Statistical Models Discover an Accurate Low-Dimensional Latent Representation of Zebrafish Neural Activity

Thijs L. van der Plas **Physics & Astronomy MSc thesis, Radboud University**

First supervisor: Bernhard Englitz (Donders Institute, Radboud University)
Second supervisor: Georges Debrégeas (Laboratoire Jean Perrin, Sorbonne University)

Original: 25 March 2019 - Minor revisions: 17 June 2019

Abstract

Present-day neuroscience maintains the growing consensus that the activity of a large population of neurons can effectively be described by a low number of latent sources. Different techniques are commonly used to infer such a low-dimensional representation, targeting specific characteristics of neural data. Here, we merge principles from statistical mechanics and machine learning, and adapt an energy-based generative approach to construct the latent space. We show that our method, a sparse Restricted Boltzmann Machine, discovers a low-dimensional latent representation that is more accurate than the prevalent Principal Component Analysis method. To this end, we use state-of-the-art large-scale calcium imaging data sets of larval zebrafish, whose activity is decomposed into a low-dimensional representation. Hence we consolidate, in a data-driven approach, the concept of low-dimensional latent signals that describe the majority of neural activity. Interpretation of these unsupervised constructed latent states promises to augment our understanding of neural processing.

Introduction

Collective neural activity in the brain underpins cognition and complex sensorimotor action. Comprehending how different ensembles of neurons interact, remains one of the fundamental research topics in neuroscience (Bassett and Sporns, 2017). Recent technological advances have strongly increased the number of neurons that can be simultaneously recorded (Panier et al., 2013; Ahrens et al., 2013; Jun et al., 2017), facilitating the exhaustive study of population activity to understand complex functioning.

In particular, one recent neuroimaging breakthrough that stands out, is the technical innovation of recording whole-brain, single-cell resolution calcium dynamics of zebrafish larvae (Panier et al., 2013; Ahrens et al., 2013; Wolf et al., 2015). These experiments use Light-Sheet Microscopy (LSM) (Keller and Ahrens, 2015) to illuminate a brain-wide horizontal 2D plane of neurons that are genetically encoded with fluorescent calcium reporters (Grienberger and Konnerth, 2012), which is

consecutively moved vertically to create a 3D imaging volume, imaging 50,000+ cells at 2-4Hz. This in vivo imaging technique can be coupled to stimulus presentation and behavior read-out to probe sensorimotor systems at the whole-brain scale (Dunn et al., 2016; Wolf et al., 2017; Migault et al., 2018; Chen et al., 2018).

These studies define each neuron's functioning by correlating its activity to environmental or behavioral variables, a classic approach that has yielded fundamental insights in the past such as tuning curves (Hubel and Wiesel, 1962) and place and grid cells (O'Keefe, 1976; Hafting et al., 2005). Though repeatedly a fruitful approach, not all neurons are directly related to variables known to the experimenter. Therefore, it is paramount to develop complementary analysis methods that address network activity from a different perspective, by characterizing their activity in an unsupervised fashion (i.e. without correlates imposed by the experimenter).

Recent work has demonstrated that substantial parts of the total variance of a population of neurons can often be explained by a small set of latent sources (Okun et al., 2015; Cunningham and Yu, 2014; Gallego et al., 2017; Huang et al., 2019). In other words, given N neurons, the activity of each is often adequately composed of an individual mapping from $K \ll N$ latent signals. Latent signals need not to have a direct physiological counterpart, but can be viewed as emergent population-level activity. To discover these latent signals, a range of standardized techniques has been developed, including Principal Component Analysis (PCA) and Factor Analysis (Bishop, 2006; Cunningham and Yu, 2014). These methods essentially create a linear mapping $y_n = W_n \cdot X + \epsilon$ between latent signals X and observable neural signals $\{y_1, y_2, ..., y_N\}$. Though successful in some applications, these techniques are inherently constrained to create orthogonal latent signals (PCA) or have independent local and global variability (FA).

In the current study, we bypass the limitations induced by these consequential constraints by working in the framework of maximum entropy models (Cimini et al., 2019; Gardella et al., 2019). This guarantees the creation of a model that is, by construction, maximally unconstrained except for the architecture and functions imposed by the experimenter that relate the model to the empirical data. We successfully apply this statistical modeling technique to spontaneous neural data of the zebrafish larva hindbrain, acquired by LSM. This is motivated by the notion that certain aspects of zebrafish behavior are stochastic, rather than deterministic, and can be accounted for using statistical models (Dunn et al., 2016; Wolf et al., 2017). Specifically, zebrafish locomotion consists of left and right turning bouts, which exhibit strong autocorrelation (Dunn et al., 2016). Sensory stimulation such as illumination can override this internal stationary statistical configuration into a different configuration (Wolf et al., 2017). Hence a statistical model is hypothesized to adequately describe spontaneous brain activity (i.e. with no sensory stimulation), which hitherto has only been assessed with (neuronal) pairwise correlation matrices (Panier et al., 2013; Ahrens et al., 2013).

Fusing such a pairwise interaction design (as in correlation matrices) with a maximum entropy model yields the Ising Model, i.e. Boltzmann Machines (when used to generate new data). Though this has successfully been applied to small neural systems (< 100 neurons) (Schneidman et al., 2006; Posani et al., 2017; Meshulam et al., 2017), it rapidly becomes computationally infeasible to train and interpret for larger systems due to the quadratic growth of parameters.

Instead, we turn to Restricted Boltzmann Machines (RBMs), a machine learning technique that includes the aforementioned low-dimensional representation, and has recently been successful in domains other than neuroscience (Tubiana and Monasson, 2017; Tubiana et al., 2018). RBMs replace pairwise neuron-to-neuron interactions (present in Boltzmann Machines) with pairwise connections between neurons and (latent) hidden units, enabling neurons to communicate via this hidden layer (Smolensky, 1986; Hinton and Salakhutdinov, 2006).

We show that by training the RBM to match experimental data statistics, it outperforms the standard dimensionality reduction method PCA in statistical accuracy, structural organization and functional (dynamic) prediction. Even though the RBM model is completely data-driven, an organization of neuronal structure emerges which is reminiscent of known physiology. Furthermore, functional prediction of neuron-to-neuron dynamics via the model's low-dimensional bottleneck performs equivalently well as the widely established, direct neuron-to-neuron modeling technique logistic regression (but also see Benjamin et al., 2018), while additionally it gains a considerable computational speed-up. These results provide evidence that the low-dimensional latent layer is an accurate and meaningful representation of the full neural system.

Hence we provide evidence that RBMs can successfully be applied to very large neural data, and capture three central characteristics of the system (statistics, structure, dynamics) significantly better than the classic dimensionality reduction technique PCA. This success arises from the dependency between low-dimensional states, which is absent in PCA, and consolidates the emerging consensus that neural functioning is governed by sparse, interacting functional processes (Song et al., 2005).

Results

Data Acquistion, Preprocessing and Model Definition

The data acquisition and preprocessing pipeline is summarized in figure 1A. Zebrafish larvae expressing nuclear green fluorescent protein calcium reporters are imaged with Light-Sheet Microscopy (LSM) (1A first panel, see Methods). This yields a 3D image stack of pixel intensity values over time, which are normalized by computing the $\Delta F/F$ signal (1A second panel, see Methods). These images are automatically segmented to identify cell locations and extract their activity. The calcium activity is deconvolved to spikes (Tubiana et al., 2017), which reduces the autocorrelation of the signal and provides a binary representation (see Methods). The third panel shows a visualization of all segmented cells (as single points) in 3D using the Fishualizer (Migault et al., 2018) and illustrates the deconvolved spiking activity. In the current study we focus on the anatomically defined region Rhombomere 1 (Randlett et al., 2015) as our region of interest (figure 1B left panel, highlighted in blue).

We perform dimensionality reduction using 4000 time points (out of a total of 5553). Evaluations are later performed on the remaining 1553 time points. The aim of any dimensionality reduction technique is to creating a mapping $\mathbf{h} = f(\mathbf{v})$ that transforms a high-dimensional visible state $\mathbf{v}$ to create a low-dimensional hidden state $\mathbf{h}$ (see figure 1B middle panel; the active neurons (bottom) are mapped to three states, differentiated by color (top)). The visible layer consists of all $N = 7993$ neurons in the system, and the hidden layer consists of $M = 70 \ll N$ latent units (illustrated in figure 1B right panel). In particular, we aim to construct a mapping that is sparse,

