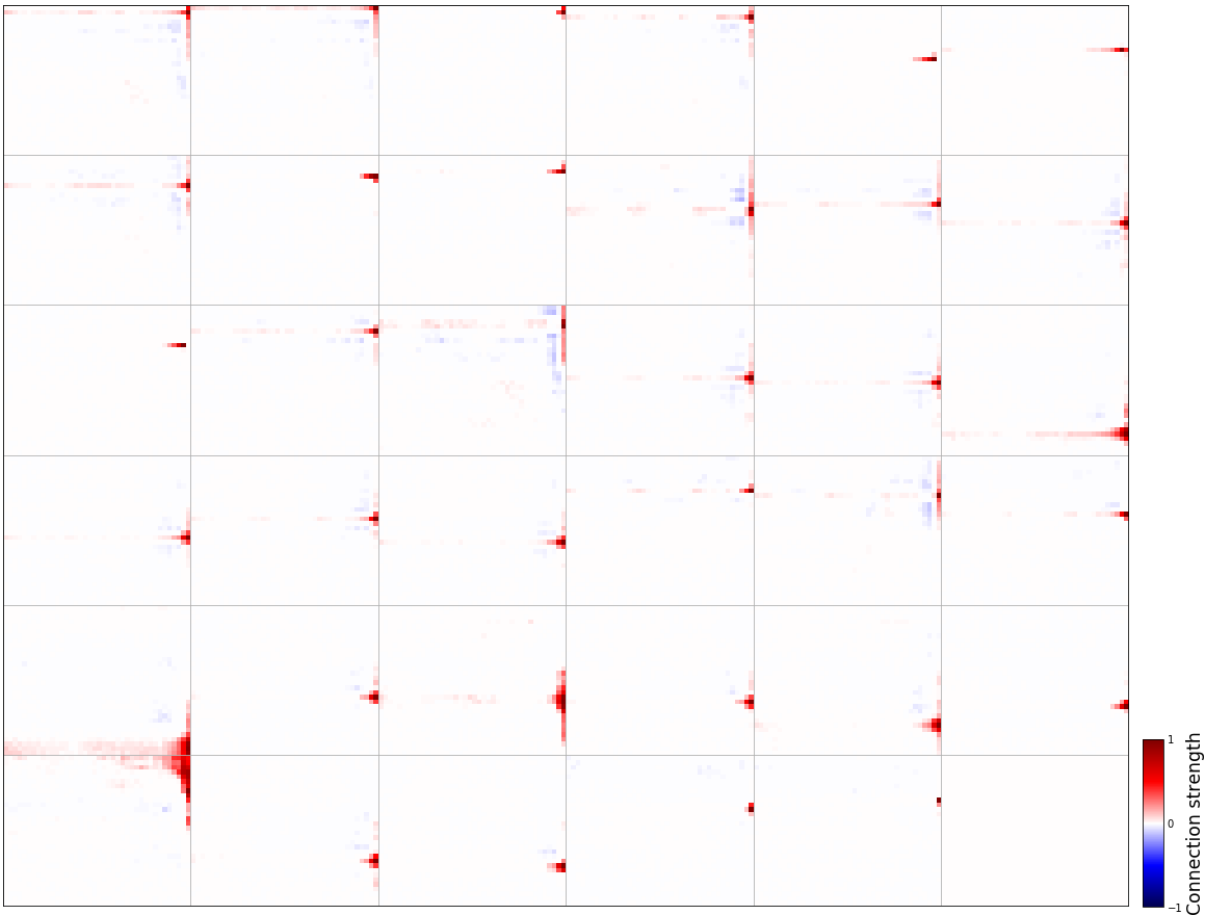
auditory inputs without added noise. Real units are the same as those shown in Fig. 2.2. Temporal prediction model units are the same as those shown in Supp. Fig. 2.4; Sparse coding model units are the same as those shown in Supp. Fig. 2.5. **a,** Each point represents a single RF (with 32 frequency and 38 time steps) which has been embedded in a 2 dimensional space using Multi-Dimensional Scaling (MDS). Red circles - real A1 neurons, black circles – temporal prediction model units, blue triangles – sparse coding model units. Colour scheme applies to all subsequent panels in the figure. **b,** Proportion of power contained in each time step of the RF, taken as an average across the population of units. **c,** Temporal span of excitatory subfields versus that of inhibitory subfields, for real neurons and temporal prediction and sparse coding model units. The area of each circle is proportional to the number of occurrences at that point. **d,** Distribution of temporal spans of excitatory subfields, taken by summing along the x-axis in **c. e,** Distribution of temporal spans of inhibitory subfields, taken by summing along the y-axis in **c. f,** Frequency span of excitatory subfields versus that of inhibitory subfields, for real neurons and temporal prediction and sparse coding model units. **g,** Distribution of frequency spans of excitatory subfields, taken by summing along the x-axis in **f. h,** Distribution of frequency spans of inhibitory subfields, taken by summing along the y-axis in **f.** The addition of noise leads to subtle changes in the RFs. Without noise, the inhibition in the temporal prediction model tends to be slightly less extended and the RFs a little less smooth (see Fig. 2.3, Supp. Fig. 2.4 for comparison).



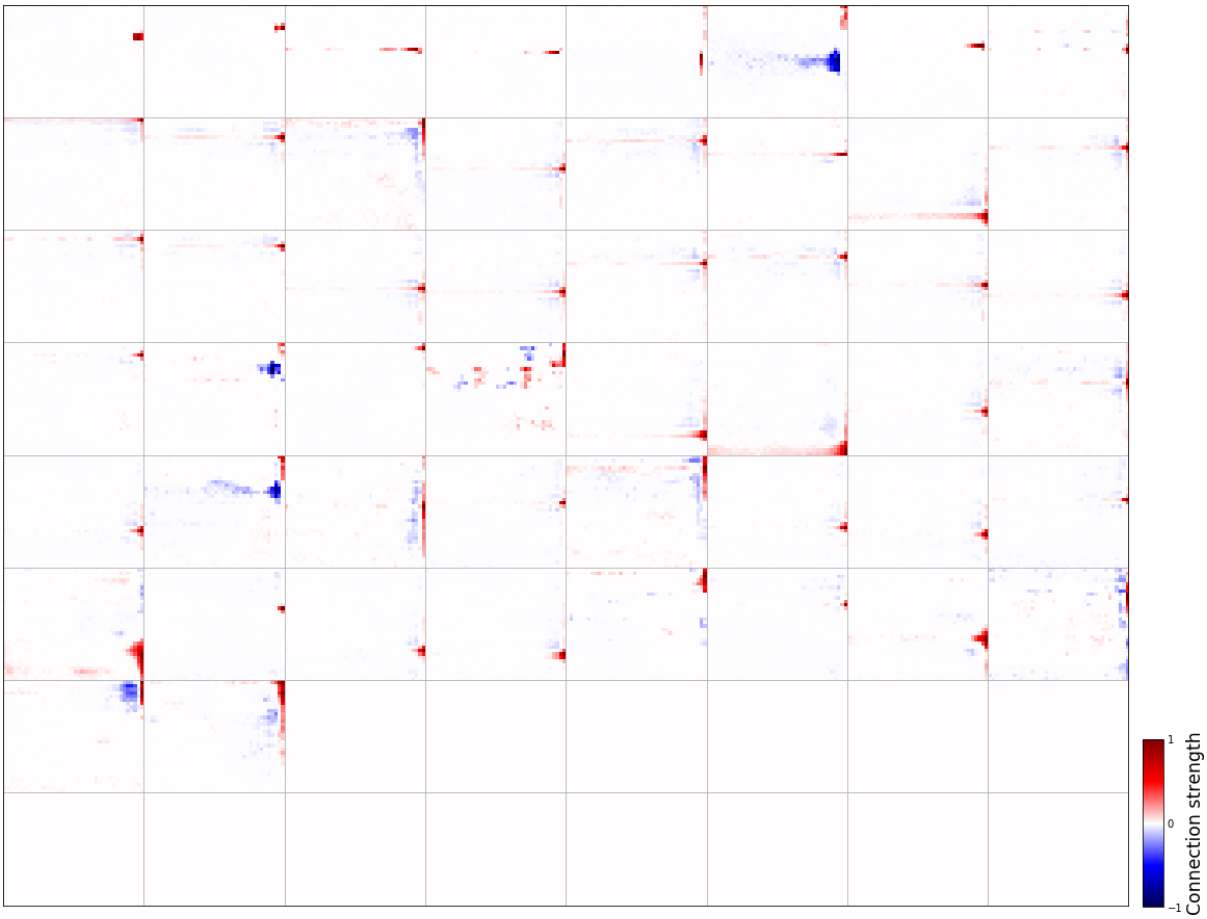
Supplementary Figure 2.7 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength 10^{-1}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



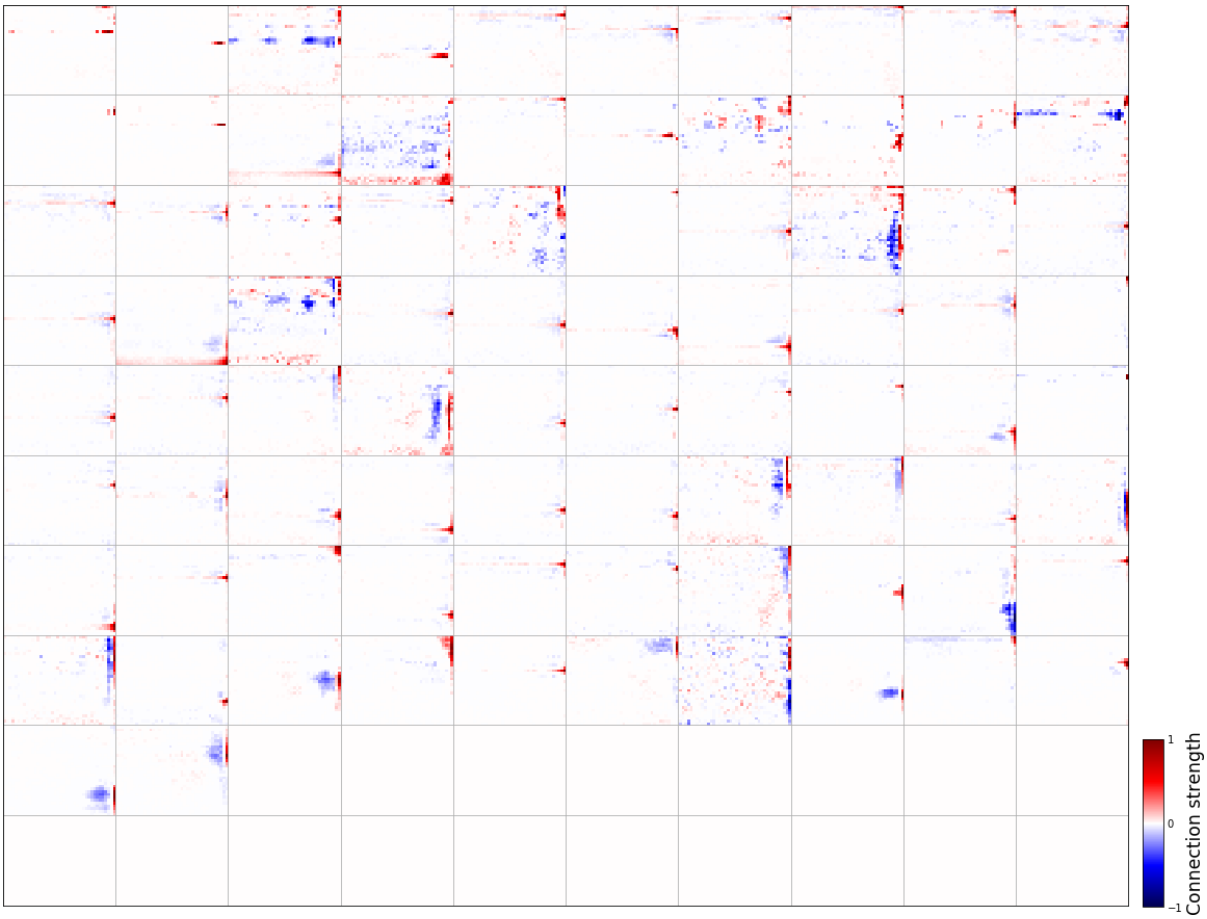
Supplementary Figure 2.8 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength $10^{-1.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



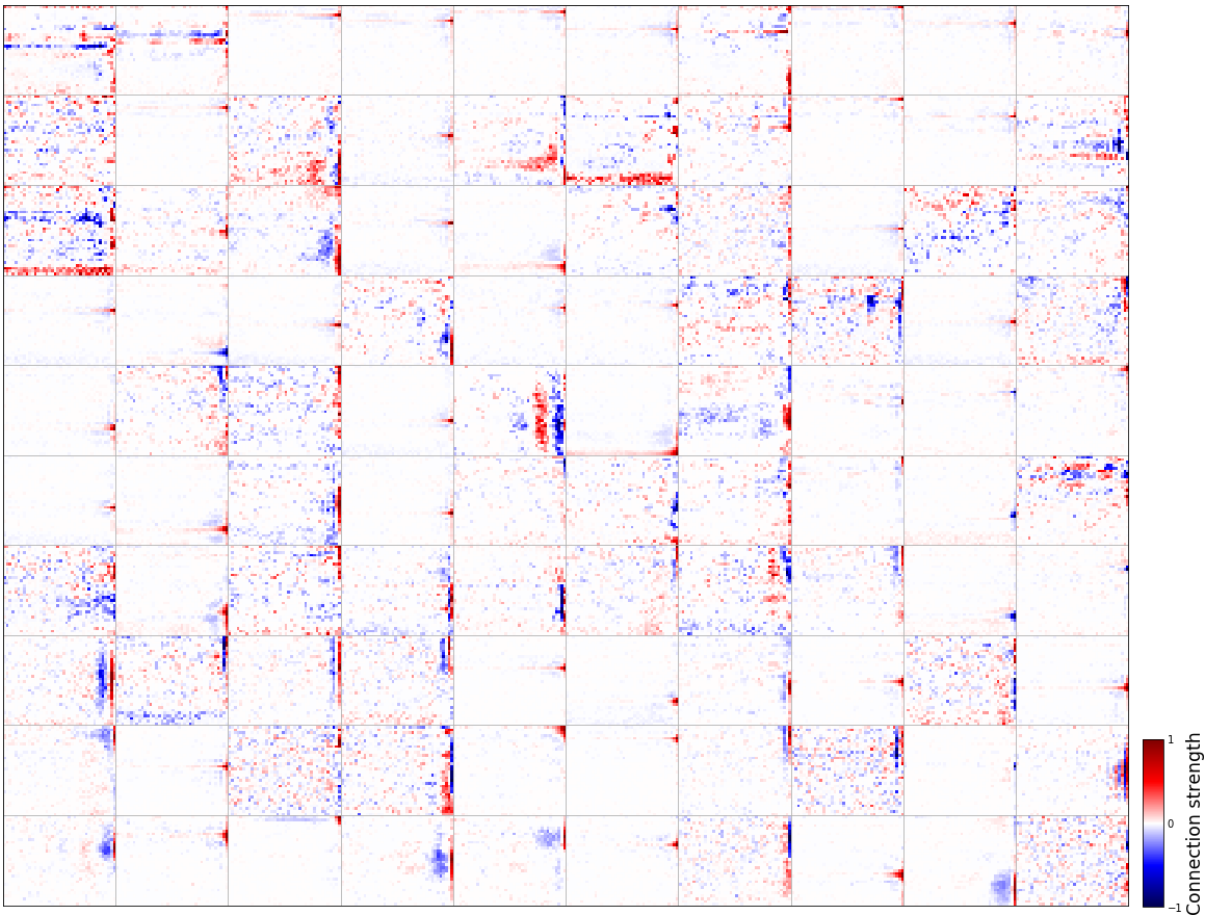
Supplementary Figure 2.9 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength 10^{-2}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



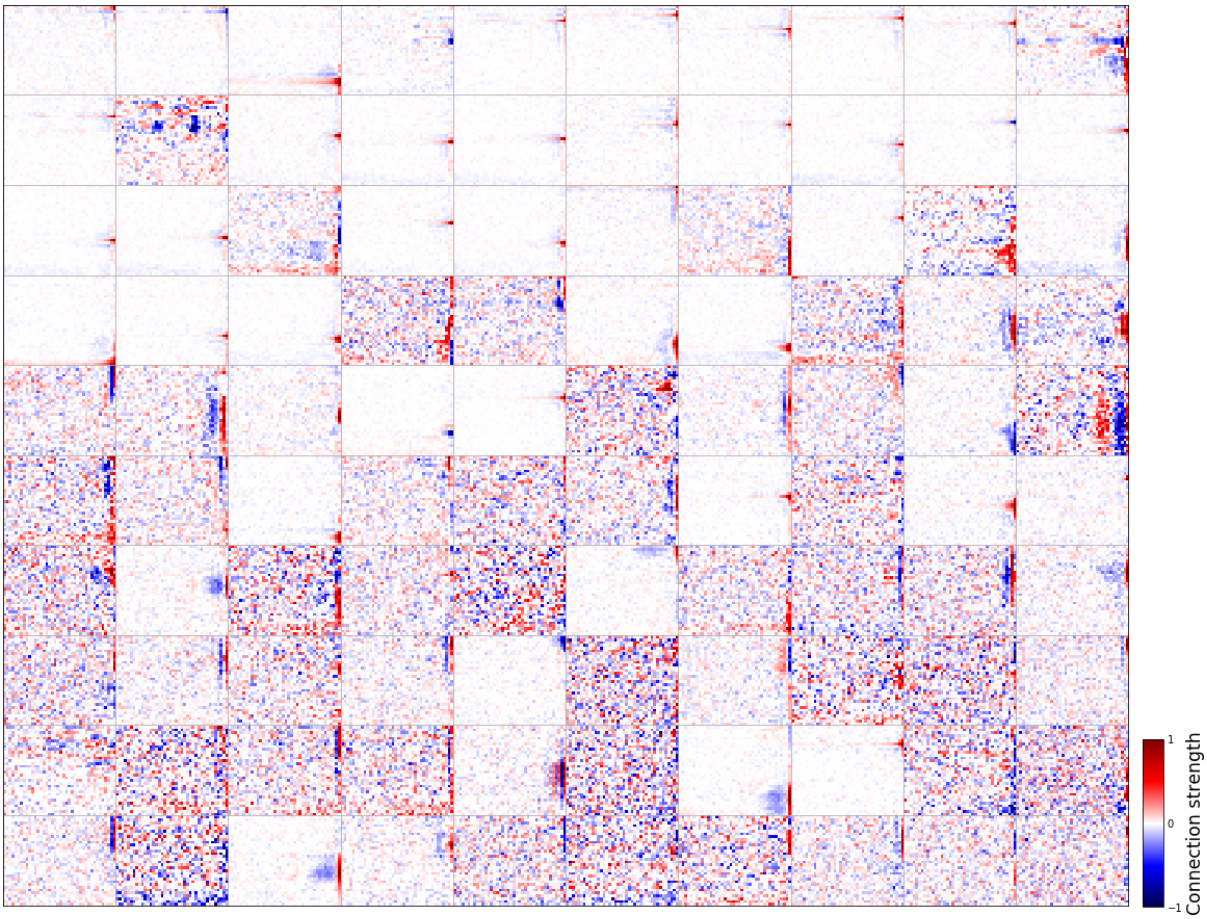
Supplementary Figure 2.10 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength $10^{-2.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



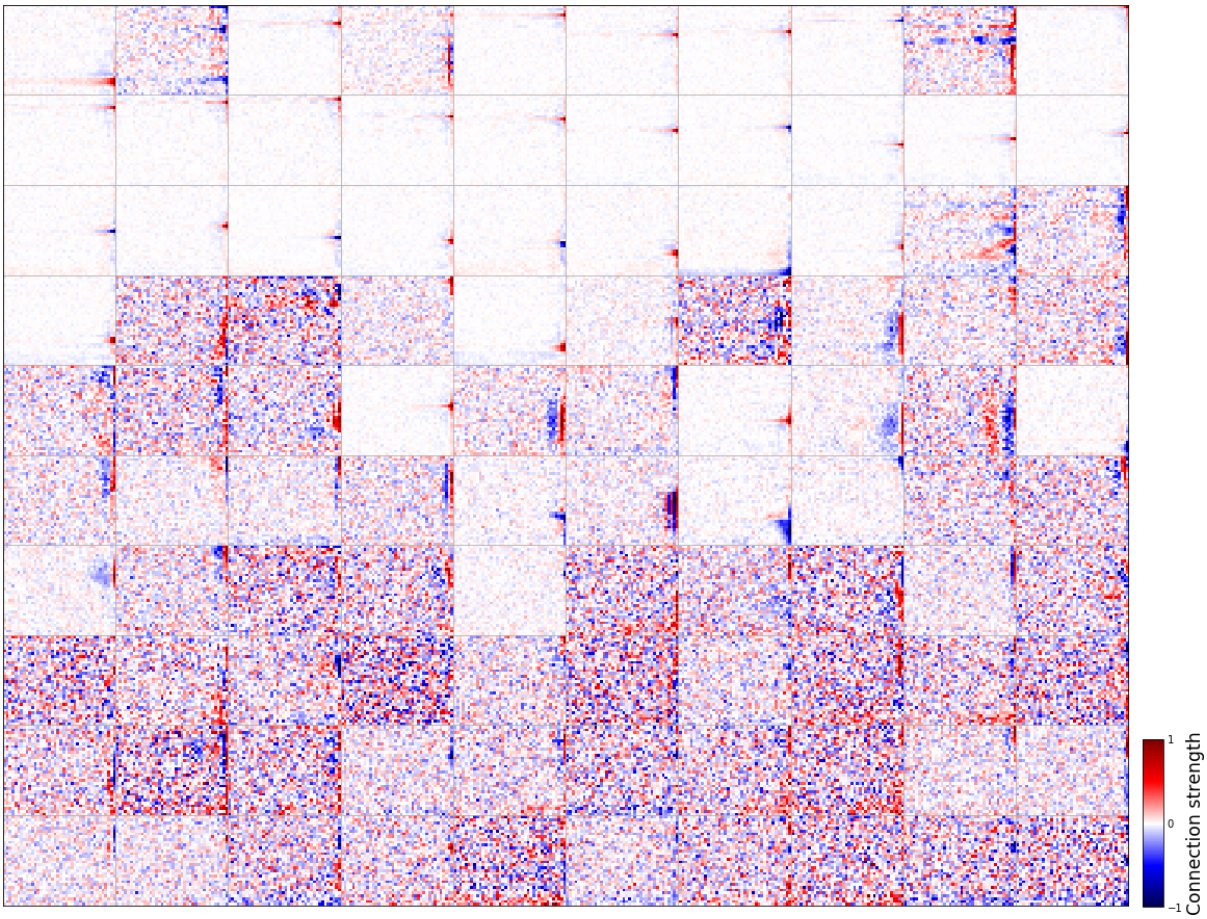
Supplementary Figure 2.11 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength 10^{-3}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



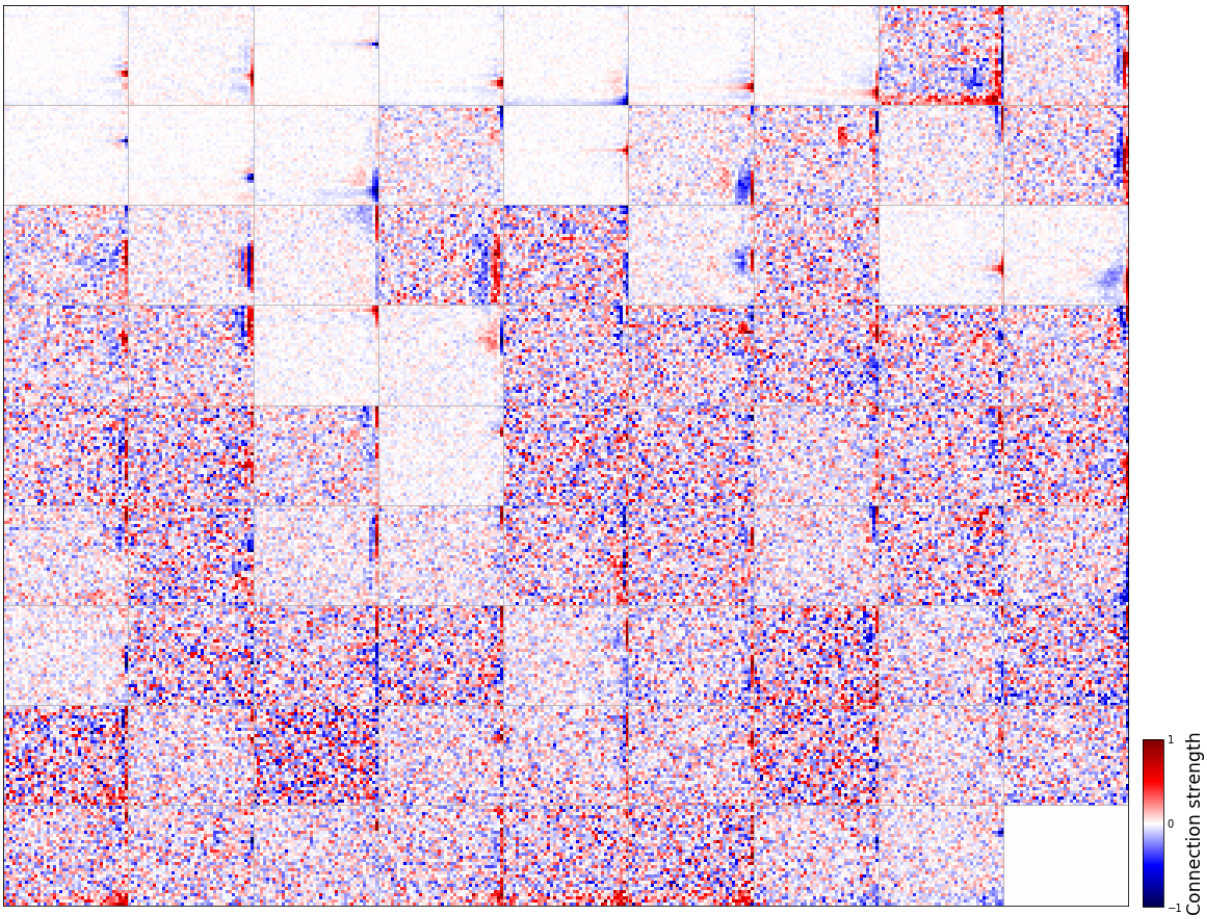
Supplementary Figure 2.12 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength $10^{-3.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



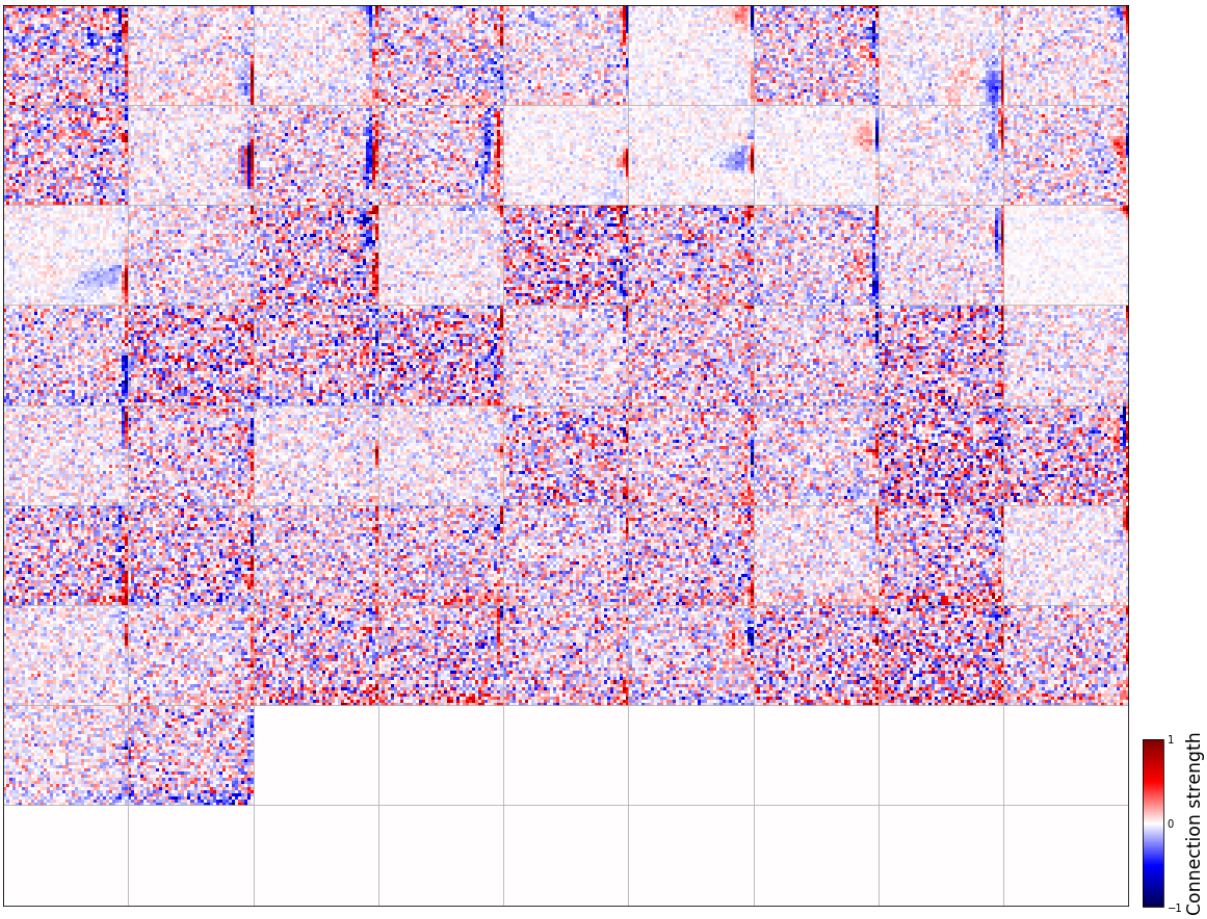
Supplementary Figure 2.13 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength 10^{-4}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



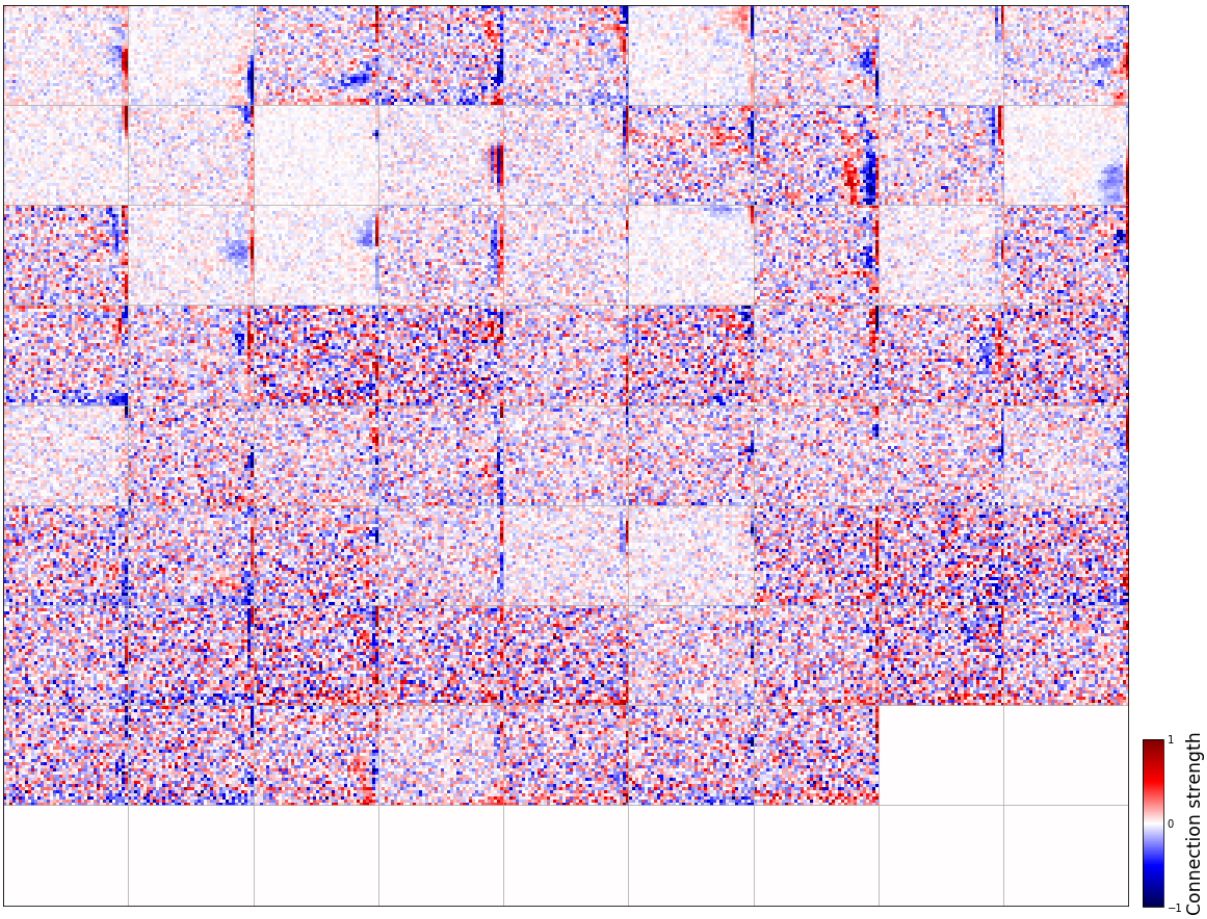
Supplementary Figure 2.14 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength $10^{-4.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



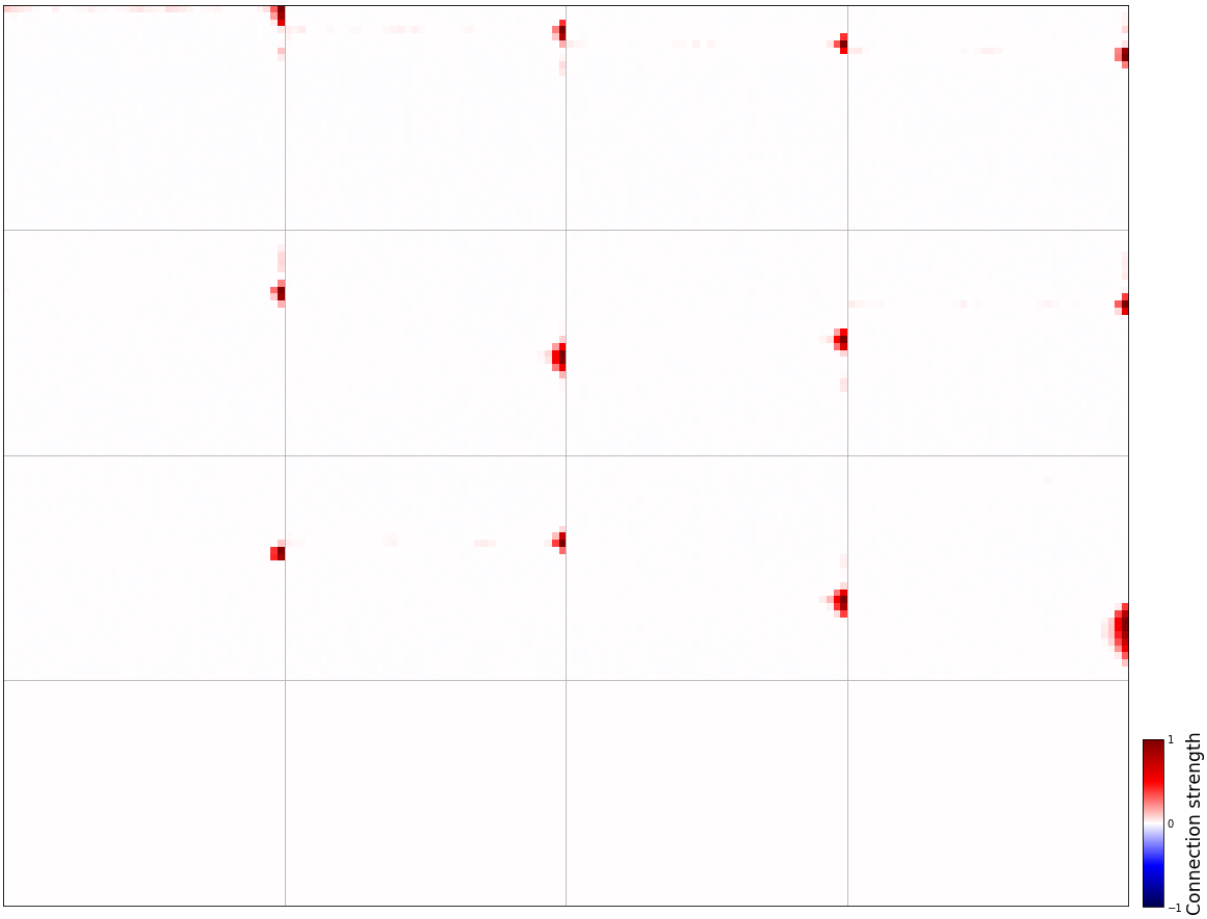
Supplementary Figure 2.15 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength 10^{-5}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



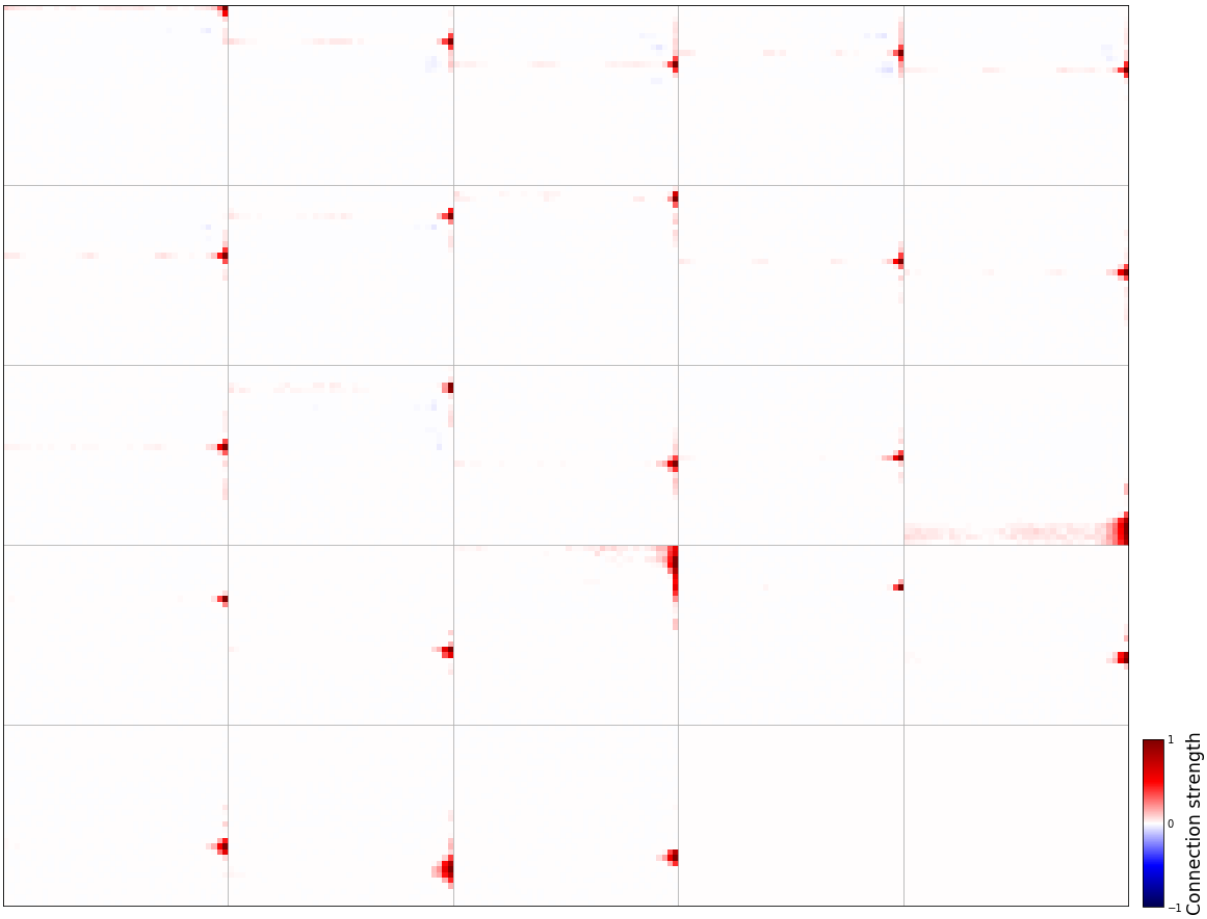
Supplementary Figure 2.16 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength $10^{-5.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



