EFFICIENT APPROXIMATION OF NEURAL POPULATION STRUCTURE AND CORRELATIONS WITH PROBABILISTIC CIRCUITS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a computationally efficient framework to model a wide range of population structures with high order correlations and a large number of neurons. Our method is based on a special type of Bayesian network that has linear inference time and is founded upon the concept of contextual independence. Moreover, we use an efficient architecture learning method for network selection to model large neural populations even with a small amount of data. Our framework is both fast and accurate in approximating neural population structures. Furthermore, our approach enables us to reliably quantify higher order neural correlations. We test our method on simulated neural populations commonly used to generate higher order correlations, as well as on publicly available large-scale neural recordings from the Allen Brain Observatory. Our approach significantly outperforms other models both in terms of statistical measures and alignment with experimental evidence.

1 INTRODUCTION

With the rise and fast growth of simultaneous neural population recording, modeling population structures and measuring correlations has become a focus of computational neuroscience Abbott & Dayan (1999); Averbeck et al. (2006); Azeredo da Silveira & Rieke (2021); Urai et al. (2022). Theoretical and Experimental works have demonstrated the necessity of measuring population correlations to investigate information coding Moreno-Bote et al. (2014); Averbeck et al. (2006), functional connectivity Dunn et al. (2015), learning Ganmor et al. (2011), and arousal Vinck et al. (2015); Doiron et al. (2016). Despite significant progress in recent years, research on measurement and analysis of population correlations still faces significant challenges Kohn et al. (2016).

Exact measurement of population correlations is an NP-hard problem in the general case since it requires computing every form of dependency among spiking neurons. As a result, researchers have tried to come up with computationally efficient ways of approximation or indirect measurement of neural correlations. Existing approaches are energy-based models rooted in statistical mechanics where the energy function incorporates couplings between subsets of variables (here neurons) Roudi et al. (2009c); Tkačik et al. (2006); Sohl-Dickstein et al. (2011); Aurell & Ekeberg (2012). However, these methods often carry auxiliary (and even unrealistic) assumptions about the neural dynamics, do not scale up for large populations, or are limited to very specific situations Roudi et al. (2009b); Averbeck et al. (2006).

Notably, generative models commonly used in other domains such as latent variable methods are often not applicable to neural populations as spiking neural data is discrete and sparse Zhao et al. (2020). Furthermore, various parameters such as behavioural and emotional state of the animal affect firing patterns of neurons even in sensory cortex Urai et al. (2022). As a result, a recording long enough to train these models contains many external variable changes and confounding factors that make drawing scientific conclusions difficult.

In this paper, we take a probabilistic approach by modeling the joint probability distribution of neural activity with Bayesian networks. Inference is NP-hard in general/classic Bayesian networks Cooper (1990), making them impractical to model the population structure. Therefore, we utilize a special family of Bayesian networks with linear inference time, first introduced as "Arithmetic circuits" Darwiche (2003); Shen et al. (2016). This family of networks has been designed to take

advantage of "context-specific independence" of variables mainly for the purpose of computational efficacy, which also makes it suitable to extract local structures in the data Boutilier et al. (1996); Shen et al. (2020). We use a modification (and equivalent Rooshenas & Lowd (2014)) of arithmetic circuits, Sum-Product Networks (SPNs). Sum product networks are more known and used by the community due to their success in image completion tasks Poon & Domingos (2011); Sanchez-Cauce et al. (2021).

In particular, we design a method that learns the architecture of the sum-product network from the spiking neural data in order to capture a wide range of population correlation/structure from local to global in polynomial time. Due to the efficiency of our architecture learning and inference in SPNs, population structure estimation is polynomial in the size of the population. In addition, we suggest a measure of high order population correlations based on our framework.

Our results include fitting on neural population simulations constructed with higher order correlations, as well as large-scale neural recording in different brain regions on more than 20 mice. Our framework outperforms both of the best energy-based and latent variable models for neural population structure estimation.