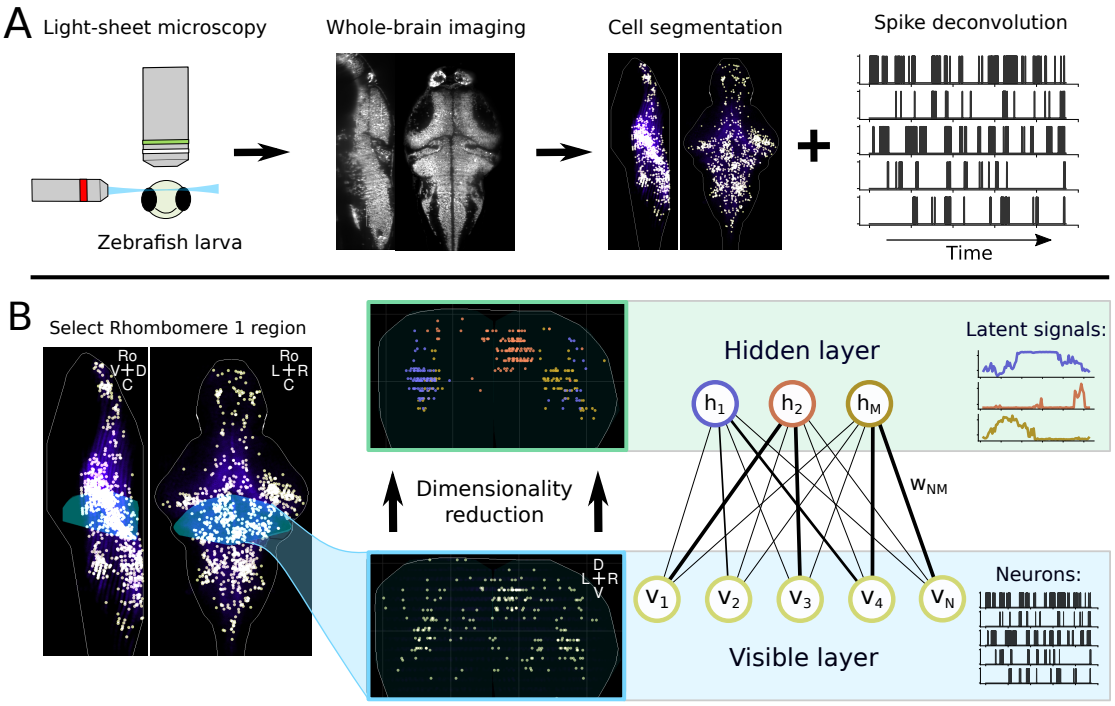

Figure 1: **Extracting low-dimensional representation using restricted boltzmann-machines. A)** An overview of the data preprocessing pipeline. From left to right: Neural data of larval zebrafish is acquired through LSM (left), in high-resolution image stacks (second). These images are automatically segmented to extract cell locations (third) and calcium activity, which is subsequently deconvolved into spiking activity (fourth). (Adapted from Migault et al., 2018). **B)** Rhombomere 1 is an anatomically defined brain region, located at the rostral side of the hindbrain (left panel, highlighted in blue). The activity of these neurons is defined as the visible layer, and dimensionality reduction aims to construct a low-dimensional mapping of this data (middle panel, 3 colors indicate the 3-dimensional mapping, other dimensions (67) are omitted for clarity). This is achieved by mapping N visible units (neurons) to M hidden units (latent signals), with varying connection strengths (right panel). The orientation of figures is indicated by the following abbreviations: Ro: Rostral, C: Caudal, D: Dorsal, V: Ventral, L: Left, R: Right.

i.e. where few connections between visible and hidden units are strong, while most are near-zero (depicted by the varying line widths in figure 1B right panel). As a result, for every visible state (i.e. the activity of all neurons at one moment in time), a small subselection of hidden units is active, while most are not.

PCA constructs this mapping by an Eigenvalue decomposition of the covariance matrix of the visible layer. The latent signals are named Principal Components (PCs), and are the Eigenvectors ranked by the magnitude of their Eigenvalues (i.e. ordered by explained variance of $\mathbf{v}$) (see Methods). This implies that by construction, the PCs are mutually orthogonal.

Alternatively, RBMs construct a mapping from the visible to hidden state by training a Maximum Entropy model, that assigns a probability to every possible $(\mathbf{v}, \mathbf{h})$ configuration via the Boltzmann distribution (see Methods for details):

$$P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp\left(-E(\mathbf{v}, \mathbf{h})\right) \tag{1}$$

For RBMs, the energy function is given by:

$$E(\mathbf{v},\mathbf{h}) = \sum_i g_i v_i - \sum_\mu U_\mu(h_\mu) + \sum_{i,\mu} w_{i,\mu} v_i h_\mu \tag{2}$$

Hidden units are obtained by sampling from $P(\mathbf{h}|\mathbf{v})$, which can be marginalized as $P(h_\mu|\mathbf{v}) \propto \exp\left(E_\mu(\mathbf{v})\right) = \exp\left(-U_\mu(h_\mu) + h_\mu \cdot \sum_i w_{i,\mu} v_i\right)$. New data can be generated in a similar fashion, by Monte Carlo sampling alternately from $P(\mathbf{h}|\mathbf{v})$ and $P(\mathbf{v}|\mathbf{h})$ (figure S1A and Methods). The most likely hidden unit activity (found by minimizing E_μ with respect to h_μ) is $h_\mu = (U'_\mu)^{-1}(\sum_i w_{i,\mu} v_i)$. Summarizing, the energy of a configuration $(\mathbf{v},\mathbf{h})$ is determined by both independent terms of ($\sum_i g_i v_i$ and $\sum_\mu U_\mu(h_\mu)$ where U_μ is the prior), and an interaction term ($\sum_{i,\mu} w_{i,\mu} v_i h_\mu$) between the visible and hidden layer.

To fit this general RBM model architecture to a specific system, the model parameters $(g_i, U_\mu, w_{i,\mu})_{i=1:N,\mu=1:M}$ must be tuned such that the correct probability is assigned to any $(\mathbf{v},\mathbf{h})$ state. This is achieved by generating new data from the model through Monte Carlo sampling, which is then compared to experimental data, after which the resulting divergence yields a gradient descent update step for the parameters. It is worth noting that sampling data is particularly fast for a RBM, compared to classic models such as Boltzmann Machines, thanks to its bipartite graph structure (figure S1A). This procedure is augmented by sparsity regularization that steers the solution to a sparse representation and avoids overfitting (Tubiana and Monasson (2017), see Methods).

Statistical Accuracy

To evaluate the performance of a RBM, we assess whether the expected values of statistics $f(\mathbf{v},\mathbf{h})$ of experimental data $\langle f(\mathbf{v},\mathbf{h})\rangle_{\text{Data}}$ match the expected value of statistics of model-generated data $\langle f(\mathbf{v},\mathbf{h})\rangle_{\text{RBM}}$. Specifically, we first evaluate the statistics used to train the model, that are expressed in the energy function (equation 2). Accordingly, the model is said to be converged if the training statistics of the generated data are highly correlated to those of the experimental data. The results of one RBM with $M = 70$ hidden units are shown in figure 2A, with scatter plots of the mean activities of visible units $\langle v_i\rangle$ (left), mean activities of hidden units $\langle h_\mu\rangle$ (middle) and pairwise interaction moments $\langle v_i h_\mu\rangle$ (right). The correlation coefficient r between the experimental and generated statistics is stated in the right bottom corner of the figures. This model has converged well, evidenced by the strong correlations between model and experimental statistics.

We further assess model performance, by surveying statistics that the model was not constrained to fit: the pairwise moments between visible units $\langle v_i v_j\rangle$, pairwise moments between hidden units $\langle h_\mu h_\nu\rangle$ and third order moments between hidden units $\langle h_\mu h_\nu h_\xi\rangle$ (figure 2B, left to right). Again, strong correlations between the model and experimental data are clearly observed for all metrics. This is critical, because it demonstrates that the model has captured collective activity, beyond the statistics it was designed to fit.

This statistical analysis was repeated for a range of RBM models, varying in size and capacity by their number of hidden units M. Figure 2C displays the correlation coefficients r for the three training statistics (left panel) and three test statistics (right panel) as function of M. RBM models that had poor correlation coefficient ($r < 0$) during training were defined to have not converged, and were omitted from these figures and further analysis. RBMs with low $M < 20$ exhibited