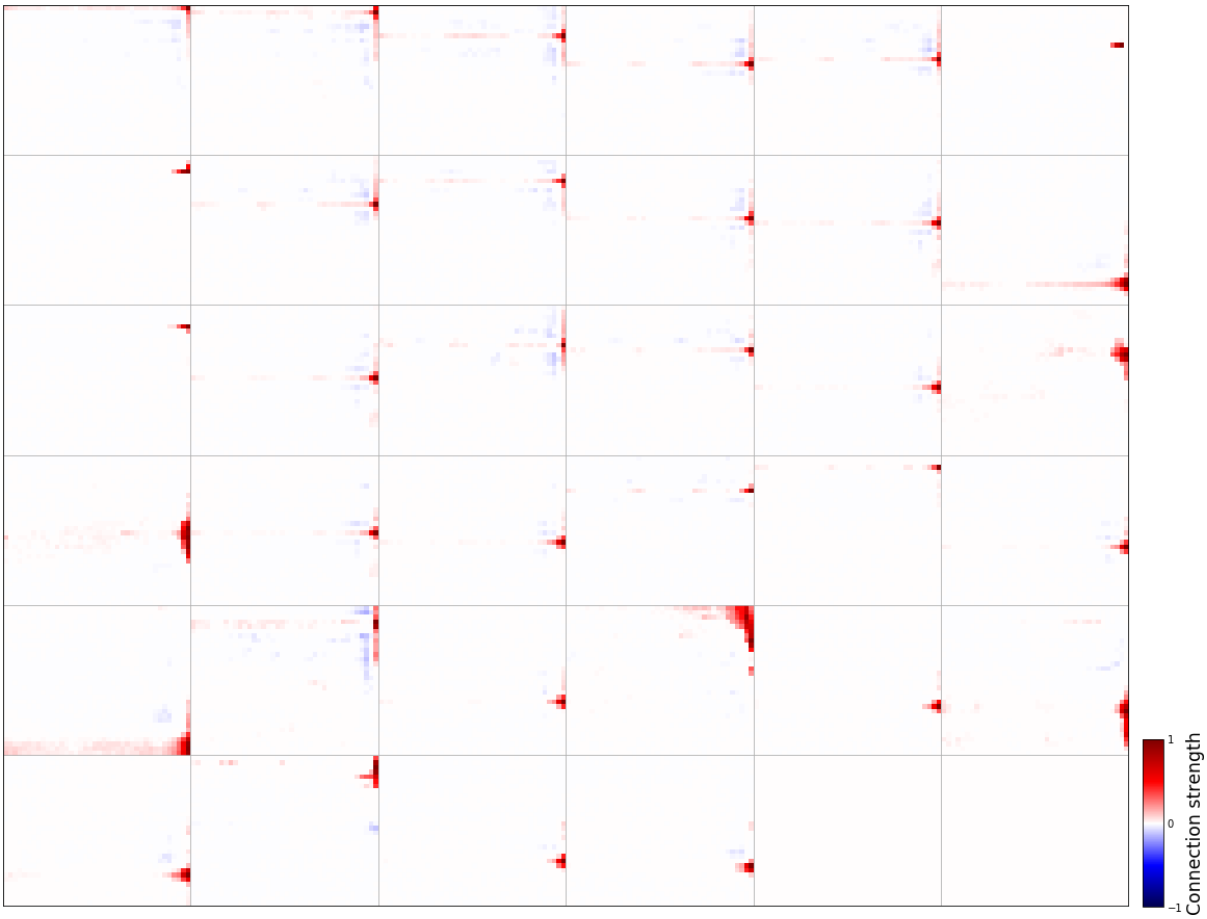
Supplementary Figure 2.17 | Full set of auditory RFs of the temporal prediction model trained with 100 hidden units and regularization strength 10^{-6}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



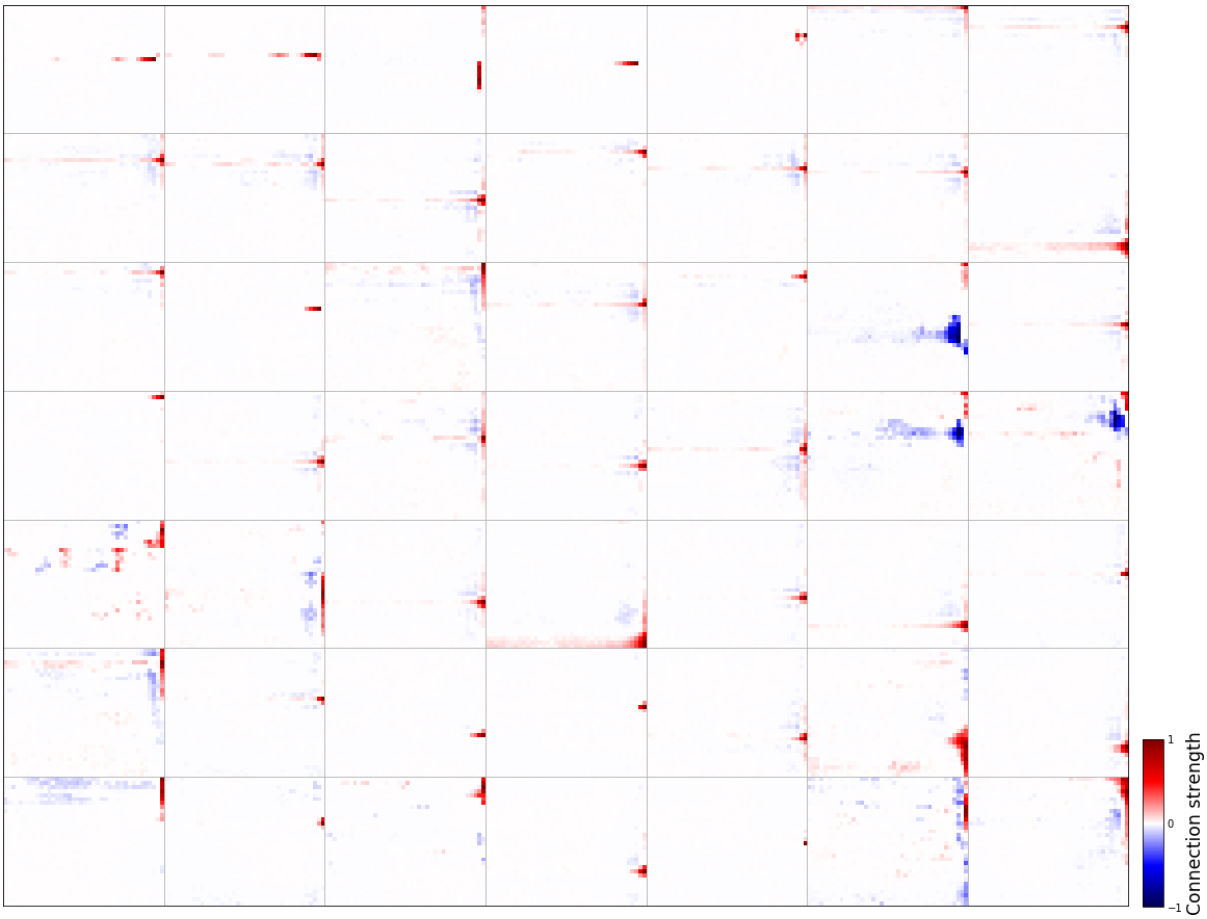
Supplementary Figure 2.18 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength 10^{-1}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



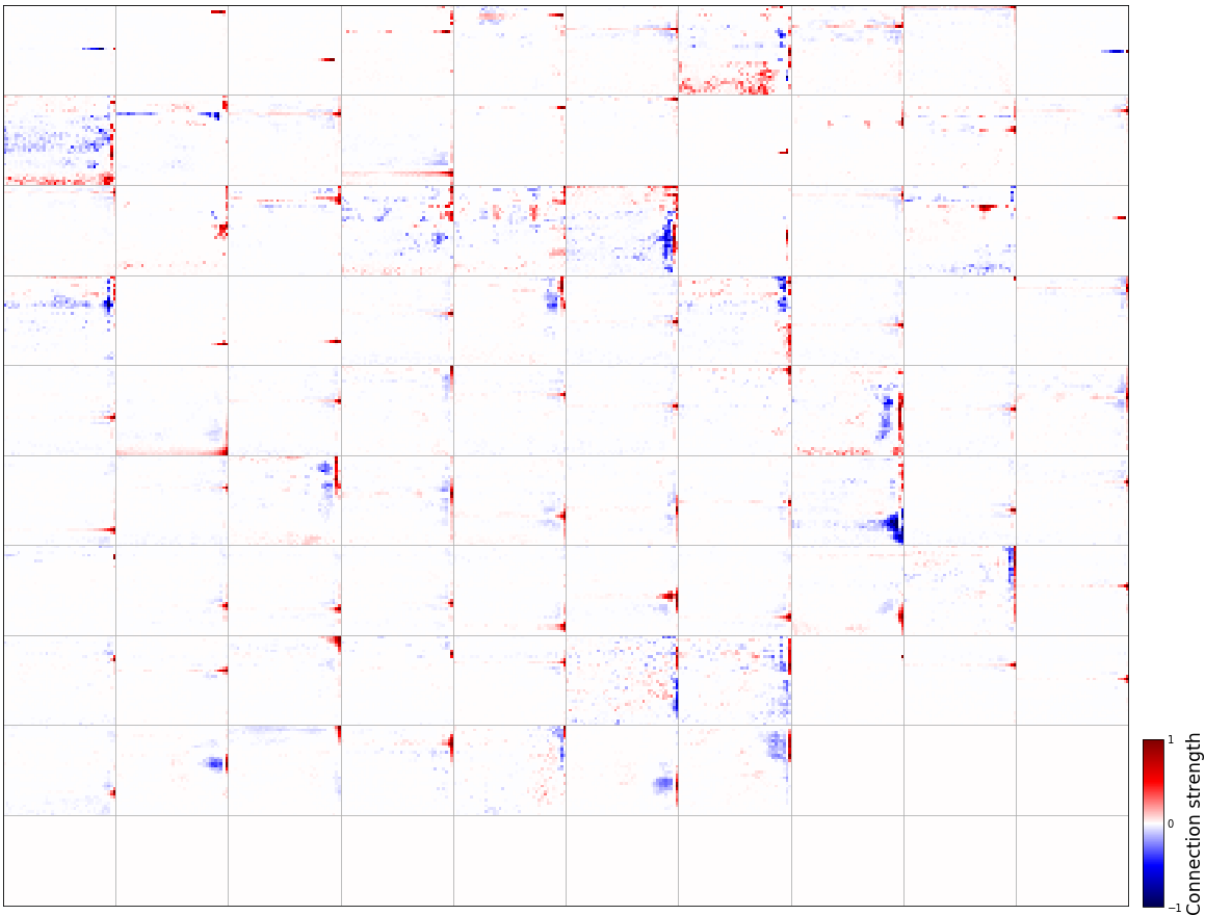
Supplementary Figure 2.19 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength $10^{-1.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



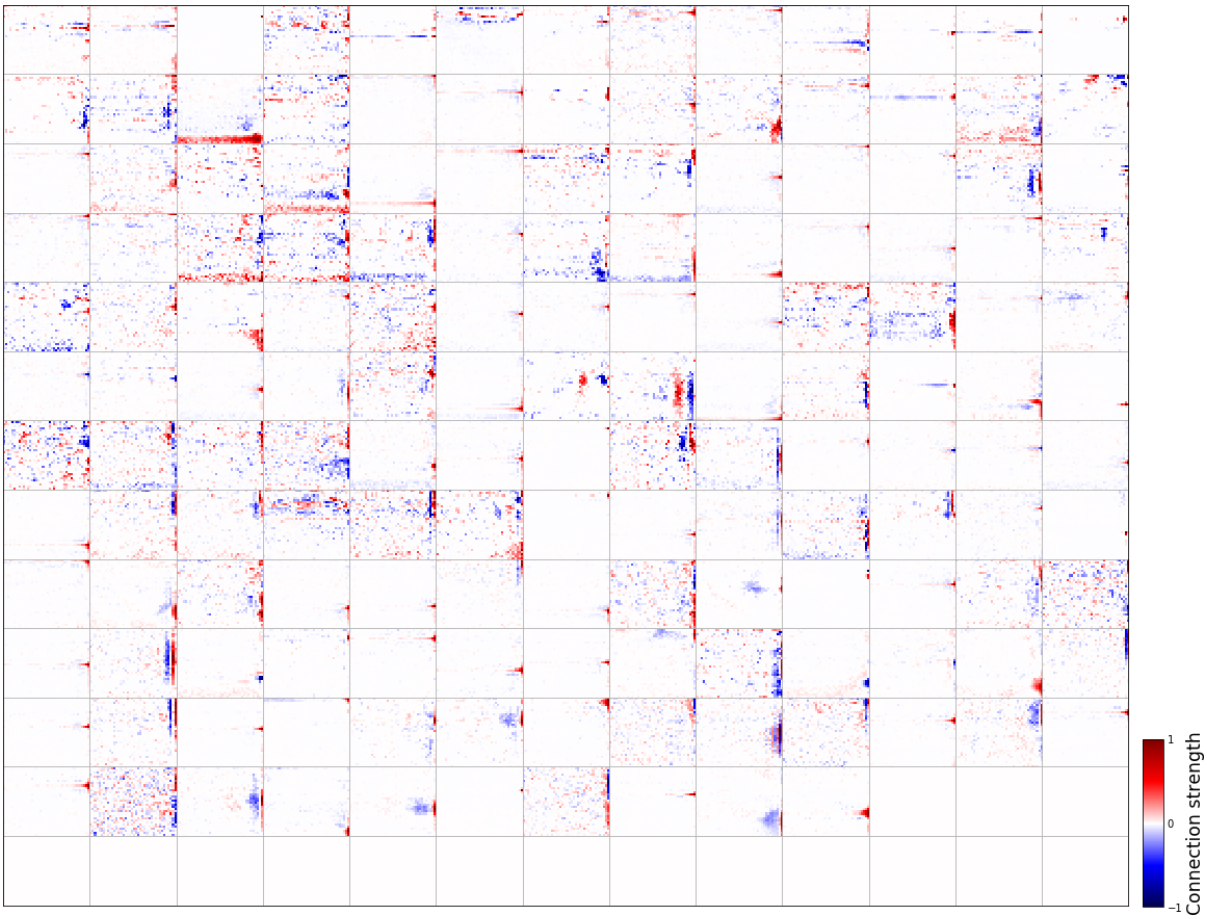
Supplementary Figure 2.20 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength 10^{-2}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



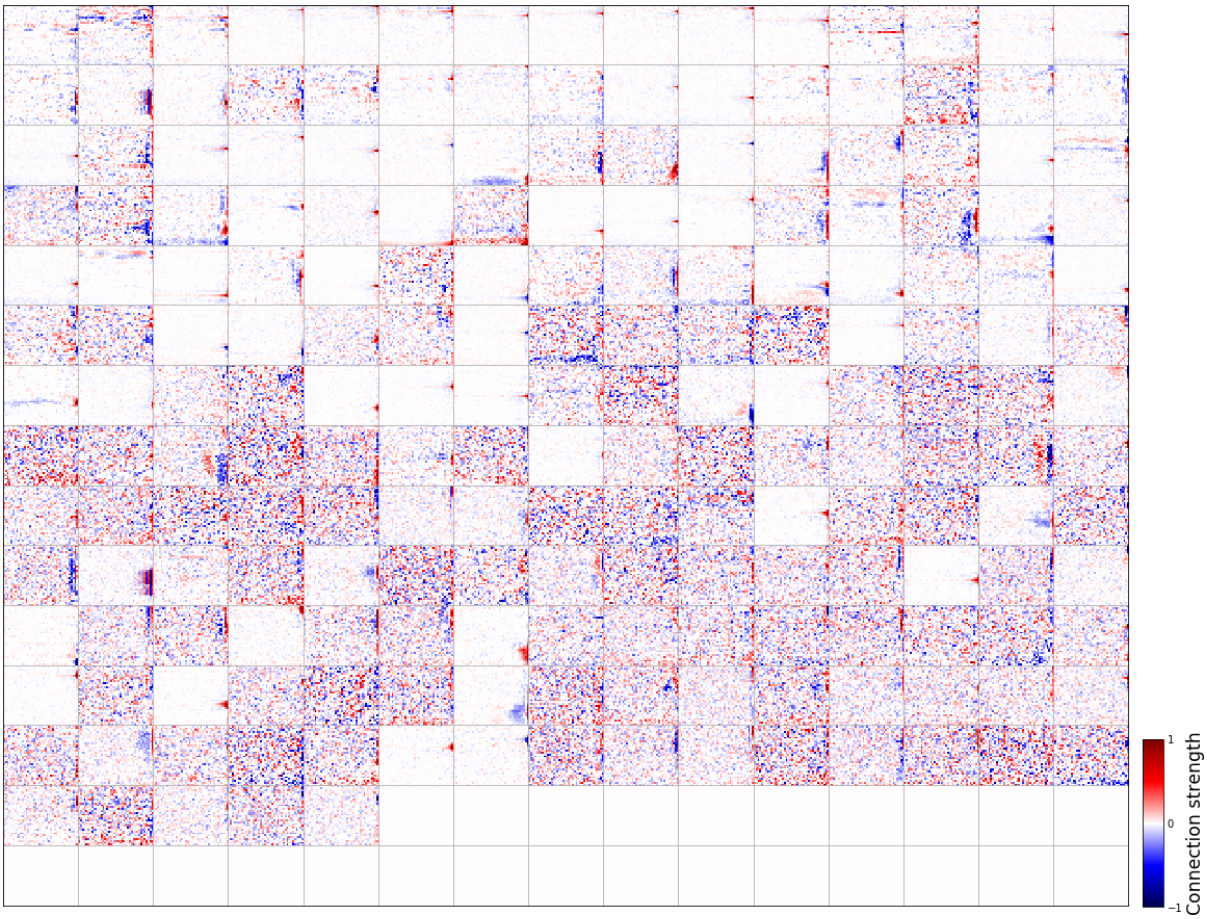
Supplementary Figure 2.21 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength $10^{-2.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



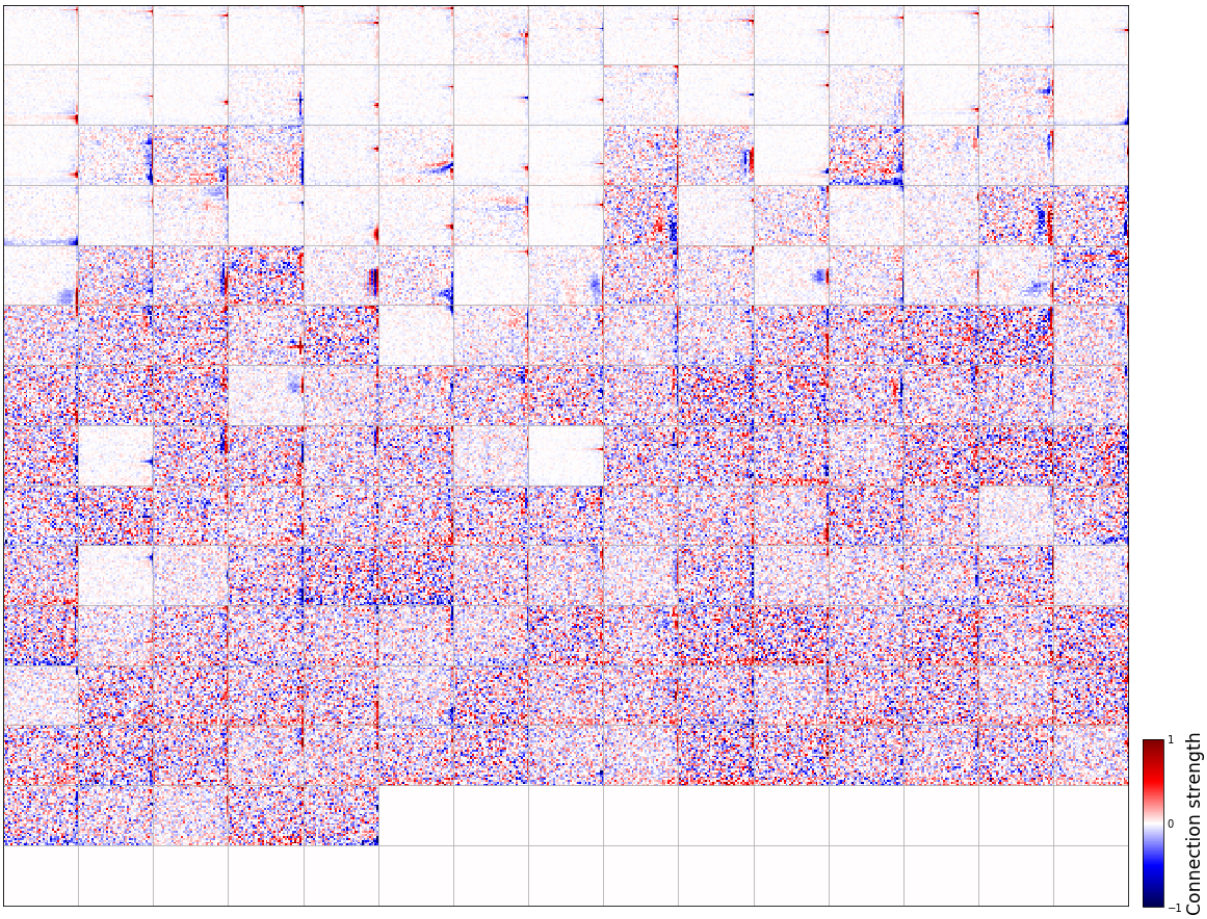
Supplementary Figure 2.22 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength 10^{-3}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



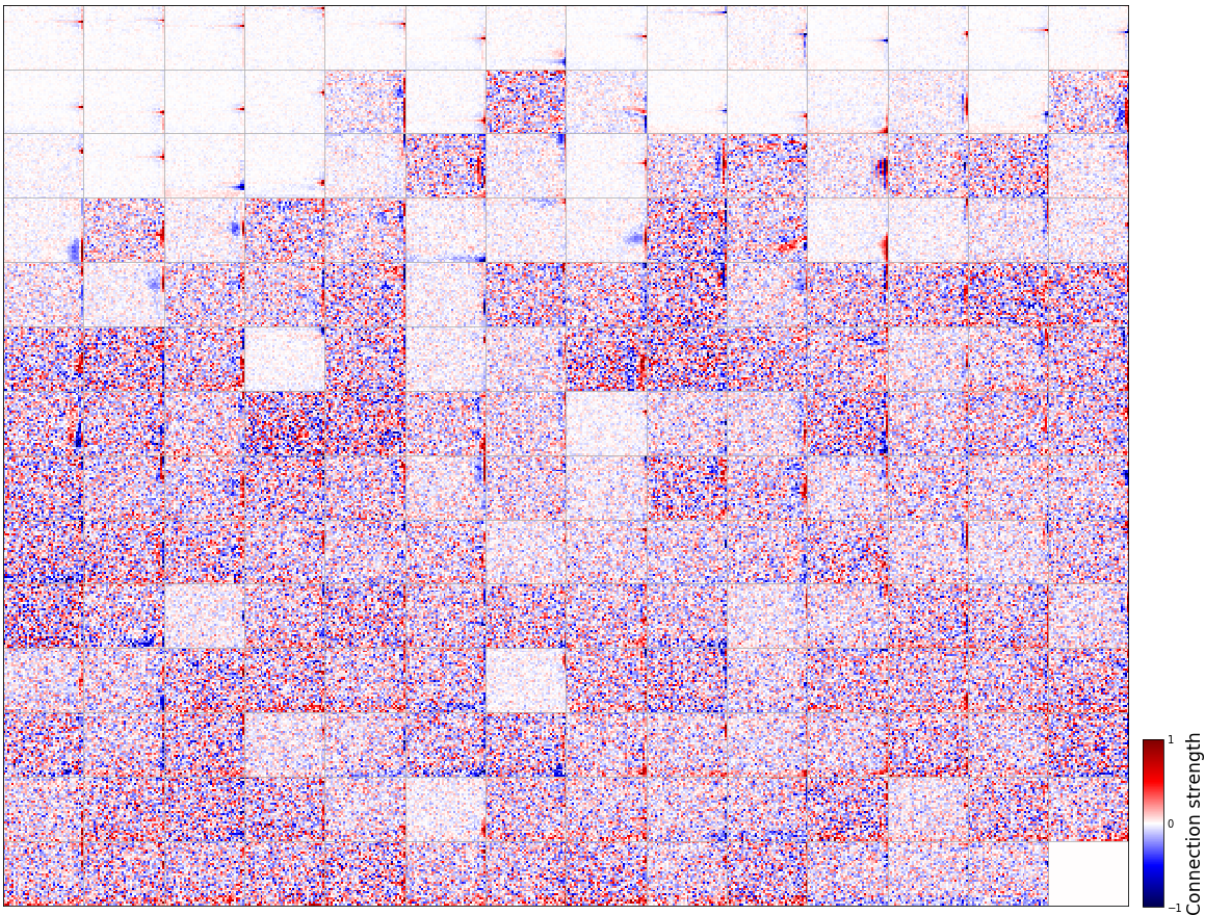
Supplementary Figure 2.23 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength $10^{-3.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



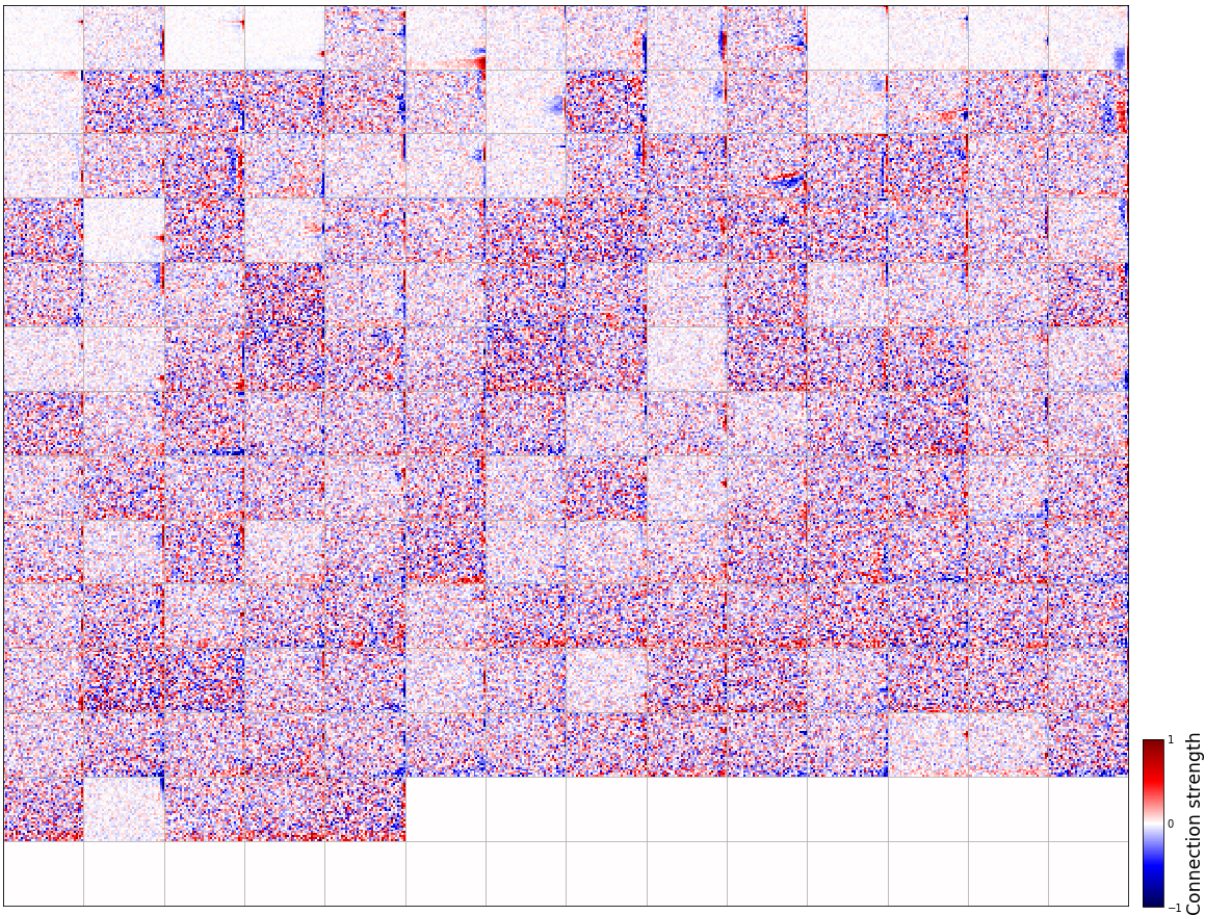
Supplementary Figure 2.24 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength 10^{-4}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



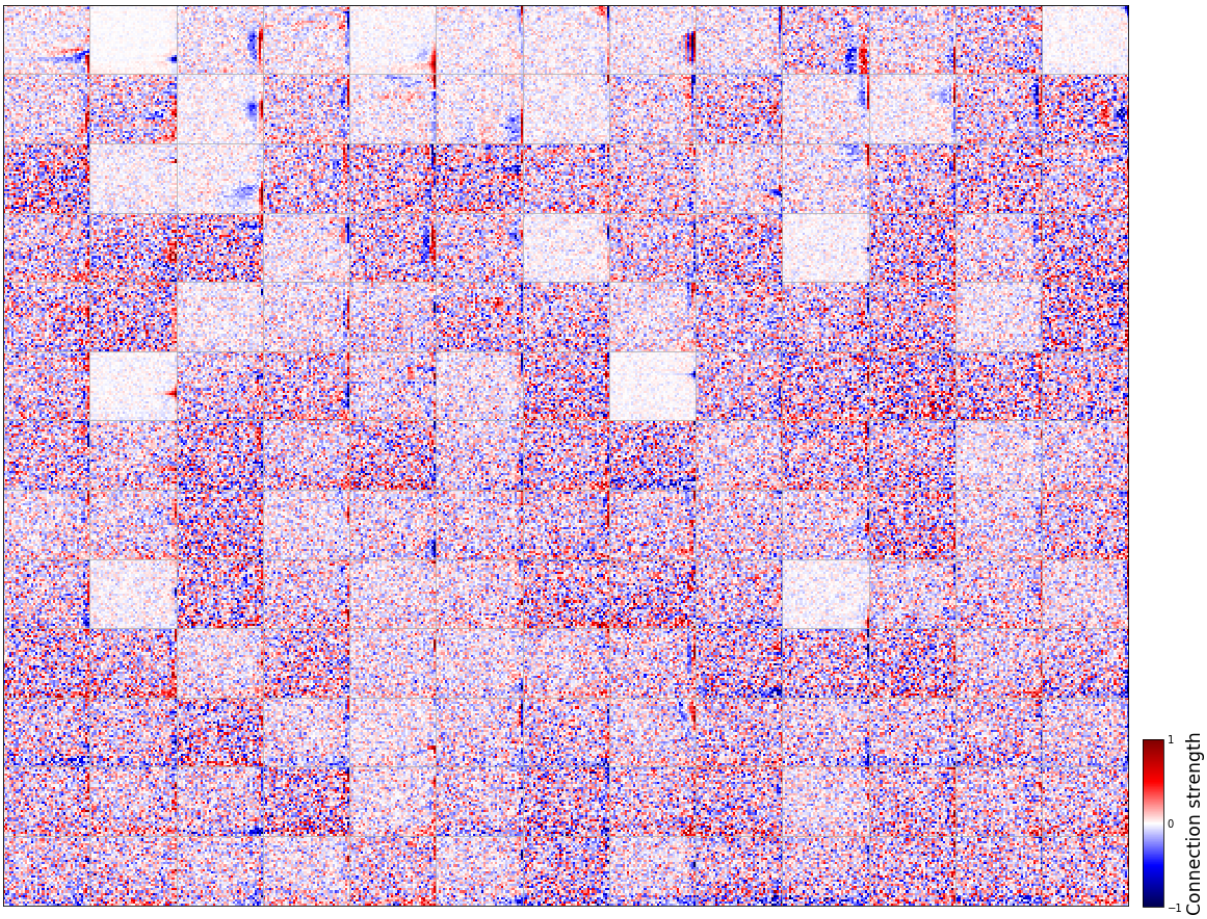
Supplementary Figure 2.25 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength $10^{-4.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



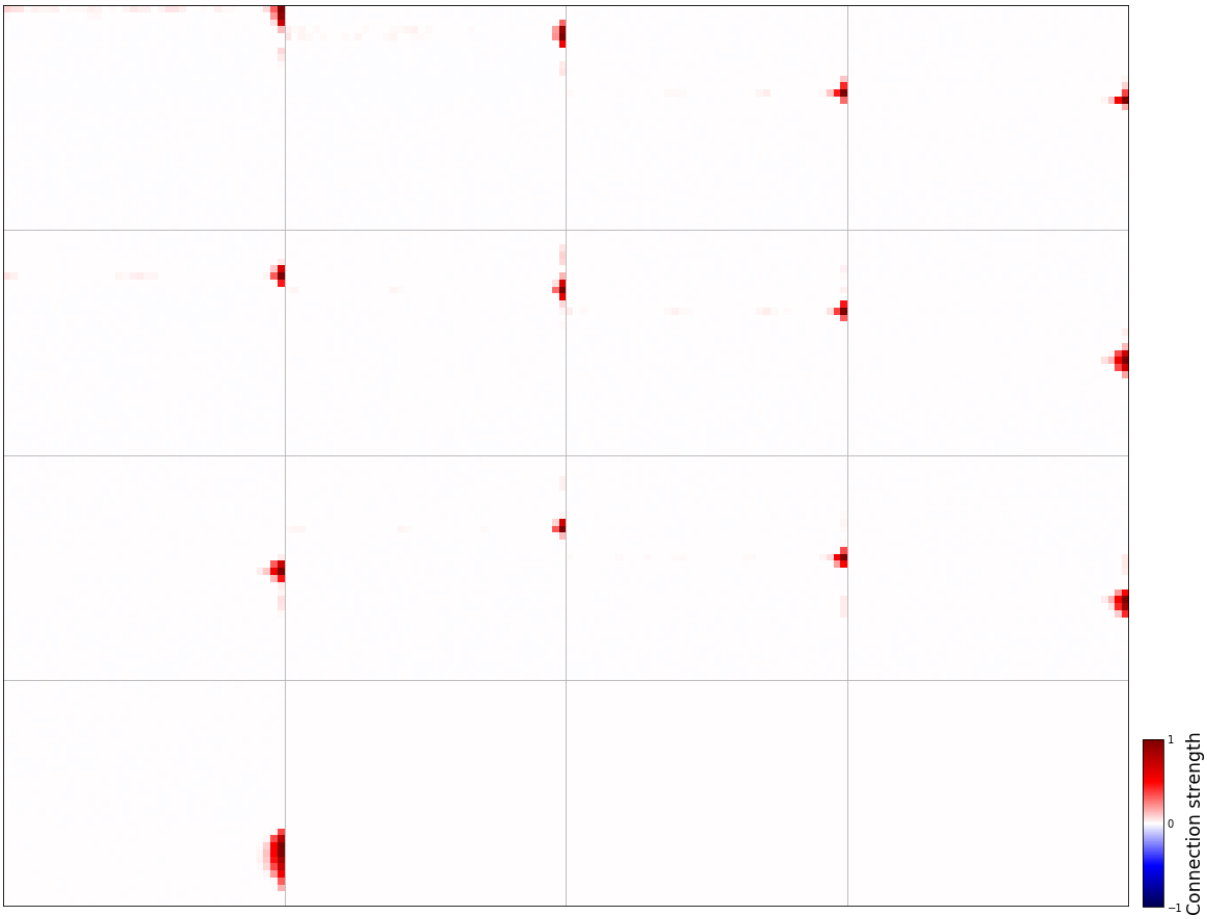
Supplementary Figure 2.26 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength 10^{-5}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



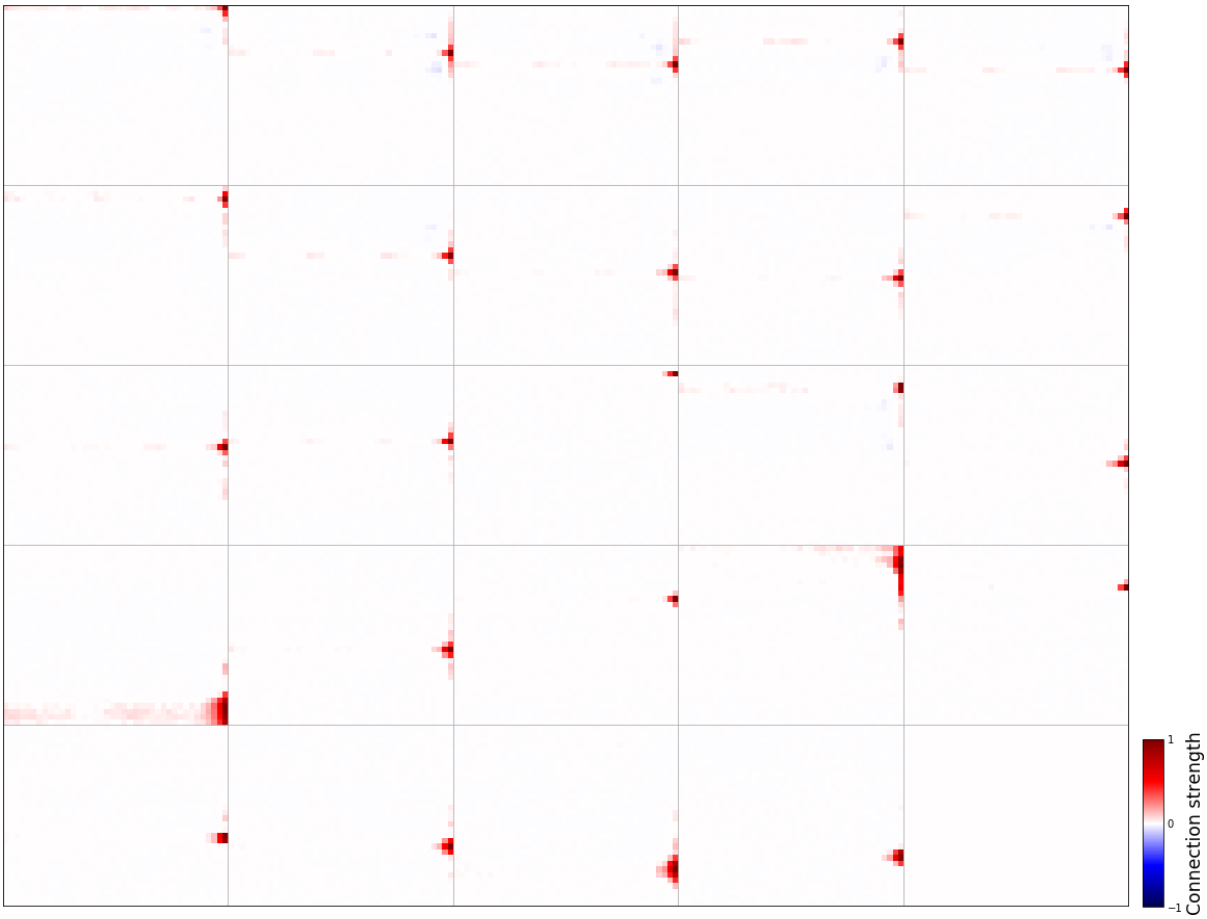
Supplementary Figure 2.27 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength $10^{-5.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



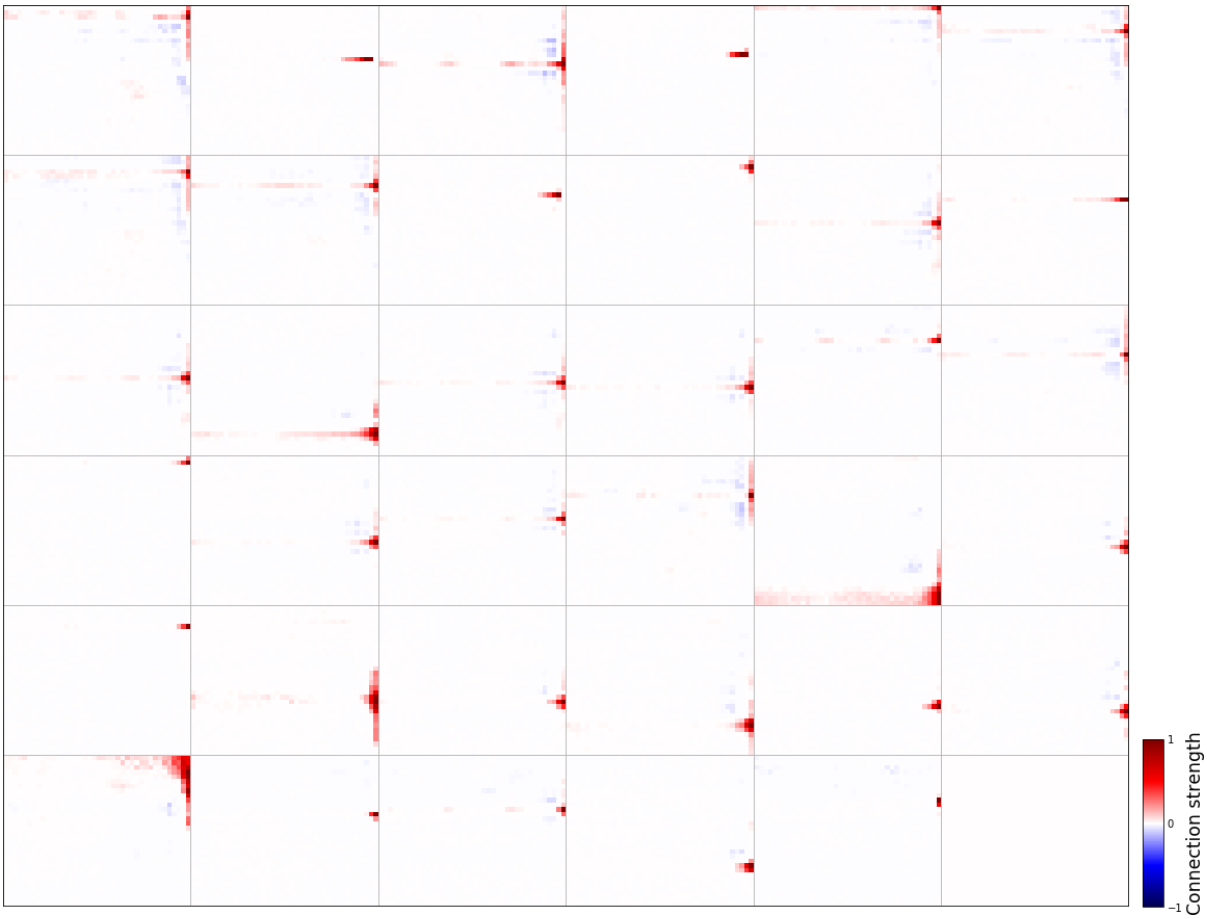
Supplementary Figure 2.28 | Full set of auditory RFs of the temporal prediction model trained with 200 hidden units and regularization strength 10^{-6}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