2 Problem Definition and Related Work

One of the critical problems in computational neuroscience is providing an accurate statistical description of spike trains in a population of neurons. As the full representation of the data, i.e. raw spike times, is high dimensional, spike trains are binned into small time windows. The time bin should be short enough so each neuron spikes at most once in each bin (with some amount of tolerance in potentially losing some spikes). In addition, this time bin should be large enough that the assumption of temporal independence of spikes holds. With this time binning strategy, each neuron's activity is a binary variable (S_i for neuron i is equal to 1 if there is a spike in the corresponding bin, otherwise 0) and each time bin represents an i.i.d sample/instance. Therefore, spike trains of N neurons for the duration of T would be represented as a binary matrix $D_{K\times N}$ where $K = T/\Delta t$ in which Δt is the bin length (Figure 1, left plot).

Consequently, the population activity has a probabilistic representation $P(S_1, \ldots, S_N)$, and the problem turns into modelling this joint distribution, given the data. More specifically, the problem is to find a model m^* from a family of models M, and optimize its free parameters Θ_m so as to satisfy the following:

$$m^*, \theta^*_m = \underset{m\in M, \theta\in\Theta_m}{\arg\max} \frac{1}{K}\sum_{k=1}^{K} \log\left(P(d^k_{1\times N}|m,\theta)\right) \tag{1}$$

In the existing approaches, M is set to maximum entropy (Ising) models, which are energy-based methods rooted in statistical mechanics Roudi et al. (2009c); Schneidman et al. (2006). Since learning maximum entropy models is computationally very expensive, these models are restricted to estimate the statistical properties of the population up to a constant order. However, going beyond second order is not computationally feasible. In fact, building the exact generative model is an intractable problem in the general case even for the second order (pairwise) correlations.

Therefore, even a Pairwise Maximum Entropy (PME) model requires further estimation where more accurate approximation algorithms requires thousands of samples for each pair, making them impossible to be used for large populations of neurons Roudi et al. (2009c); Tkačik et al. (2006); Sohl-Dickstein et al. (2011); Aurell & Ekeberg (2012). Moreover, there exist plausible scenarios, such as a dichotomized common input to loosely coupled neurons, in which pairwise correlations are negligible compared to higher order correlations Macke et al. (2011); Amari et al. (2003). Overall, PME is neither computationally efficient nor accurate in population structure estimation.

To improve the structure estimation (and also as further evidence for inaccuracy of PME models), some methods modelled higher order correlations through the energy based approaches, e.g. "k-pairwise" correlationsTkačik et al. (2014), Restricted Boltzmann Machines (RBMs) on top of pair-wised correlations (semi RBMS (sRBMs)) Köster et al. (2014), and sparse low-order correlations Ganmor et al. (2011). Founded upon energy based models, these methods also needs computationally expensive sampling strategies to estimate the partition function. Moreover, to approximate the

partition function, these methods often rely on pseudo-likelihoods which needs extensive amount of data points to be a accurate.

Here we use a special type of Bayesian network with linear inference, and consequently fast learning time, to estimate distributions of joint neural activity. Our network approximates these distributions by ignoring correlations based on their effect size, rather than statistical order. Moreover, the normalization process to obtain real probability values is also linear in the size of the network. As a result, it is significantly more successful than energy-based models both in terms of efficiency and accuracy.

3 MODEL

Our framework is based on Sum-Product Networks (SPNs). An SPN is a rooted directed acyclic graph representing a joint probability distribution of given variables. This distribution is the result of a hierarchical combination of alternating mixtures (sum nodes) and factorizations (product nodes), with given variables as the leaf nodes of the network Poon & Domingos (2011); Sanchez-Cauce et al. (2021). Specifically, each *leaf* node (a node with no children) represents a univariate probability distribution. When the variable is categorical, the leaf node is a variable indicator ($\mathbb{I}(x)$). The *scope* of each leaf node is a singleton where its element is the variable that the leaf represents. Other nodes are either *sum* or *product*. A *product* node represents the product of its children (connected by a link). A *sum* node represents the (normalized) weighted sum of its children. The weight of each child is shown as the label of the link from the sum node to that child. The scope of sum and product nodes is the union of scopes of their children.