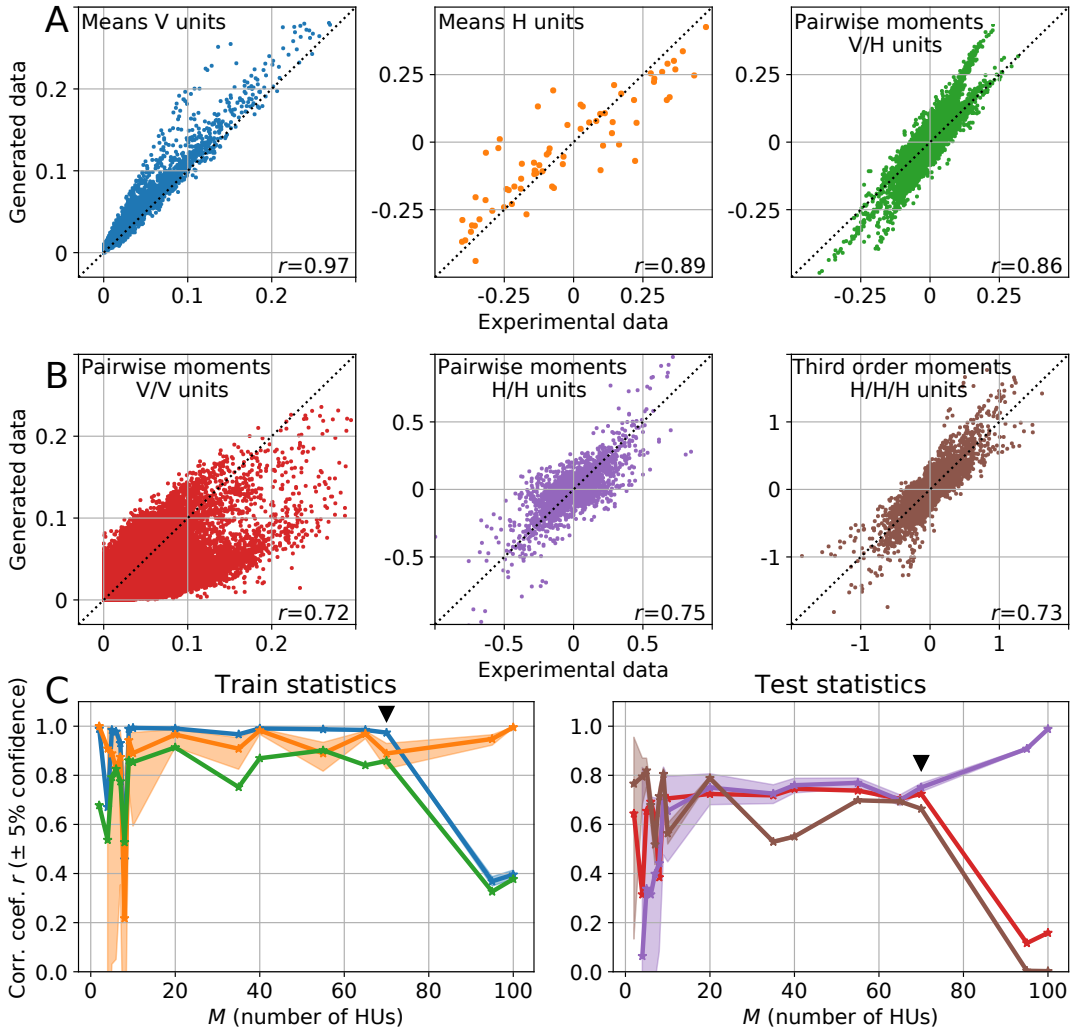

Figure 2: **RBM converges seen and unseen data statistics. A)** Scatter plots of the computed moments $\langle f \rangle$ of experimental training data $\langle f \rangle_{\text{Data}}$ (x-axis) versus generated data $\langle f \rangle_{\text{RBM}}$ (y-axis), that are used to train the RBM (see equation 2). From left to right: the mean activity of visible units $\langle v_i \rangle$ (blue), the mean activity of hidden units $\langle h_\mu \rangle$ (orange) and the pairwise moments between visible and hidden units $\langle v_i h_\mu \rangle$ (green). The correlation coefficient r is stated in the right bottom corner, and a $y = x$ dotted line is added for reference. **B)** Scatter plots of the computed moments that the RBM is not constrained to fit, of experimental testing data (x-axis) versus generated data (y-axis). From left to right: the pairwise moments between visible and visible units $\langle v_i v_j \rangle$ (red), the pairwise moments between hidden and hidden units $\langle h_\mu h_\nu \rangle$ (violet) and the third order moments of hidden units $\langle h_\mu h_\nu h_\xi \rangle$ (brown). Figure A and B were computed for a RBM with $M = 70$ hidden units. These computed moments yield a correlation coefficient r that indicates how well the model resembles the data (stated in the right bottom corner of each figure). Figure **C)** shows these coefficients as a function of M, the number of hidden units, for the moments used training (left) and testing (right). A black triangle indicates the r values for $M = 70$. The same color code is used across the panels.

smaller correlations for test statistics, and sometimes training statistics, than RBMs with higher M. Furthermore we observe that for high $M > 70$, performance is worse than for the intermediate range of M. These failures result from practical limitations, rather than theoretical incapability. The sampling procedure might have insufficiently covered the data space, due to the large number

of parameters to fit. The number of iterations (during training) was fixed, and increasing this would likely enhance results for large M. However, model performance already converged for the intermediate M values (figure 2C), so this was not pursued.

Local Organization of Hidden Units

After having fulfilled convergence criteria, we investigate the structural organization of the RBM and PCA models, by assessing their weights $\mathbf{W} = (w_{i,\mu})_{i=1:N,\mu=1:M}$ that connect the visible and hidden layer. The distribution of all weights is plotted in figure 3A for both RBM (purple) and PCA (green). The RBM entails a much sparser representation, as 88% of its weights are contained in the near-zero peak $|w| < 0.002$ (see zoom inset, notice the log y-scale). Importantly,

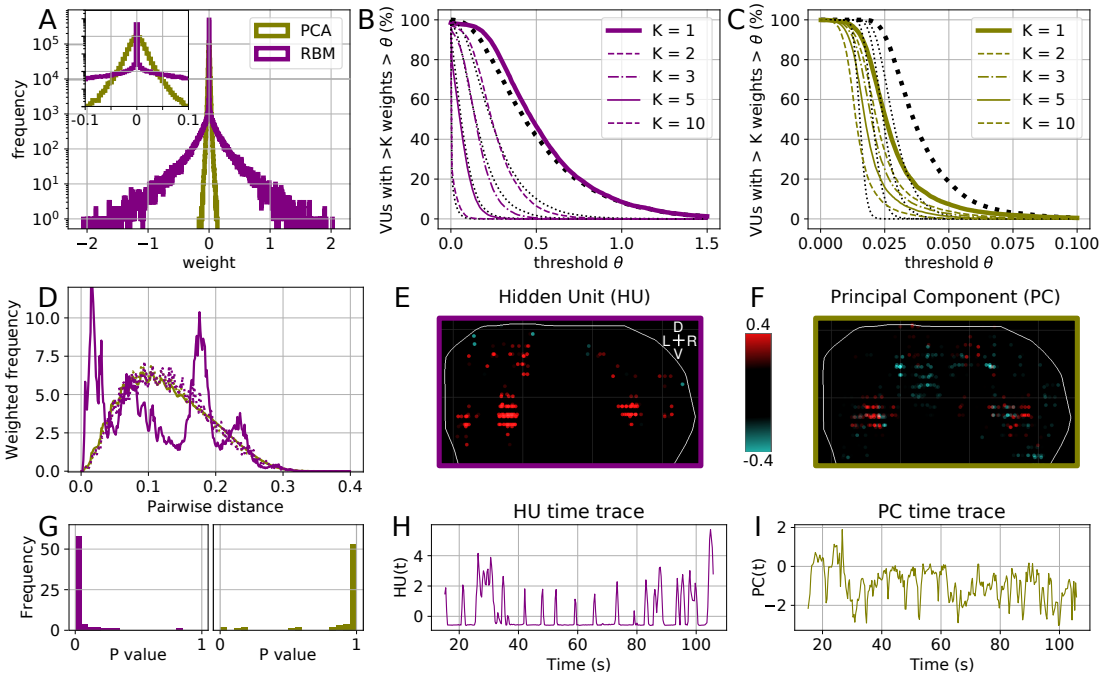

Figure 3: **RBM uncovers sparse structural components. A)**: Log-distribution of weights $w_{i,\mu}$ between the visible and hidden layer for RBM (purple) and PCA (green), with a zoom inset for $|w| < 0.1$. Both distributions have the same integral of $N \cdot M$. This color code is preserved throughout the figures. **B)**: Percentage of neurons that have at least K weight magnitudes $|w|$ greater than θ, which is varied along the x-axis. Results are plotted for $K = 1, 2, 3, 5, 10$ (legend in plot). For each K value the weights were shuffled and the same metric is plotted (black dashed lines). A two-sided Kilmogorov-Smirnov test determines that all distributions are significantly indifferent from their shuffled distribution (P values: $P = 0.82$ for $K = 1$, $P = 0.13$ for $K = 2$, $P = 0.19$ for $K = 3$, $P = 0.67$ for $K = 5$ and $P = 0.97$ for $K = 10$). **C)** Equivalent figure as (B), analyzed for PCA weights. All distributions significantly differ from their shuffled counterparts (KS test, P values: $P = 7{\cdot}10^{-4}$ for $K = 1$, $P = 2{\cdot}10^{-11}$ for $K = 2$, $P = 1{\cdot}10^{-19}$ for $K = 3$, $P = 2{\cdot}10^{-31}$ for $K = 5$ and $P = 9{\cdot}10^{-28}$ for $K = 10$). **D)** Normalized distributions of weighted pairwise distances between all neurons, for one hidden units (E) and one principal component (F). Weights ω were defined as the multiplication of the RBM weights that connect neurons to the same hidden unit $\omega_{i,j} = w_{i,\mu} \cdot w_{j,\mu}$. This is plotted for RBM, PCA (solid lines), and their coordinate-shuffled data sets (dashed lines). **E)** The projection of the example hidden unit to the visible layer. The weights are shown, at the location of their visible units, according to the blue-red color bar. **F)** The projection of the example principal component (the 6^{th} principal component). **G)** Bar diagram for RBM (left) and PCA (right) of all KS test P values between their hidden units/principal components and shuffled counterparts (as in figure D). **H)** Activity time trace of of HU depicted in panel E). **I)** Activity time trace of PC depicted in panel F).

the weights that are nonzero encompass two strong, symmetrical distribution tails. The salience of this result becomes apparent in comparison to PCA, that creates a non-sparse mapping (only 20% are near-zero, see zoom inset in figure 3A).

Strong weights are typically assigned to active neurons: The distribution of summed absolute weights per neuron $\sum_\mu w_{i,\mu}$ correlates with mean activity $\langle v_i \rangle$ (Pearson correlation coefficient 0.74) and with variance $Var(v_i)$ (Pearson correlation coefficient 0.76) (data not plotted). Hence, hidden units can facilitate neurons with different activation profiles.

Next, we asked whether this sparse set of weights is distributed uniformly across all neurons, or confined to a subset of highly connected neurons. Figure 3B shows the percentage of neurons that have at least one weight with a magnitude $|w_{i,\mu}|$ greater than a threshold θ that is varied along the x-axis (thick purple line). This is compared to a uniformly randomly shuffled weight distribution (thick dashed black line), and the difference between these two distributions is found to be statistically insignificant (Kilmogorov-Smirnov (KS) test, with $P > 0.05$, see Methods and figure legend). This result was reproduced for equivalent analyses of distributions for visible units with at least 2, 3, 5 or 10 weights greater than θ (thin lines, figure 3B). This means that RBM weights are distributed uniformly, assuring that no substantial subset of the neuronal population is detached from the hidden layer.

This result is not generic: in comparison, PCA weights differed significantly from their uniformly shuffled versions (figure 3C, KS test, all $P < 0.001$), meaning that all distributions significantly differ from their uniformly shuffled equivalent.

Even though RBM weights are distributed uniformly across neurons, they are not connected randomly. To test this, we consider the projections to the visible layer per hidden unit. To assess the local density of neurons that strongly connect to a hidden unit, we compute the pairwise Euclidean distance $d_{i,j}$ between all pairs of neurons i, j and weigh these by a factor $\omega_{i,j} = w_{i,\mu} \cdot w_{j,\mu}$. In figure 3D the distribution of weighted connections $\omega_{i,j} \cdot d_{i,j}$ is shown for one hidden unit (figure 3E) and one principal component (figure 3F). The weighted distribution is shown for both the actual neurons (thick lines) and coordinate-shuffled data (dashed lines) for both RBM and PCA.

This particular hidden unit and principal component were hand-selected to resemble each other structurally, as can be seen from the midline-symmetric nuclei in red (figure 3E/F). However because the RBM is much sparser, there is less scattering of other medium-connected neurons, and the weights of the strongly-connected neurons heavily outweigh the weakly-connected neurons compared to PCA. As a result, the RBM distribution differs from its shuffled distribution, with two prominent peaks at the within-cluster distance and between-cluster distance. Meanwhile, PCA overlays with its shuffled distribution. This is quantified by the KS statistics of all 70 hidden units and principal components, whose P values are plotted in a bar histogram in figure 3G. This shows that the majority of RBM hidden units have a significantly different density compared to their shuffled distribution, while the majority of principal components do not.