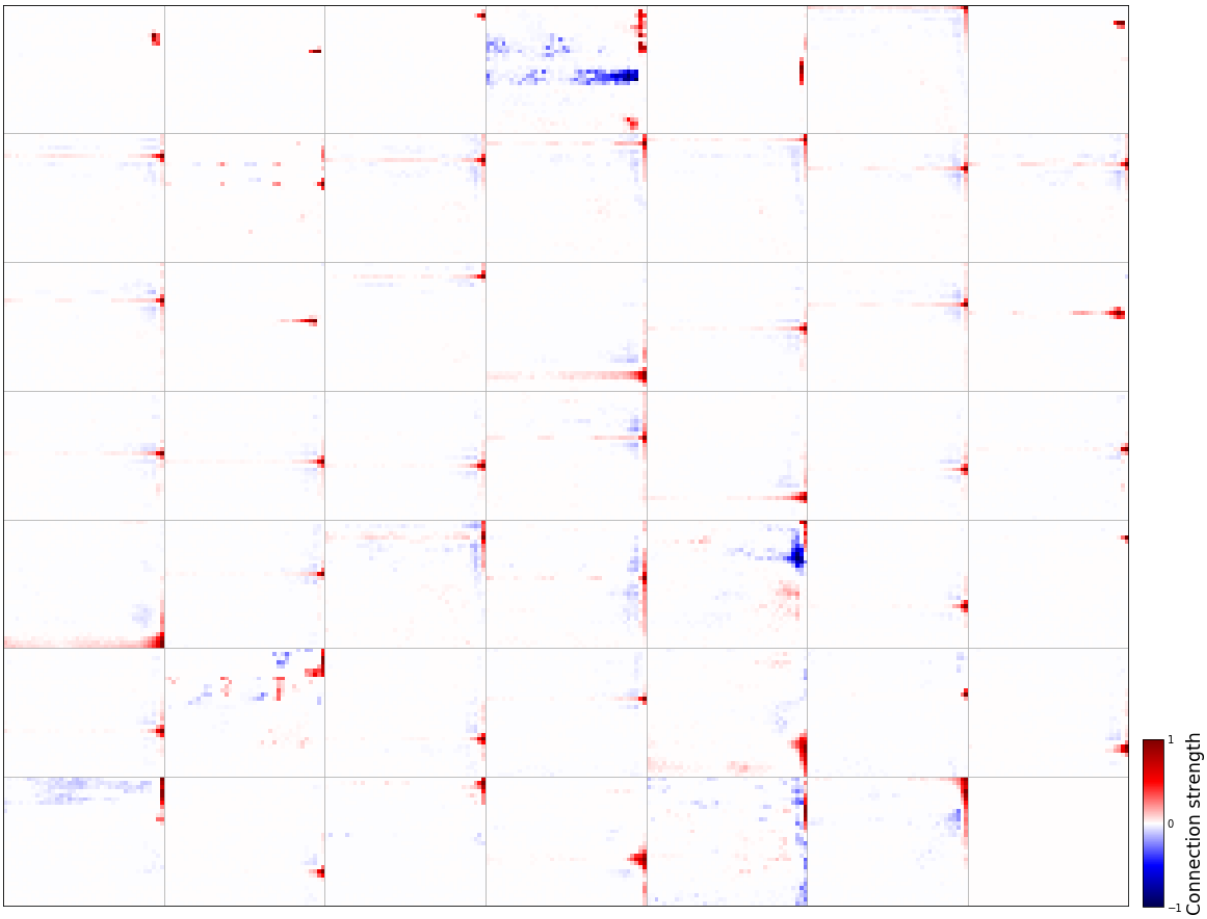
Supplementary Figure 2.29 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength 10^{-1}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



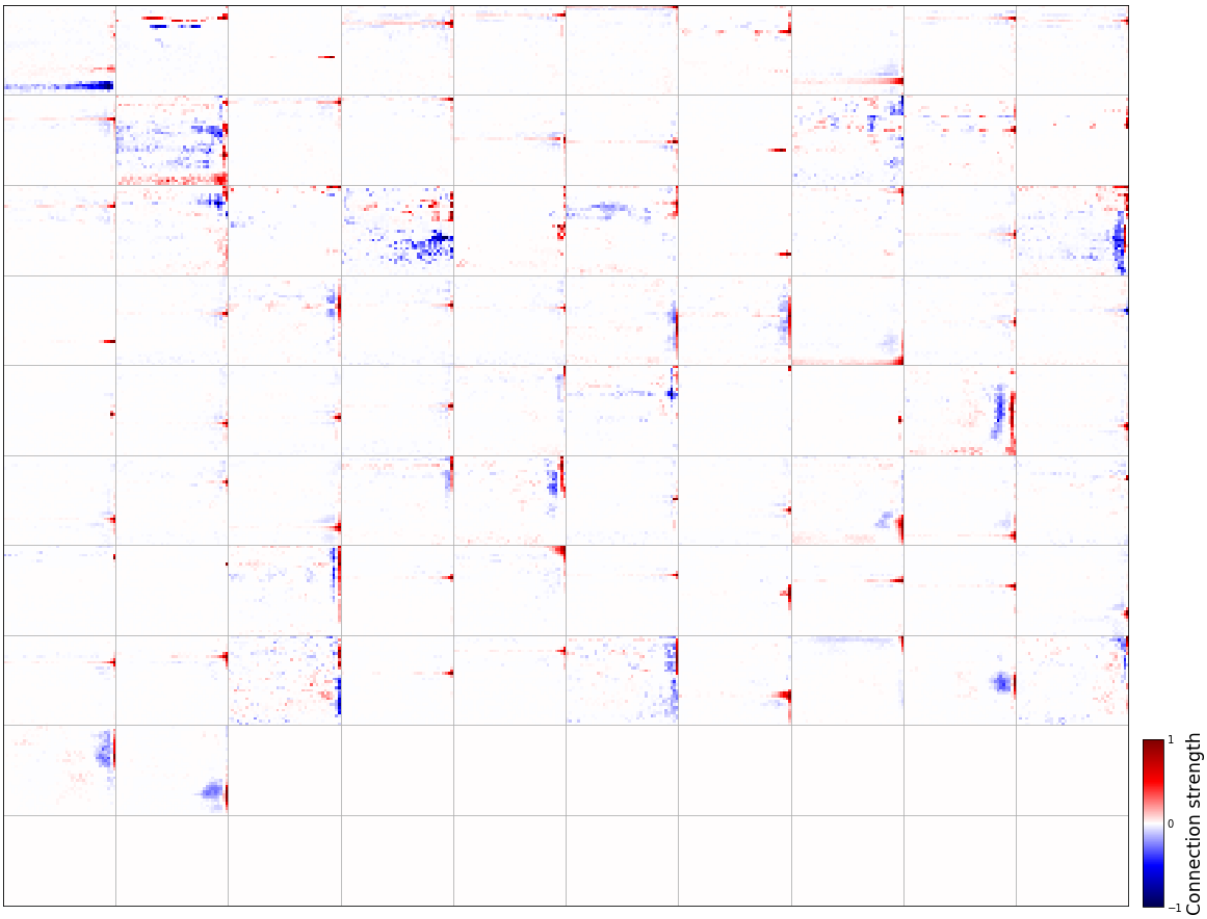
Supplementary Figure 2.30 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength $10^{-1.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



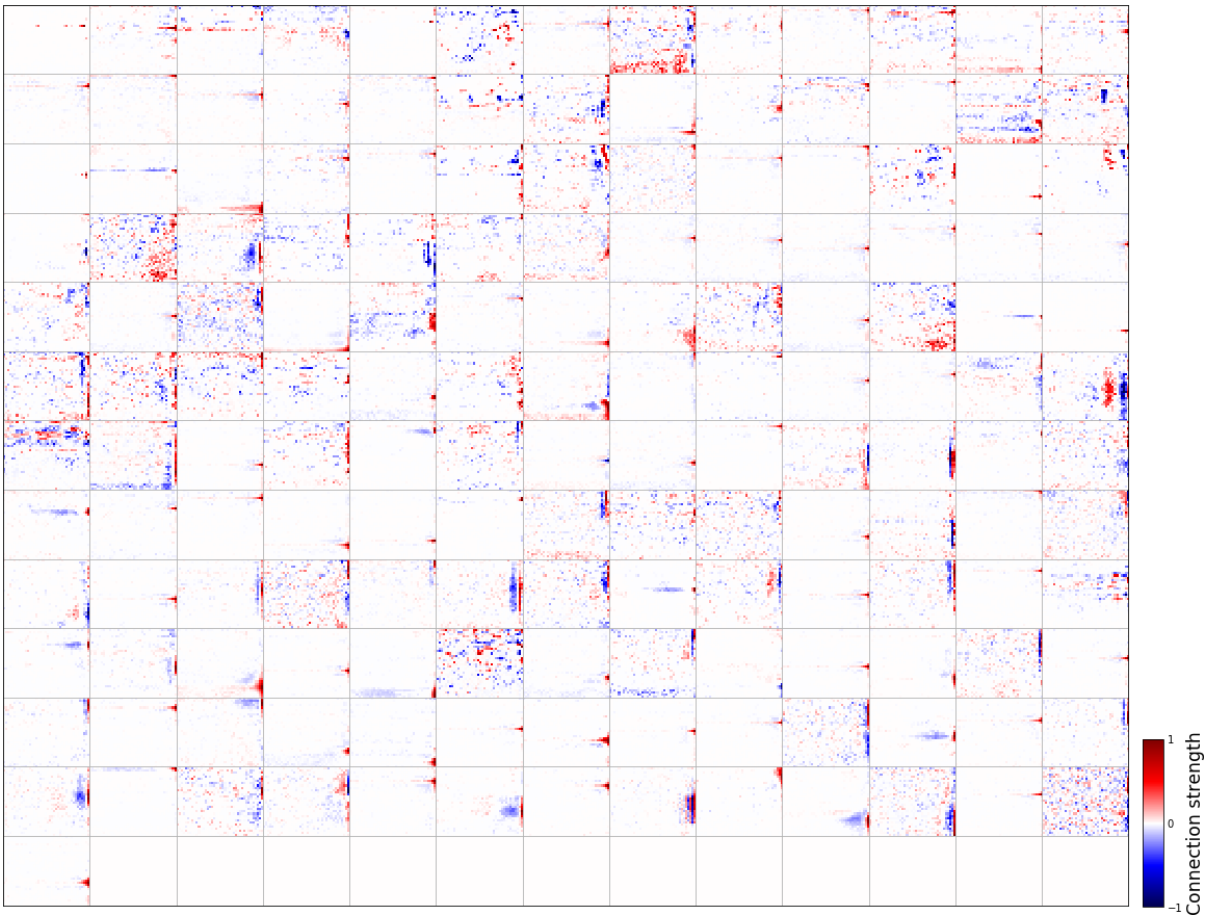
Supplementary Figure 2.31 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength 10^{-2}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



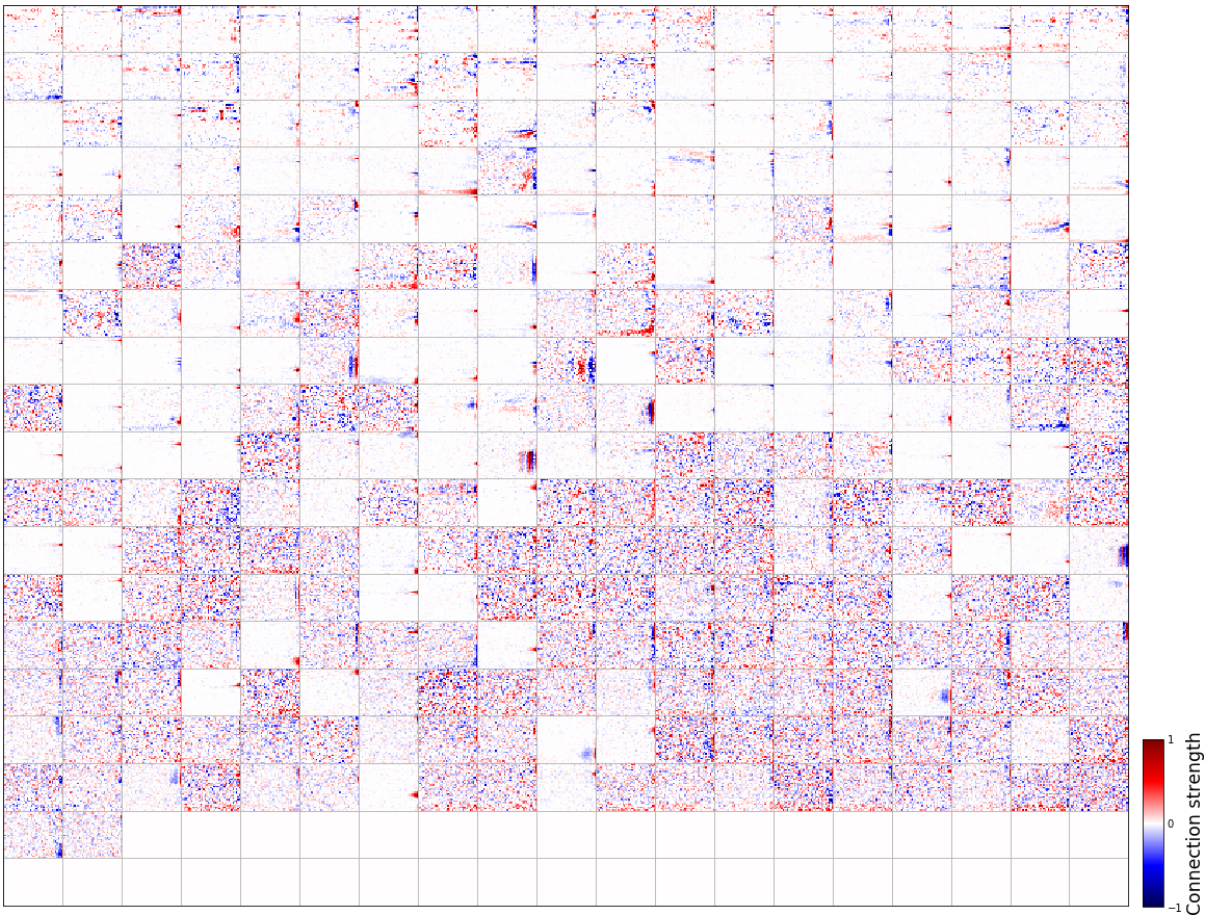
Supplementary Figure 2.32 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength $10^{-2.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



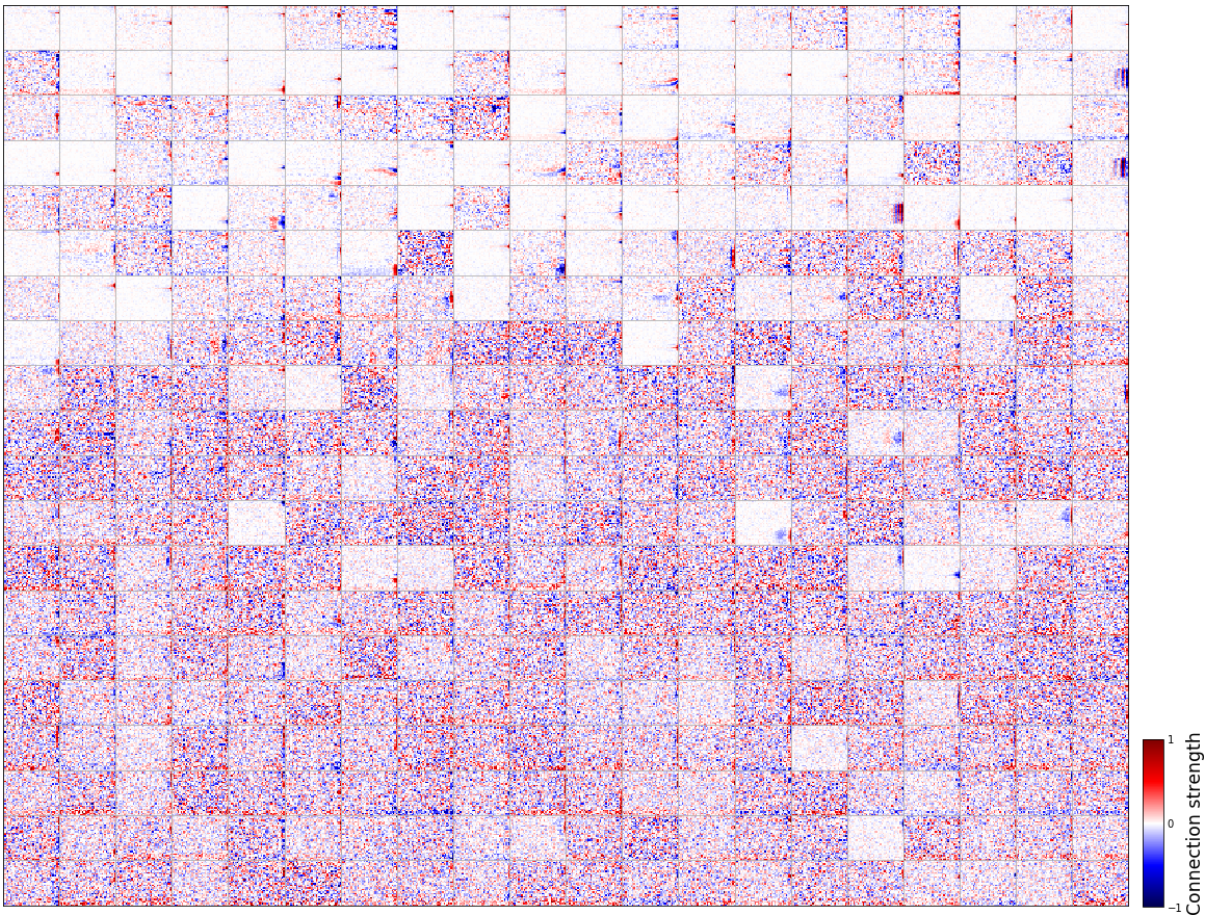
Supplementary Figure 2.33 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength 10^{-3}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



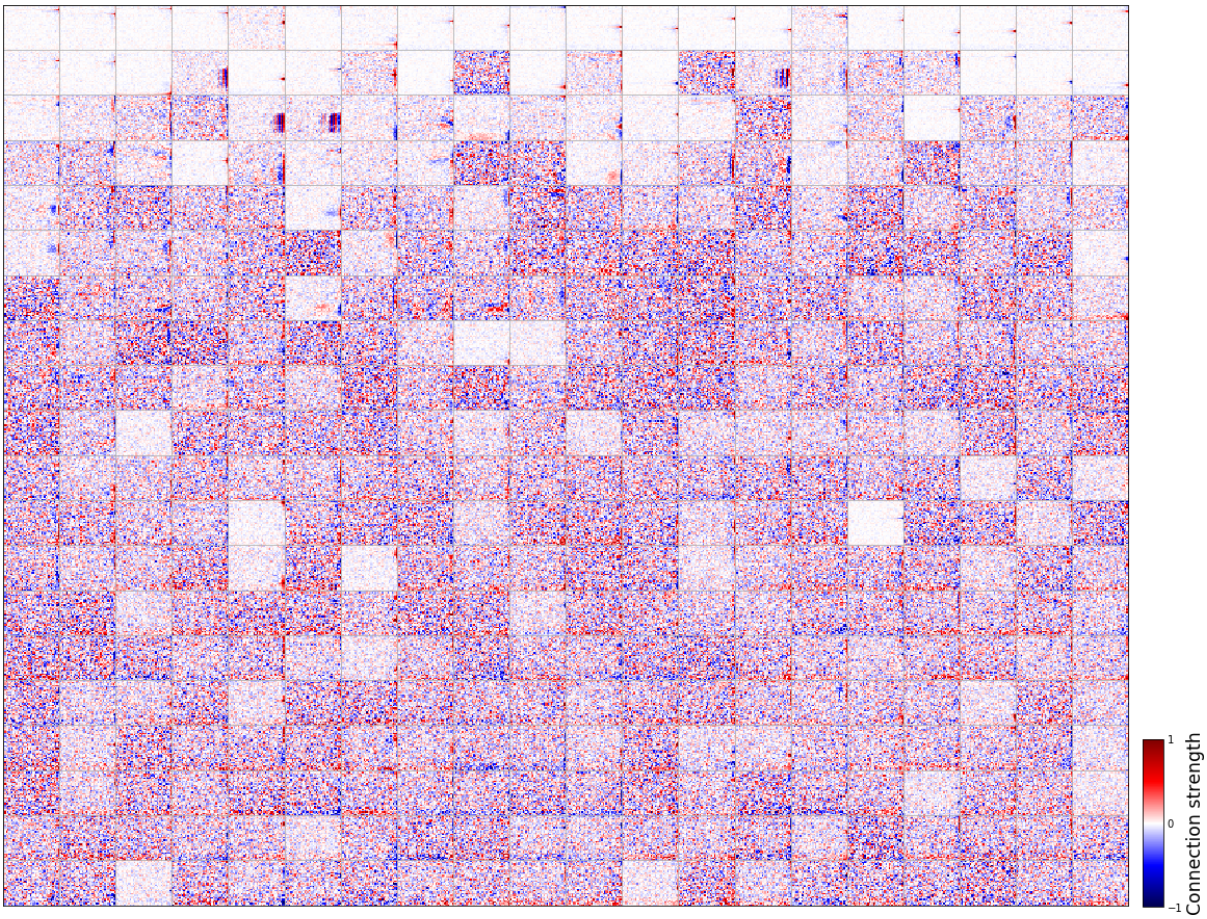
Supplementary Figure 2.34 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength $10^{-3.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



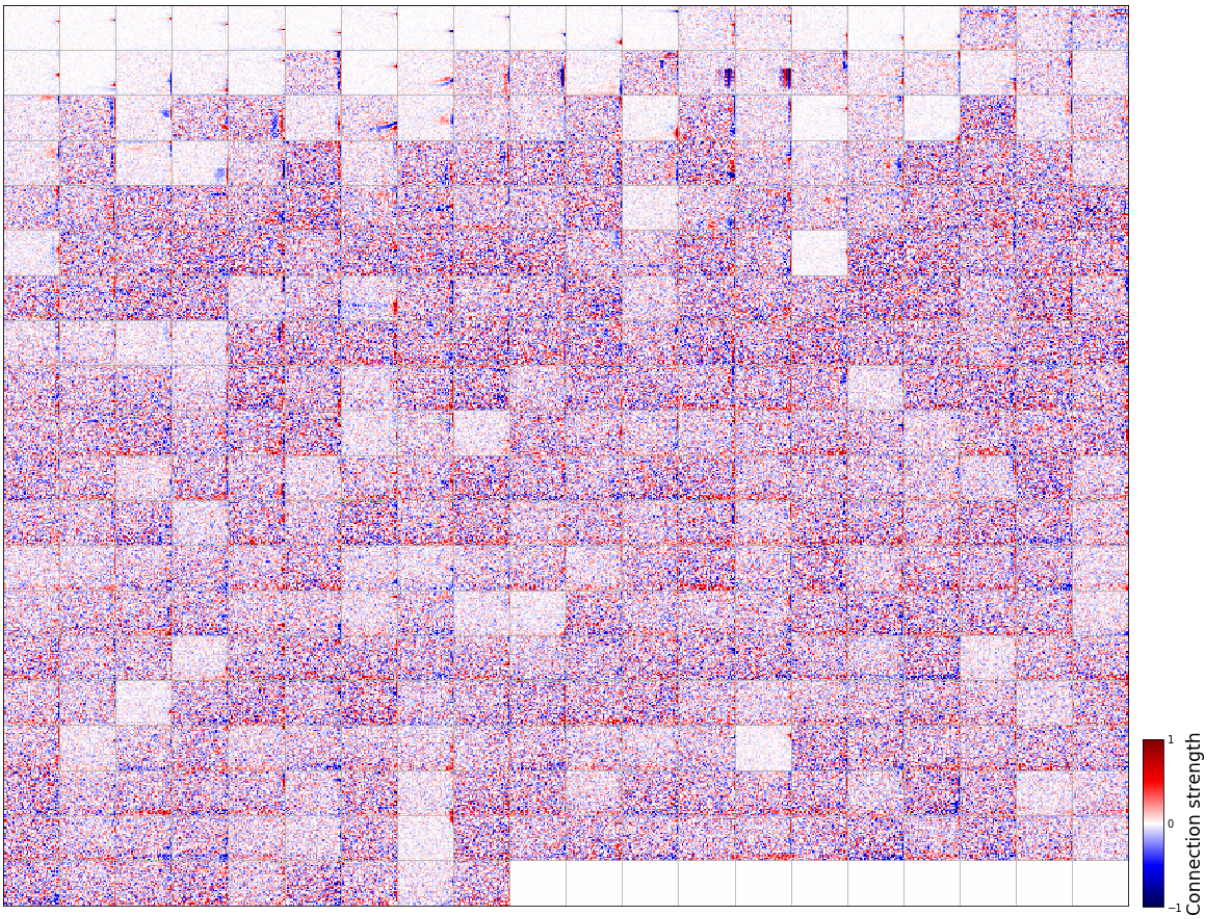
Supplementary Figure 2.35 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength 10^{-4}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



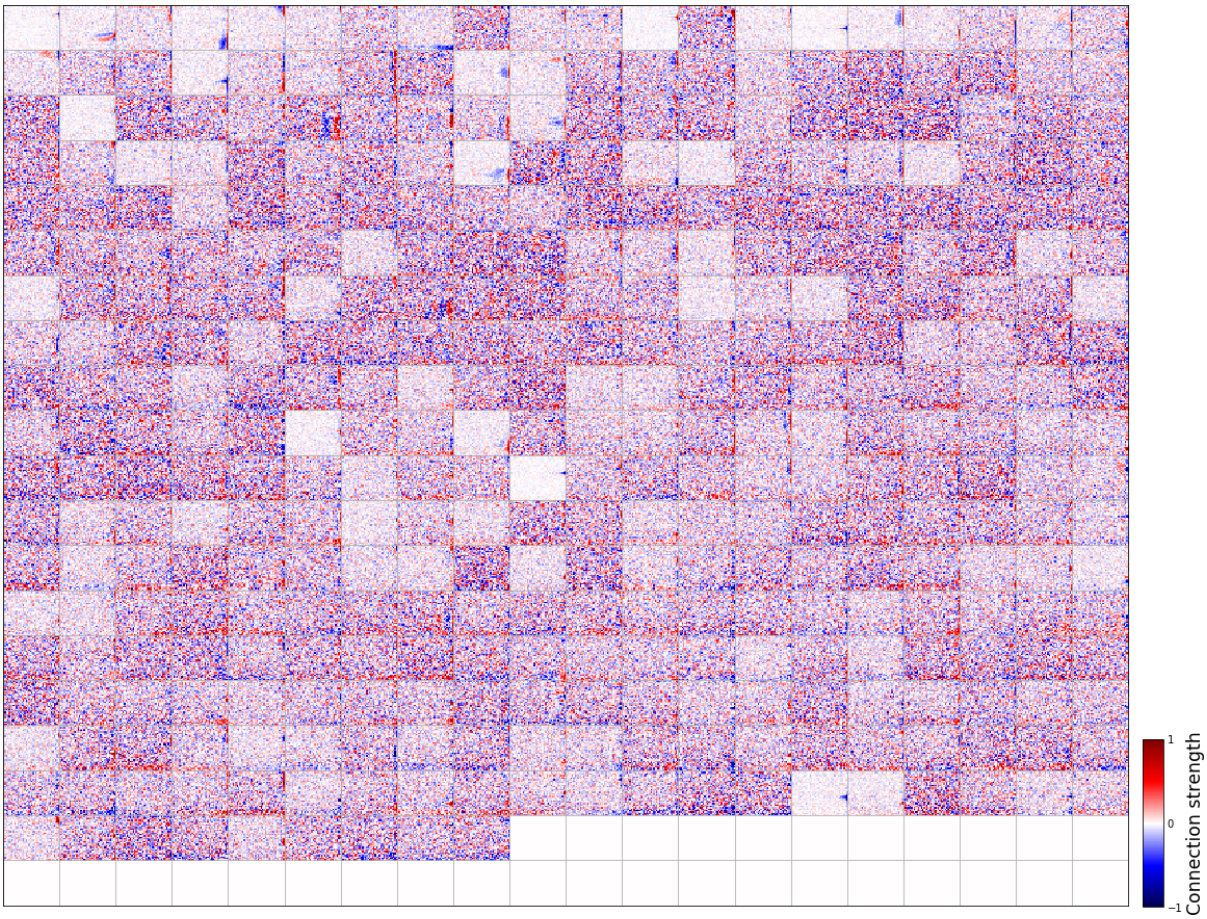
Supplementary Figure 2.36 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength $10^{-4.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



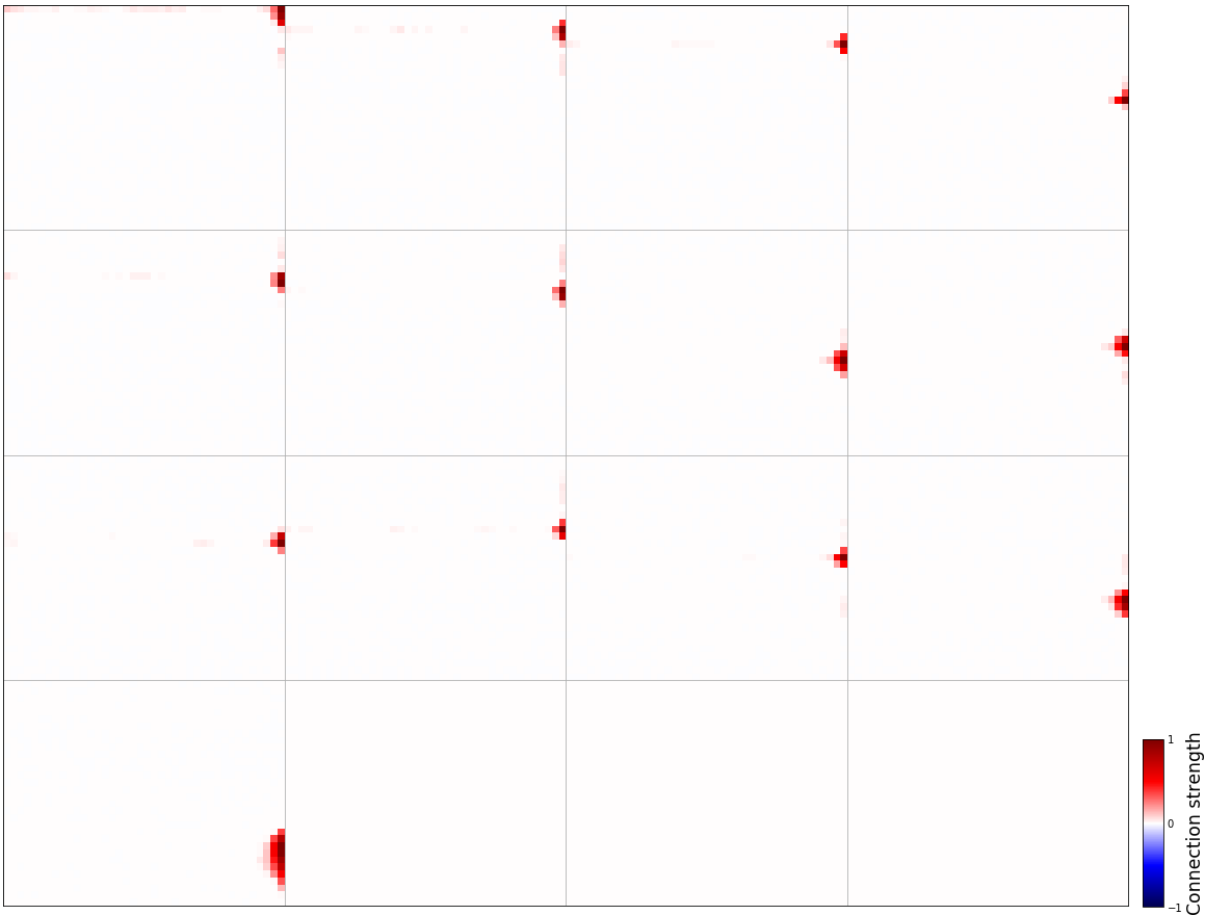
Supplementary Figure 2.37 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength 10^{-5}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



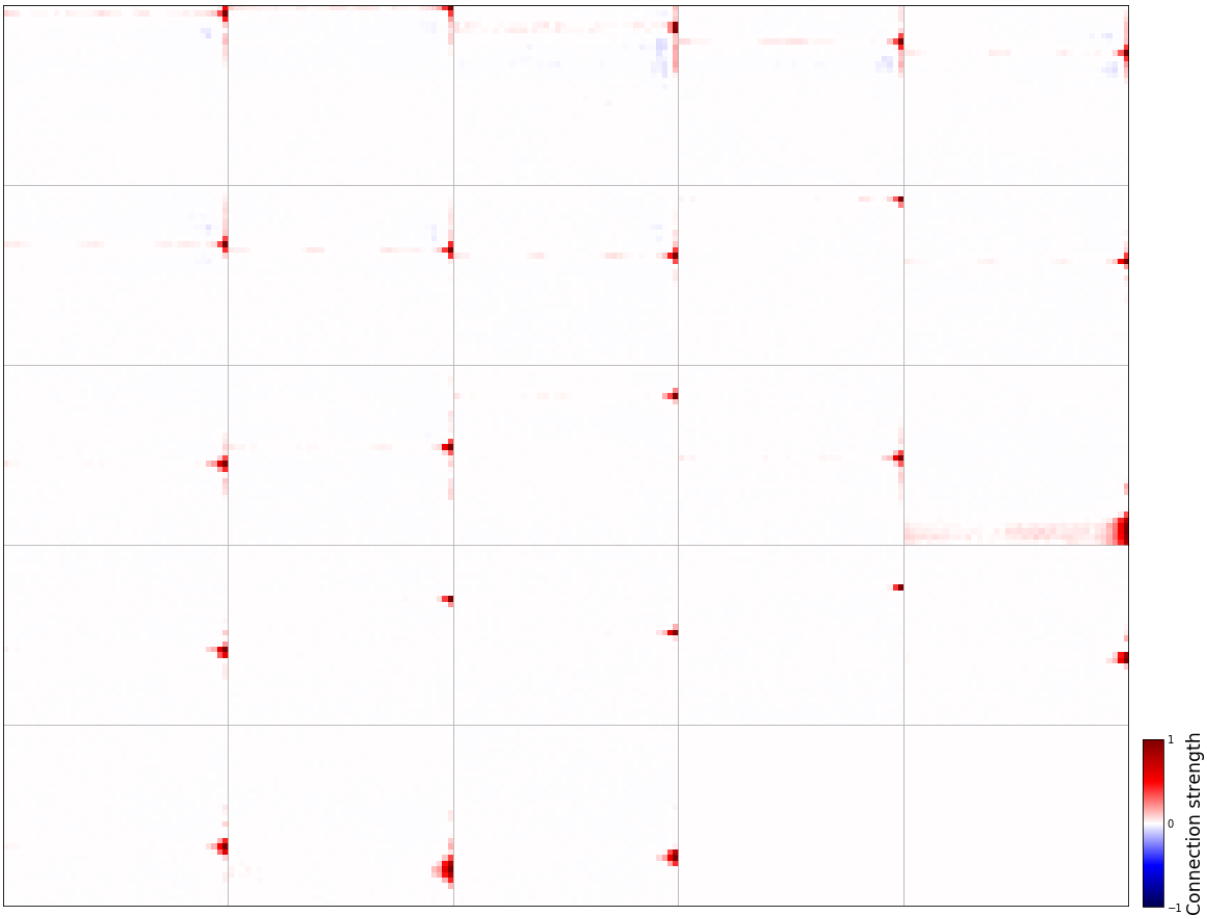
Supplementary Figure 2.38 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength $10^{-5.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



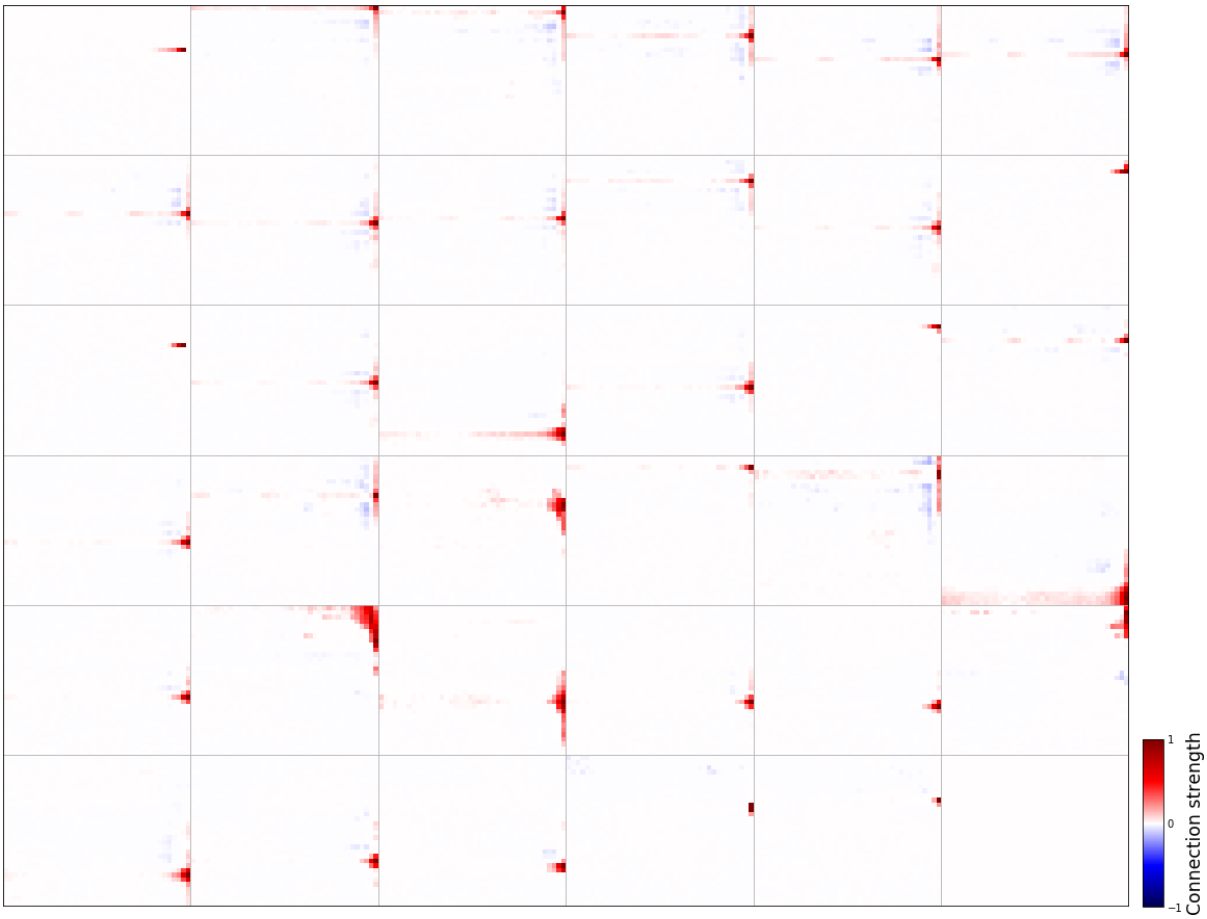
Supplementary Figure 2.39 | Full set of auditory RFs of the temporal prediction model trained with 400 hidden units and regularization strength 10^{-6}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



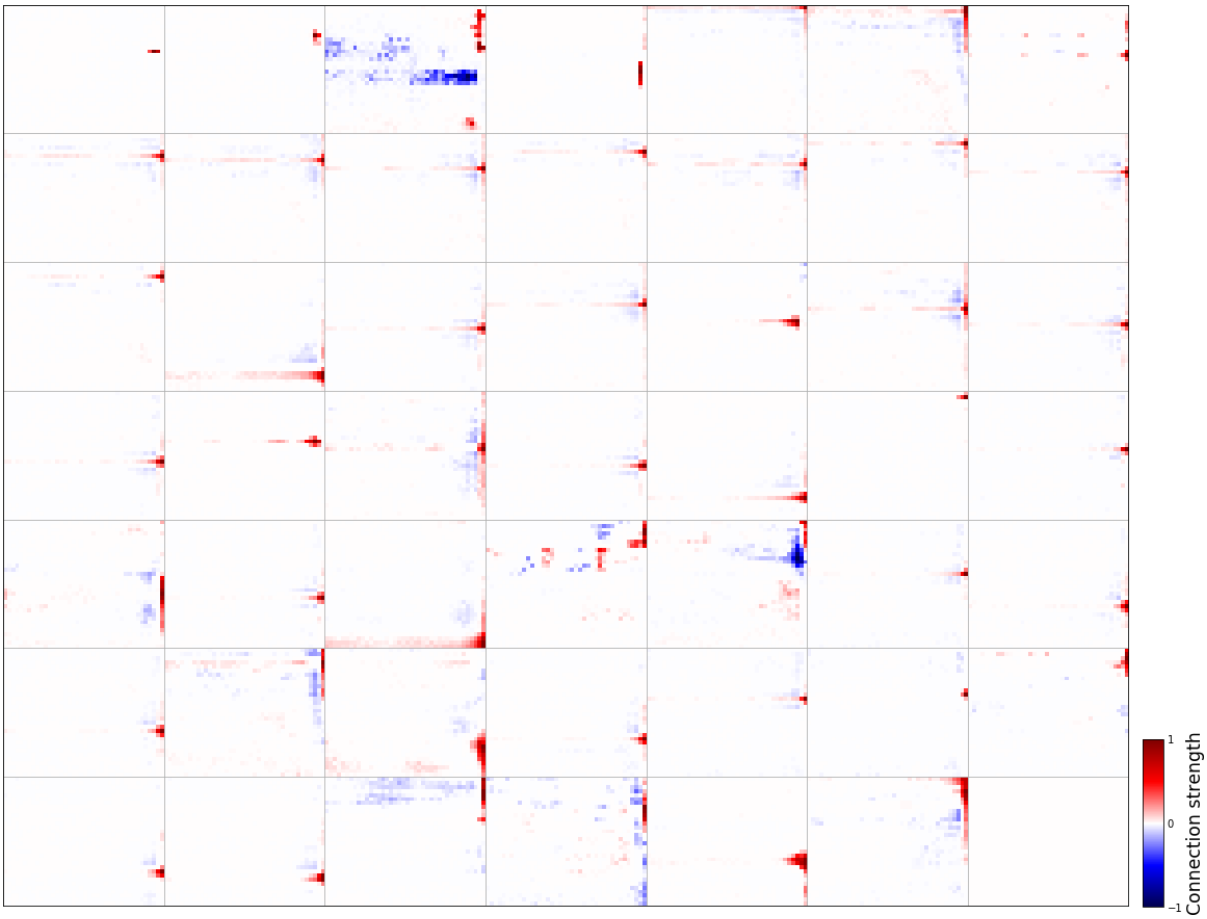
Supplementary Figure 2.40 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength 10^{-1}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