Starting from a leaf as a one-node SPN, sum product networks can be built bottom-up by combining smaller SPNs through sum or product nodes. The root of the SPN represents a valid joint probability distribution if the SPN is *complete* and *decomposable*. An SPN is complete *iff* all children of each sum node have the same scope. A sum-product network is decomposable *iff* all children of each product node have disjoint scopes.

Parameter Learning: To model the joint probability distribution of the given variables, the weights on the links of sum nodes should be learned with a learning algorithm such as gradient descent or Expectation-Maximization (EM), given the data Sanchez-Cauce et al. (2021). Similar to other Bayesian networks, the structure of the graph needs to be determined before parameter learning. One approach is to use a random dense graph and rely on the parameter learning algorithm to select important components. This strategy requires data sets orders of magnitudes larger than that gathered in neural experiments.

Structure Learning: Network structure can also be learned from the data directly based on the general computational properties of sum and product nodes Gens & Pedro (2013). Sum nodes represent the sum of probability distributions over the same set of variables (see *completeness* above). This means that they act as a union of sets. Therefore, they explain the data best if their child distributions represent disjoint sets of data (instances), or in other words different "contexts" in the data. A Product node computes the joint distribution of its children with a product, explicitly carrying an assumption of independence among the factors contributing to the product. This means that recursive splittings of the data based on instances (e.g. by a clustering method) producing a sum node, and based on variables (e.g. via performing independence tests) resulting in a product node would give us a suitable SPN structure for the given data. While there exist few algorithms for structure learning Gens & Pedro (2013); Vergari et al. (2015), we developed our own method due to our specific aim which is learning correlations in neural population.

3.1 NEURAL POPULATION ANALYSIS WITH *nSPN*S

We construct a family of SPNs that we call Neural SPNs (*nSPN*) to model the joint probability distribution of spiking in populations of neurons. These models are represented by directed graphs where leaves of an *nSPN* representing spiking activity of N neurons correspond to $\mathbb{I}_{s_i}$ and $\mathbb{I}_{\bar{s_i}}$ ($1 \leq i \leq N$) indicating whether neuron i fired or not in a given time bin (see section 2). We suggest a systematic way to build the overall architecture of an $nSPN$ suitable to capture population correlations even with a limited number of data points. First we describe a shallow *nSPN* with three layers and then a mechanism to construct deep architectures that impose regularization for better

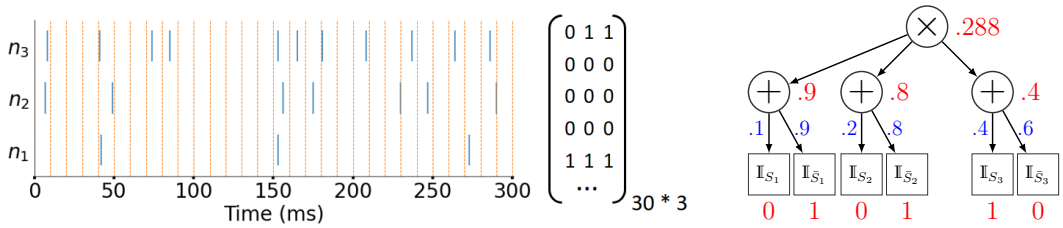

Figure 1: **Modelling Spiking activity with Sum Product Networks(SPN)** Left: Neural population activity is often represented by a binary matrix, obtained from binning each neuron's spike train (here at 10 ms). Here, the firing rates of neurons 1 to 3 are 10, 20, and 40 Hz respectively. Due to the high correlation of these neurons, bins with all 1, or all 0 are more frequent compared to the activity of 3 independent neurons with the same firing rates. Right: An spn tree (here $nSPN^1$) fit to the population in the left plot. Since it assumes independence, its weights reflect the mean probability of a spike in each time bin after training (shown in blue). Given a data instance (here 001), each node represents the probability of its children, meaning that the root represents the probability (likelihood) of the instance.

fits. These deep architectures are obtained by clustering based on the mutual information between different neurons. Finally, we describe our measure of high order correlations and describe a mixture of *nSPN* models that aims to capture correlations with deep structures.