Taken together, these results show that RBM weights 1) are sparse, 2) distribute uniformly across all neurons but 3) connect to the hidden layer non-randomly. In fact, hidden units project to

sparse, locally dense ensembles of neurons (figure 3E). More example units are shown in figure S3.

Because RBMs are probabilistic models, the dimensionality reduction is not deterministic, but given by $P(\mathbf{h}|\mathbf{v})$. However, we can compute the most likely $\mathbf{h}(t)$ to obtain a single (best) estimate, as derived earlier. For PCA, dimensionality reduction is deterministic (see Methods), so a single estimate is readily available. Figures 3H/I show two such example traces, of the hidden unit of 3E (in 3H) and the principal component of of 3F (in 3I). Because both latent representations have partial overlap (e.g. the aforementioned symmetric nuclei in red), the time traces exhibit transient peaks at the same time points (with inverted signs). However, crucially, the hidden unit's time trace remains very stable at a baseline value (3H), unless it is activated for either a transient or a sustained period, while the principal component fluctuates heavily between peaks (3I). This difference is caused by the difference in sparsity - because the principal component has non-sparse weights it receives additional input other than the coordinated peak activity. Regardless of the physiological context of these signals, it is clear that the hidden unit represents an on/off state, while the principal component shows a continuously changing signal. Both represent the same core ensemble of neurons, so the RBM is much more effective at distilling certain parts of information and suppressing others. This is typical, as evidenced by more examples of hidden units in figure S3.

Prediction of Neuronal Dynamics

To further gauge model accuracy, we investigate whether RBMs are able to predict the binary neuronal dynamics $\mathbf{v}(t)$. An established method to predict binary dynamics is logistic regression, a Generalized Linear Model (GLM) that estimates the probability $p(t) = P(x_t = 1)$, using a sigmoid transfer function that ensures that $p(t) \in (0, 1)$ (see Methods for details). We use an optimized GLM to estimate the maximally achievable performance by full regression, i.e. by regressing each neuron $v_i(t)$ against all others $\mathbf{v}(t-1)$ with a single time step delay. This corresponds to the function g_1 in figure 4A. This is a computationally hard task, because the number of parameters scales with N^2 and the number of time points is small compared with the number of parameters.

Using RBMs, we can also perform prediction on the visible units, via the low-dimensional bottleneck. If the latter captures the essential dynamics, it could provide a more noise-resistant, more performant representation. This is exactly what we find: for this purpose, each visible state $\mathbf{v}(t)$ is first transformed to its low-dimensional latent representation $\mathbf{h}(t)$, after which the next hidden state $\mathbf{h}(t+1)$ is predicted, that is transformed to $\mathbf{v}(t+1)$ via the inverse dimensionality reduction (illustrated by $f \rightarrow g_2 \rightarrow f^{-1}$ in figure 4A). Ordinary Least Squares linear regression is used to predict $\mathbf{h}(t+1)$ from $\mathbf{h}(t)$ (i.e. function g_2 in figure 4A). The dimensionality reduction f is performed with either RBM or PCA, and can trivially be inverted to f^{-1}. GLM and RBM directly estimate $p(t)$, but PCA does not. To facilitate comparison with PCA, the PCA-reconstructed signal $\mathbf{v}(t+1)$ is transformed to a probability $p(t)$ by using a sigmoid transfer function with two free parameters that are optimized, this is referred to as sigmoid-PCA (s-PCA) (see Methods and figure S1D for details).

The quality of the reconstructed dynamics is quantified by the log likelihood LLH of the predicted dynamics of unseen test data (see Methods). Figure 4B shows the distributions of LLH of all neurons for RBM, GLM and s-PCA. Performance of RBM and GLM is equivalent (not significantly different in KS test), even though RBM forces the dynamics to go through a M-dimensional

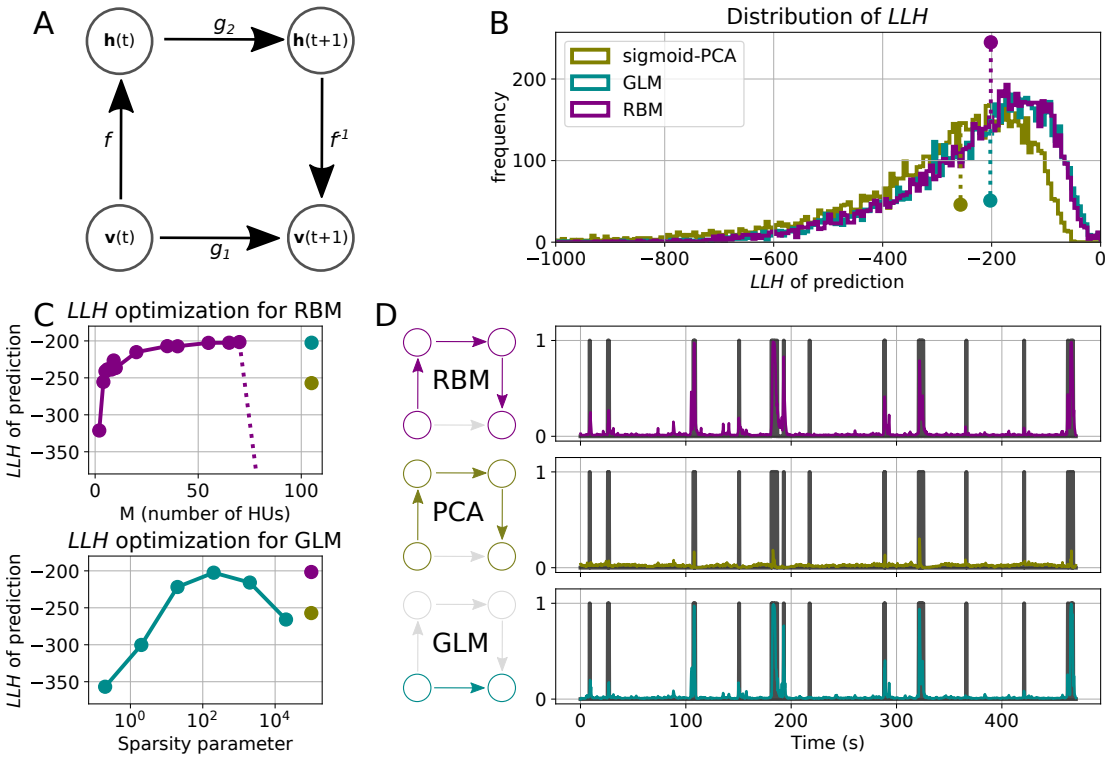

Figure 4: **Bottle-neck prediction via RBM exceeds PCA and matches GLM prediction. A)** A schematic of the two possible routes to predict $\mathbf{v}(t+1)$ from $\mathbf{v}(t)$. Dynamic prediction is performed with the regression functions g_1 and g_2. GLM computes the full visible-to-visible prediction with logistic regression as g_1. Alternatively, $\mathbf{v}(t)$ is transformed to its low-dimensional representation $\mathbf{h}(t)$ via f. Regression is performed in the hidden layer via g_2, and the predicted hidden signal $\mathbf{h}(t+1)$ is transformed back to the visible layer $\mathbf{v}(t+1)$ via the inverse dimensionality reduction f^{-1}. **B)** The distributions of LLH of all neurons for sigmoid-PCA (s-PCA), GLM and RBM. The medians are indicated by the circle. The distribution of s-PCA differs significantly from RBM (KS test, P value 0.002) and from GLM (KS test, P value 0.005), but RBM and GLM do not significantly differ (KS test, P value 1.00). **C)** Top: Optimization of RBM performance, by evaluating the median LLH as a function of M. The LLH of high M were respectively $LLH(M = 95) = -742$, $LLH(M = 100) = -633$ and are left out for visibility (hence the dotted line). Bottom: Optimization of GLM performance, by evaluating the median LLH as a function of the L_2 sparsity regularization parameter. Top RBM performance was achieved for $LLH(M = 70) = -202$ and for GLM also $LLH(200) = -202$. s-PCA is optimized in figure S1D, and the top models of each category are shown in figure 4B. **D)** Predicted dynamics for one example trace (top: RBM, middle: s-PCA, bottom: GLM).

bottleneck, while PCA performs worse than both RBM and GLM (respective KS test P values of 0.002 and 0.005).

The performance of RBM was evaluated against the number of hidden units M in figure 4C (top panel). The RBMs show a smooth, convergent increase of LLH for increasing M. Even though it can be argued that the statistics of figure 2C converge earlier than $M = 70$, dynamic prediction in figure 4C continues to improve. Figure 4C (bottom panel) shows the optimization of the GLM against its L_2 sparsity regularization parameter. Hence, GLM has reached its maximum performance, while RBM performance could be further improved by incorporating more delays in the regression model g_2 (as in a multivariate autoregressive model). This is (computationally) possible because of the low number of dimensions, while it is not for the high-dimensional GLM.

The dynamic prediction of one representative example neuron is shown in figure 4D for GLM, RBM and PCA.

Strikingly, the predictive power of s-PCA is fully dependent on PC autocorrelations. When regression g_2 was repeated without autoregression (i.e. PC_μ is regressed against $\mathbf{PC}_{\nu\neq\mu}$), s-PCA typically predicts a flat line (evidenced by the average variance of predicted neuronal dynamics decreasing from $1.1 \cdot 10^{-2} \rightarrow 9.2 \cdot 10^{-6}$), and its median likelihood decreases from $LLH = -257 \rightarrow LLH = -297$. RBM, however, is robust and shows no dependence on autocorrelation: RBM log likelihood remains at $LLH = -202$ when regressed without autocorrelation. This demonstrates that the nature of the hidden layer regression is very different between RBM and PCA.