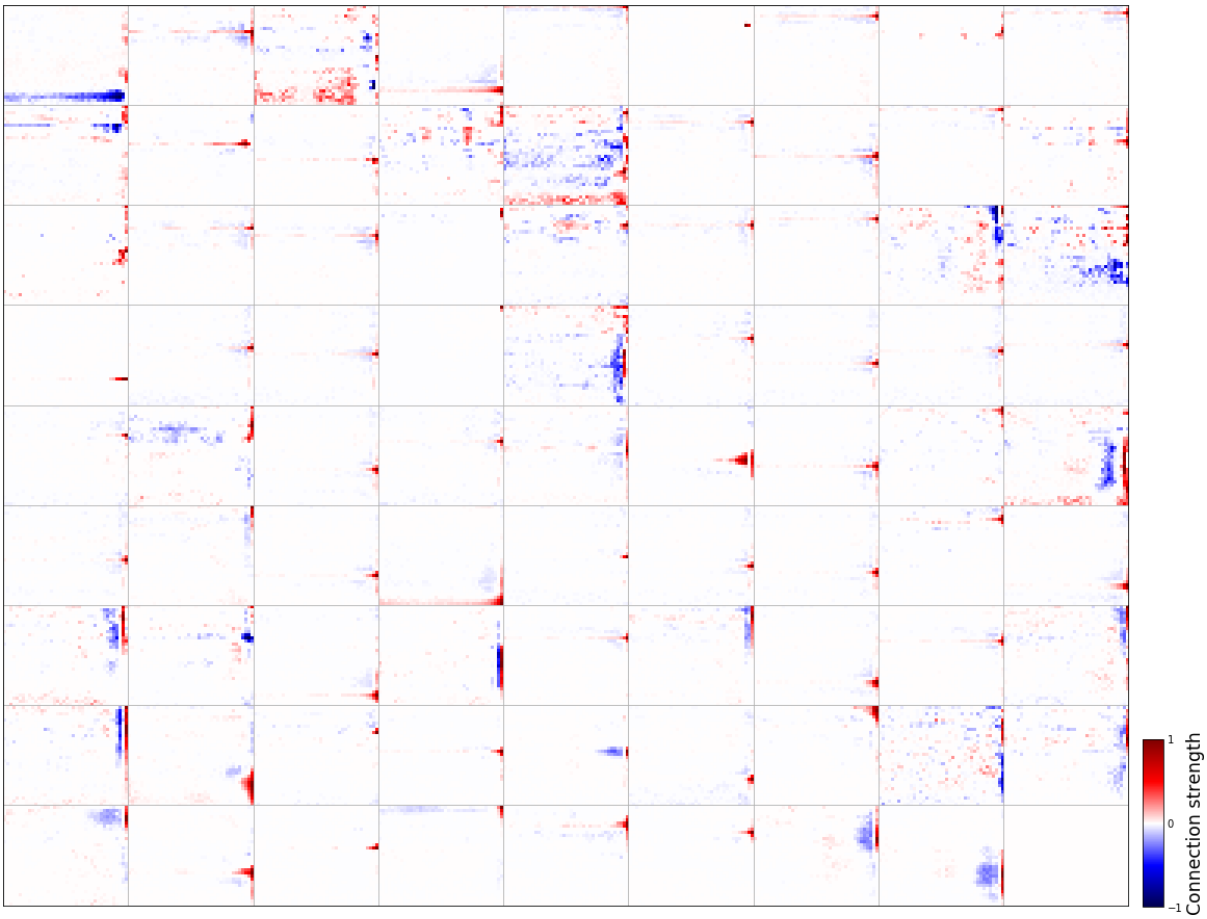
Supplementary Figure 2.41 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength $10^{-1.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



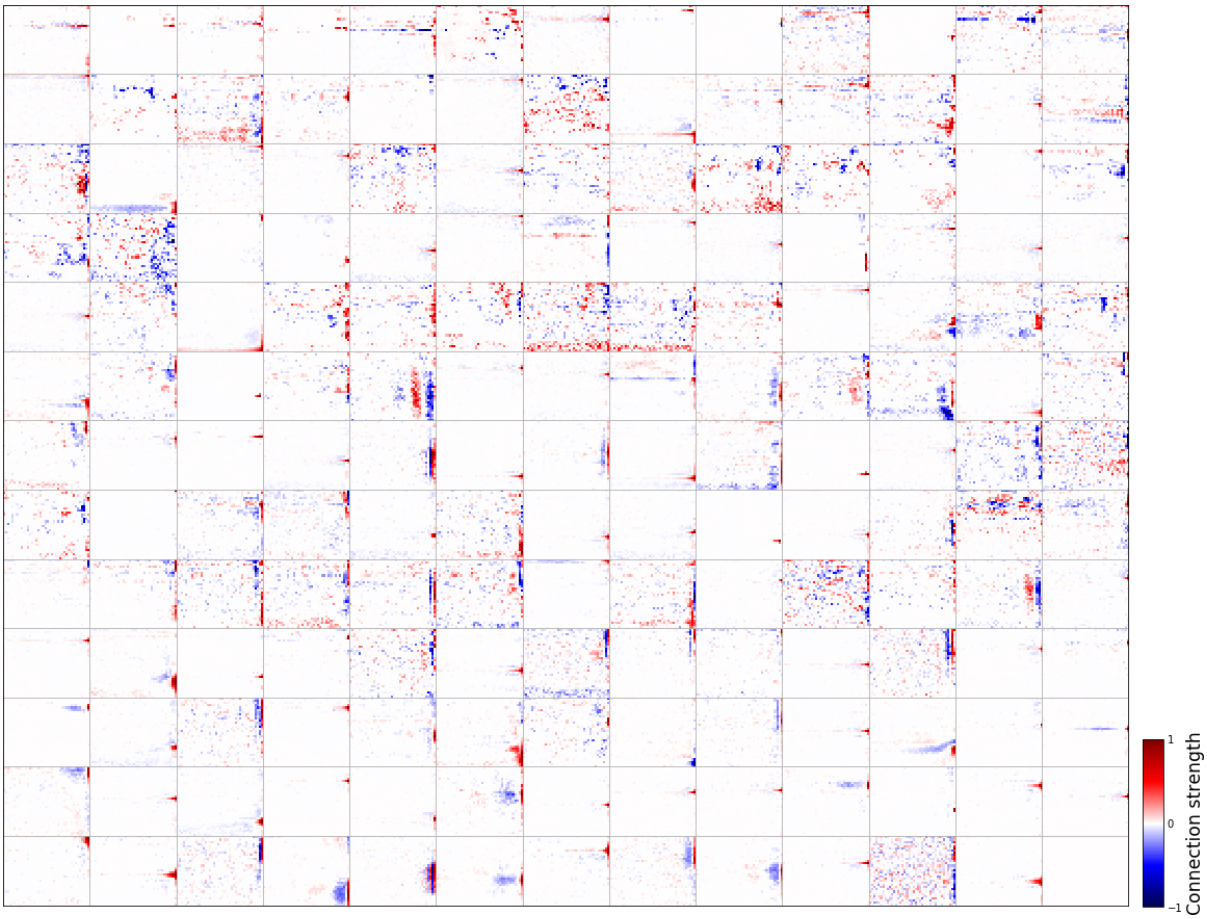
Supplementary Figure 2.42 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength 10^{-2}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



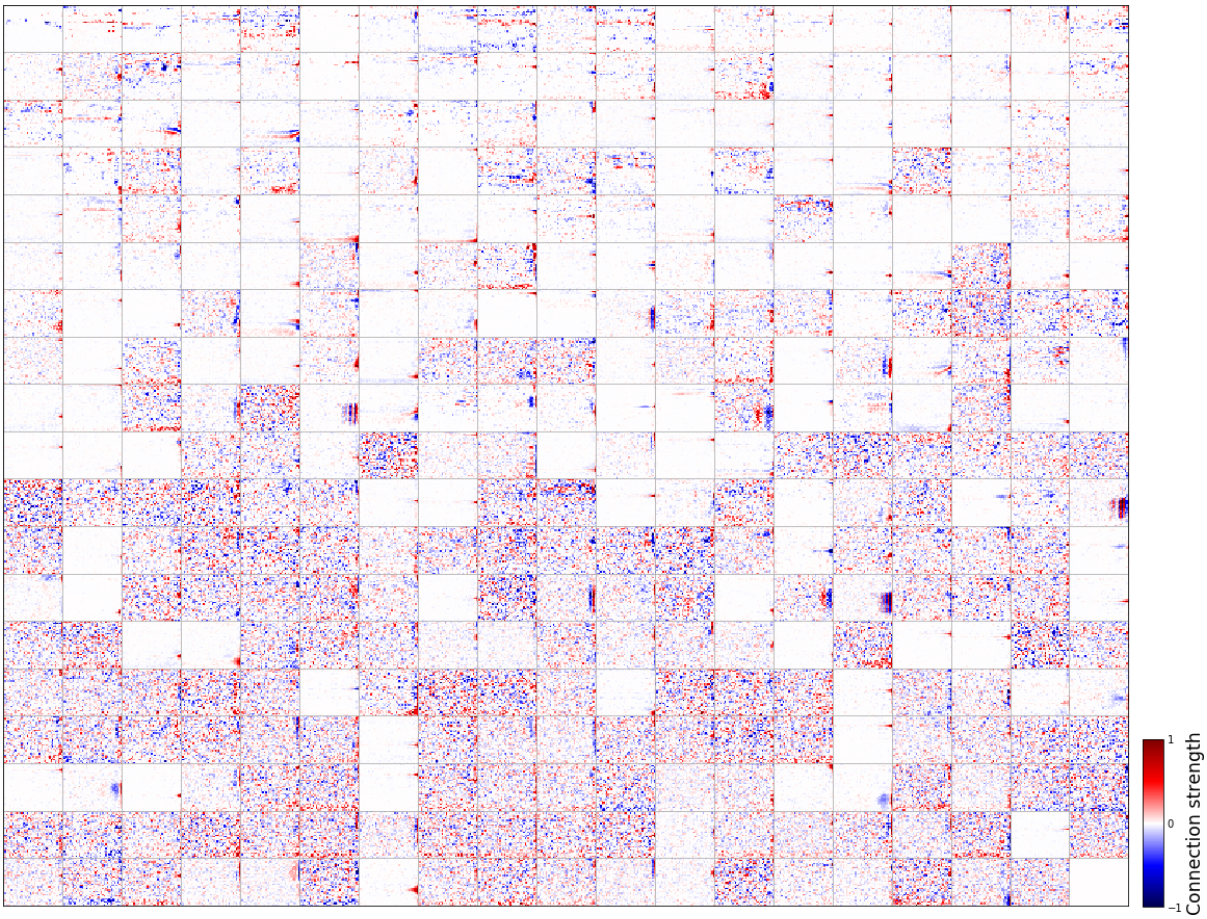
Supplementary Figure 2.43 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength $10^{-2.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



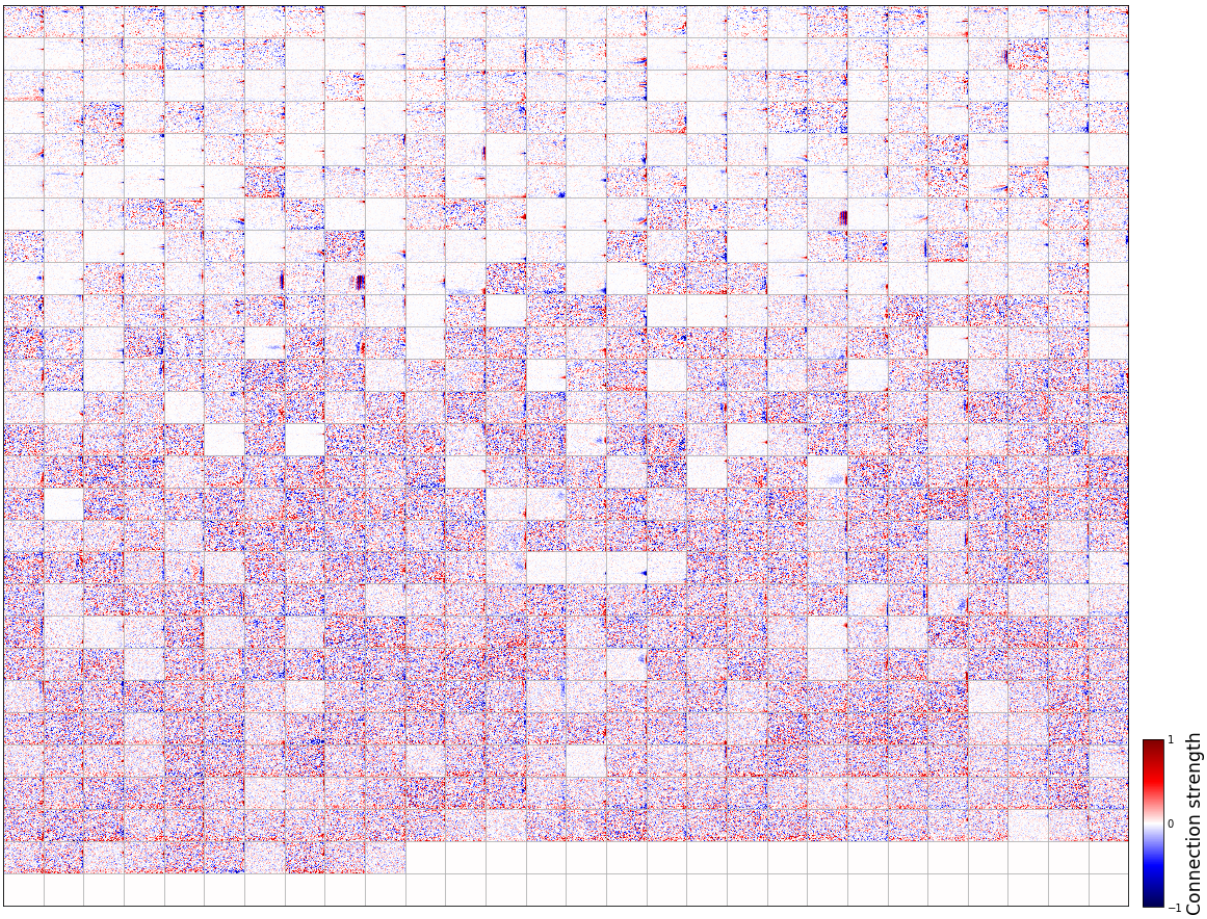
Supplementary Figure 2.44 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength 10^{-3}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



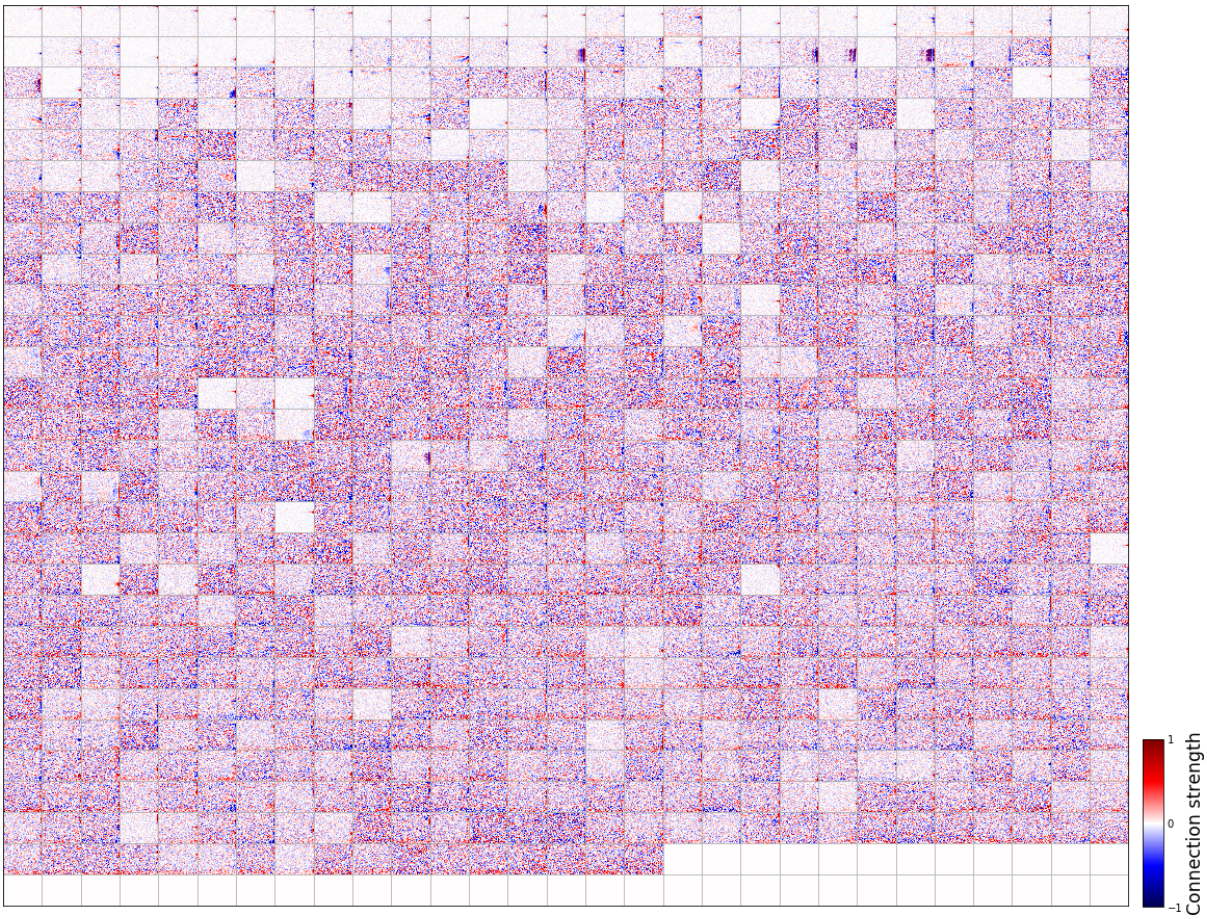
Supplementary Figure 2.45 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength $10^{-3.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



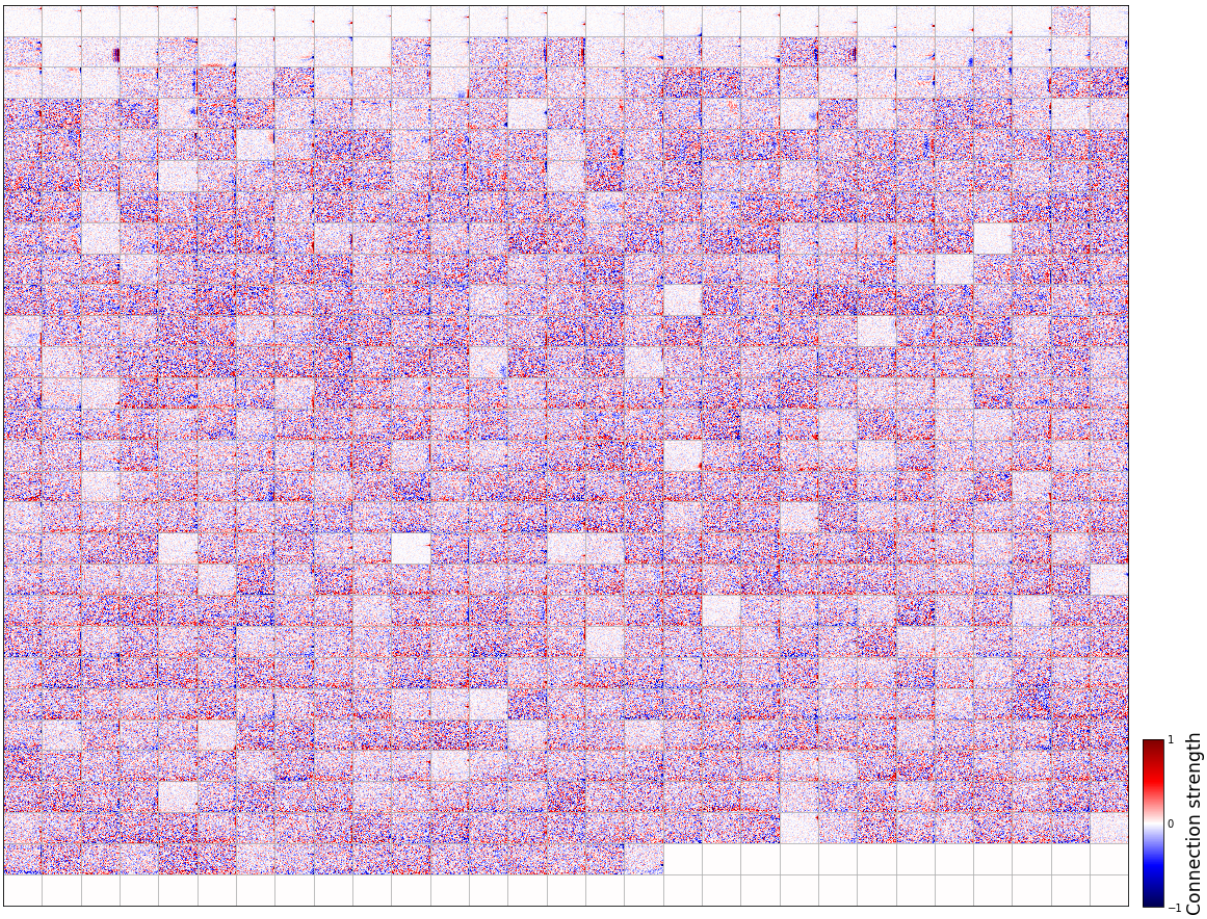
Supplementary Figure 2.46 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength 10^{-4}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



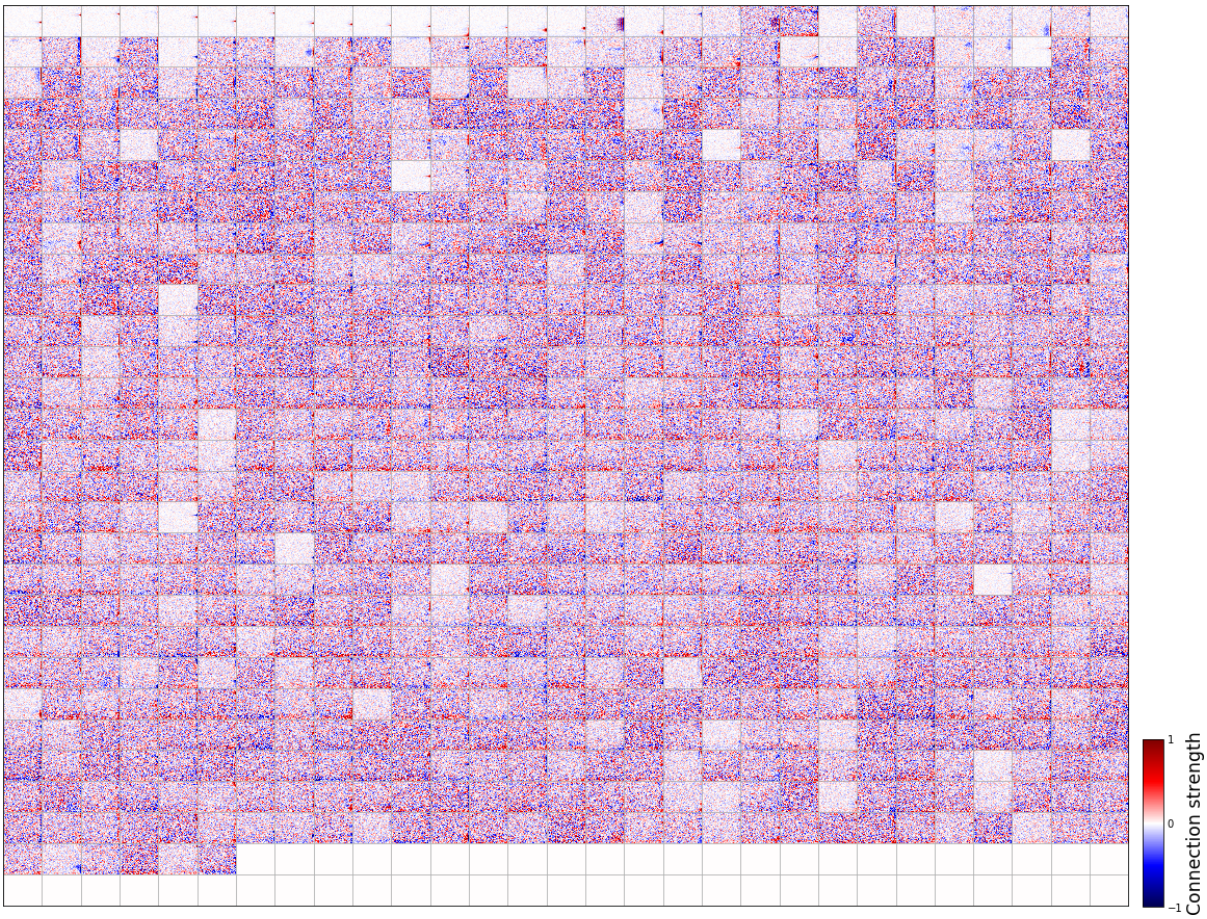
Supplementary Figure 2.47 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength $10^{-4.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



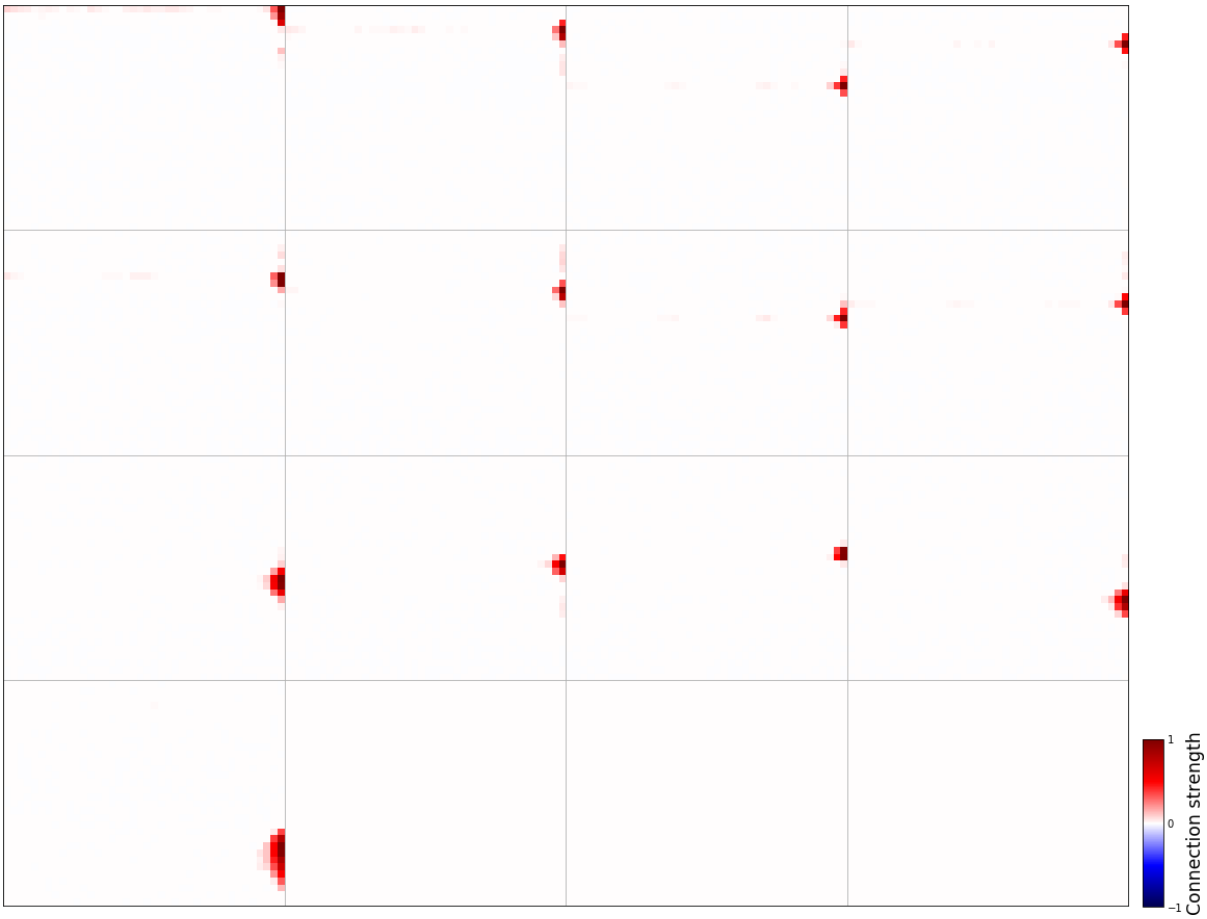
Supplementary Figure 2.48 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength 10^{-5}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



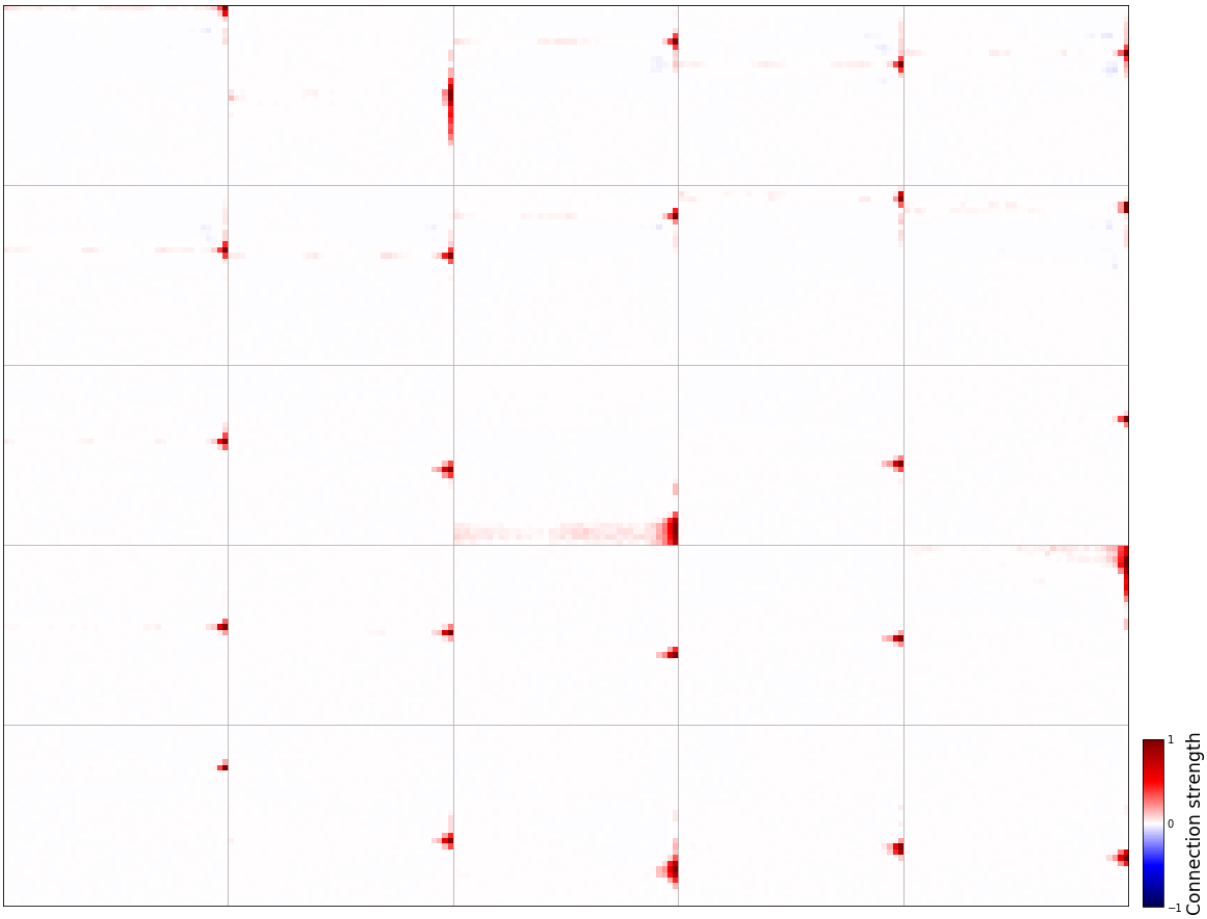
Supplementary Figure 2.49 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength $10^{-5.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



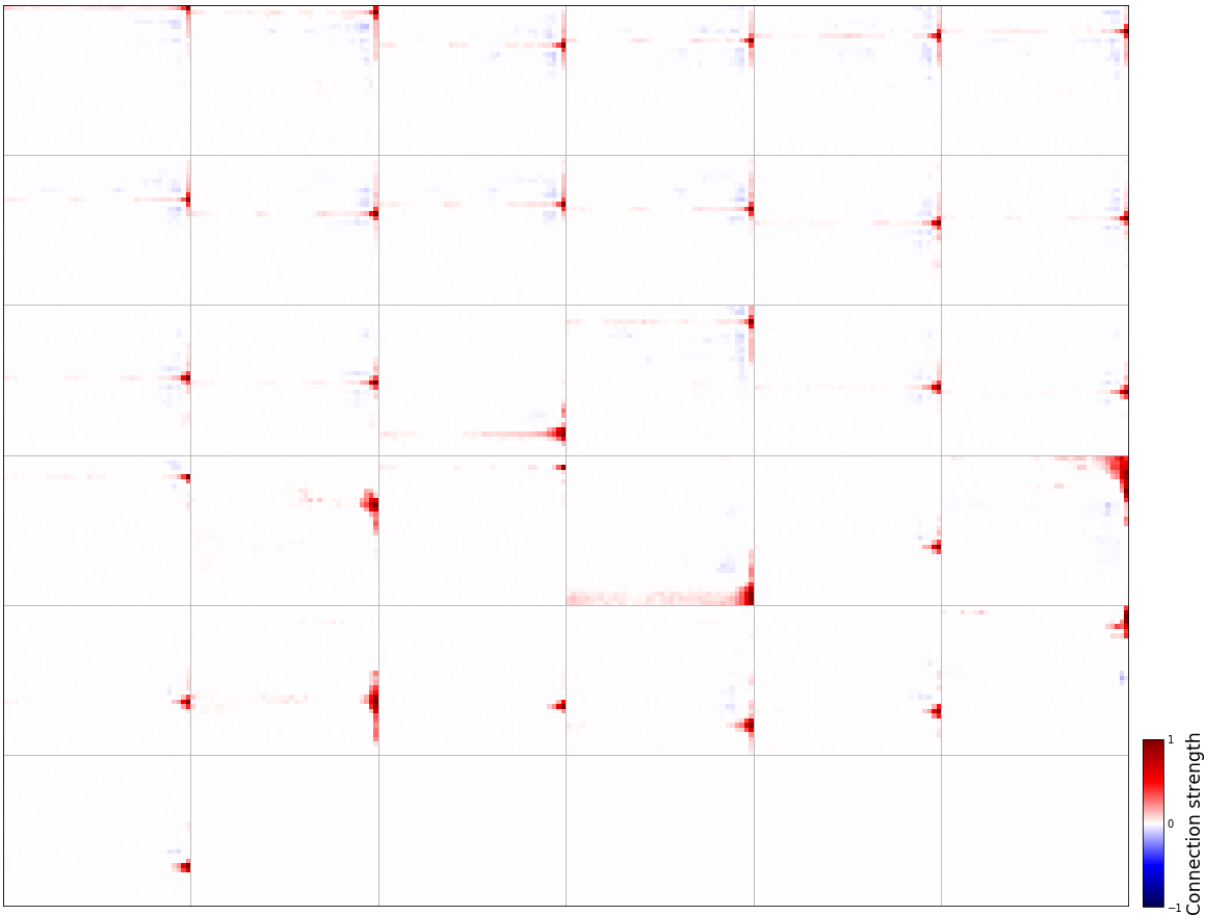
Supplementary Figure 2.50 | Full set of auditory RFs of the temporal prediction model trained with 800 hidden units and regularization strength 10^{-6}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



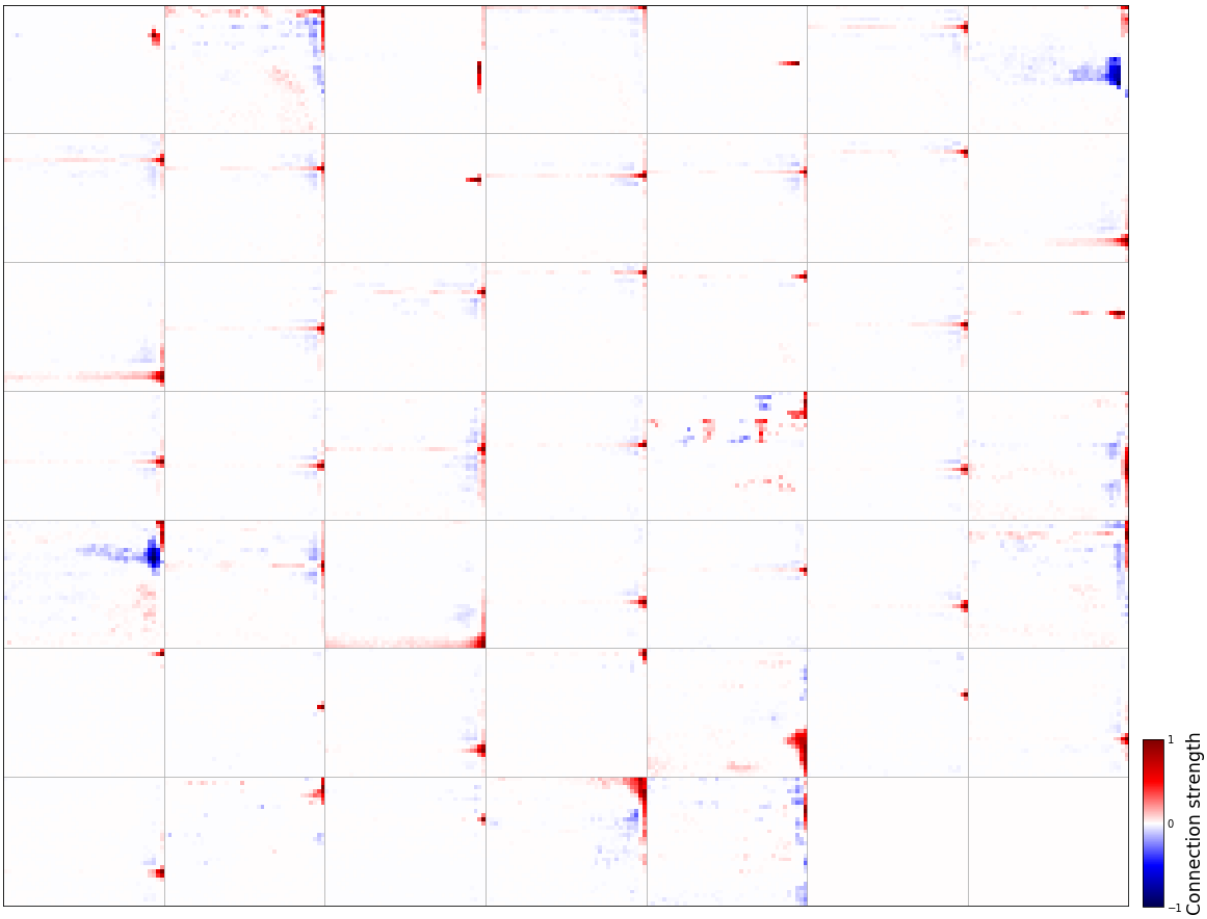
Supplementary Figure 2.51 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength 10^{-1}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



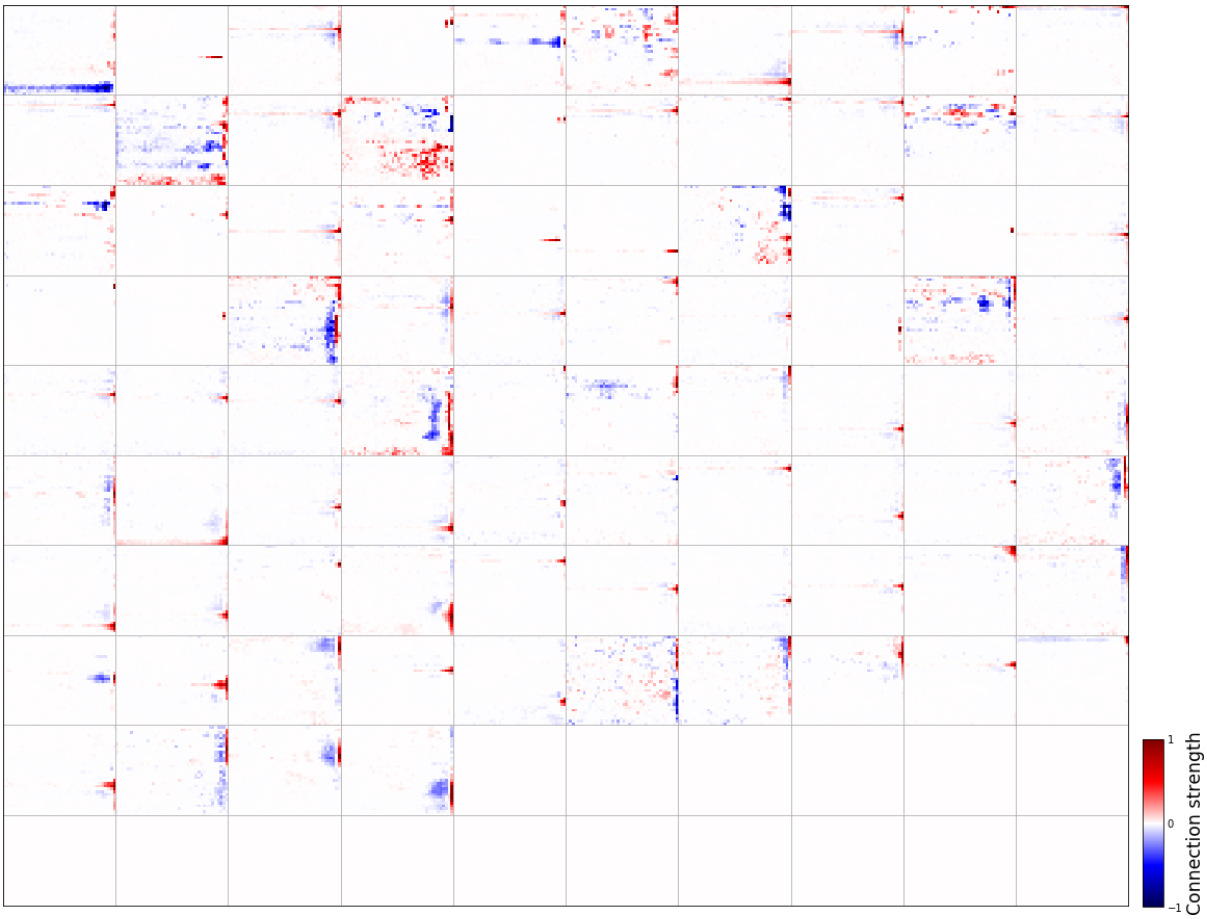
Supplementary Figure 2.52 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength $10^{-1.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