Shallow architecture of *nSPN*s: The simplest structure of an $nSPN$ (in terms of graph complexity) consists of a product node as a root, linked to N sum nodes. Each sum node represents the spike probability of a neuron i, linked to $\mathbb{I}_{s_i}$ and $\mathbb{I}_{\bar{s}_i}$ leaf nodes. This model, shown as $nSPN^1$ is basically a Naive Bayes model, assuming complete independence among neurons. The right plot in figure 1 shows an example $nSPN^1$ trained on the joint activity of three neurons in the left plot. As mentioned before, each node of an SPN represents the probability of the leaves of the (sub-)tree of which it is the root. As a result, (normalized) link weights of each sum node are aligned with the mean spike probability of the related neuron (note the assumption of independence). Furthermore, given an assignment of probabilities to the leaf nodes and a configuration of spiking activity, the root represents the probability of that configuration's occurrence according to the model (shown by the red number near the root node).

Within the SPN framework, a simple extension of the Naive Bayes model involves the introduction of multiple *contexts*, with population spiking being described by a Naive Bayes assumption *conditioned on the context*. If there are two such contexts with associated probability, the model consists of a sum node as the root linked to two product nodes, each of which a root of an $nSPN^1$ (sub)-tree. Similarly, a model can include an arbitrary number of contexts by increasing the number of product nodes under the root, e.g. $nSPN^b$ for b product nodes/contexts.

In principle, one can construct a shallow graph sufficient to model any neural data set by adding enough product nodes. One problem with this approach is that there exist simple networks with deeper structure for which a three-layer model like we have described will require an exponential number of nodes Delalleau & Bengio (2011), which in addition to tractability also raises the issue of data limitations. Moreover, all existing parameter learning algorithms for SPNs find local optima and shallow networks get stuck in local optima more frequently. Most importantly, when the data is limited deeper architectures of our framework avoid over-fitting as their structure implements regularization.

Deep architecture of *nSPN*s: As computing all correlations is intractable, we seek an approach for constructing graphs that trades some fidelity for efficiency. Our approach is to identify correlated ensembles of neurons and build deeper graphs that reflect these ensembles. Since product nodes in SPNs describe a factorization into independent distributions with separate scopes, a natural approach, given a scope of neurons $\{S_1, \ldots, S_N\}$, is to assign neurons to a number of groups based on a clustering algorithm, and form a model that has a product node as the root, with children that form the models for each group in the clustering. Consider that such a clustering says to divide the neurons $\{S_1, \ldots, S_N\}$ into $\{S_1, \ldots, S_j\}$ and $\{S_{j+1}, \ldots, S_N\}$. One can then choose, for example, to model each group via the graphs $nSPN^2(S_1, \ldots, S_j)$ and $nSPN^2(S_{j+1}, \ldots, S_N)$. This model would