Discussion

Large-scale data analysis methods have become quintessential in modern neuroscience. To this end, neuroscience increasingly benefits from machine learning methods (Glaser et al., 2019), that adopt general systems that learn to recreate a biological network and its functioning, such as feature extraction in vision (Güçlü and van Gerven, 2015; Yamins and DiCarlo, 2016) or the emergence of grid cells for navigation tasks (Banino et al., 2018). Furthermore, maximum entropy models that directly model pairwise connectivity between neurons (i.e. with Ising models) have frequently been used to describe collective neural activity (Schneidman et al., 2006; Watanabe et al., 2014; Tavoni et al., 2017; Posani et al., 2017; Meshulam et al., 2017; Cocco et al., 2017), however, this pairwise interaction model design is computationally infeasible for large systems such as zebrafish LSM recordings.

In the current study we leverage RBMs to create a mapping from neural data to a low-dimensional latent space, motivated by recent successes in other domains (Tubiana and Monasson, 2017; Tubiana et al., 2018). Previously, Köster et al. (2014) used an RBM to model a system of 10 neurons, including 10 delayed time steps per neuron, and showed that modeling higher-order interactions outperformed the second-order Ising model. Plis et al. (2014) use a L_1-sparse RBM to decompose fMRI data, and show equivalent performance to ICA.

In this study we have focused our comparison on PCA instead, and have successfully demonstrated that the RBM application is superior to PCA in terms of statistics, structure and dynamics. Figure 2 shows that the model has captured the statistics of individual and collective neural activity, and new data can be generated fulfilling the same characteristics. Figure 4 strengthens this statement, by showing that RBMs are also able to predict neuronal dynamics equally accurate as a fully-connected, optimized GLM, even though activity is fed through a low-dimensional bottleneck. This is in stark contrast to PCA, which fails to predict dynamics.

The RBM bottleneck consists of hidden units, whose strong connections have been shown to be uniformly distributed, so that no significant proportion of neurons is detached from the hidden layer. Due to their sparsity of weights with non-random connectivity, hidden units facilitate insightful read-out. Hidden unit S3A strongly resembles elements of the Hindbrain Oscillator (Ahrens et al., 2013; Dunn et al., 2016; Wolf et al., 2017). It encompasses two midline-symmetric, mutually inhibiting nuclei of neurons. Its activity displays alternating periods of up and down

activity, with oscillation periods in accordance with previously found results (Wolf et al., 2017). Other examples of hidden units show diverse temporal signatures, such as frequent, transient peaks (3H) rare, transient peaks (S3B), frequent, sustained peaks (S3D) or mixed signals (S3C). Because we show that these hidden units constitute a functionally and structurally valid dimensionality reduction, these emergent latent structures are likely to account for real physiological processes.

In conclusion, our RBM approach discovers a low-dimensional representation of a large neural data set that learns to accurately match multiple key facets of the neural system. This suggests that the constructed latent space is biologically relevant - contrary to PCA whose orthogonal components prevent interaction at the mesoscale, though crucial for neural processing. Our results suggest that a more accurate representation can be constructed with RBMs, which could yield central insights of key mechanisms that determine neural functioning.

Acknowledgements

Over the past year, I have been incredibly fortunate to work with a large group of extraordinarily bright and motivated people, and the work described in this thesis was done together with them. I deeply thank my supervisors, Bernhard and Georges, for their close, time-consuming and tireless supervision, that has propelled my (scientific) ambition and curiosity. I have enormously learned from and enjoyed our many lengthy discussions about science, and life outside of science. In practice, I have always felt as if I had four supervisors: I am very thankful to Volker for his unparalleled energy and inviting us to join his super exciting line of research, and to Rémi Proville, for at least hundred hours on skype, and two hours in person(!), of detailed advice.

The topic of this thesis essentially combines the hard experimental work of Guillaume and Geoffrey, and the theoretical developments of Jérôme and Rémi Monasson, and I feel very fortunate to be a part of their work.

In Paris, I have been most welcomed by the LJP zebrafish team, taken to the strangest choice of pub many times, and I could not have asked for more - Georges, Volker, Geoffrey, Guillaume, Natalia, Sophia, Hugo, Benjamin, Thomas, Raphaël and all other LJP students and staff, it was fantastic - I thank you! In Nijmegen, I have enjoyed tremendous help from a small army of friends, family and colleagues, during all stages of my project: Han, Tim, Freek, Lisanne, Karol, Lando, Séba, Oli, Lennie, Maarten, Arthur, Uphoff, Jojo, Luka and Fons in particular have been especially supportive.

Methods

Data acquisition Spontaneous (i.e. in the absence of sensory stimulation) neural recordings were obtained using Light-Sheet Microscopy (LSM) of larval zebrafish (Panier et al., 2013; Wolf et al., 2015). The fluorescence intensity values F are normalized to $\Delta F/F = (F - \langle F \rangle)/(\langle F \rangle - F_0)$ where $\langle F \rangle$ is the baseline signal per pixel and F_0 is the overall background intensity. Automated cell segmentation is performed (Panier et al., 2013), which yields the $\Delta F/F$ calcium activity of 54334 cells. We identify the neurons in the anatomical subset Rhombomere 1, as defined by the ZBrainAtlas (Randlett et al., 2015). This yields a population of 7933 neurons, with 5553 consecutive $\Delta F/F$ measurements in time with a sampling rate of $3.3Hz$.

Spikes (i.e. action potentials) underlie calcium activity and their relation is commonly estimated by an exponential decay function (Yaksi and Friedrich, 2006; Friedrich et al., 2017; Tubiana et al., 2017). We use Blind Sparse Deconvolution (BSD) to deconvolve calcium data to spikes, by fitting spike trains with both a fast exponential rise function and a slow exponential decay function. BSD estimates the most likely binary spike train by minimizing the L_2 norm of the difference between the convolved estimated spike train and the true calcium data, via L_1 sparsity regularization and online hyperparamater optimization (Tubiana et al., 2017). This procedure transforms the continuous calcium signal into a binary spike signal (i.e either spike (1) or no spike (0)). The calcium signal is heavily autocorrelated due to its intrinsic (exponential decay) dynamics: the average autocorrelation with a single time step delay $\rho_1(x(t)) = r(x_{1:T-1}(t), x_{2:T}(t)) = 0.88$ for calcium dynamics (of all neurons in Rhombere 1). Spike deconvolution reduces autocorrelation to an average of $\rho_1 = 0.22$, which may well be a realistic value based on the dynamics alone..

Data visualization It is notoriously difficult to efficiently visualize high-dimensional data. LSM records functional activity of a very large number of neurons, that are simplified (by cell segmentation) to points in space. Visualizing these functional data in matrix format eradicates anatomical structure, while showing structural images hinders functional insights. To overcome these limitations we have previously developed a 4D (space + time) functional data viewer (the *Fishualizer*) that preserves spatial structure of neurons (Migault et al., 2018). This was used in figures 1, 3E and S2.

Principal Component Analysis Principal Component Analysis (PCA) transforms the data $\mathbf{V} = \mathbf{v}_{t=1:T}$ into a mutually orthogonal set of principal components $\mathbf{PC} = PC_{k=1:K}$ which are sorted by explained variance (of $\mathbf{V}$) (Bishop, 2006). We set $K = 70$ to match the number of hidden units $M = 70$ of the example RBM for all considered analyses. PCA is achieved by an Eigenvalue decomposition of the covariance matrix of the training data $\mathbf{V}$. Hence we acquire a linear mapping between $\mathbf{PC} = W \cdot \mathbf{V}$. Figure S1B shows the cumulative explained variance (of $\mathbf{V}$) as a function of the number of principal components K.

For the analysis of figure 4, we extend our PCA model to enable dynamic prediction. To obtain an estimate that a spike occurs at time t, $p(t) = P(x_t = 1)$, we pass the reconstructed signal $v_i = W_i^{-1} \cdot \mathbf{PC}$ through a sigmoid function $\sigma(x) = 1/(1 + \exp(-a \cdot (x + b)))$. This function has two free parameters a and b that can transpose and scale the signal if necessary. These were optimized through a dual parameter sweep (see figure S1D) and were found to maximize the Log likelihood (LLH) at $a = 13,\ b = -3.25$.

Maximum Entropy modeling The Maximum Entropy model $P(\mathbf{x})$ of state $\mathbf{x}$ is defined as $P(\mathbf{x}) = \frac{1}{Z}\exp\left(-E(\mathbf{x})\right)$, where the energy E incorporates all imposed constraints. Subsequently the entropy of $P(\mathbf{x})$ is maximized, which yields an expression for P as function of the constraints (see Appendix A for a full derivation). This general framework can be combined with any graphical model design (Gardella et al., 2019), including the bipartite structure of the RBM.

Restricted Boltzmann Machines Restricted Boltzmann Machines (RBMs) incorporate a bipartite graphical model design (figure 1B right panel), consisting of a visible layer, containing the deconvolved binary activity of all neurons $\mathbf{v} = (v_1, v_2, ..., v_N)$ and a hidden layer containing the continuous hidden units $\mathbf{h} = (h_1, h_2, ...h_M)$. The total state of the system at time t is $\mathbf{x}_t = (\mathbf{v}_t, \mathbf{h}_t)$ and is modelled by $P(\mathbf{x})$ (continuing from the Maximum Entropy equation S15):

$$P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z}\exp\left(-E(\mathbf{x})\right) = \frac{1}{Z}\exp\left(\sum_i g_i v_i - \sum_\mu U_\mu(h_\mu) + \sum_{i,\mu} w_{i,\mu} v_i h_\mu\right) \tag{3}$$

Note that (because of the bipartite structure) visible or hidden units can readily be marginalized by:

$$P(\mathbf{v}|\mathbf{h}) = \prod_i P(v_i|\mathbf{h}) \propto \prod_i \exp\left(g_i v_i + \sum_\mu w_{i,\mu} v_i h_\mu\right) \tag{4}$$

$$P(\mathbf{h}|\mathbf{v}) = \prod_\mu P(h_\mu|\mathbf{v}) \propto \prod_\mu \exp\left(-U_\mu(h_\mu) + \sum_i w_{i,\mu} v_i h_\mu\right) \tag{5}$$

This results in an intuitive and fast Monte Carlo sampling procedure, as depicted in figure S1A, where $\mathbf{v}^{k+1} \sim P(\mathbf{v}|\mathbf{h}^k) = (v_i^{k+1} \sim P(v_i|\mathbf{h}^k) \ \ \forall\, i)$, and vice versa for $\mathbf{h}^{k+1}$. Double-rectified linear units are used as hidden unit prior U_μ, because it can flexibly be tuned to allow hidden units to model higher-order couplings between visible neurons (i.e. neurons) (Tubiana and Monasson, 2017; Tubiana et al., 2018).