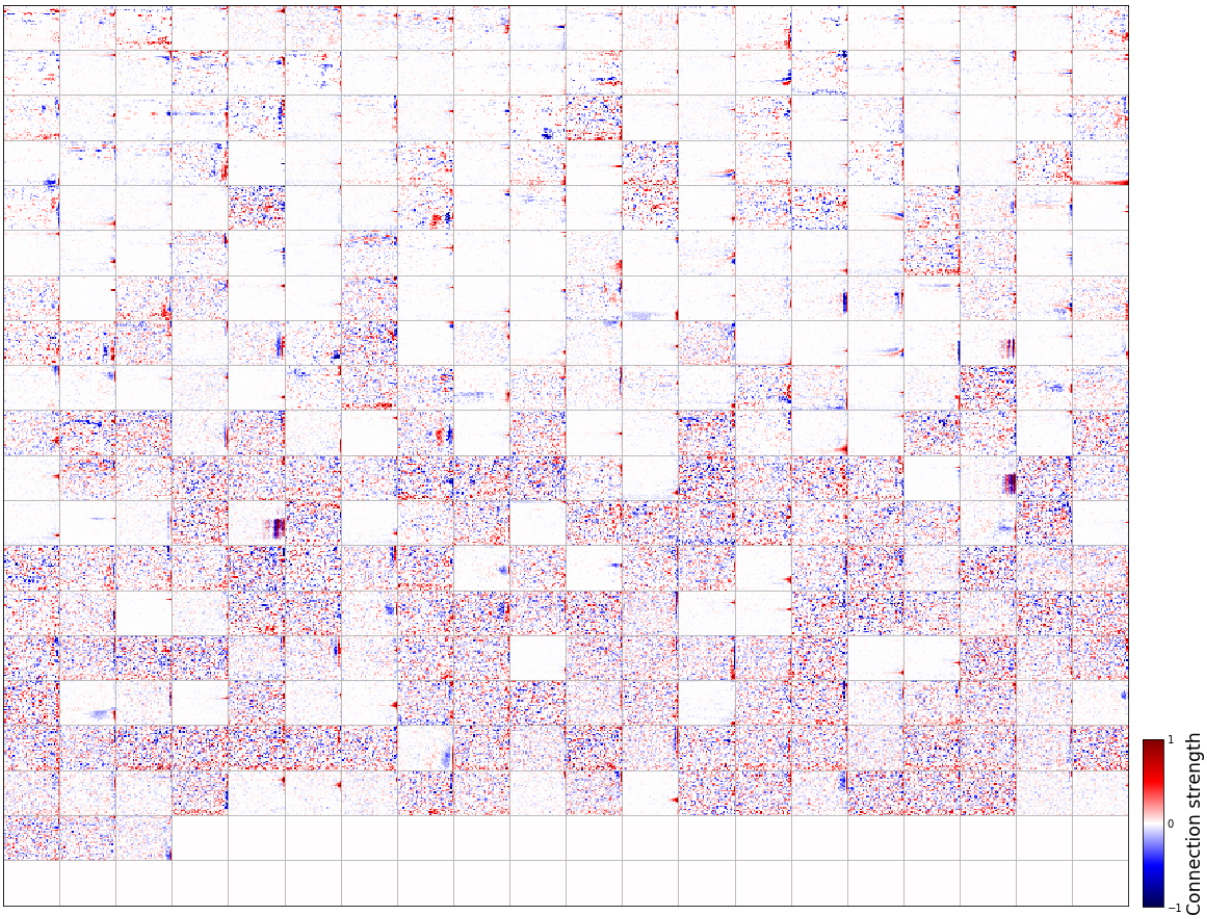
Supplementary Figure 2.53 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength 10^{-2}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



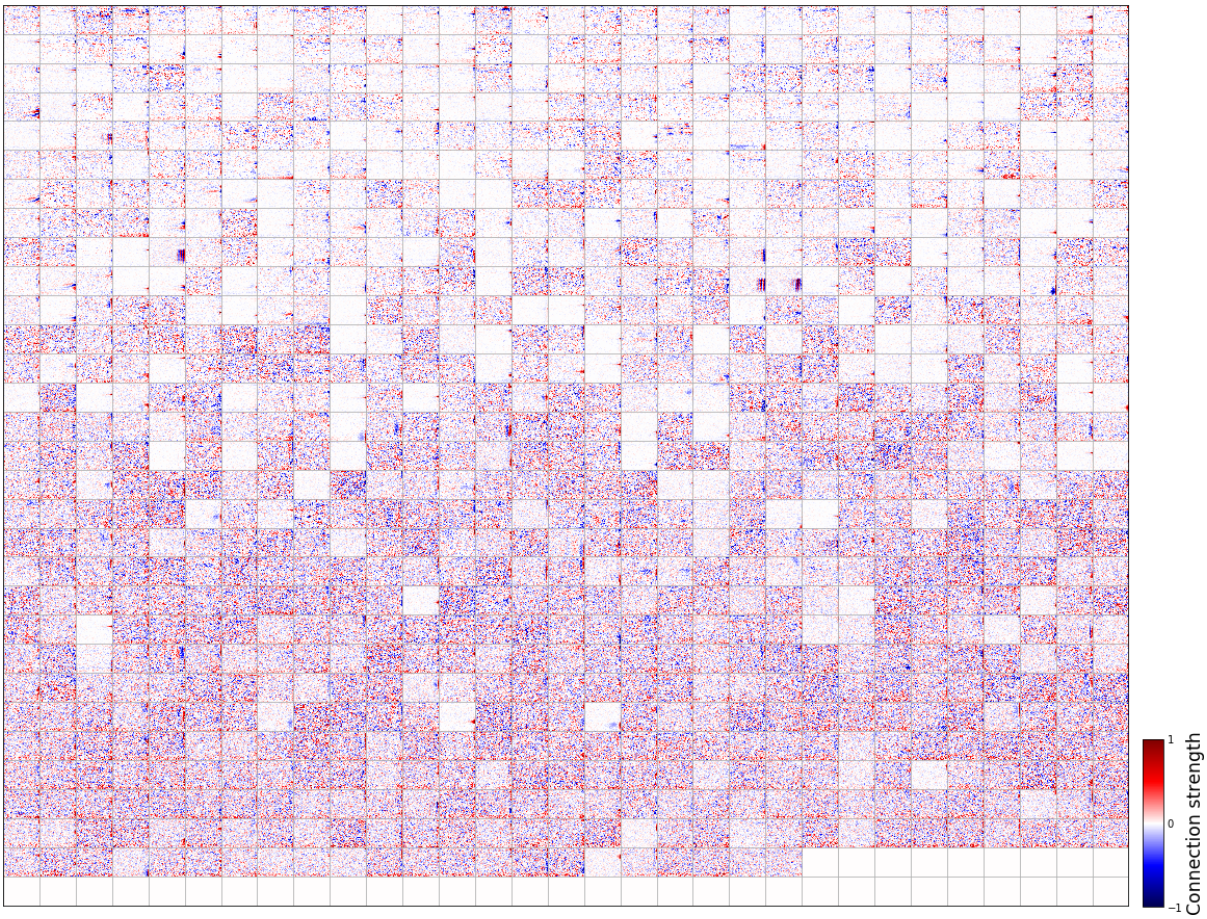
Supplementary Figure 2.54 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength $10^{-2.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



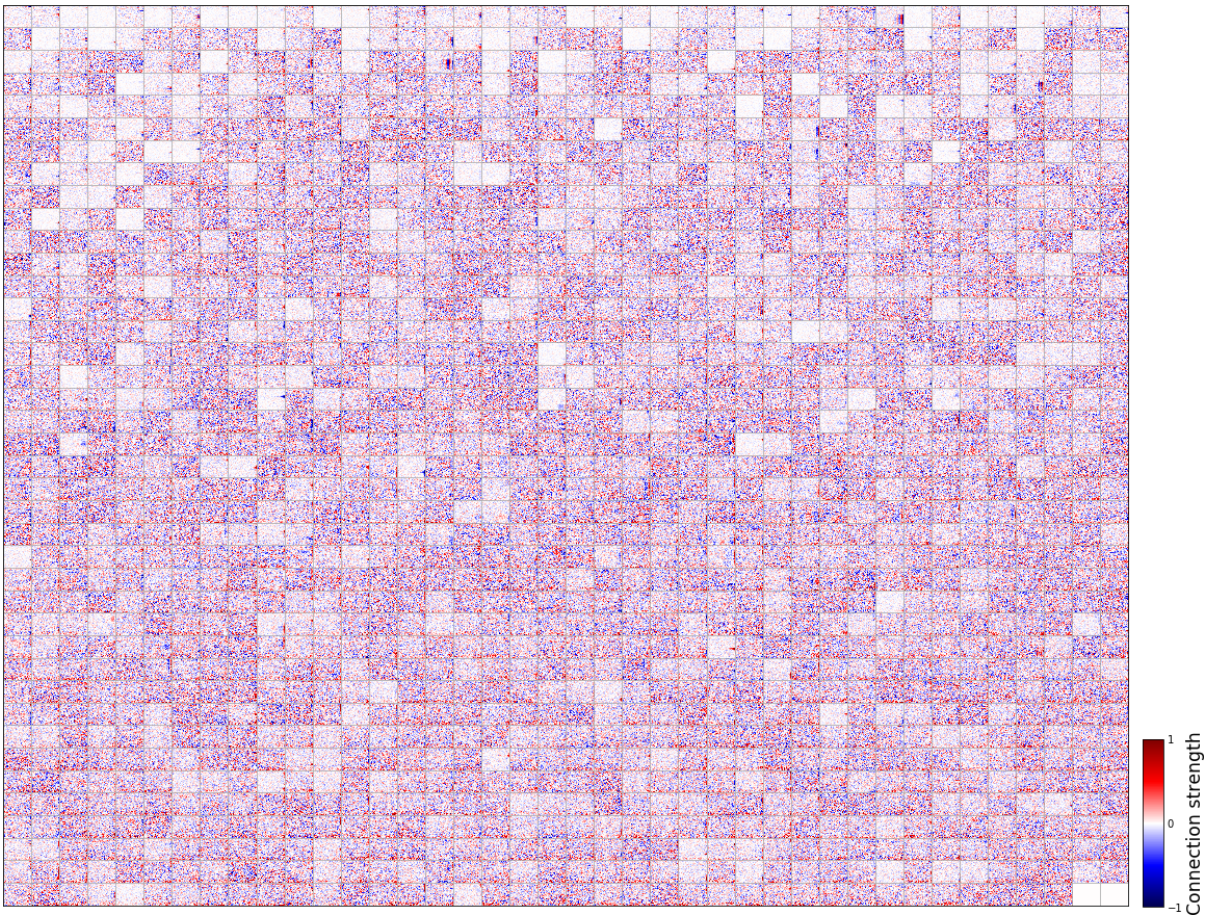
Supplementary Figure 2.55 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength 10^{-3}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



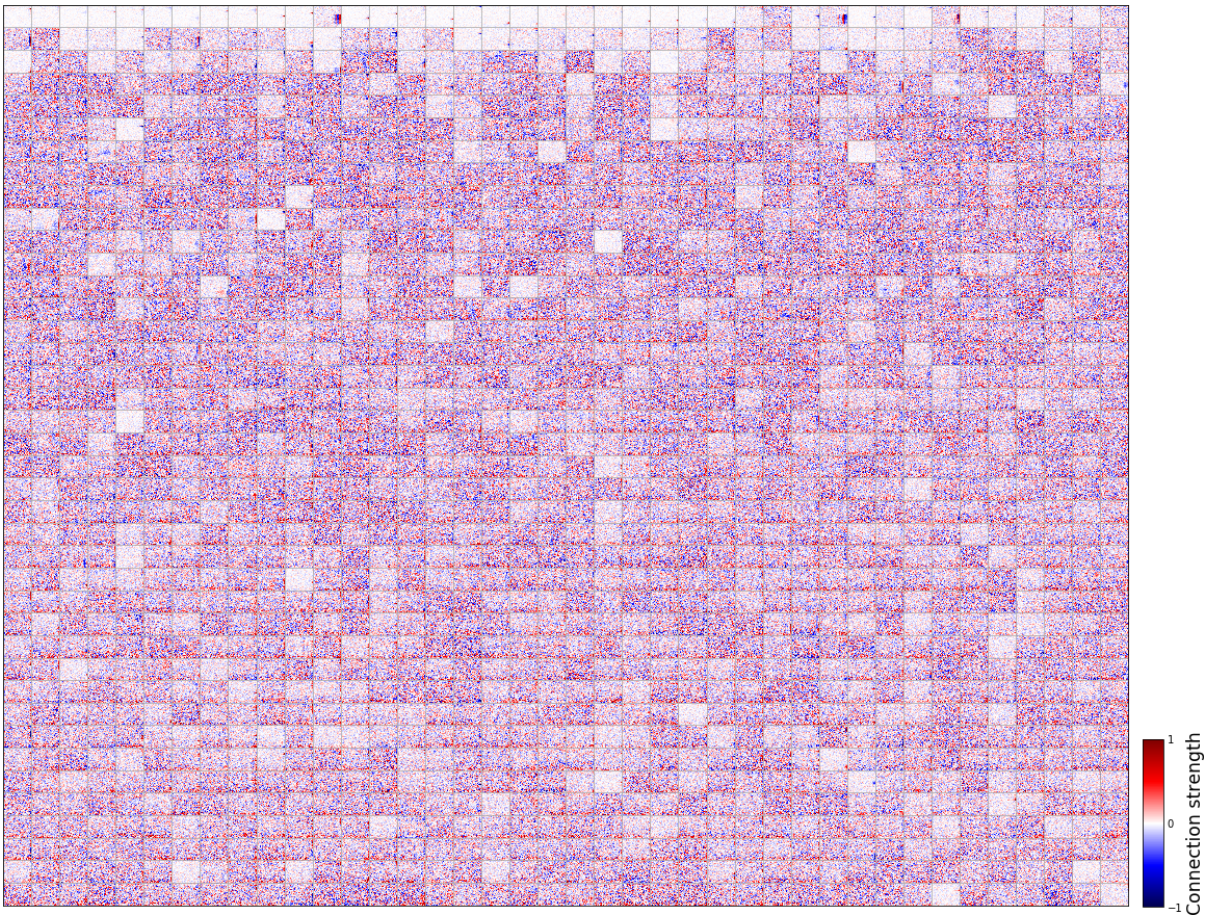
Supplementary Figure 2.56 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength 10^{-4}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



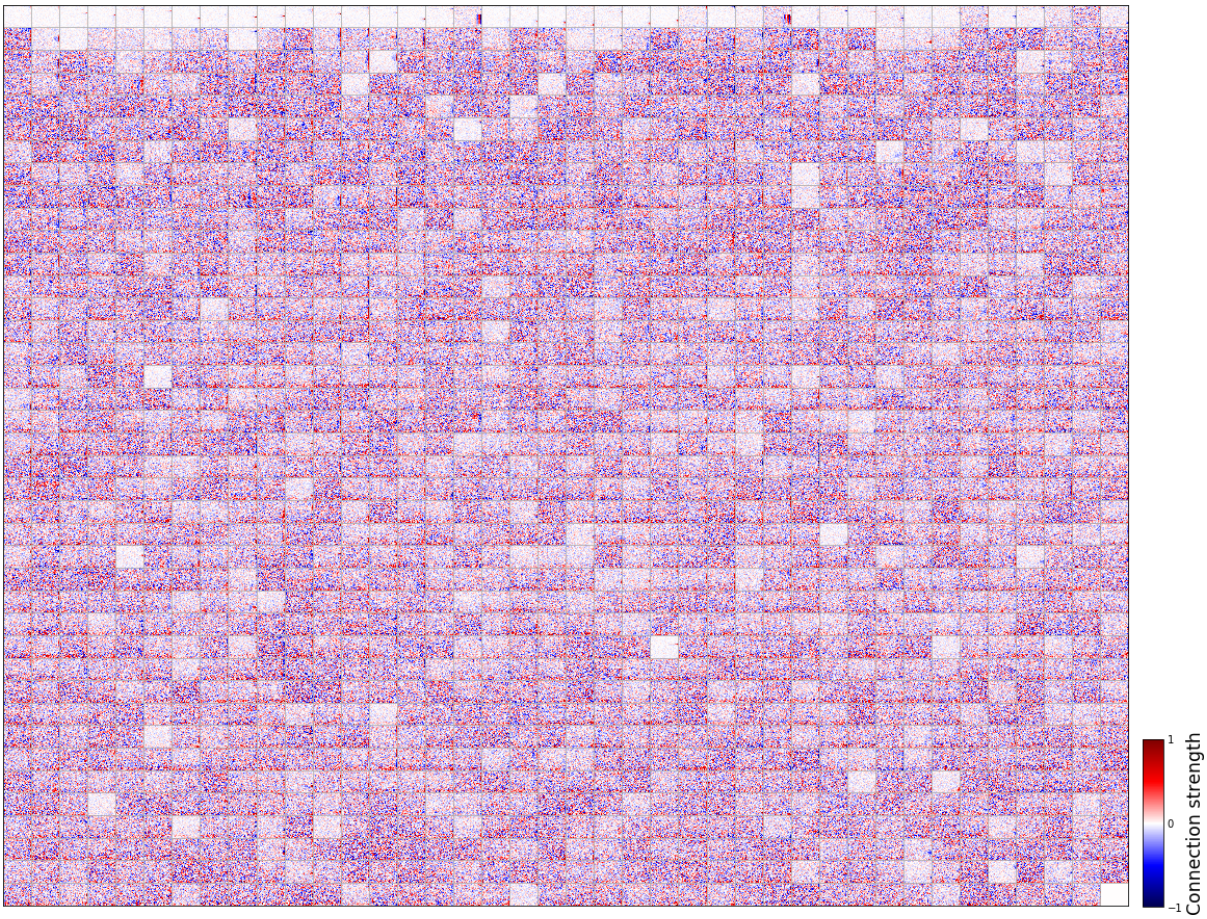
Supplementary Figure 2.57 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength $10^{-4.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



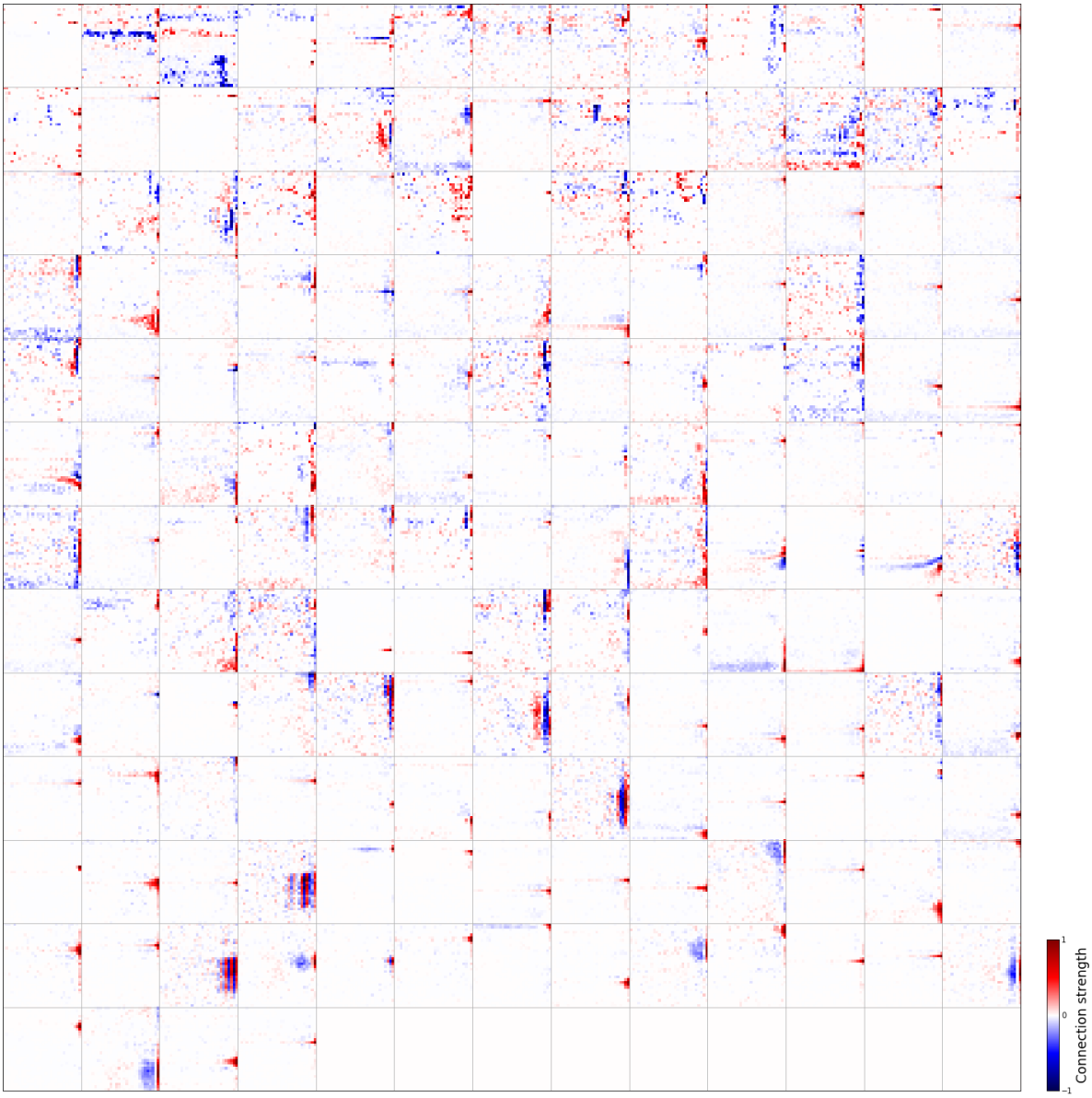
Supplementary Figure 2.58 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength 10^{-5}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



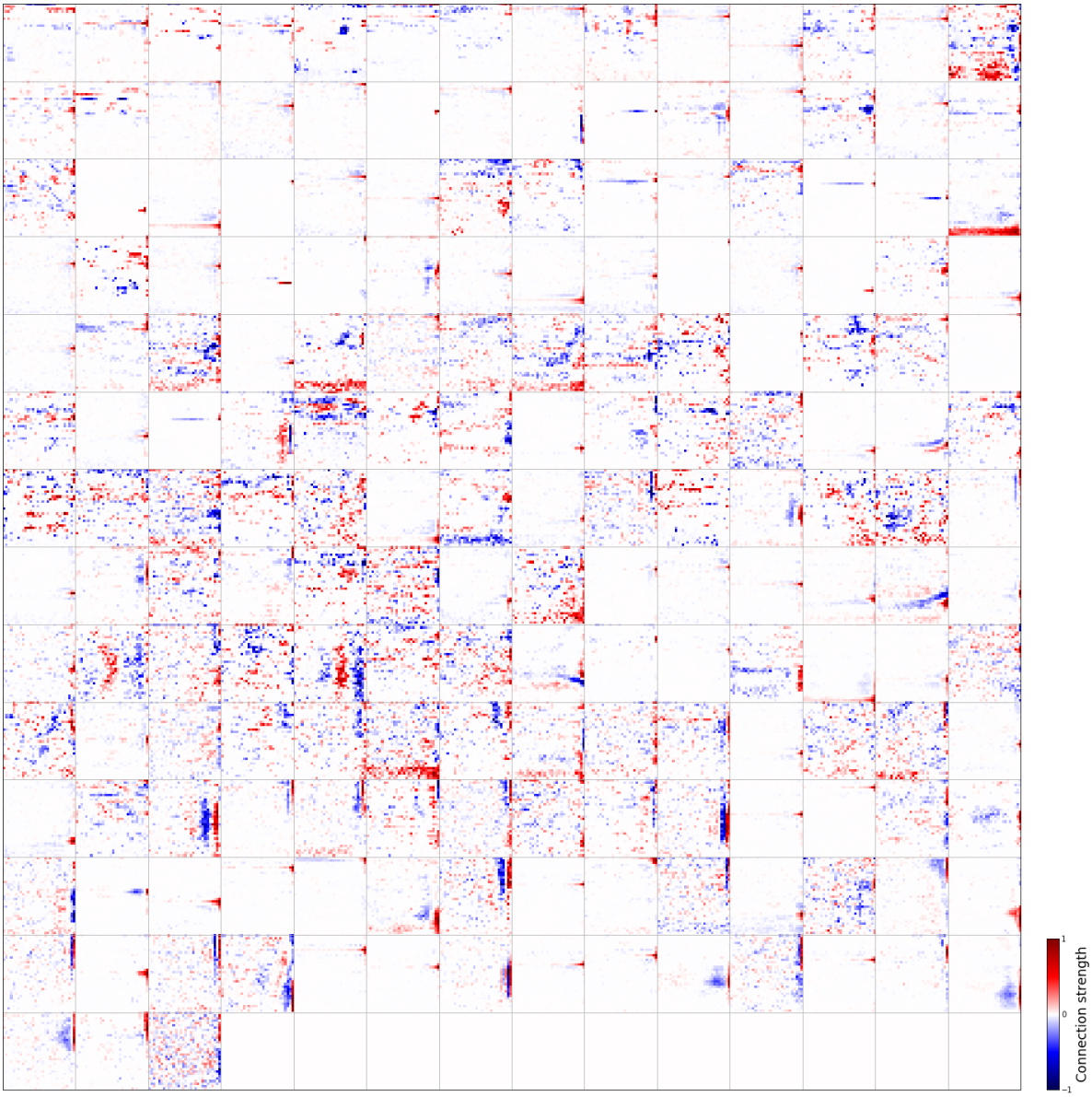
Supplementary Figure 2.59 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength $10^{-5.5}$. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



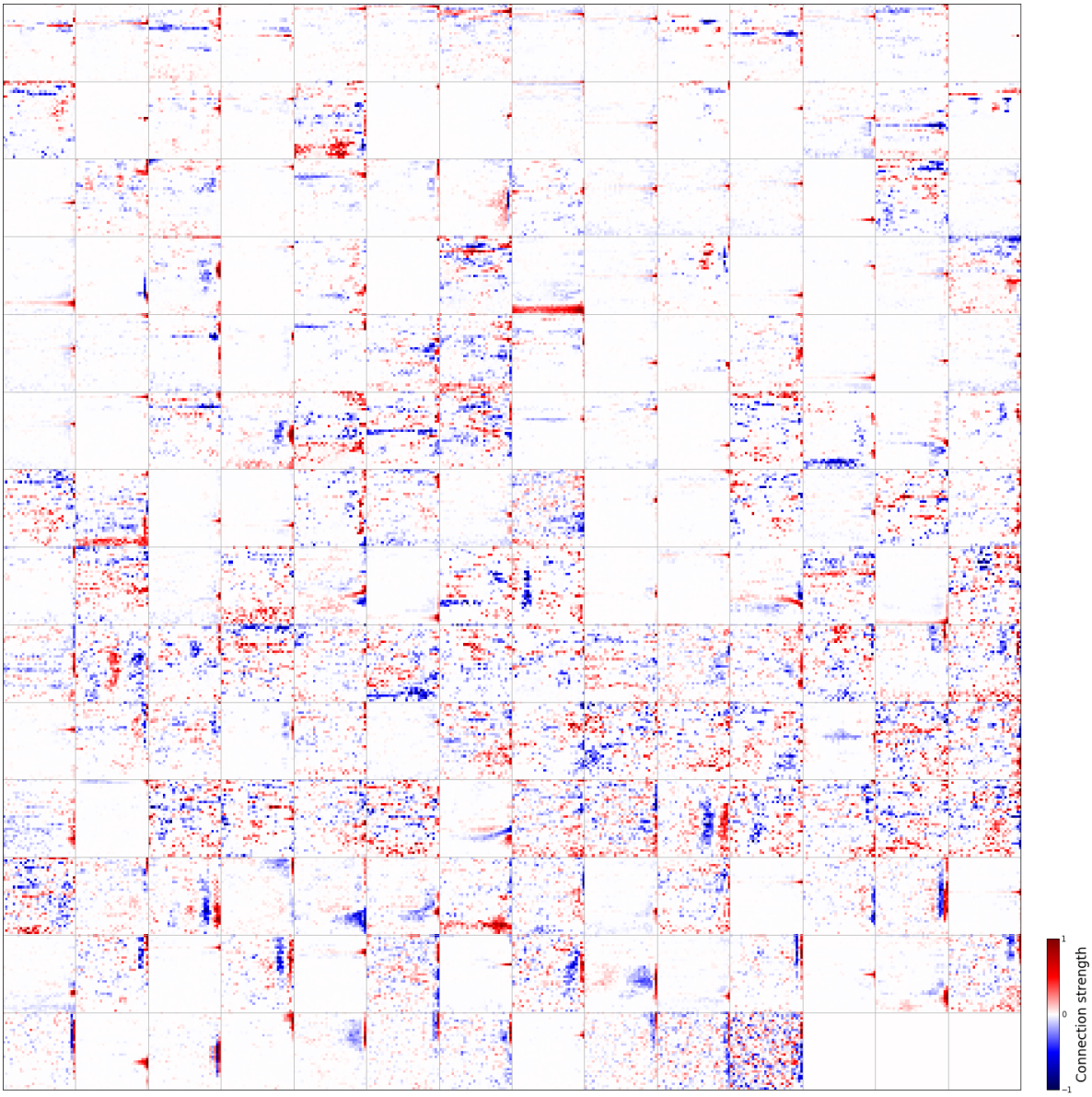
Supplementary Figure 2.60 | Full set of auditory RFs of the temporal prediction model trained with 1600 hidden units and regularization strength 10^{-6}. Most hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving only a few active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time in msec and the y-axis denoting frequency in kHz. The scale of the axes is the same as that shown in Figure 2.3.



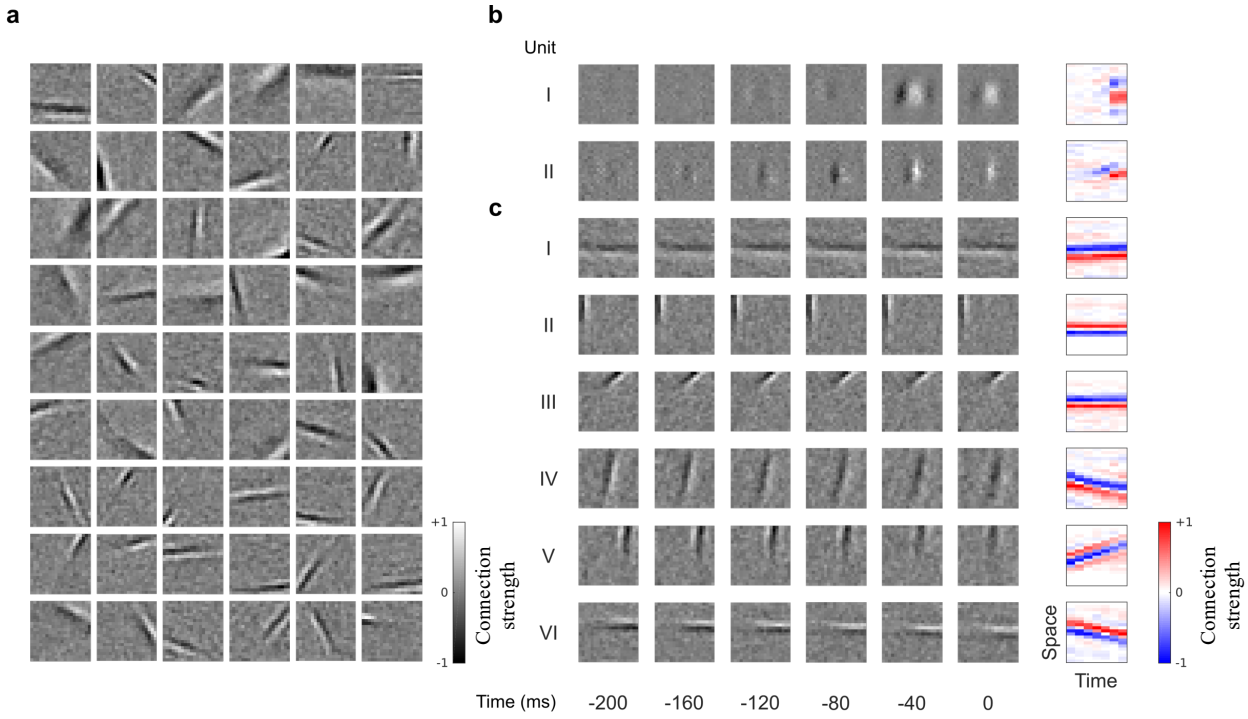
Supplementary Figure 2.61 | Full set of auditory RFs of the temporal prediction model units when network is trained to predict one time-step into the future. Units were obtained by training the model with 1600 hidden units on auditory inputs. The hidden unit number and L_1weight regularization strength ($10^{-5.5}$) was chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving 160 active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time and the y-axis denoting frequency. The scale of the axes is the same as is shown in Figure 2.3.



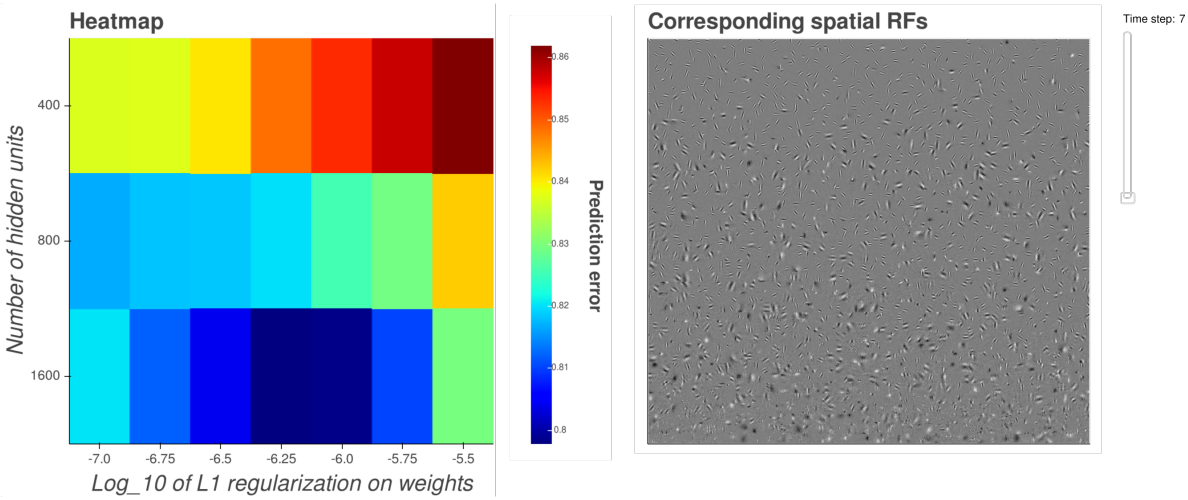
Supplementary Figure 2.62 | Full set of auditory RFs of the temporal prediction model units when network is trained to predict six time-steps into the future. Units were obtained by training the model with 1600 hidden units on auditory inputs. The hidden unit number and L_1weight regularization strength ($10^{-5.5}$) was chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving 185 active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time and the y-axis denoting frequency. The scale of the axes is the same as is shown in Figure 2.3.



Supplementary Figure 2.63tt | Full set of auditory RFs of the temporal prediction model units when network is trained to predict one time-step into the future. Units were obtained by training the model with 1600 hidden units on auditory inputs. The hidden unit number and L_1weight regularization strength ($10^{-5.5}$) was chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Most hidden units' weight matrices decayed to near zero during training (due to the L_1regularization), leaving 193 active units. Inactive units were excluded from analysis and are not shown. Red – excitation, blue - inhibition. Each box shows the RF of a single model unit, with the x-axis denoting time and the y-axis denoting frequency. The scale of the axes is the same as is shown in Figure 2.3.



Supplementary Figure 3.1 | Visual RFs for real V1 neurons and sparse coding model units. a, Model units are the same as those used in Fig. 3.5. Example spatial RFs of randomly selected units at their best time step. **b-c,** Example 3D and corresponding 2D spatiotemporal RFs at most recent 6 time steps of (**b**) (I, space-time separable, and II, space-time inseparable) real[23] V1 neurons and (**c**) (I-III, space-time separable, and IV-VI, space-time inseparable) sparse coding model units.



Supplementary Figure 3.2 | Interactive figure exploring the relationship between the strength of L_1 regularization on the network weights and the structure of the RFs the network produces when the network is trained on visual inputs.

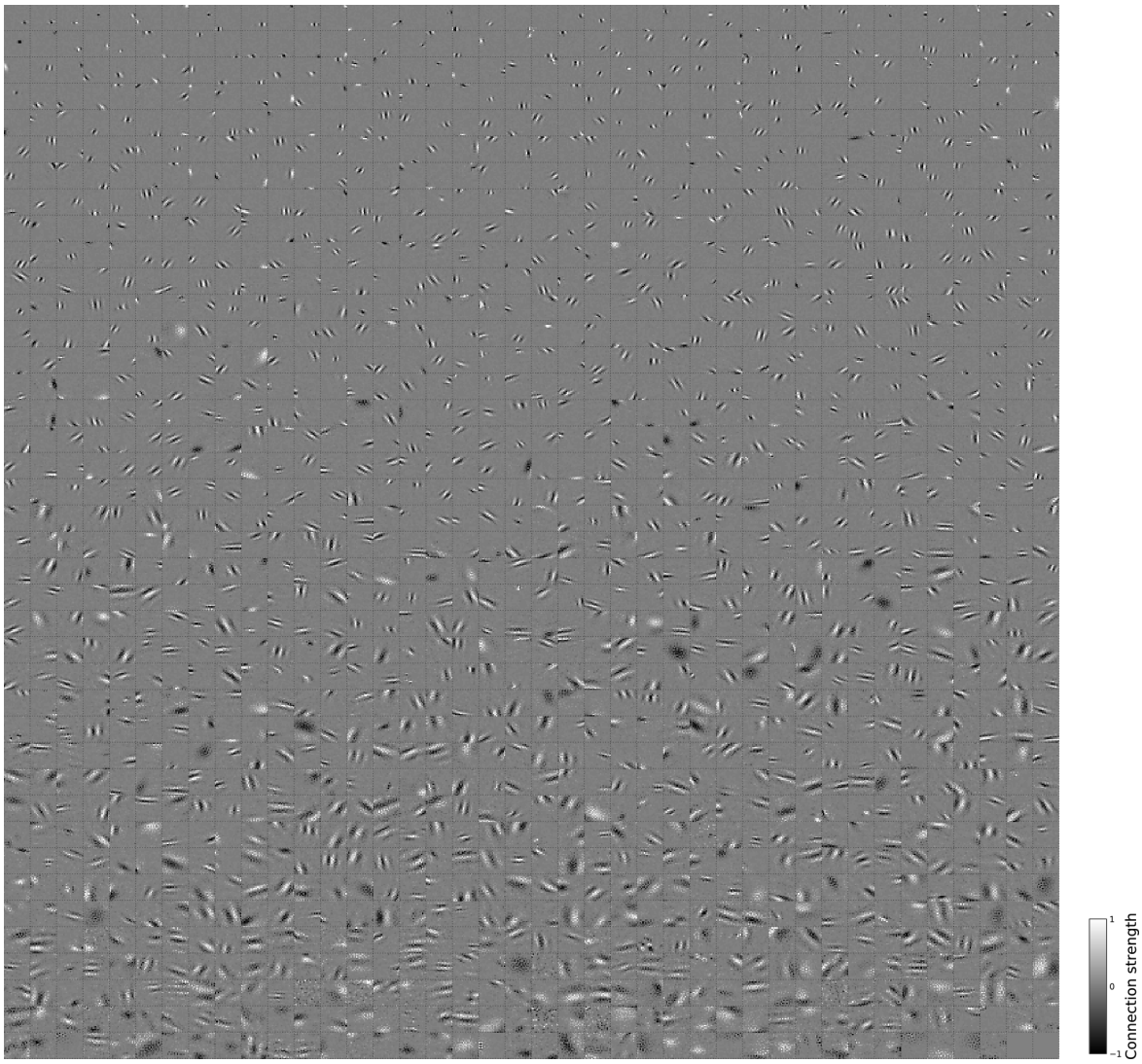
The interactive version of this figure can be found at:

https://yossing.github.io/temporal_prediction_model/figures/interactive_supplementary_figures.html

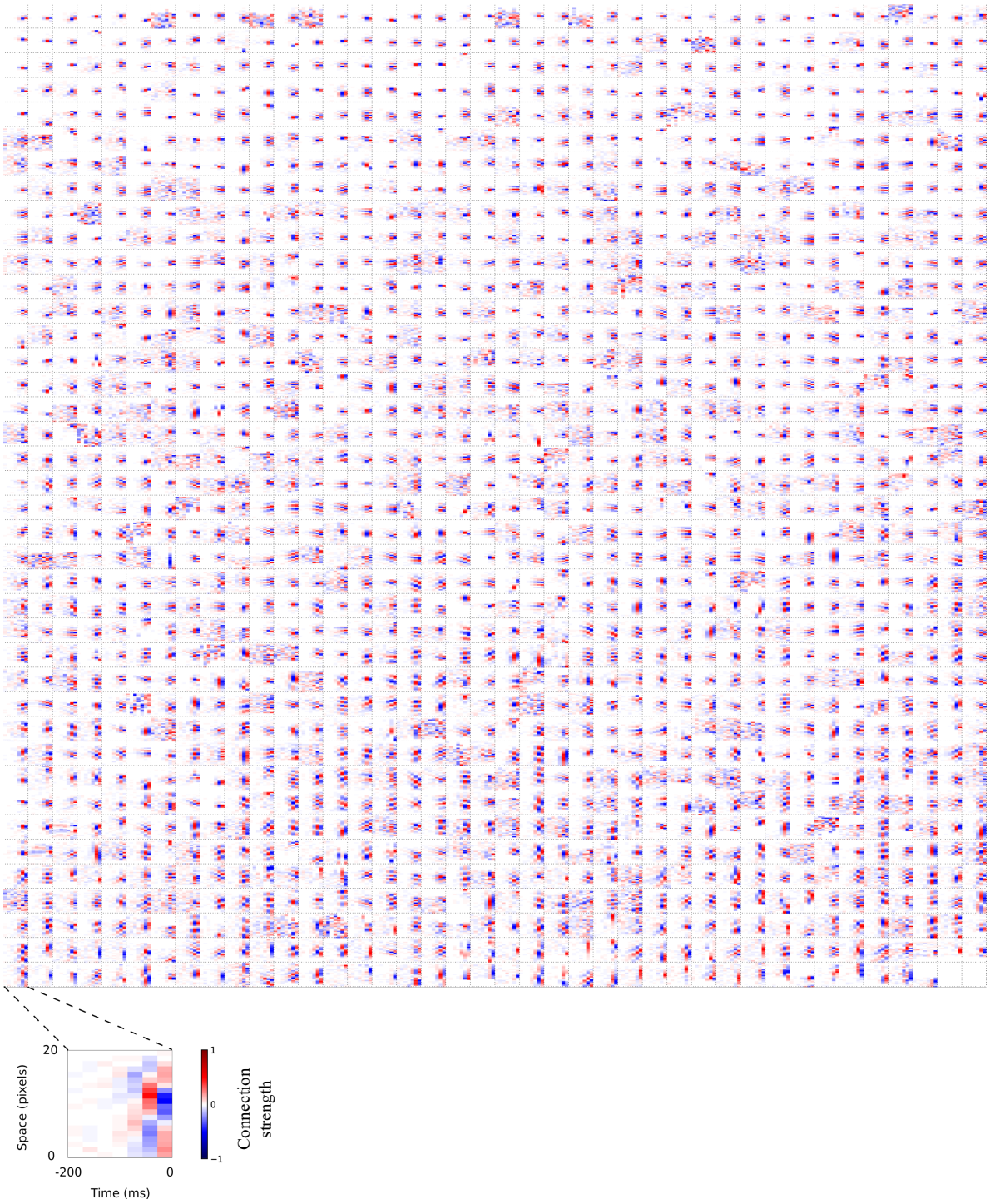
The left panel shows the performance of the network with the hyperparameter settings specified on the x and y axes. The x axis signifies the strength of L_1 regularization placed on the weights of the network during training. The y axis signifies the number of hidden units in the network. The colour represents the performance of model measured by the reconstruction error on held out validation set at the end of training.