This probabilistic model $P(\mathbf{x})$ is generative because it can generate new data by the aforementioned sampling procedure (i.e. $\mathbf{v}^{k-1} \rightarrow \mathbf{h}^k \rightarrow \mathbf{v}^k \rightarrow \mathbf{h}^{k+1}$). The RBM must be trained to fit the empirical data by tuning its parameters accordingly, and exploits this generative property for this purpose. Here, new data $\{\mathbf{v}^1, \mathbf{v}^2 ...\}$ is generated from an initial real data state $\mathbf{v}^0$, so that its divergence from $\mathbf{v}^0$ is used in a log likelihood gradient descent step that updates the weights $\mathbf{w}$, visible fields $\mathbf{g}$ and transfer function hyperparameters $\{\theta_U\}_{\mu=1,..,M}$.

Additionally, sparsity regularization is included in the gradient descent step to avoid overfitting and to bias the solution toward a sparse weight representation. This results in a mapping between the visible and hidden layer that strongly activates only a limited number of hidden units per visible state, while diminishing others. This is called the compositional phase of the model (Tubiana and Monasson, 2017) and greatly facilitates interpretability by delineating visible layer activity into a defined subset of hidden units. Specifically, a L_1^2 term $\propto \sum_\mu \left(\sum_i |w_{i,\mu}|\right)^2$ is added to the cost function (the log likelihood of the data $LLH(\mathbf{v}, \mathbf{h}) = \log\left(\prod_k P(\mathbf{v}^k, \mathbf{h}^k)\right)$):

$$Cost = C = LLH(\mathbf{v}, \mathbf{h}) + \lambda \cdot \sum_\mu \left(\sum_i |w_{i,\mu}|\right)^2 \tag{6}$$

where λ is a learning rate that geometrically decays during training. The gradient descent step is defined by $\Delta w_{i,\mu} = \eta \cdot \frac{\delta C}{\delta w_{i,\mu}}$. The initial sparsity parameter λ was determined by log-likelihood cross validation (where the model was trained on 4000 samples and evaluated on 1553 samples), for $M = 50, 100, 250$ hidden units (see supplemental figure S1C). This yielded an optimal sparsity parameter, that was used throughout the rest of this study, of $\lambda = 0.008$.

Dynamic prediction The prediction of deconvolved time traces was evaluated as follows. Let $p(t)$ be the continuous instantaneous firing probability with $p_t \in [0, 1]$. The likelihood function of one particular time trace $x(t)$ with discrete $x_t \in \{0, 1\}$ is (Bishop, 2006):

$$P(x_{t=1:T}) = \prod_{t=1}^{T} (x_t \cdot p_t + (1 - x_t) \cdot (1 - p_t)) \tag{7}$$

Hence the log likelihood $LLH(x, \hat{x})$ of a continuous estimate $\hat{x}(t)$ with $\hat{x}_t \in [0, 1]$. is given by:

$$LLH(x, \hat{x}) = \sum_t \log (x_t \cdot \hat{x}_t + (1 - x_t) \cdot (1 - \hat{x}_t)) \tag{8}$$

Logistic regression We used logistic regression (Bishop, 2006) to obtain baseline performance, from an established, straightforward method for full (i.e. all neurons versus all neurons) dynamic prediction of neuronal activity. Logistic regression is a Generalized Linear Model (GLM) that estimates $\hat{\mathbf{x}}$ by $\hat{\mathbf{x}} = \phi(\beta \cdot X)$ where $\phi(x) = 1/(1 + \exp(-x))$ is the sigmoid function such that all $\hat{\mathbf{x}} \in (0, 1)$ (i.e. it convolves a linear regression $\beta \cdot X$ with a nonlinear transfer function ϕ). It trains by gradient descent with respect to the likelihood function (equation 8). To avoid overfitting we used L_2 norm (i.e. Ridge) regression, with a sparsity parameter that was numerically optimized for best cross-validation performance (figure 4C bottom panel).

Accounting for neural activity bursts LSM recordings of spontaneous neural activity contain transients bursts of activity that activate the majority of neurons in Rhombomere 1. This is thought to be induced externally, e.g. a motor efference copy of swimming movement (to escape the agarose the sample is encapsulated in). In maximum entropy models, high activity states are assigned low probability because of the high (combinatorial) number of possible configurations that can account for high activity (maximum at $\sum_i^N v_i = N/2$) (Schneidman et al., 2006; Meshulam et al., 2017). We observe the same effect - these activation bursts are ascribed a very low log likelihood. As a consequence, this state subspace is very unlikely to be sampled from in the model (during Monte Carlo sampling), and hence the RBM does not reproduce these bursts of activity. This is in correspondence with the presumption that bursts are activated outside of the Rhombomere 1 system. As a result, model statistics (of figure 2) would be biased because the experimental data contains bursts (which yields slightly higher average activity), while generated data does not.

We quantify activity bursts by determining the maximum curvature in the sorted population activity curve. This is determined by finding the maximum distance between the data and its secant (the vector from the first to last data point), see figure S2A left panel. Using this threshold the activity bursts are effectively filtered (figure S2A right). Regardless, we train all RBMs with all data available, but perform the statistical evaluation in figure 2 (but not in other figures) with exclusion of activity bursts. The equivalent of figure 2C that includes bursts is shown in figure

S2B. Correlation coefficients are generally smaller when bursts are included, but approximately retain relative performance as a function of M.

Statistical tests We use the two-sided Kilmogorov-Smirnov (KS) test to assess whether two distribution density functions $f_1(x)$ and $f_2(x)$ significantly differ. We find the cumulative distribution functions $F_i(x) = \sum_{y=-\infty}^{x} f_i(y)$ (because f_i is discrete), after which test statistic $D_{1,2}$ is defined as: $D_{1,2} = \sup_x |F_1(x) - F_2(x)|$. The null hypothesis that f_1 and f_2 are samples from the same distribution is rejected with significance level α if $D_{1,2} > \sqrt{-\frac{1}{2}\ln(\alpha)(\frac{N_1+N_2}{N_1 N_2})}$ where N_i is the sample size of f_i.

Code All modeling and data analysis (post processing) was done in Python 2.7 and 3.7, using various standard libraries (numpy, scipy, sklearn, pandas). The RBM was adapted from (Tubiana and Monasson, 2017; Tubiana et al., 2018) where it was used for random RBMs and protein analysis.

Supplementary Figures

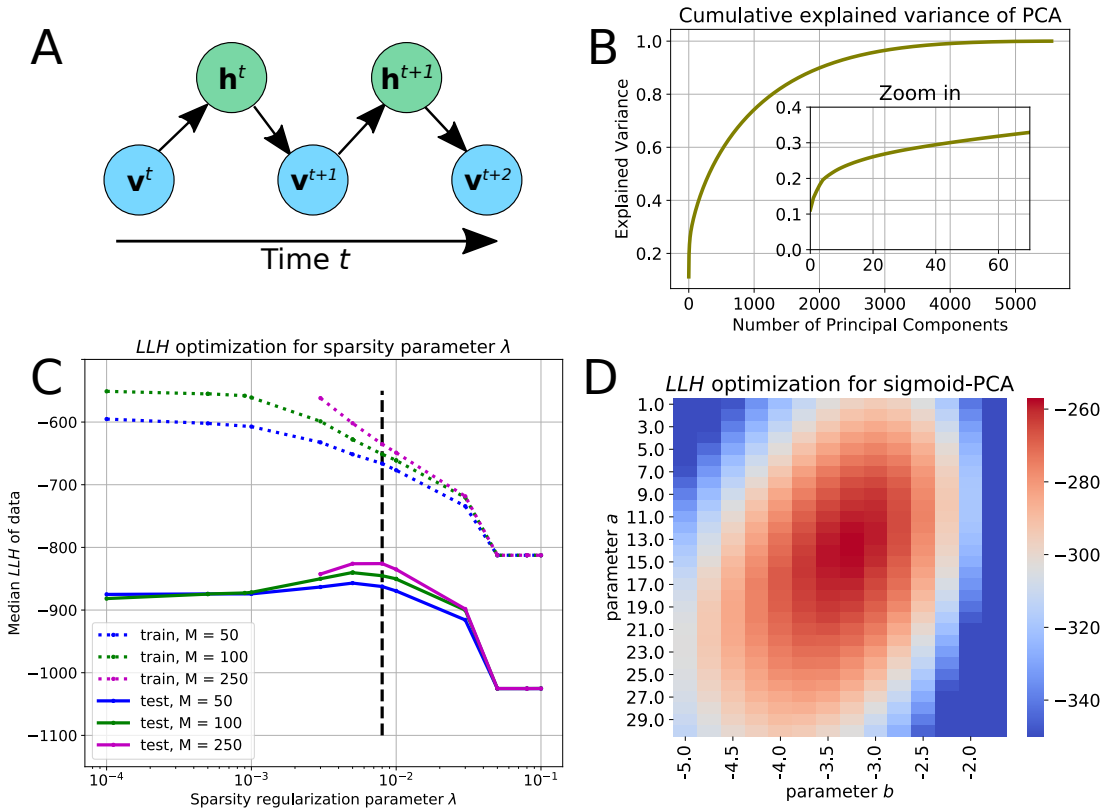

Figure S1: **Additional methods details A)** Schematic illustration of Monte Carlo sampling procedure. Because of the bipartite graphic design of the RBM, the visible layer can be sampled as $P(\mathbf{v}|\mathbf{h})$ and the hidden layer as $P(\mathbf{h}|\mathbf{v})$. **B)** Cumulative explained variance of the PCA decomposition of the data for all components and a zoom-in on 70 components (inset). **C)** Sparsity regularization optimization of RBMs. The sparsity parameter λ is varied along the x-axis, and the likelihood of three different RBMs ($M = 50, 100, 250$) is evaluated (y-axis). The likelihood is computed based on equation 1, for training data (dotted lines) and test data (solid lines). The vertical dotted black line indicates $\lambda = 0.008$, the extracted parameter value. It was chosen because it is the largest of two (adjacent) λ values that maximize the three test LLH curves. **D)** Optimization of the two free parameters a and b for sigmoid-PCA (as defined in text). The optimum was found for $LLH(a = 13, b = -3.25) = -257$.