How to interact with the figure:

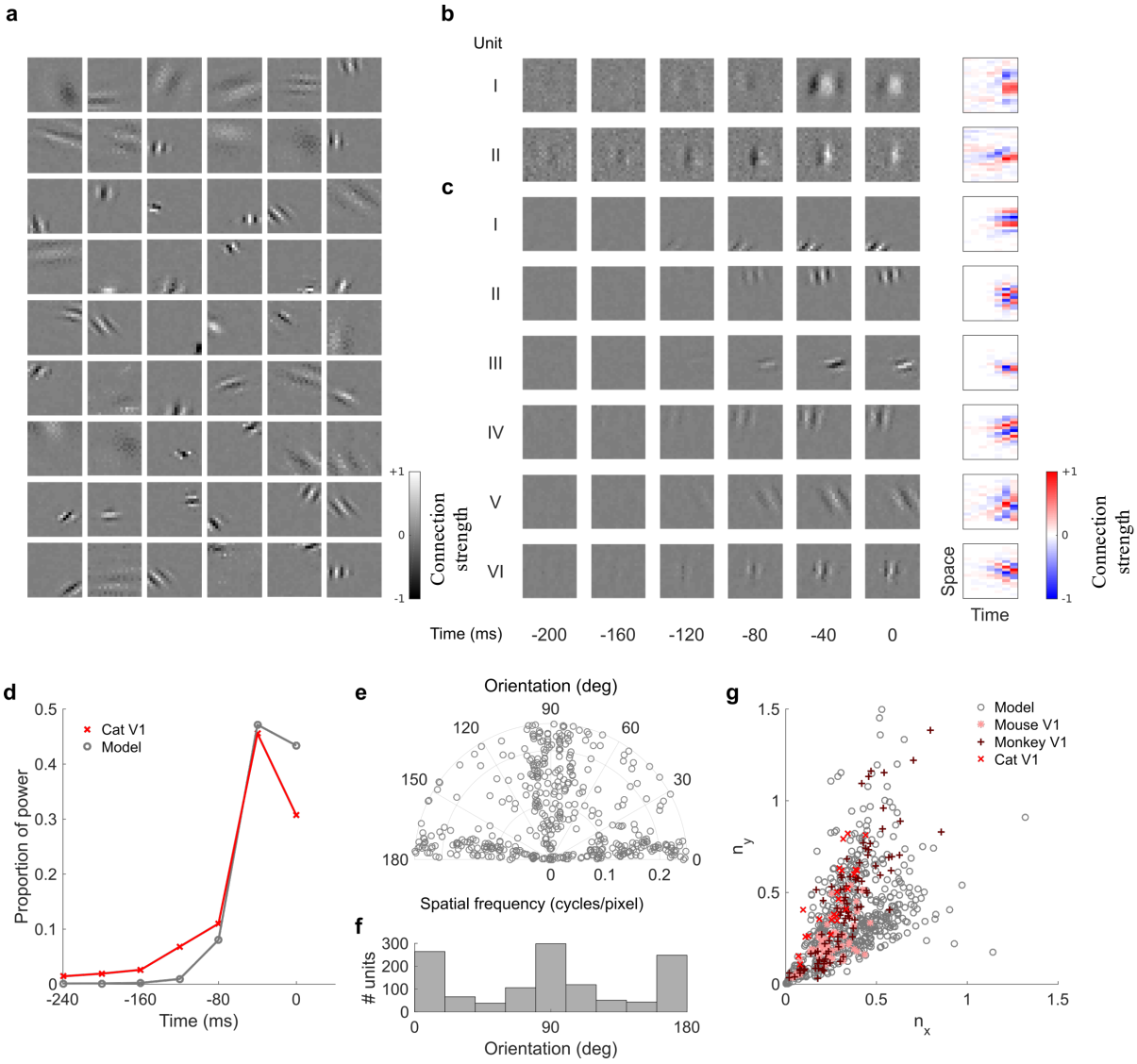
Hover over a point in the left panel to show the corresponding spatial receptive fields of the network in the right panel. Using the settings on the right of the right hand panel, zoom, pan and reset the image to explore the shapes of the spatial receptive fields. Change the slider labelled 'time step' to change the time step of the spatial receptive fields being shown in the right hand panel. Some hidden units' weight matrices decayed to near zero during training. Inactive units were excluded from analysis and are not shown



Supplementary Figure 3.3 | Full set of visual spatial RFs of temporal prediction model units trained on visual inputs without added noise. Model units were obtained by training the model with 1600 hidden units on visual inputs. The hidden unit number and L_1 weight regularization strength (10^{-4}) was chosen because it results in the lowest MSE on the prediction task, as measured using a cross validation set. Some hidden units' weight matrices decayed to near zero during training (due to the L_1 regularization), leaving 1585 active units, which were included in analysis. Inactive units were excluded from analysis. Each square represents the spatial RF of a single unit, shown at its best time step. The best time step was determined by selecting the time step for which the power (sum of squares) of the RF was greatest. White – excitation, black - inhibition.

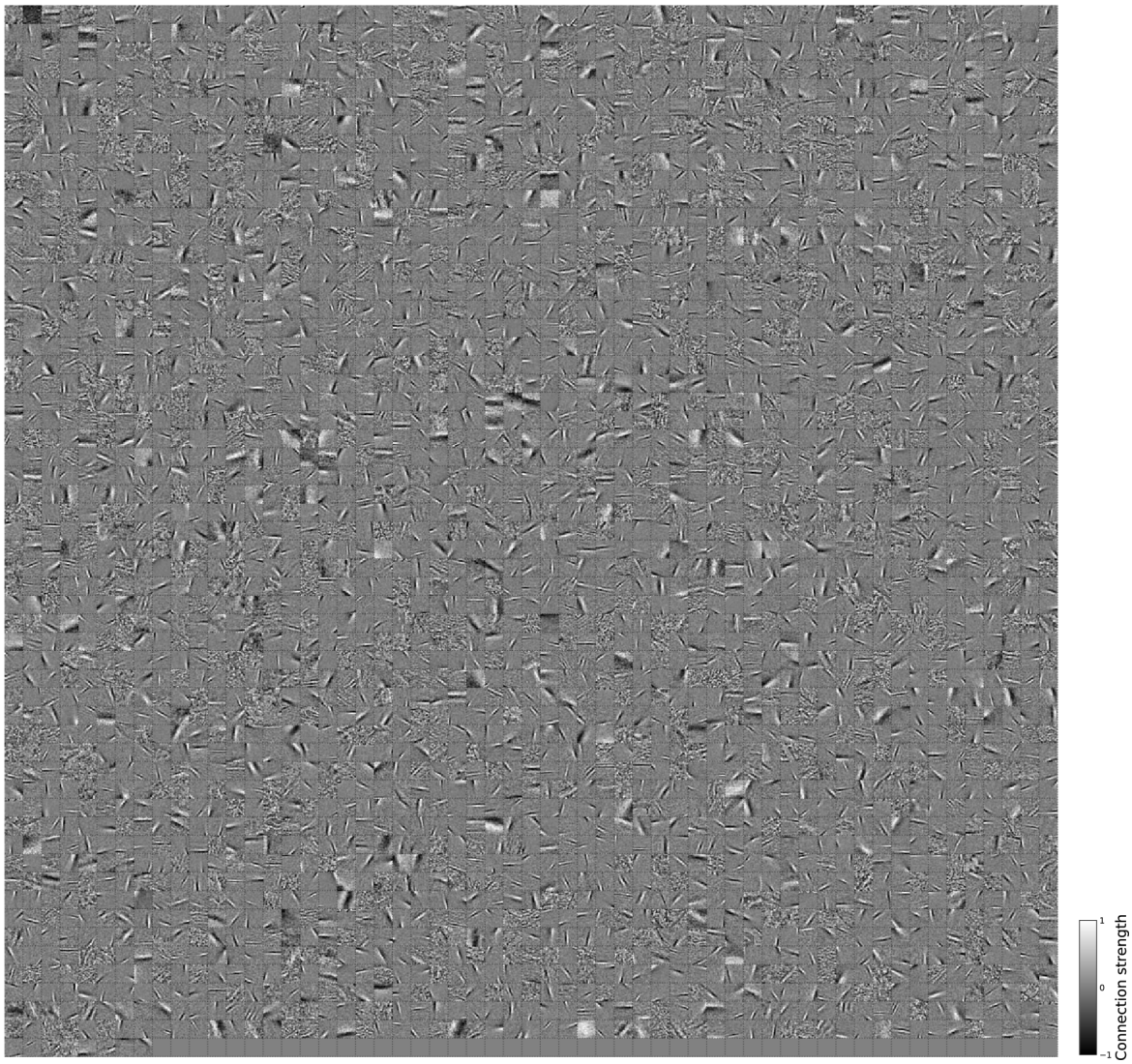


Supplementary Figure 3.4 | 2D (space-time) visual spatiotemporal RFs of temporal prediction model units trained on visual inputs without added noise. Obtained from the same units shown in Supp. Fig. 3.3 using methods outlined in Fig 3.4. Red – excitation, blue - inhibition. Inset shows axes.

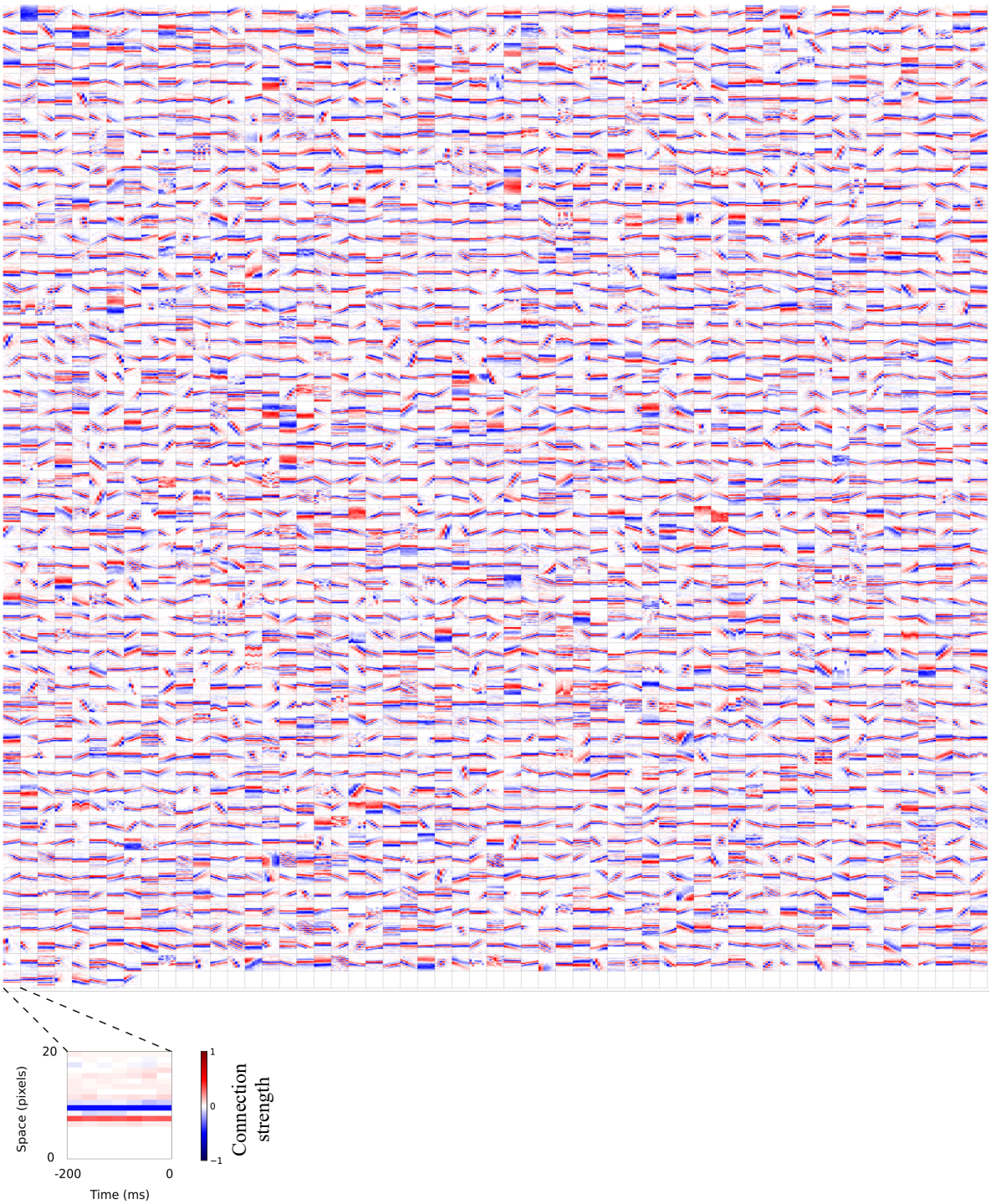


Supplementary Figure 3.5 | Visual RFs and population measures for real V1 neurons and temporal prediction model units trained on visual inputs without added noise. a, Model units are the same as those used in Sup. Fig. 3.3,3.4. Example spatial RFs of randomly selected units at their best time step. **b-c,** Example 3D and corresponding 2D spatiotemporal RFs at most recent 6 time steps of (I, space-time separable, and II, space-time inseparable) real (Ohzawa et al., 1996) V1 neurons and (**c**) (I-III, space-time separable, and IV-VI, space-time inseparable) sparse coding model units. **d**, Proportion of power (sum of squared weights over space and averaged across units) in each time step, for real and model populations. **e**, Joint distribution of spatial frequency and orientation tuning for population of model units. **f,** Distribution of orientation tuning for population of model units. **g**, Distribution of RF shapes for real neurons (cat (Jones and Palmer, 1987b), mouse (Niell and Stryker, 2008) and monkey (Ringach, 2002)) and model units. For **e-g**, only units that could be well approximated by Gabor functions (n = 1246 units; see Methods) were included in the analysis. Of these, only model units that were space-time separable (n = 569) are shown in **f** to be comparable with the neuronal data (Ringach, 2002). The addition of noise only leads to subtle changes in the

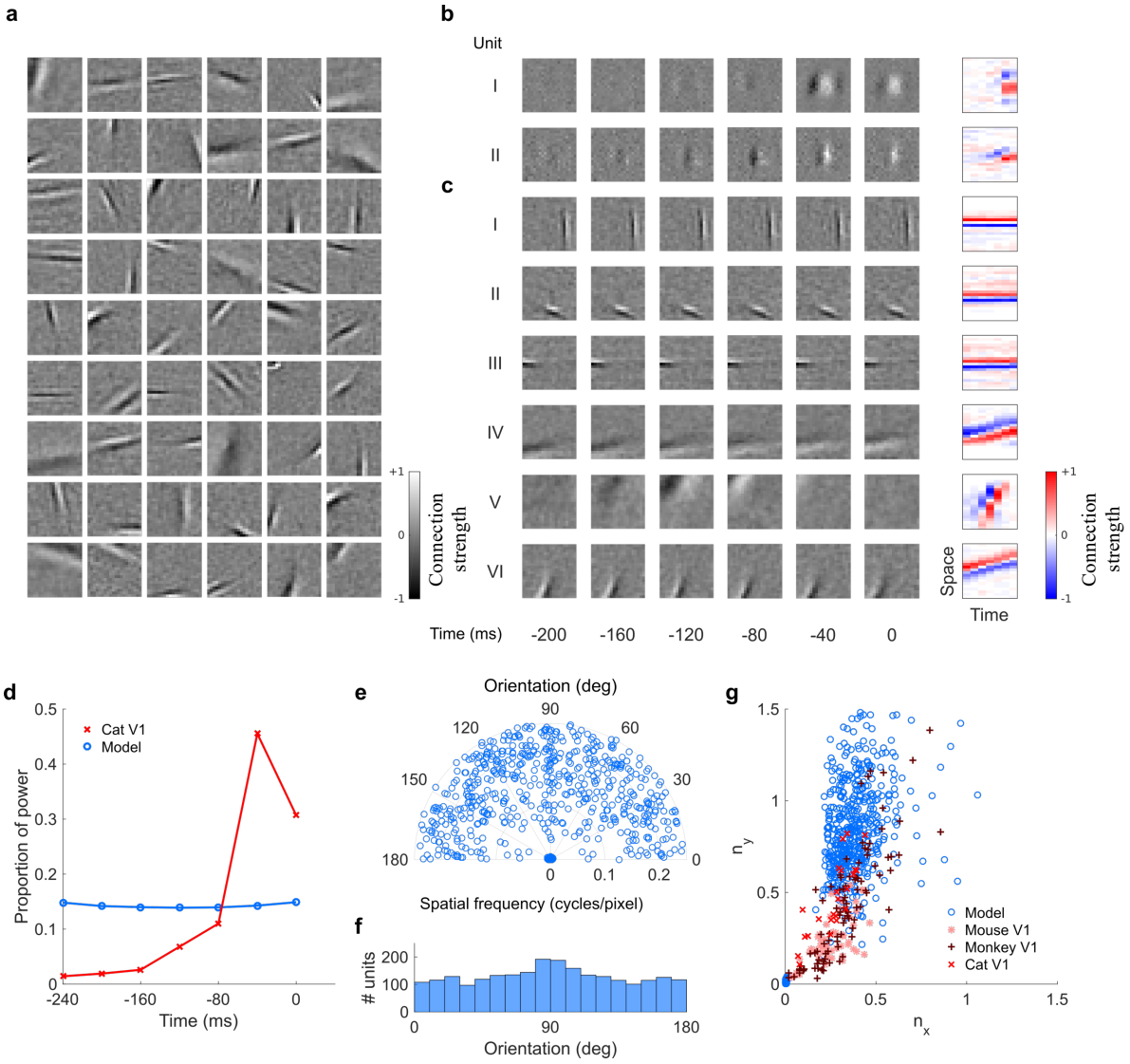
RFs; most apparently, there are more units with RFs comprising multiple short subfields (forming an increased number of points towards the lower right quadrant of **g)** than is seen in the case when noise is used.



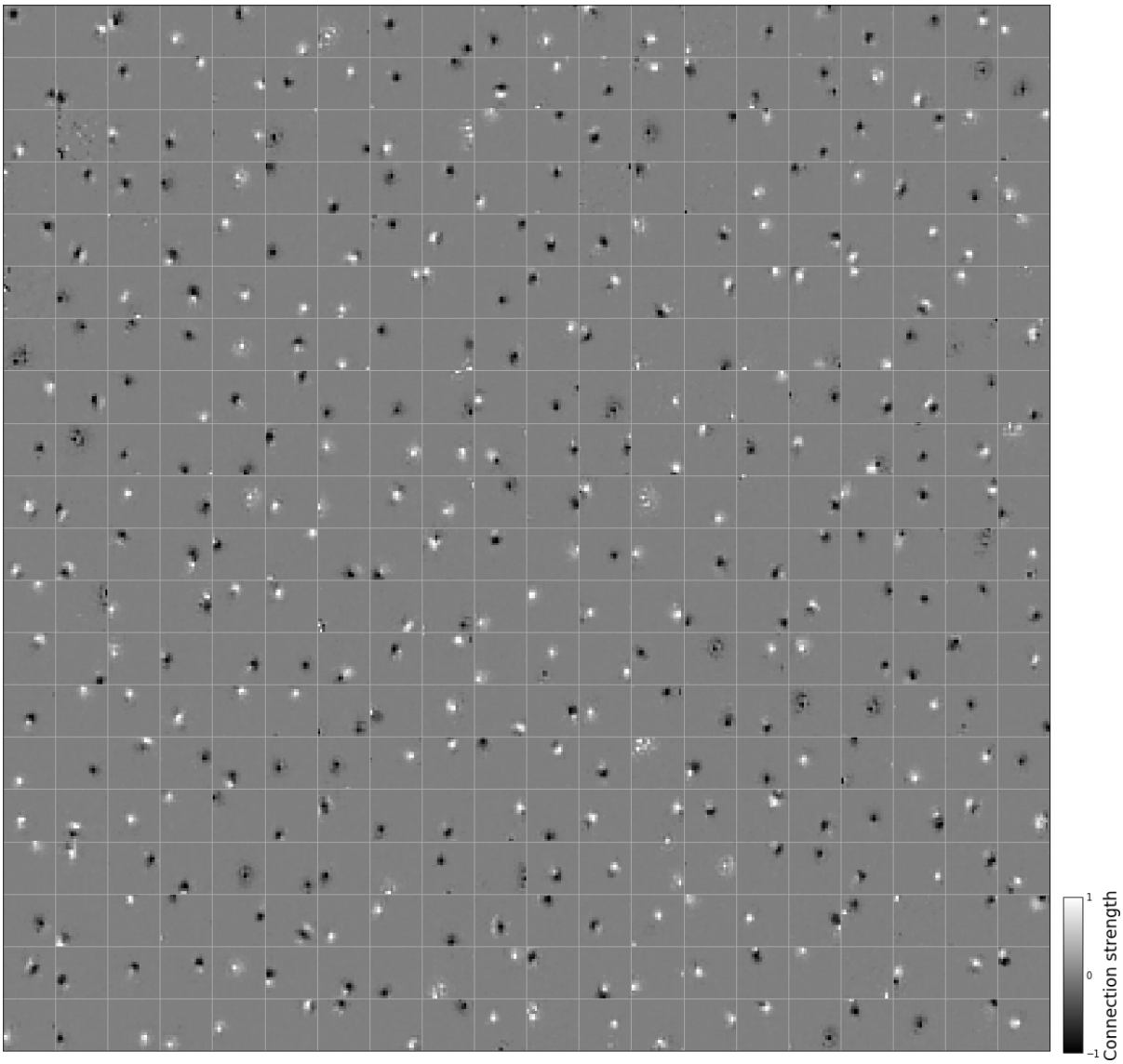
Supplementary Figure 3.6 | Full set of visual spatial RFs of sparse coding model units trained on visual inputs without added noise. Model units were obtained by training the sparse coding model with 3200 units on identical visual inputs used to train the temporal prediction model. The model configuration (3200 units, L_1 sparsity strength of $10^{0.5}$ on the unit activities) was chosen because it resulted in the RFs that look most like the RFs of V1 simple cells as determined by visual inspection. Each square represents the spatial RF of a single unit, shown at its best time step. The best time step was determined by selecting the time step for which the power (sum of squares) of the RF was greatest. White – excitation, black - inhibition.



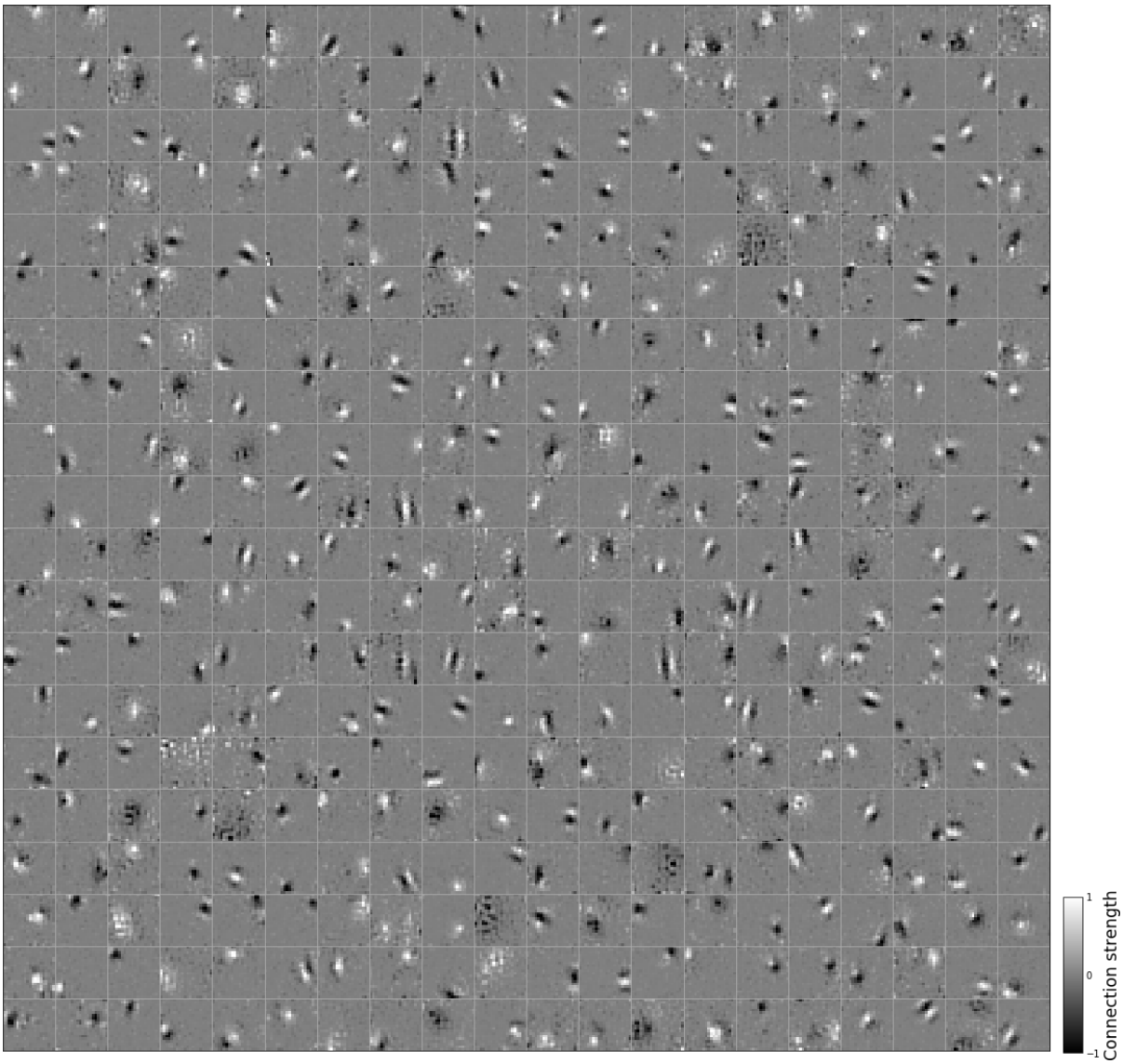
Supplementary Figure 3.7 | 2D (space-time) visual spatiotemporal RFs of sparse coding model units trained on visual inputs without added noise. Obtained from the same units shown in Supp. Fig. 3.6 using methods outlined in Fig 3.4. Red – excitation, blue - inhibition. Inset shows axes.



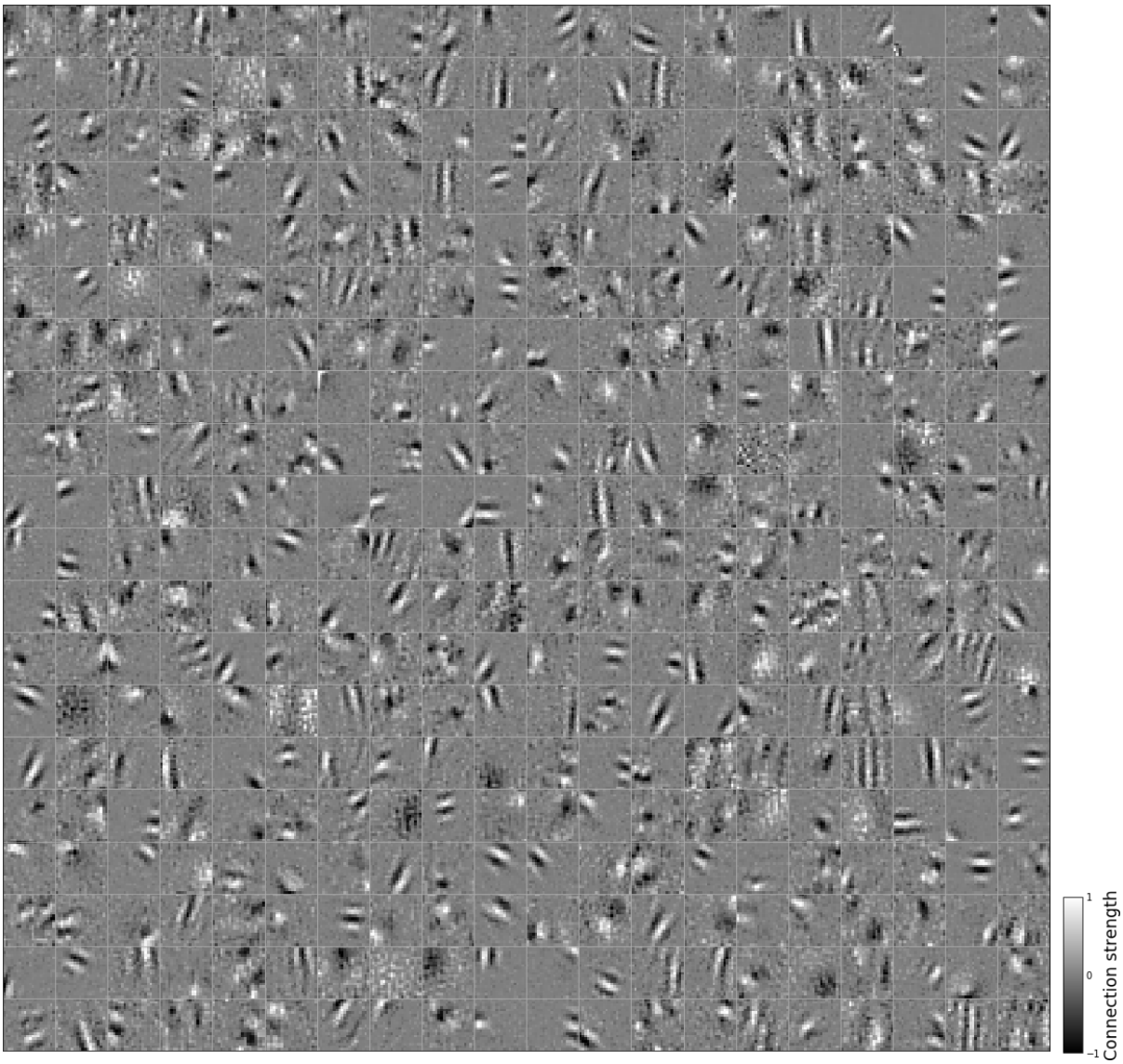
Supplementary Figure 3.8 | Visual RFs and population measures for real V1 neurons and sparse coding model units trained on visual inputs without added noise. a, Model units are the same as those used in Supp. Figs. 3.6,3.7. Example spatial RFs of randomly selected units at their best time step. **b-c,** Example 3D and corresponding 2D spatiotemporal RFs at most recent 6 time steps of (**b**) (I, space-time separable, and II, space-time inseparable) real (Ohzawa et al., 1996) V1 neurons and (**c**) (I-III, space-time separable, and IV-VI, space-time inseparable) sparse coding model units. **d**, Proportion of power (sum of squared weights over space and averaged across units) in each time step, for real and model populations. **e**, Joint distribution of spatial frequency and orientation tuning for population of model units. **f,** Distribution of orientation tuning for population of model units. **g**, Distribution of RF shapes for real neurons (cat (Jones and Palmer, 1987b), mouse (Niell and Stryker, 2008) and monkey (Ringach, 2002)) and model units. For **e-g**, only units that could be well approximated by Gabor functions (n = 2482 units; see Methods) were included in the analysis. Of these, only model units that were space-time separable (n = 860) are shown in **f** to be comparable with the neuronal data (Ringach, 2002).



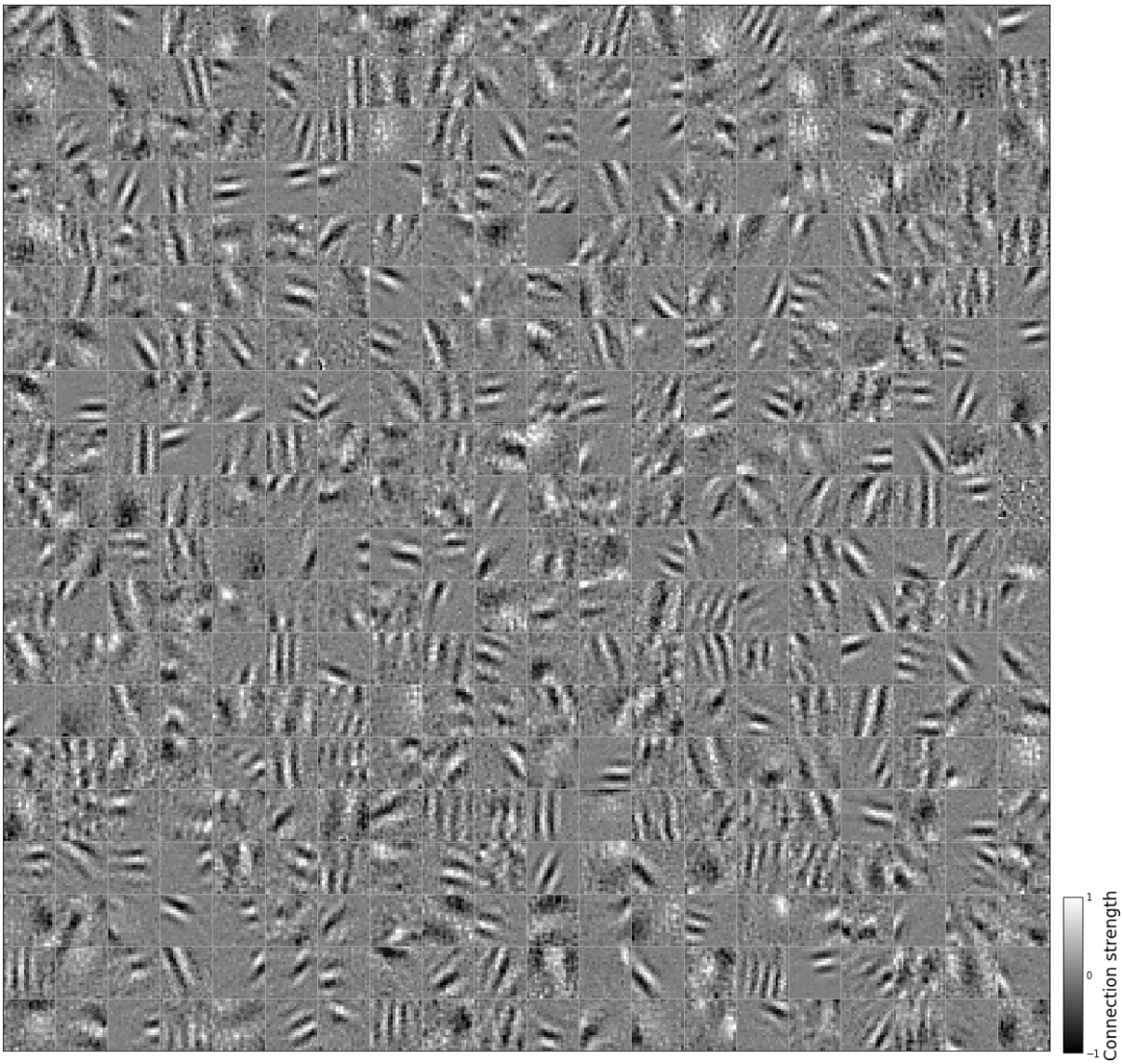
Supplementary Figure 3.9 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength ($10^{-5.5}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



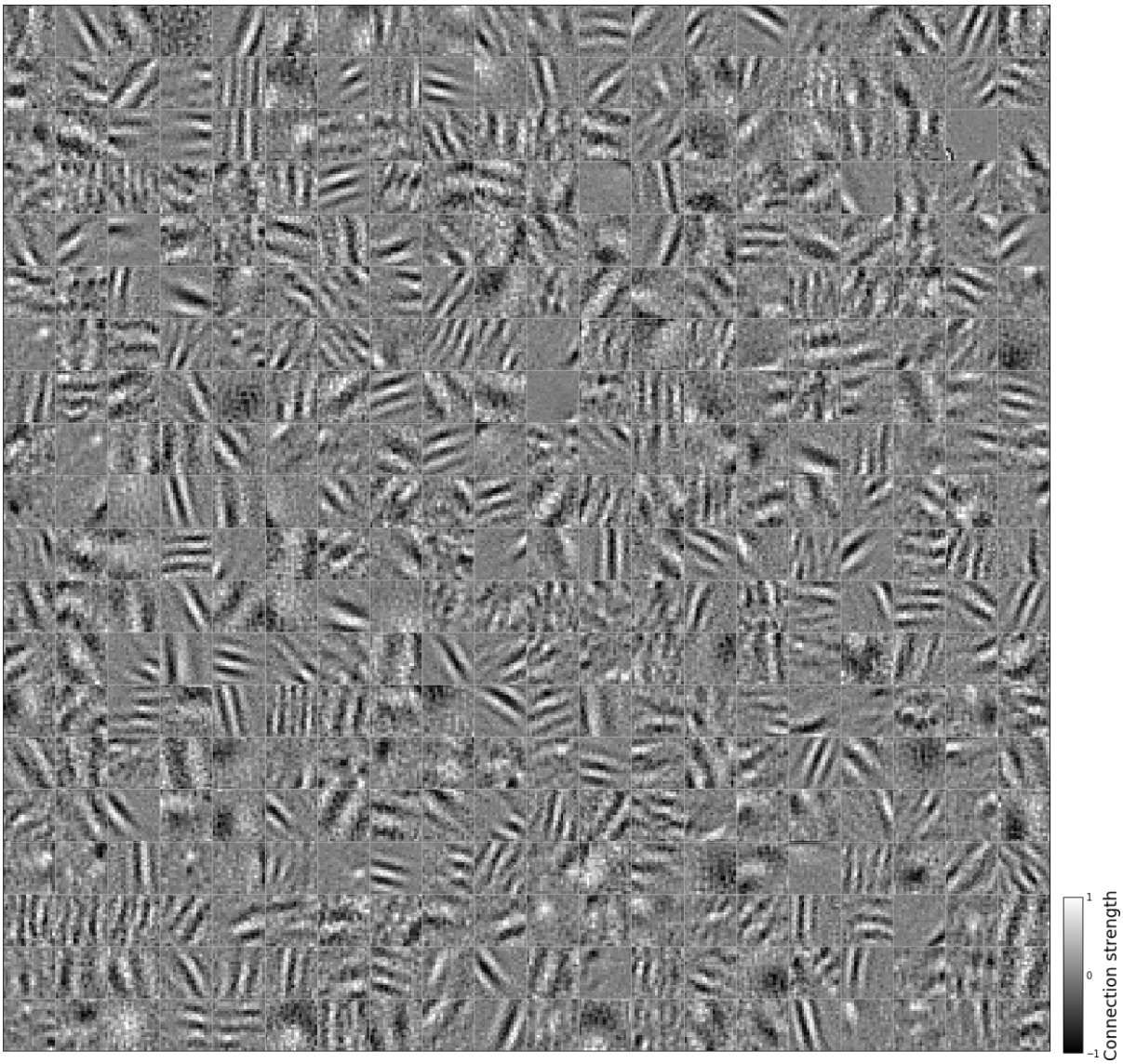
Supplementary Figure 3.10 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength ($10^{-5.75}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



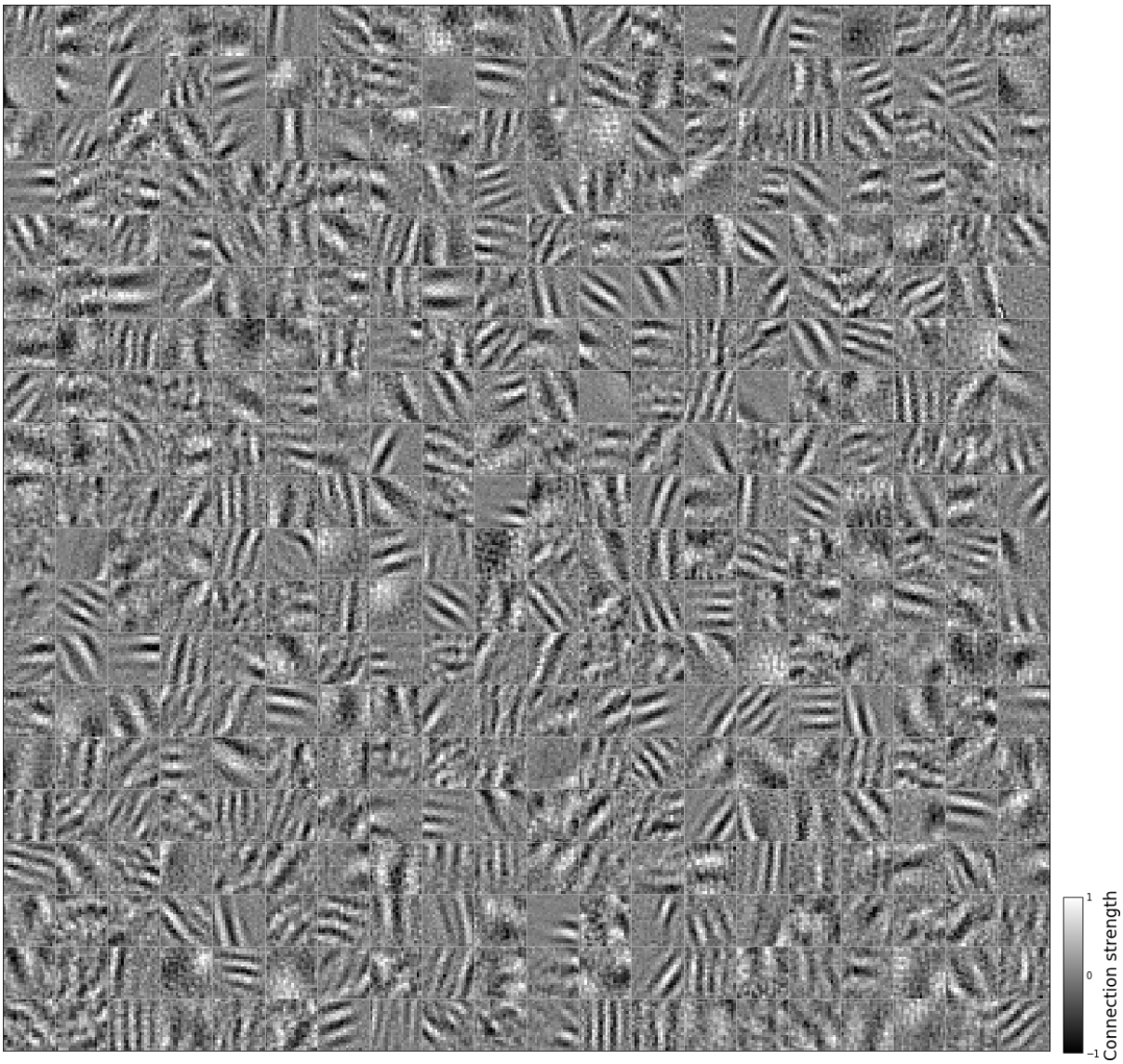
Supplementary Figure 3.11 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength (10^{-6}). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



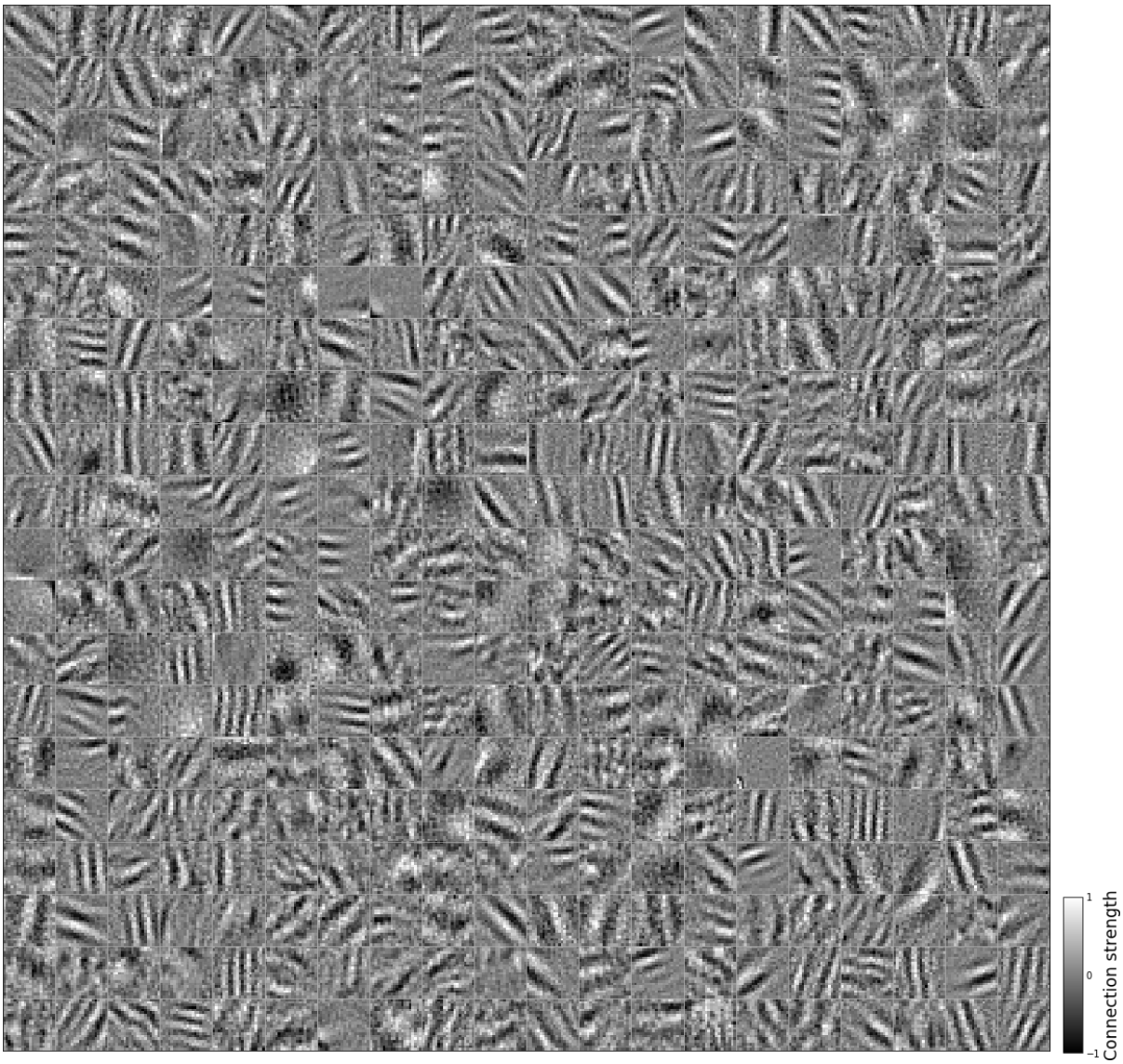
Supplementary Figure 3.12 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength ($10^{-6.25}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



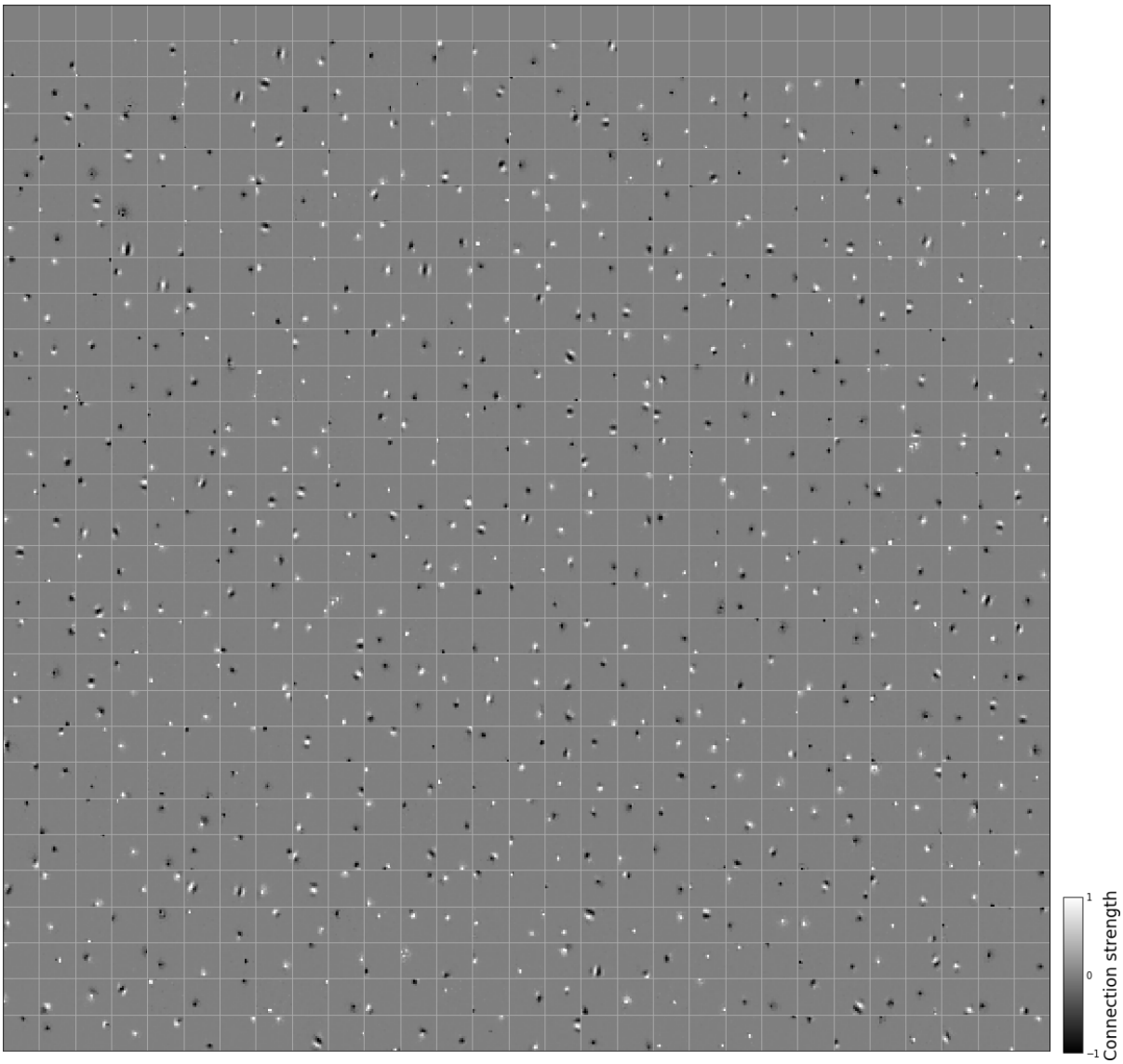
Supplementary Figure 3.13 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength ($10^{-6.5}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



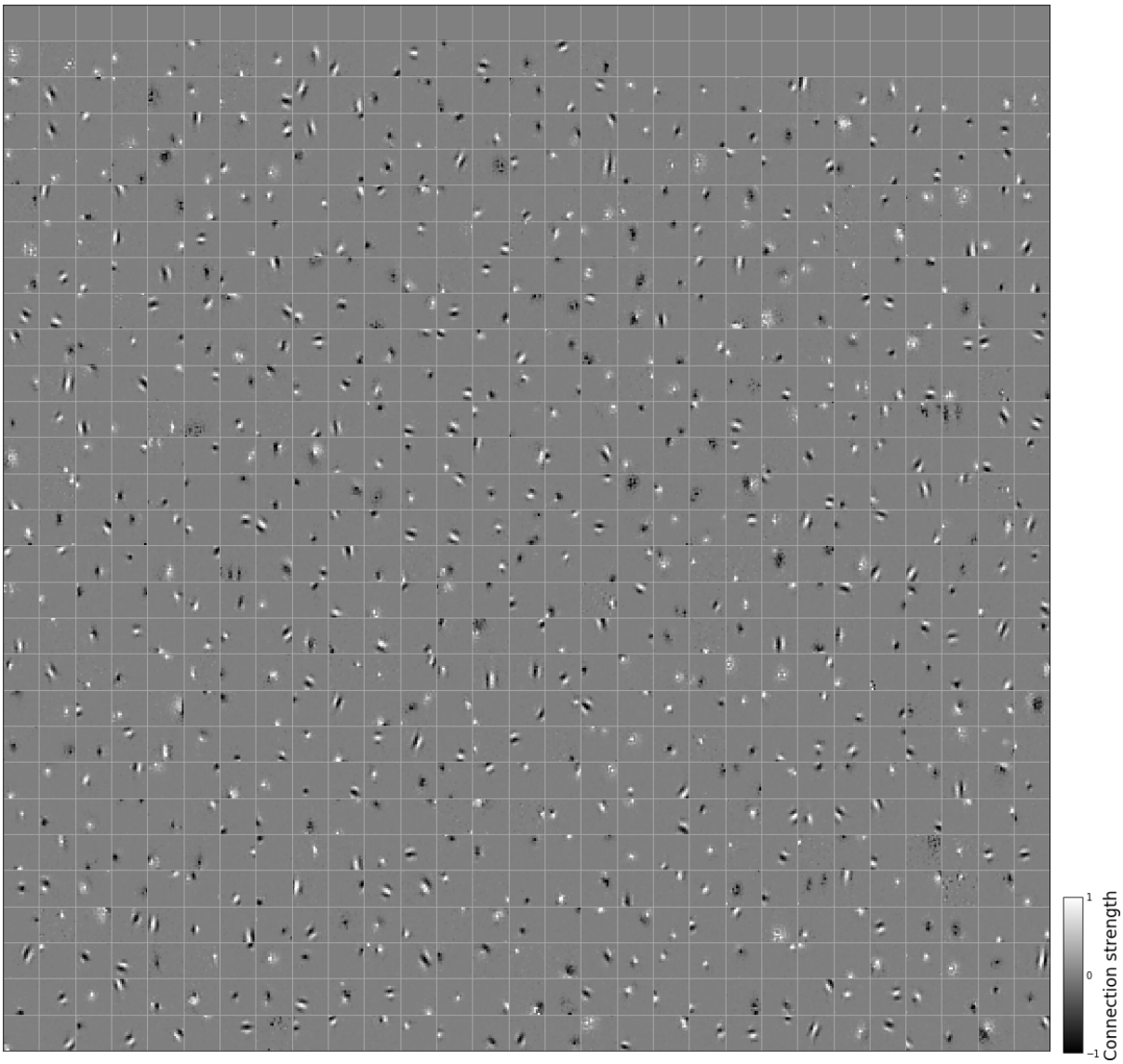
Supplementary Figure 3.14 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength ($10^{-6.75}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



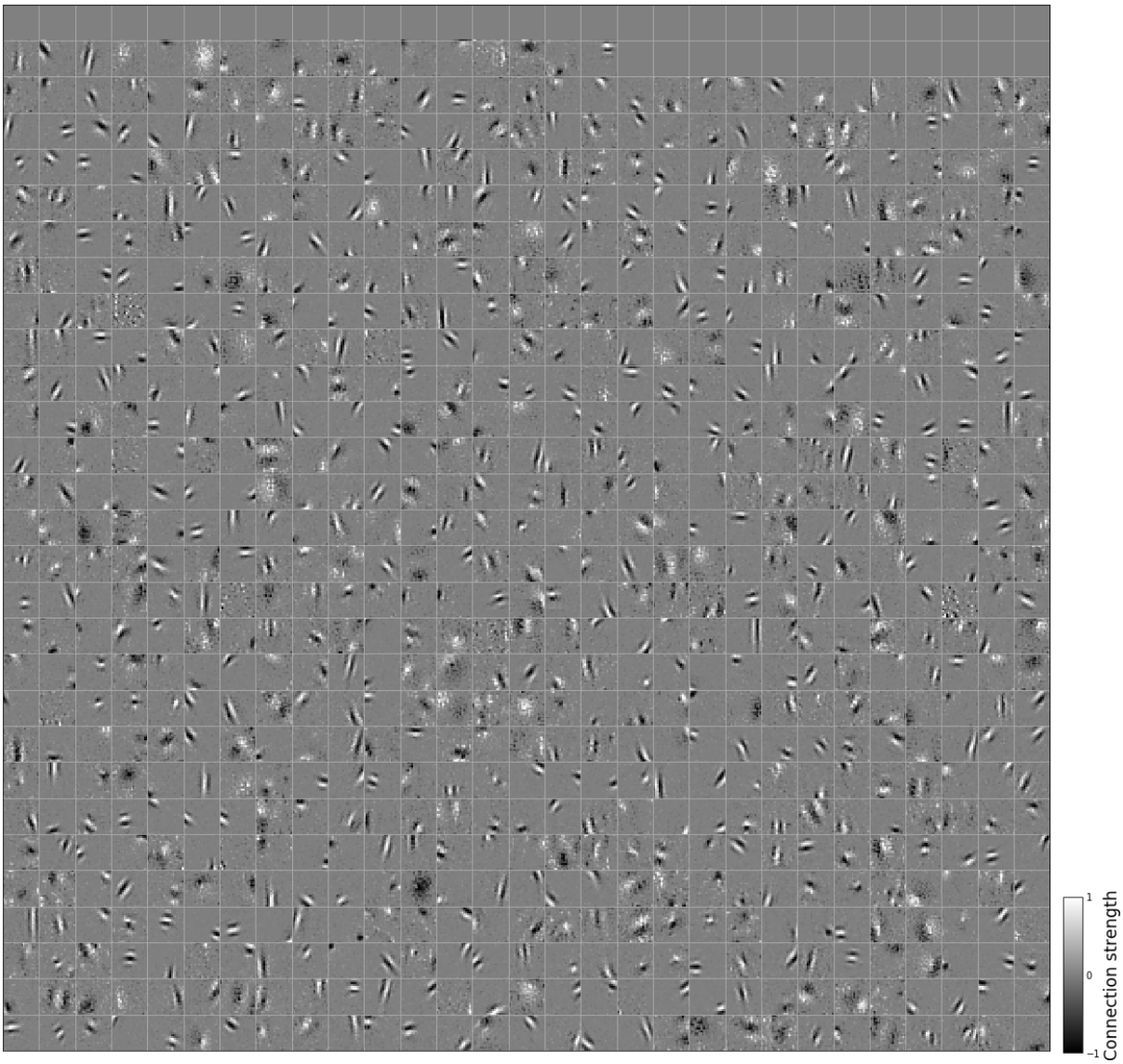
Supplementary Figure 3.15 | Full set of visual spatial RFs of the temporal prediction model trained with 400 units and L_1 weight regularization strength (10^{-7}). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



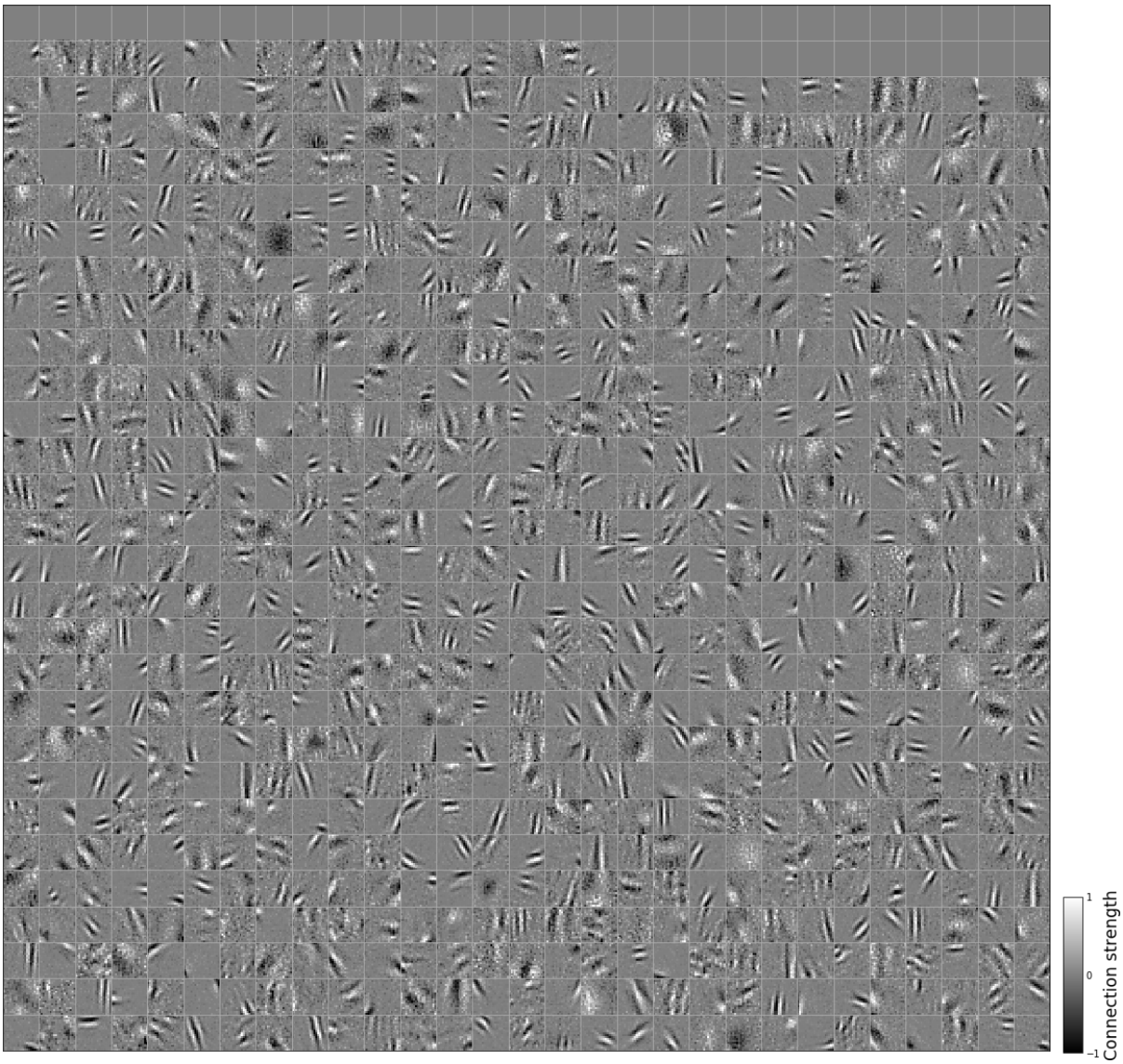
Supplementary Figure 3.16 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength ($10^{-5.5}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



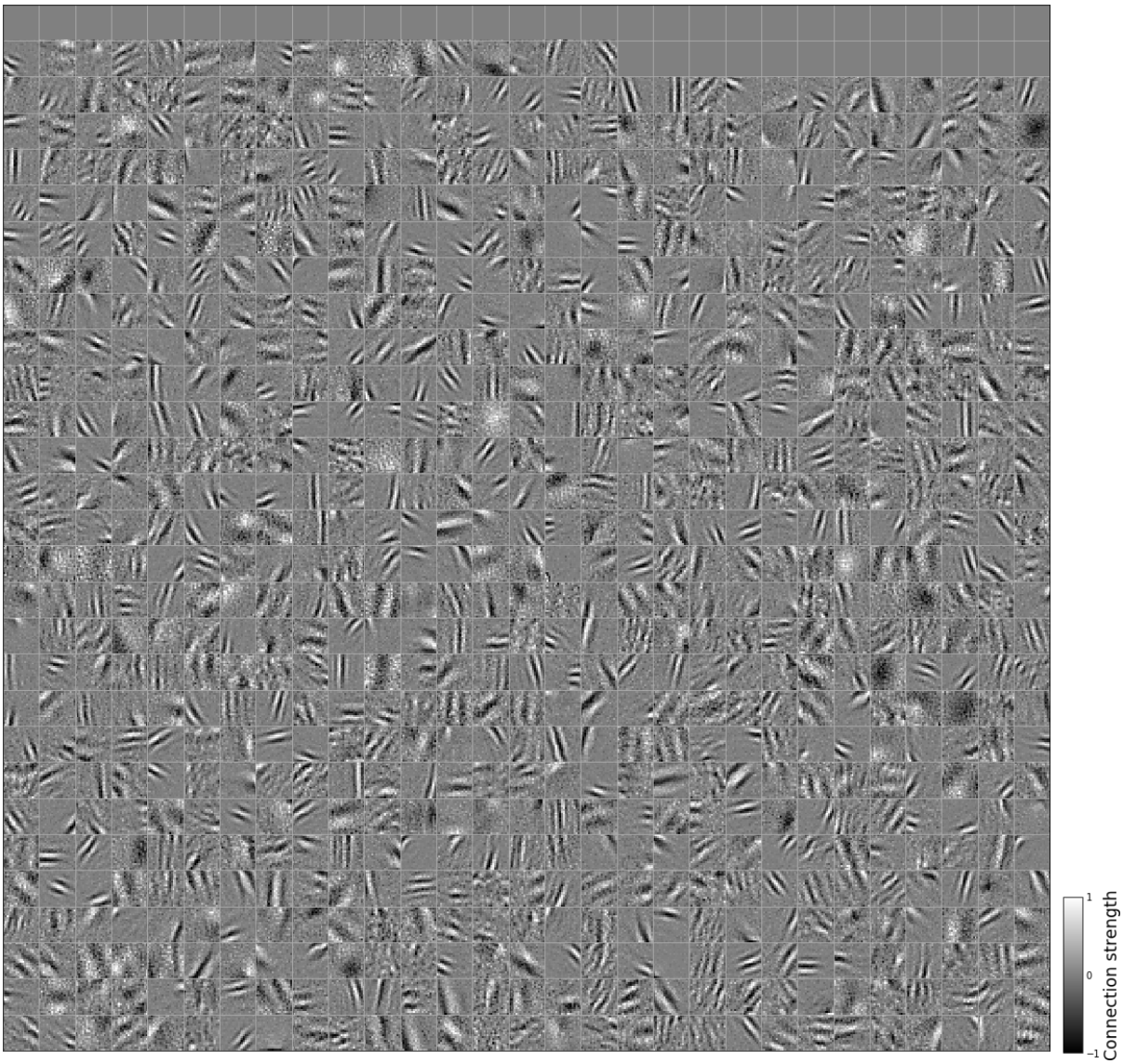
Supplementary Figure 3.17 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength ($10^{-5.75}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



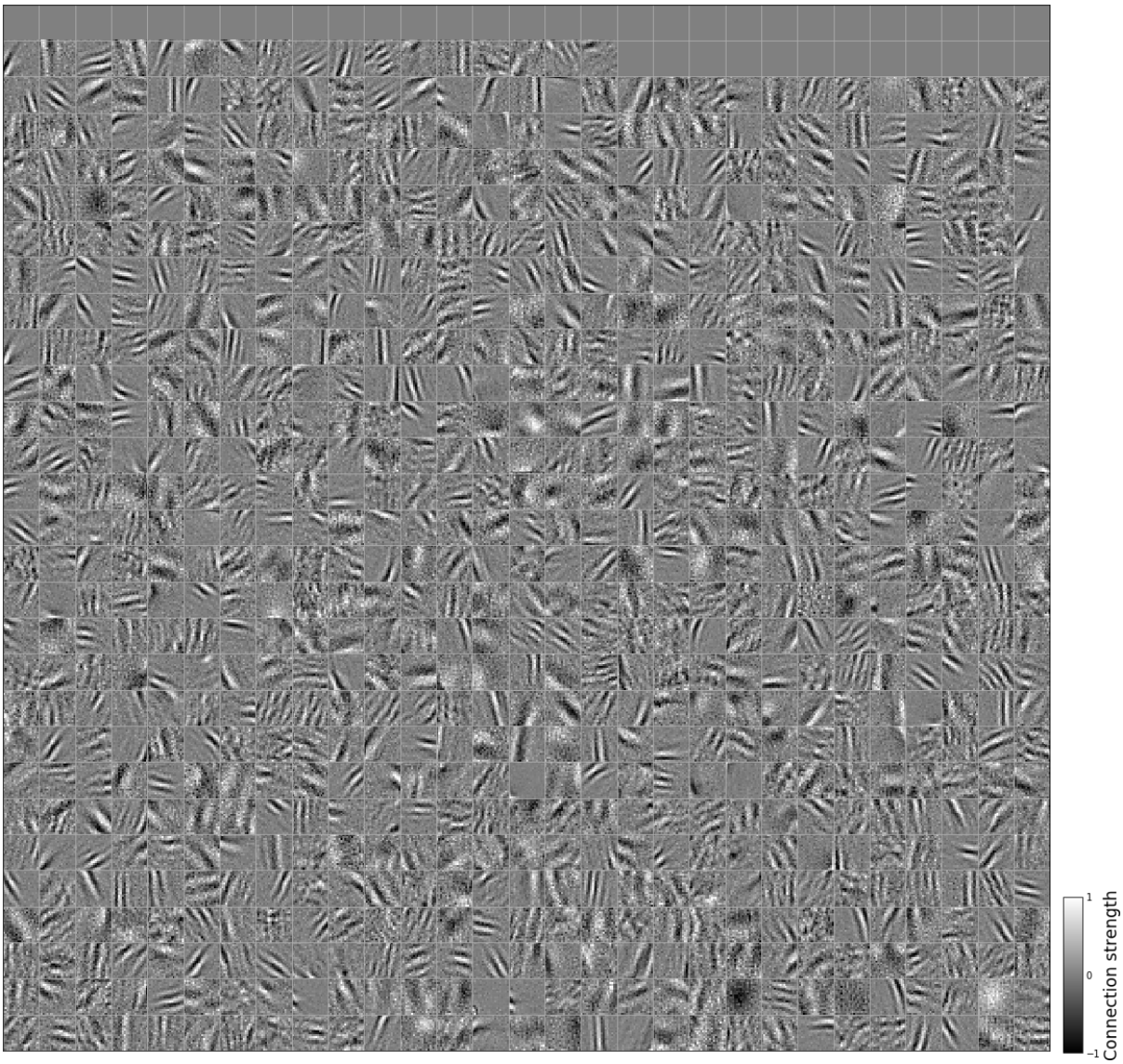
Supplementary Figure 3.18 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength (10^{-6}). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



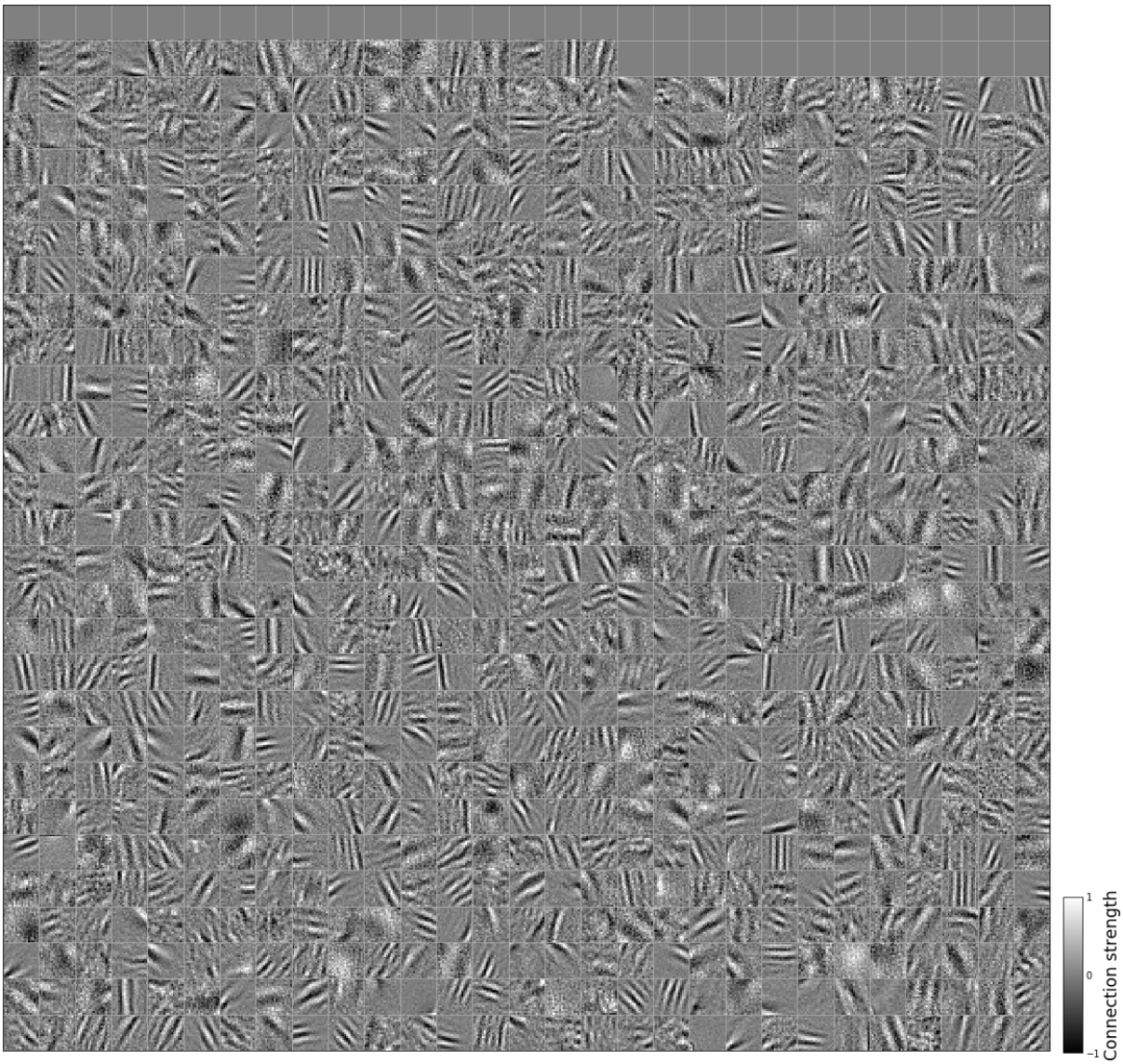
Supplementary Figure 3.19 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength ($10^{-6.25}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



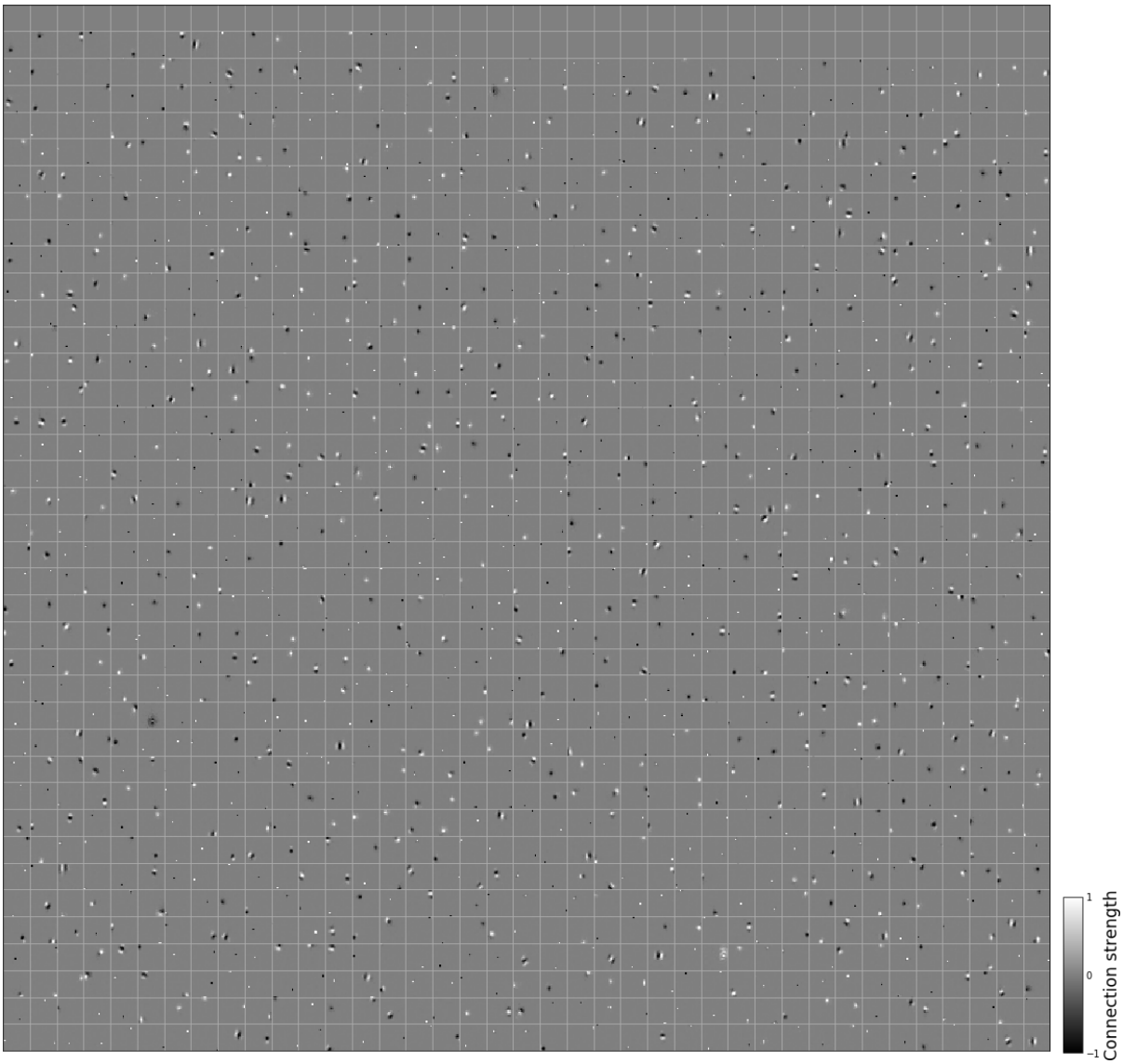
Supplementary Figure 3.20 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength ($10^{-6.5}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



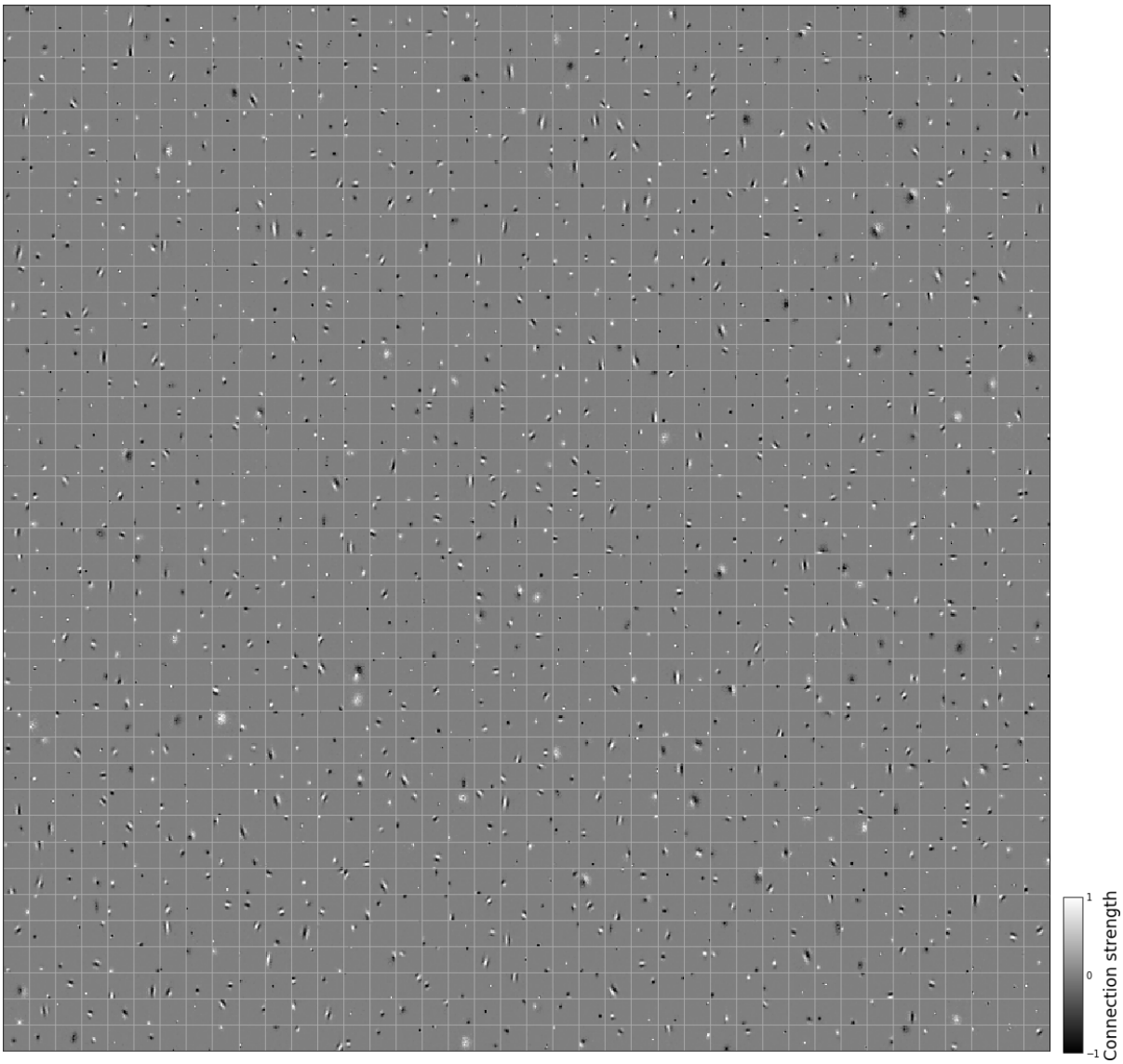
Supplementary Figure 3.21 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength ($10^{-6.75}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