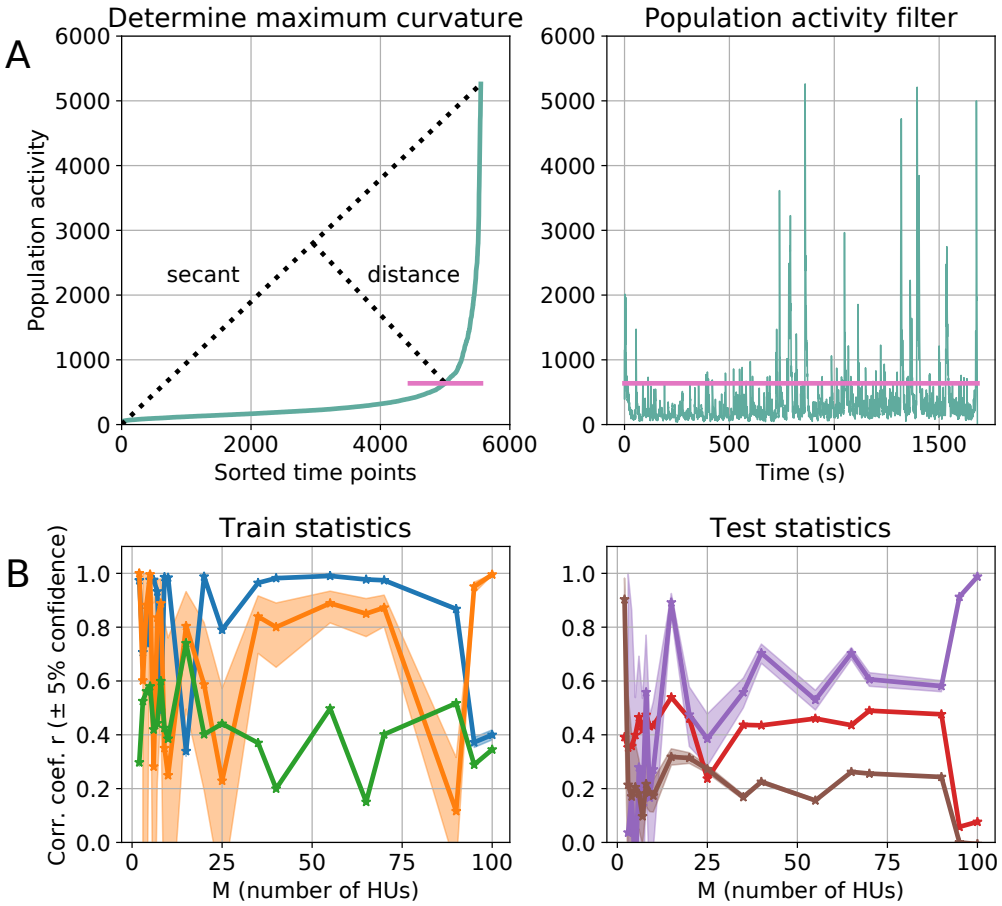

Figure S2: **Activity burst filtering A)** Automatic detection of threshold to filter activity bursts. All time points are sorted by population activity (pale green), and to locate the maximum curvature in the curve, the point is sought that maximizes the parallel distance to the secant of the curve (dotted lines). This yields a population activity threshold value of 637 spikes (magenta) that is used to filter bursts by excluding all time points with population activity greater than the threshold. **B)** Equivalent statistics of figure 2C, including bursts. The same color code as figure 2 is used.

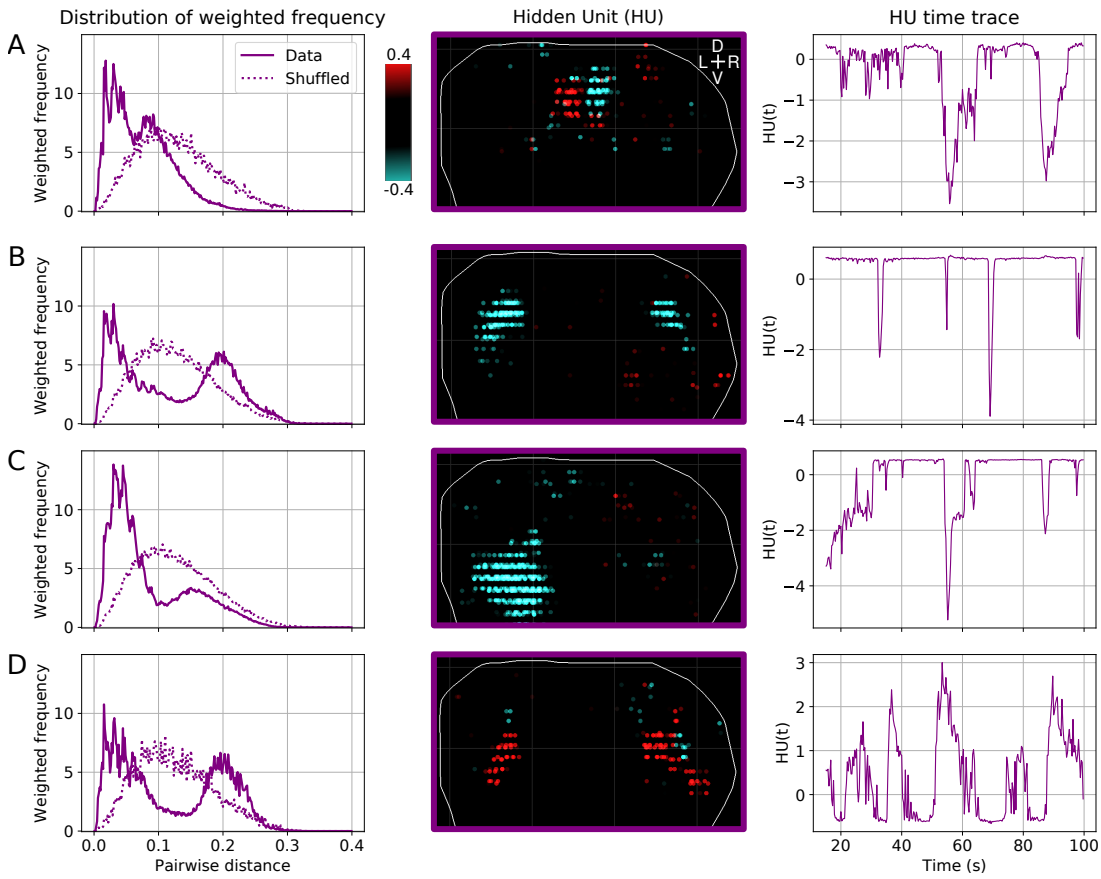

Figure S3: **Four examples of hidden units, in addition to figure 3.** The distribution of weighted pairwise distances (left), projection to visible units (middle) and activity time trace (right) are shown.

Appendix A: Derivation of Maximum Entropy model

This appendix extensively derives the general Maximum Entropy model (Bialek, 2012). The entropy $H = H(x)$ for a state x is defined as:

$$H = \sum_x P(x) \log\left(\frac{1}{P(x)}\right) = -\sum_x P(x) \log\left(P(x)\right) \tag{S1}$$

where log indicates a log base 2. For every function $f_k(x)$ of the state x (where k indicates the k-th function), we can write the expectation value of the model $\langle f_k \rangle_{P(x)}$ and of the data $\langle f_k \rangle_{\text{Data}}$ as:

$$\langle f_k \rangle_{P(x)} = \sum_x P(x) f_k(x), \qquad \langle f_k \rangle_{\text{Data}} = F_k \tag{S2}$$

where F_k is a scalar (because it is directly calculated from the data). Next we constrain that the expectation value of the model should be equal to the empirical expectation value:

$$\langle f_k \rangle_{P(x)} = \langle f_k \rangle_{\text{Data}} \iff \sum_x P(x) f_k(x) - F_k = 0 \tag{S3}$$

For example, a general constraint is:

$$\sum_x P(x) = 1 = F_0, \quad f_0(x) = 1 \tag{S4}$$

For mean activity this constraint becomes:

$$\sum_x P(x) x_i = \langle x_i \rangle = F_{1,i}, \quad f_{1,i}(x) = x_i \tag{S5}$$

For second order moments (pairwise interactions), the constraint becomes:

$$\sum_x P(x) x_i x_j = \langle x_i x_j \rangle = F_{2,ij}(x), \quad f_{2,ij}(x) = x_i x_j \tag{S6}$$

The model $P(\mathbf{x})$ is constructed to be maximally unconstrained otherwise, by maximizing its entropy H, given the constraints $\langle f_k \rangle = F_k$. This is achieved by adding the constraints as Lagrange multipliers with parameter λ_k;

$$\widetilde{H} = -\sum_x P(x) \log\left(P(x)\right) - \sum_k \lambda_k \left(\sum_x P(x) f_k(x) - F_k\right) \tag{S7}$$

$$= -\frac{1}{\ln 2} \sum_x P(x) \ln\left(P(x)\right) - \sum_k \lambda_k \left(\sum_x P(x) f_k(x) - F_k\right) \tag{S8}$$

We seek the solution $P(x)$ that maximizes $\widetilde{H}$. This is found by taking the functional derivative of $\widetilde{H}$ with respect to the function $P(x)$ to find the maximum. Change of variable yields:

$$\frac{\delta \widetilde{H}}{\delta P(x)} = \frac{\delta}{\delta P(x)} \left(\frac{-1}{\ln 2} \sum_y P(y) \ln\left(P(y)\right) - \sum_k \lambda_k \left(\sum_y P(y) f_k(y) - F_k\right)\right) \tag{S9}$$