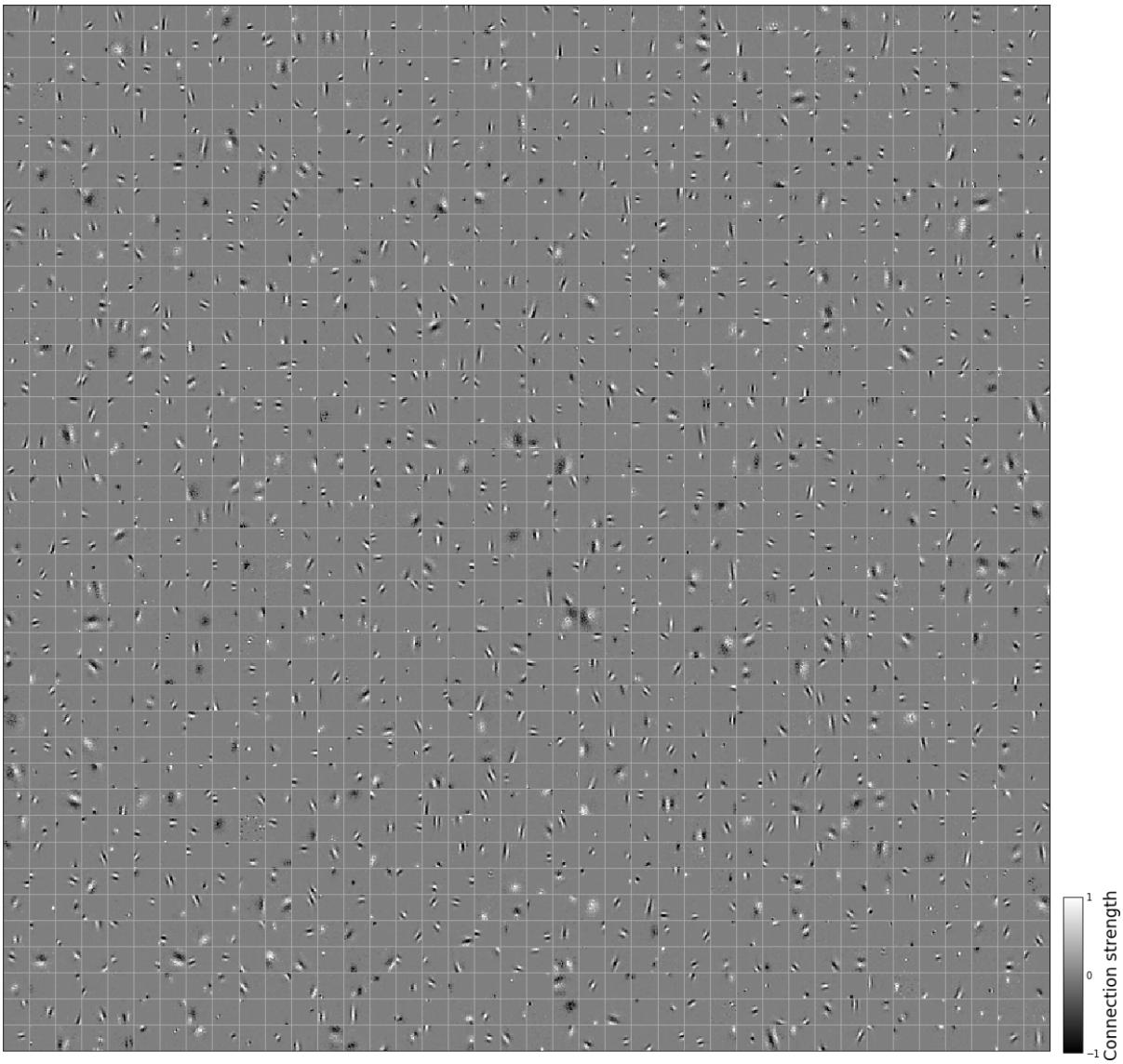
Supplementary Figure 3.22 | Full set of visual spatial RFs of the temporal prediction model trained with 800 units and L_1 weight regularization strength (10^{-7}). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



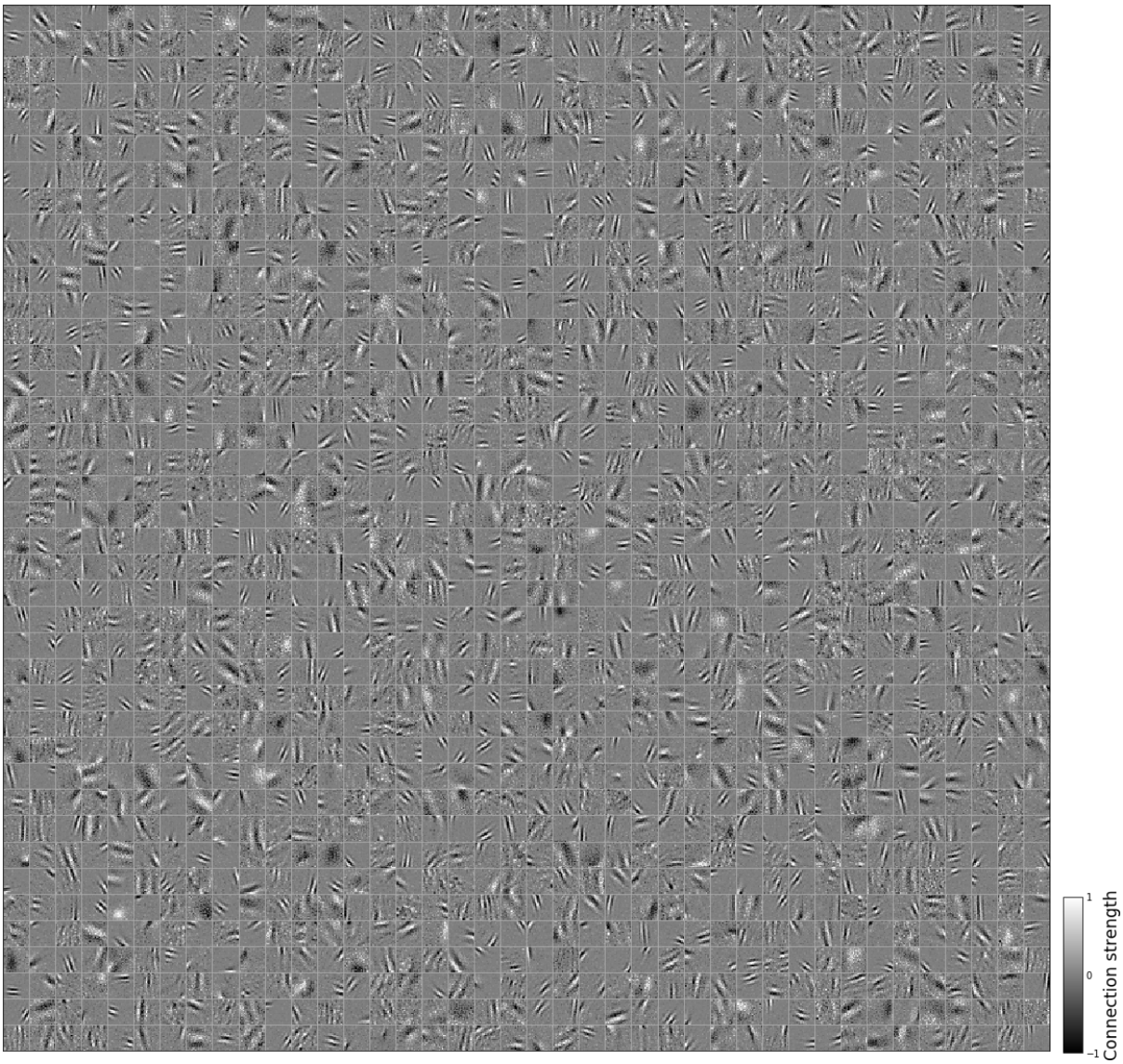
Supplementary Figure 3.23 | Full set of visual spatial RFs of the temporal prediction model trained with 1600 units and L_1 weight regularization strength ($10^{-5.5}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



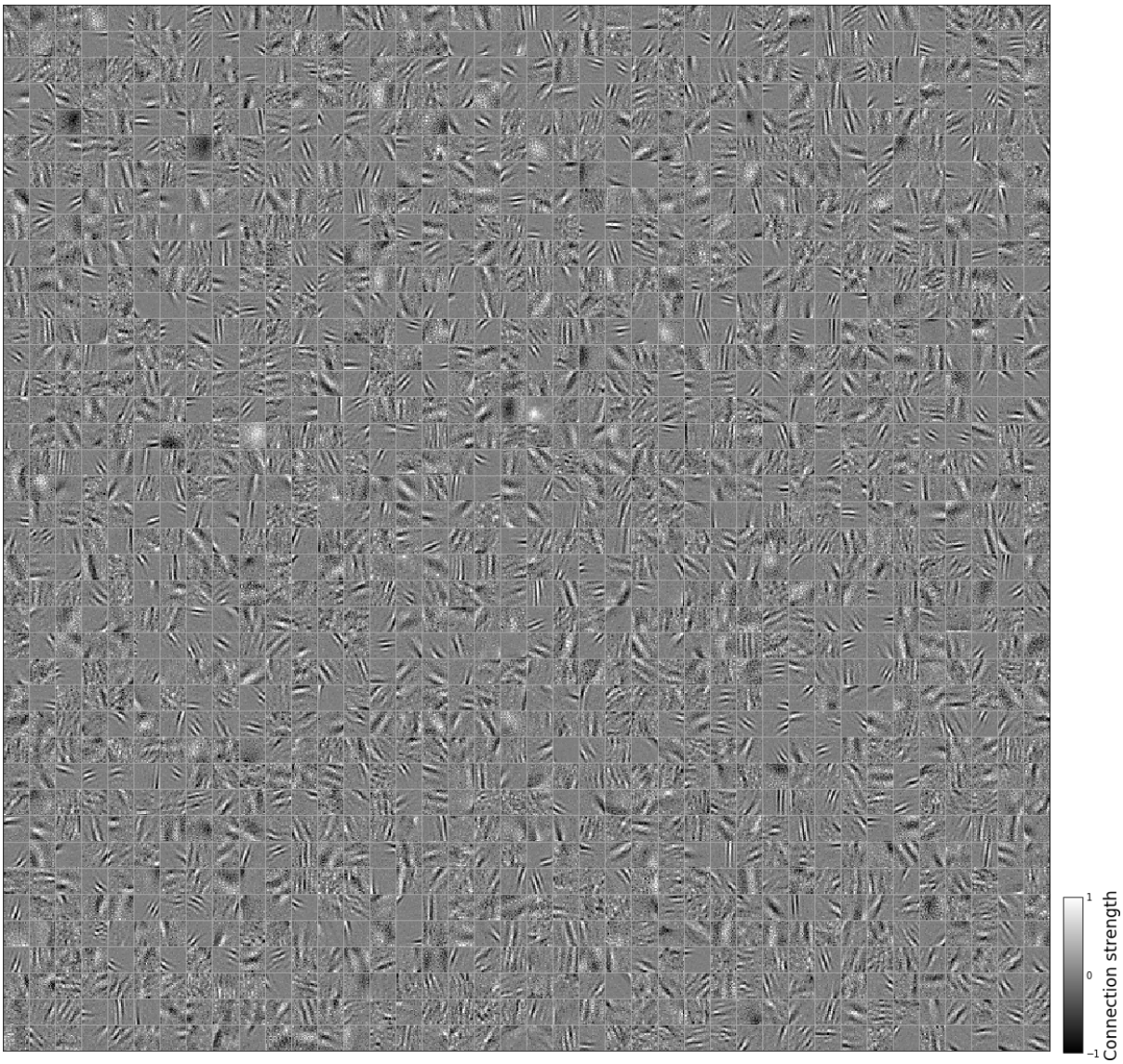
Supplementary Figure 3.24 | Full set of visual spatial RFs of the temporal prediction model trained with 1600 units and L_1 weight regularization strength ($10^{-5.75}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



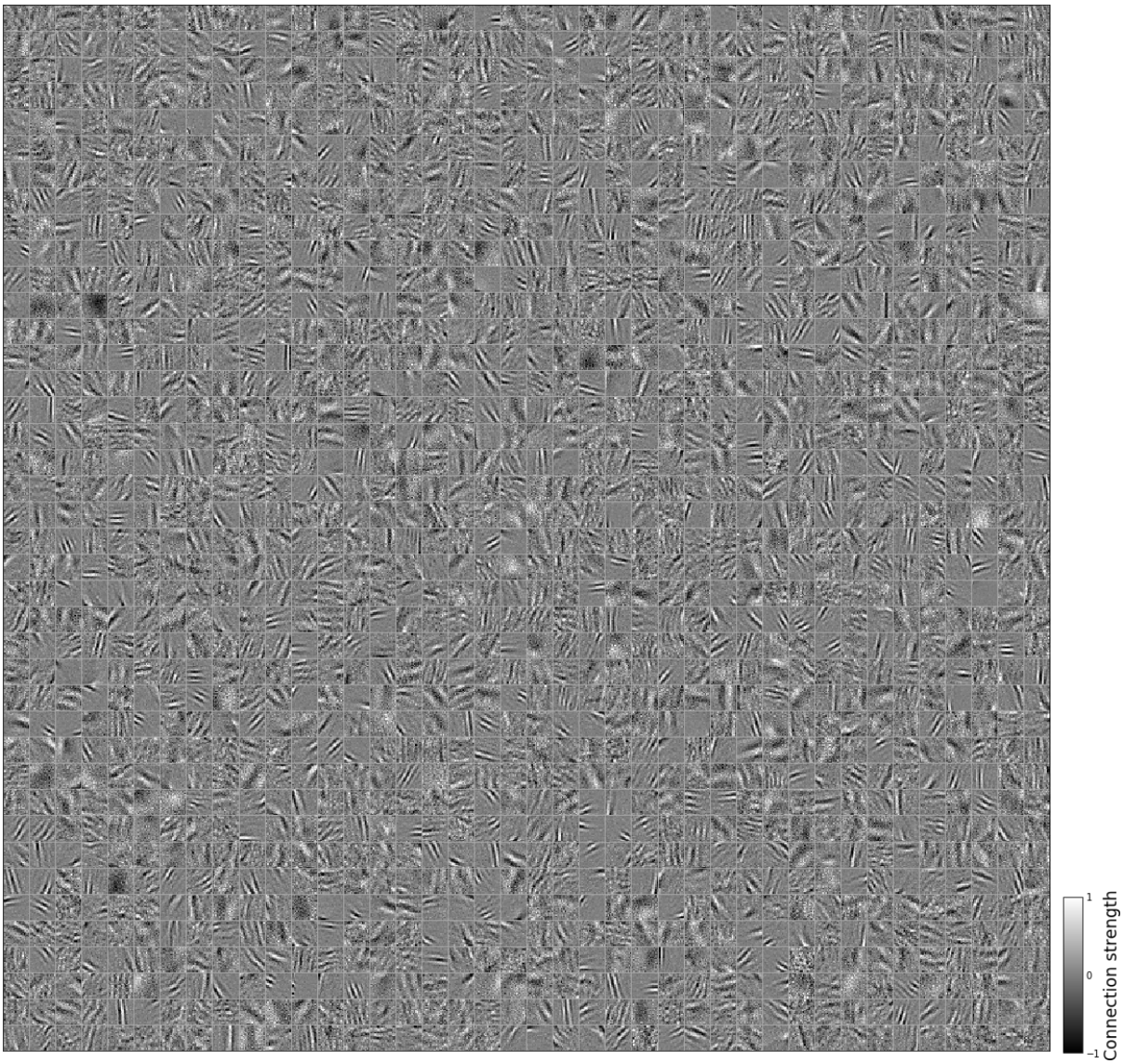
Supplementary Figure 3.25 | Full set of visual spatial RFs of the temporal prediction model trained with 1600 units and L_1 weight regularization strength (10^{-6}). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



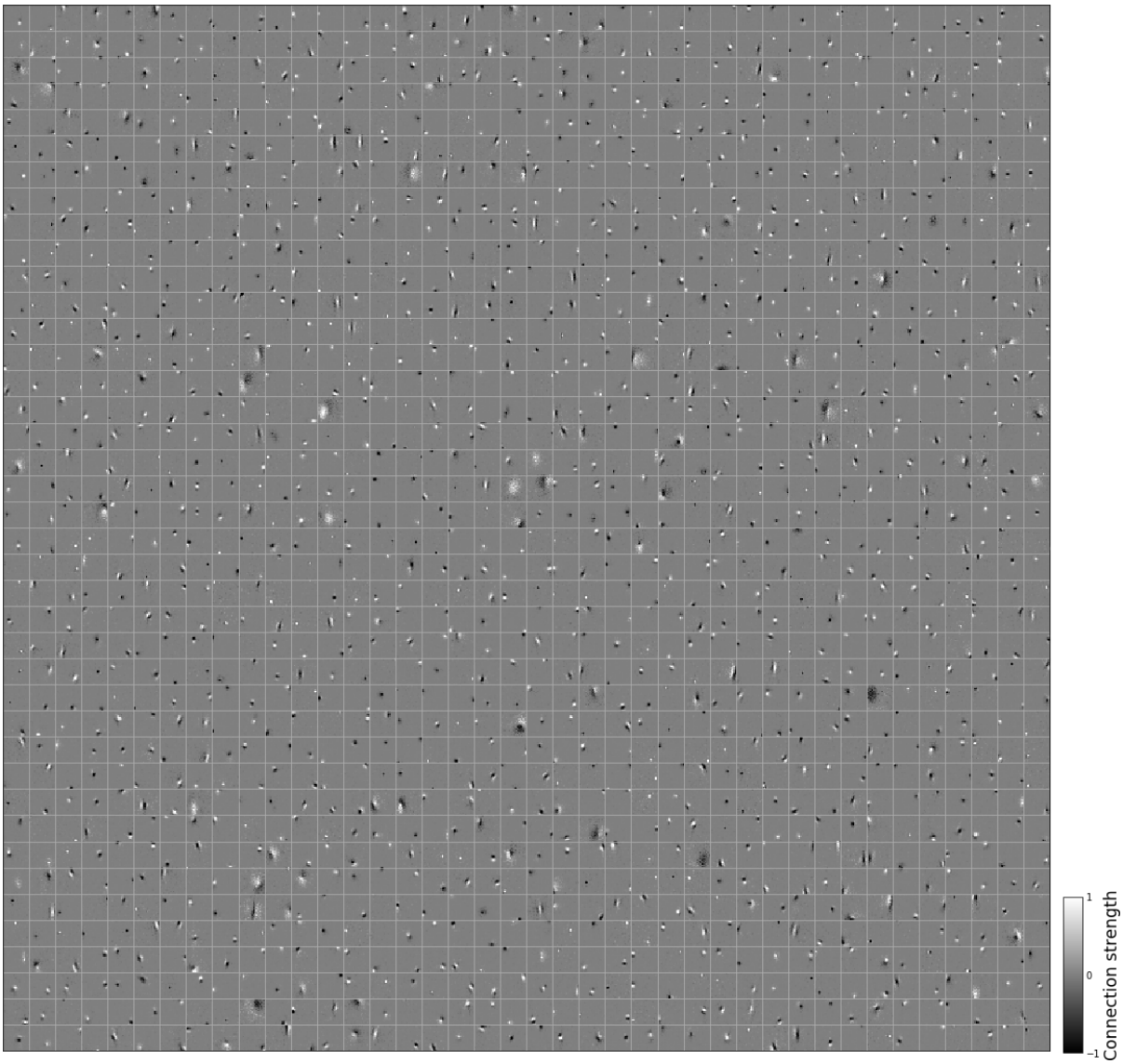
Supplementary Figure 3.26 | Full set of visual spatial RFs of the temporal prediction model trained with 1600 units and L_1 weight regularization strength ($10^{-6.5}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



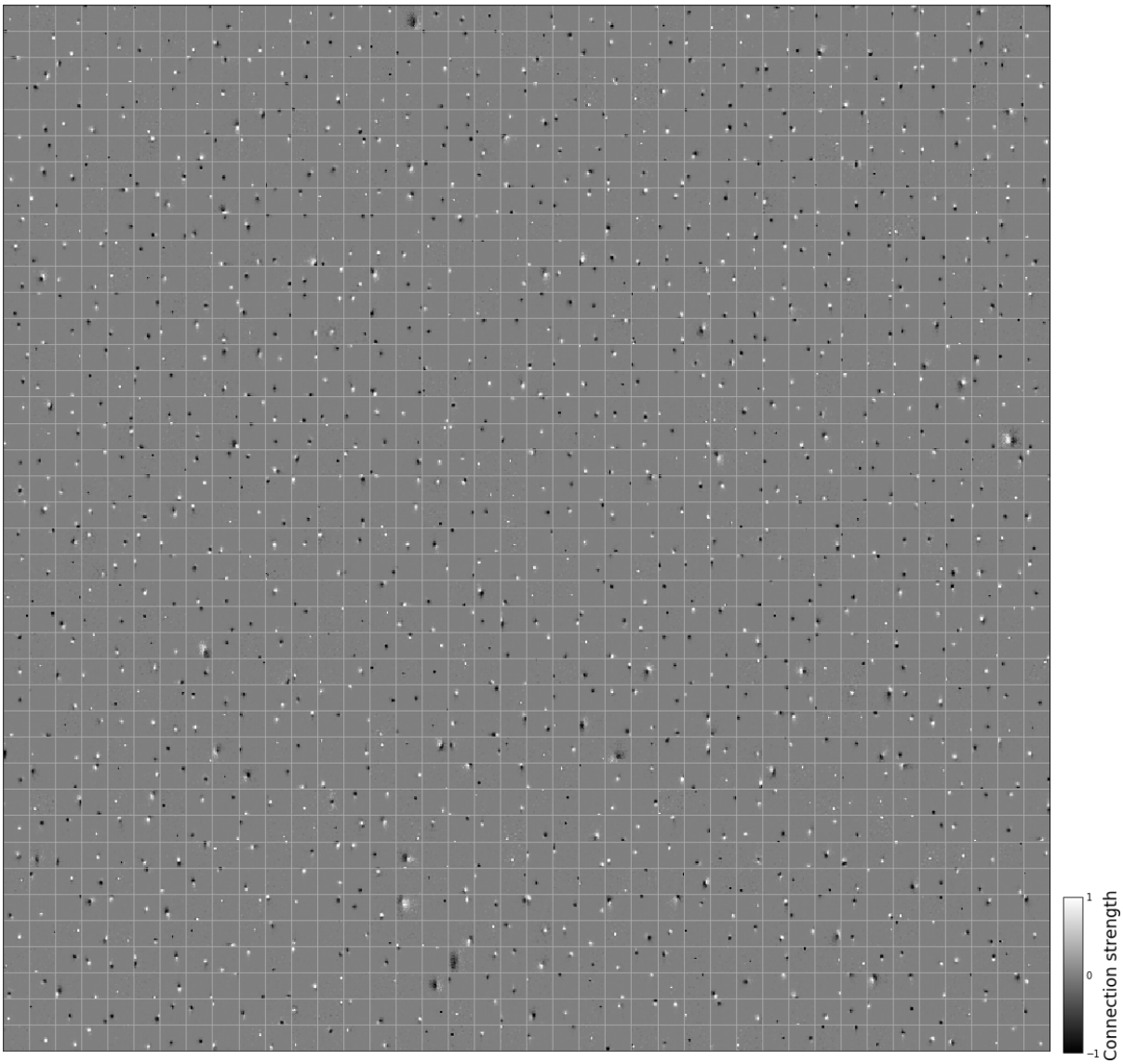
Supplementary Figure 3.27 | Full set of visual spatial RFs of the temporal prediction model trained with 1600 units and L_1 weight regularization strength ($10^{-6.75}$). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



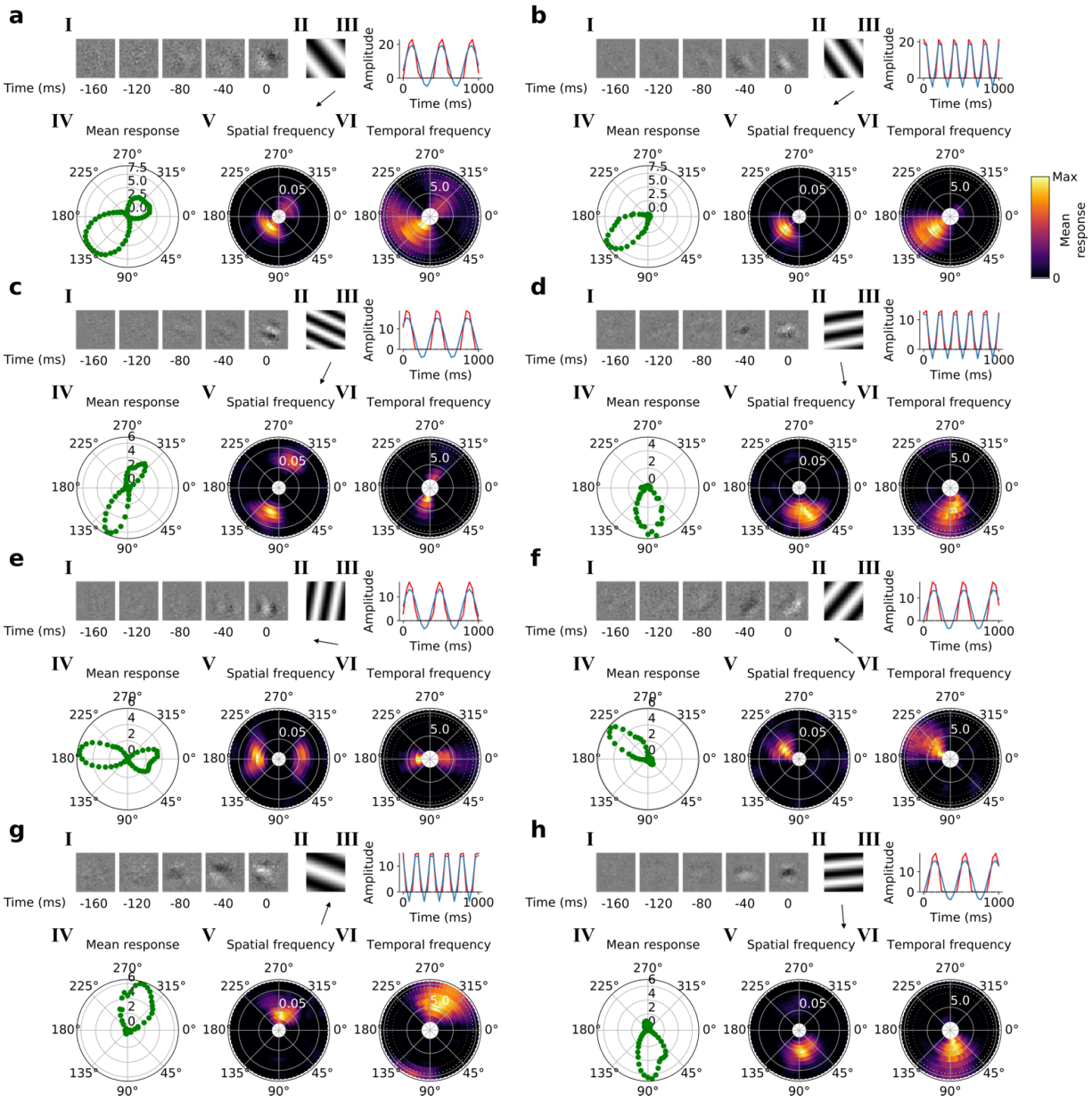
Supplementary Figure 3.28 | Full set of visual spatial RFs of the temporal prediction model trained with 1600 units and L_1 weight regularization strength (10^{-7}). Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



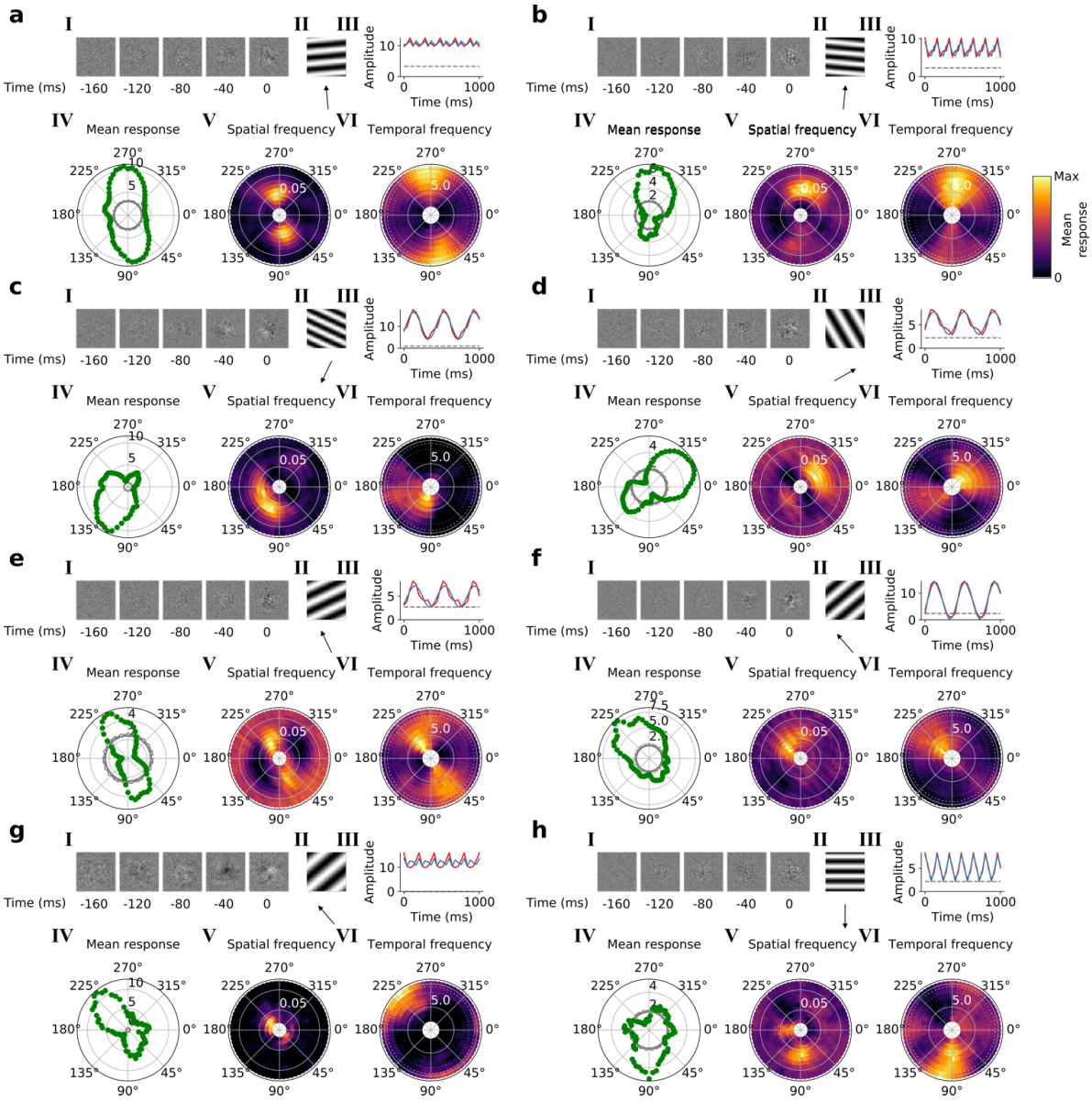
Supplementary Figure 3.29 | Full set of visual spatial RFs of the temporal prediction model trained when network is trained to predict 3 time-steps into the future. Units were obtained by training the model with 1600 hidden units on visual inputs. The hidden unit number and L_1weight regularization strength (10^{-6}) was chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



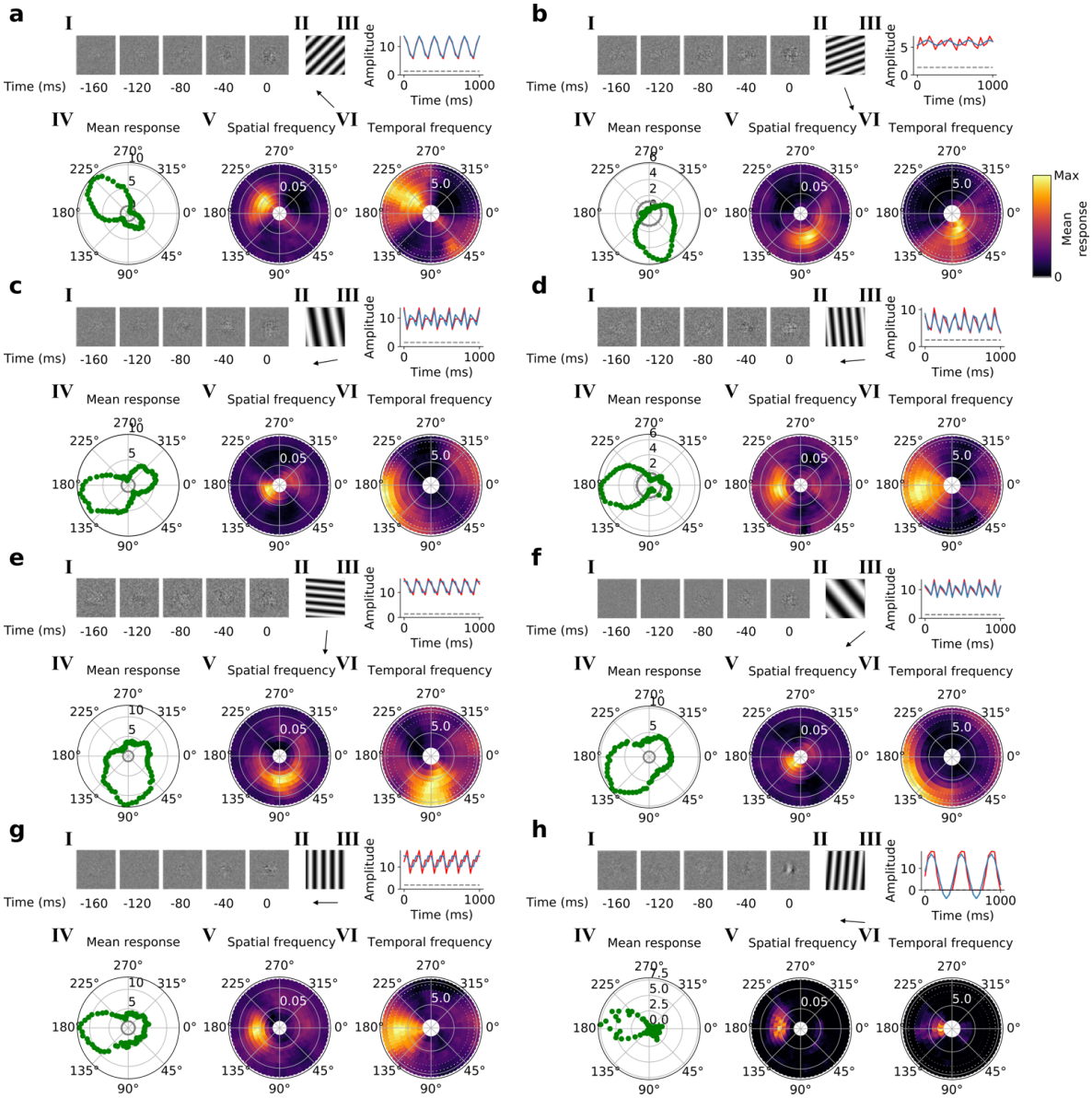
Supplementary Figure 3.30 | Full set of visual spatial RFs of the temporal prediction model trained when network is trained to predict 6 time-steps into the future. Units were obtained by training the model with 1600 hidden units on visual inputs. The hidden unit number and L_1weight regularization strength (10^{-6}) was chosen because they result in the lowest MSE on the prediction task, as measured using a cross validation set. Each square represents the spatial RF of a single unit, shown at its best time step. White – excitation, black - inhibition.



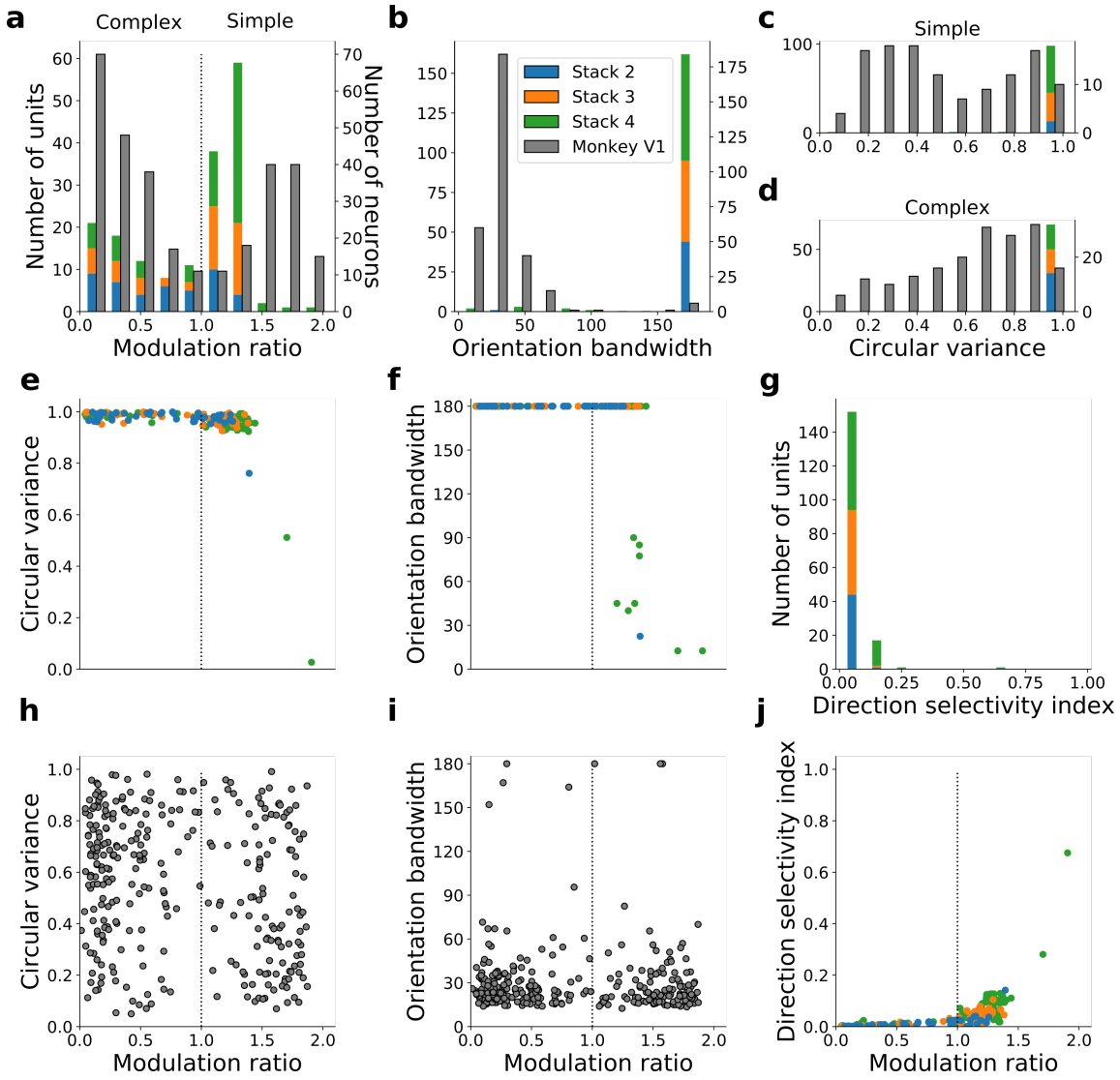
Supplementary Figure 4.1 | Tuning properties of example units in stack 2. ***a-h,*** (I) the linear RF, (II) the drifting grating that best stimulates this unit and (III) the amplitude of the unit's response to this grating over time. Red line: unit response; blue line: best-fitting sinusoid; gray dashed line: response to blank stimulus (sometimes overlapping with x-axis). (IV) The unit's mean response over time plotted against orientation (in degrees) for gratings presented at its optimal spatial and temporal frequency. (V, VI) Tuning curves showing the joint distribution of responses to (V) orientation (in degrees) and spatial frequency (in cycles/pixel) at the preferred temporal frequency and to (VI) orientation and temporal frequency (in Hz) at the preferred spatial frequency. In V and VI the color represents the mean response over time to the grating presented.



Supplementary Figure 4.2 | Tuning properties of example units in stack 3. ***a-h,*** (I) the linear RF, (II) the drifting grating that best stimulates this unit and (III) the amplitude of the unit's response to this grating over time. Red line: unit response; blue line: best-fitting sinusoid; gray dashed line: response to blank stimulus (sometimes overlapping with x-axis). (IV) The unit's mean response over time plotted against orientation (in degrees) for gratings presented at its optimal spatial and temporal frequency. (V, VI) Tuning curves showing the joint distribution of responses to (V) orientation (in degrees) and spatial frequency (in cycles/pixel) at the preferred temporal frequency and to (VI) orientation and temporal frequency (in Hz) at the preferred spatial frequency. In V and VI the color represents the mean response over time to the grating presented.



Supplementary Figure 4.3 | Tuning properties of example units in stack 4. ***a-h,*** (I) the linear RF, (II) the drifting grating that best stimulates this unit and (III) the amplitude of the unit's response to this grating over time. Red line: unit response; blue line: best-fitting sinusoid; gray dashed line: response to blank stimulus (sometimes overlapping with x-axis). (IV) The unit's mean response over time plotted against orientation (in degrees) for gratings presented at its optimal spatial and temporal frequency. (V, VI) Tuning curves showing the joint distribution of responses to (V) orientation (in degrees) and spatial frequency (in cycles/pixel) at the preferred temporal frequency and to (VI) orientation and temporal frequency (in Hz) at the preferred spatial frequency. In V and VI the color represents the mean response over time to the grating presented.



Supplementary Figure 4.4 | Quantitative tuning properties of control model units in stacks 2-4 in response to drifting sinusoidal gratings and corresponding measures of macaque V1 neurons. The control model was trained to reproduce the final time step of the input rather than predicting the immediate future time step, but otherwise was identical to the temporal prediction model. **a-d,** Histograms showing tuning properties of model and macaque V1 neurons as measured using drifting gratings. Units from stacks 2-4 are plotted on top of each other to also show the distribution over all stacks; this is because no stack can be uniquely assigned to V1. Modulation ratio measures how simple or complex a cell is; orientation bandwidth and circular variance are measures of orientation selectivity. **e, f,** Joint distributions of tuning measures for model units. Each dot is a model unit. **g,** Distribution of direct selectivity for model units. **h, i,** Joint distributions of tuning measures for V1 data **j,** Joint distribution of direction selectivity and modulation ratio for model units.