The functional derivative $\frac{\delta}{\delta P(x)}$ is solved by incorporating the kronecker delta $\delta_{i,j} = 1 \iff i = j, \;\; otherwise \;\; \delta_{i,j} = 0$.

$$= \left(\frac{-1}{\ln 2} \sum_y \left(\ln P(y) + 1 \right) \delta_{x,y} - \sum_k \lambda_k \left(\sum_y f_k(y) \delta_{x,y} \right) \right) \tag{S10}$$

Subsequently the kronecker delta's are eliminated by resolving the sum (that includes $y = x$):

$$= \left(\frac{-1}{\ln 2} \left(\ln P(x) + 1 \right) - \sum_k \lambda_k \left(f_k(x) \right) \right) \tag{S11}$$

So we find:

$$\frac{\delta \widetilde{H}}{\delta P(x)} = -\frac{1}{\ln 2} \left(\ln \left(P(x) \right) + 1 \right) - \sum_k \lambda_k f_k(x) = 0 \tag{S12}$$

$$\ln \left(P(x) \right) = -\ln 2 \sum_k \lambda_k f_k(x) - 1 \tag{S13}$$

$$P(x) = \exp \left(-\ln 2 \sum_k \lambda_k f_k(x) - 1 \right) = \frac{1}{e} \exp \left(-\ln 2 \sum_k \lambda_k f_k(x) \right) \tag{S14}$$

$$= \frac{1}{Z} \exp \left(-\sum_k \lambda'_k f_k(x) \right) = \frac{1}{Z} \exp \left(-E(x) \right) \tag{S15}$$

where $Z = \sum_x \exp(-E(x))$ to normalize the probability distribution. This is the Boltzmann distribution and defines the maximum entropy model $P(x)$.

References

Ahrens, M. B., Orger, M. B., Robson, D. N., Li, J. M., and Keller, P. J. (2013). Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nature methods*, 10(5):413–420.

Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., Pritzel, A., Chadwick, M. J., Degris, T., Modayil, J., et al. (2018). Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705):429–433.

Bassett, D. S. and Sporns, O. (2017). Network neuroscience. *Nature neuroscience*, 20(3):353 – 364.

Benjamin, A. S., Fernandes, H. L., Tomlinson, T., Ramkumar, P., VerSteeg, C., Chowdhury, R. H., Miller, L. E., and Kording, K. P. (2018). Modern machine learning as a benchmark for fitting neural responses. *Frontiers in computational neuroscience*, 12.

Bialek, W. (2012). *Biophysics: searching for principles*. Princeton University Press.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Chen, X., Mu, Y., Hu, Y., Kuan, A. T., Nikitchenko, M., Randlett, O., Chen, A. B., Gavornik, J. P., Sompolinsky, H., Engert, F., et al. (2018). Brain-wide organization of neuronal activity and convergent sensorimotor transformations in larval zebrafish. *Neuron*, 100(4):876–890.

Cimini, G., Squartini, T., Saracco, F., Garlaschelli, D., Gabrielli, A., and Caldarelli, G. (2019). The statistical physics of real-world networks. *Nature Reviews Physics*, 1(1):58–71.

Cocco, S., Monasson, R., Posani, L., and Tavoni, G. (2017). Functional networks from inverse modeling of neural population activity. *Current Opinion in Systems Biology*, 3:103–110.

Cunningham, J. P. and Yu, B. M. (2014). Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509.

Dunn, T. W., Mu, Y., Narayan, S., Randlett, O., Naumann, E. A., Yang, C.-T., Schier, A. F., Freeman, J., Engert, F., and Ahrens, M. B. (2016). Brain-wide mapping of neural activity controlling zebrafish exploratory locomotion. *Elife*, 5:e12741.

Friedrich, J., Zhou, P., and Paninski, L. (2017). Fast online deconvolution of calcium imaging data. *PLoS computational biology*, 13(3):e1005423.

Gallego, J. A., Perich, M. G., Miller, L. E., and Solla, S. A. (2017). Neural manifolds for the control of movement. *Neuron*, 94(5):978–984.

Gardella, C., Marre, O., and Mora, T. (2019). Modeling the correlated activity of neural populations: A review. *Neural computation*, 31(2):233–269.

Glaser, J. I., Benjamin, A. S., Farhoodi, R., and Kording, K. P. (2019). The roles of supervised machine learning in systems neuroscience. *Progress in neurobiology*, 175:126–137.

Grienberger, C. and Konnerth, A. (2012). Imaging calcium in neurons. *Neuron*, 73(5):862–885.

Güçlü, U. and van Gerven, M. A. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014.

Hafting, T., Fyhn, M., Molden, S., Moser, M.-B., and Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507.

Huang, C., Ruff, D. A., Pyle, R., Rosenbaum, R., Cohen, M. R., and Doiron, B. (2019). Circuit models of low-dimensional shared variability in cortical networks. *Neuron*, 101(2):337–348.

Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154.

Jun, J. J., Steinmetz, N. A., Siegle, J. H., Denman, D. J., Bauza, M., Barbarits, B., Lee, A. K., Anastassiou, C. A., Andrei, A., Aydın, Ç., et al. (2017). Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232.

Keller, P. J. and Ahrens, M. B. (2015). Visualizing whole-brain activity and development at the single-cell level using light-sheet microscopy. *Neuron*, 85(3):462–483.

Köster, U., Sohl-Dickstein, J., Gray, C. M., and Olshausen, B. A. (2014). Modeling higher-order correlations within cortical microcolumns. *PLoS computational biology*, 10(7):e1003684.

Meshulam, L., Gauthier, J. L., Brody, C. D., Tank, D. W., and Bialek, W. (2017). Collective behavior of place and non-place neurons in the hippocampal network. *Neuron*, 96(5):1178–1191.

Migault, G., van der Plas, T. L., Trentesaux, H., Panier, T., Candelier, R., Proville, R., Englitz, B., Debrégeas, G., and Bormuth, V. (2018). Whole-brain calcium imaging during physiological vestibular stimulation in larval zebrafish. *Current Biology*, 28(23):3723–3735.

O'Keefe, J. (1976). Place units in the hippocampus of the freely moving rat. *Experimental neurology*, 51(1):78–109.

Okun, M., Steinmetz, N. A., Cossell, L., Iacaruso, M. F., Ko, H., Barthó, P., Moore, T., Hofer, S. B., Mrsic-Flogel, T. D., Carandini, M., et al. (2015). Diverse coupling of neurons to populations in sensory cortex. *Nature*, 521(7553):511–515.

Panier, T., Romano, S. A., Olive, R., Pietri, T., Sumbre, G., Candelier, R., and Debrégeas, G. (2013). Fast functional imaging of multiple brain regions in intact zebrafish larvae using selective plane illumination microscopy. *Frontiers in neural circuits*, 7.

Plis, S. M., Hjelm, D. R., Salakhutdinov, R., Allen, E. A., Bockholt, H. J., Long, J. D., Johnson, H. J., Paulsen, J. S., Turner, J. A., and Calhoun, V. D. (2014). Deep learning for neuroimaging: a validation study. *Frontiers in neuroscience*, 8:229.

Posani, L., Cocco, S., Ježek, K., and Monasson, R. (2017). Functional connectivity models for decoding of spatial representations from hippocampal ca1 recordings. *Journal of computational neuroscience*, 43(1):17–33.

Randlett, O., Wee, C. L., Naumann, E. A., Nnaemeka, O., Schoppik, D., Fitzgerald, J. E., Portugues, R., Lacoste, A. M., Riegler, C., Engert, F., et al. (2015). Whole-brain activity mapping onto a zebrafish brain atlas. *Nature methods*, 12(11):1039–1046.

Schneidman, E., Berry II, M. J., Segev, R., and Bialek, W. (2006). Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007 – 1012.

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of harmony theory. Technical report, Colorado Univ at Boulder Dept of Computer Science.

Song, S., Sjöström, P. J., Reigl, M., Nelson, S., and Chklovskii, D. B. (2005). Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS biology*, 3(3):e68.

Tavoni, G., Ferrari, U., Battaglia, F. P., Cocco, S., and Monasson, R. (2017). Functional coupling networks inferred from prefrontal cortex activity show experience-related effective plasticity. *Network Neuroscience*, 1(3):275–301.

Tubiana, J., Cocco, S., and Monasson, R. (2018). Learning protein constitutive motifs from sequence data. *arXiv preprint arXiv:1803.08718*.

Tubiana, J. and Monasson, R. (2017). Emergence of compositional representations in restricted boltzmann machines. *Physical review letters*, 118(13):138301.

Tubiana, J., Wolf, S., and Debrégeas, G. (2017). Blind sparse deconvolution for inferring spike trains from fluorescence recordings. *bioRxiv*, page 156364.

Watanabe, T., Hirose, S., Wada, H., Imai, Y., Machida, T., Shirouzu, I., Konishi, S., Miyashita, Y., and Masuda, N. (2014). Energy landscapes of resting-state brain networks. *Frontiers in neuroinformatics*, 8.

Wolf, S., Dubreuil, A. M., Bertoni, T., Böhm, U. L., Bormuth, V., Candelier, R., Karpenko, S., Hildebrand, D. G., Bianco, I. H., Monasson, R., and Debrégeas, G. (2017). Sensorimotor computation underlying phototaxis in zebrafish. *Nature Communications*, 8(1):651.

Wolf, S., Supatto, W., Debrégeas, G., Mahou, P., Kruglik, S. G., Sintes, J.-M., Beaurepaire, E., and Candelier, R. (2015). Whole-brain functional imaging with two-photon light-sheet microscopy. *Nature methods*, 12(5):379–380.

Yaksi, E. and Friedrich, R. W. (2006). Reconstruction of firing rate changes across neuronal populations by temporally deconvolved ca2+ imaging. *Nature methods*, 3(5):377–383.

Yamins, D. L. and DiCarlo, J. J. (2016). Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, 19(3):356–365.