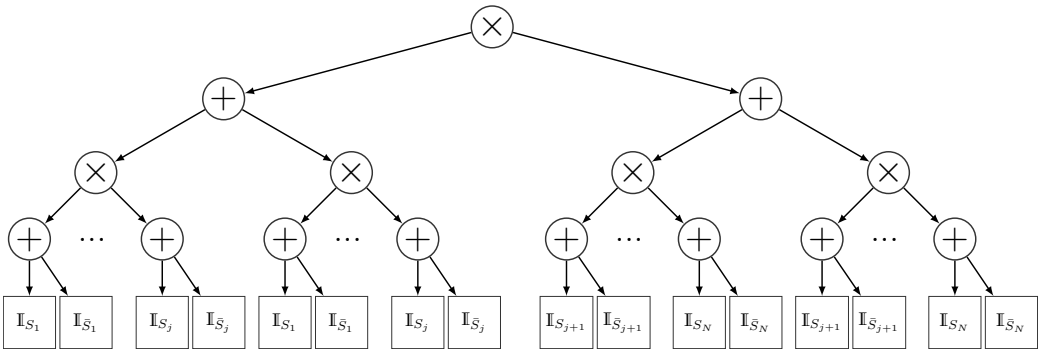

Figure 2: **Constructing *nSPN*s from clustering**. In deep architectures of our nSPNs, variables are divided into two groups at the top (higher than level 2) product nodes. The division is done through a splitting algorithm trying to find two complimentary subsets of variables with the least correlation. Without loss of generality, we can reorder variable indices so consequent incidences are always in the same group. The illusterated tree is a four-layer $nSPN$ consists of a product node as a root, assuming independence between two groups of variables $\{S_1, \ldots, S_j\}$ and $\{S_{j+1}, \ldots, S_N\}$. Correlation between variables of each group is estimated with a three-later shallow structure.

give the graph shown in Figure 2. Roughly speaking, this model describes j neurons with two common contexts and $N - j$ neurons with two *different* contexts that vary independently of those for the j neurons.

We can iterate this process to produce a graph with a finer scale. Instead of using $nSPN^2$ to describe the distribution of $\{S_1, \ldots, S_j\}$ we can iterate the clustering approach and separate these into $\{S_1, \ldots, S_k\}$ and $\{S_{k+1}, \ldots, S_j\}$. These in turn can either be described by an $nSPN^b$ model or we can continue separating into clusters. For example, $nSPN^b_d$ is an SPN in which d levels of clustering by product nodes is performed (the tree has 2d + 3 layers), and each sum node connected to intermediate nodes has b branches. Note that, one can easily use different number of branches for each node, or another criterion for splitting the variables without loss of generality (e.g minimum scope size for the clustering).

Correlation-based clustering of neurons: A product node assumes its children are independent. Therefore, to lose the least amount of population correlation, we would ideally minimize the mutual information between the two groups Gens & Pedro (2013). Formally, splitting a variable set S into nonempty subsets of V and $S \setminus V$ is optimal *iff* $V = \arg\min_{X \subset S} I(X, S \setminus X)$ where $I(X, Y) = H(X) + H(Y) - H(X, Y)$ and $H(X) = -\sum_{i=1}^{n} P(x_i) \log P(x_i)$. Since mutual information is a sub-modular function, the optimal split can be achieved in polynomial time by Queyranne's algorithm if querying $H(X, Y)$ is in polynomial time, e.g. through an oracle Queyranne (1998); Narasimhan et al. (2005); Gens & Pedro (2013); Hidaka & Oizumi (2018). This is not the case for us. In fact, such query has the exact nature of our main goal, i.e. estimating the joint probability of the set. Therefore, we use the marginal mutual information between pairs, $I(s_i, s_j)(1 \leq i, j \leq N)$ instead. With such a pairwise marginal entropy estimate, approaches that are based on maximum/minimum linkage criterion such as minimum/maximum spanning trees are the most suitable. Particularly, we use agglomerative hierarchical clustering with the distance metric of $\log(2) - I(s_i, s_j)$ between variable i and j and linkage criterion of average distance.

Measure of Neural Correlation and mixture-of-*nSPNs*: The simplest architecture, $nSPN^1$, assumes independence between all neurons. On the other hand, the more complicated structures in our framework only assume independence within each *context* where the number of contexts are always more than 1. This means that in stationary periods where the firing patterns of neurons do not change significantly, $nSPN^1$ fits to the data as well as more complex $nSPN$ structures if and only of all neurons are independent of each other. Any improvement in goodness of fit for more complicated structures (relative to $nSPN^1$) is due to the existence of neural correlations. As a result, we can use the difference between the average log-likelihood (equation 1) of the best fitting $nSPN$ and $nSPN^1$ as a measure for neural correlations. For example, if our best model is $nSPN^{b^*}$, our measure for population neural correlation is:

$$\Delta ll_{SPN} = ll_{nSPN^{b*}} - ll_{nSPN^{1}} \tag{2}$$

Any shallow structure SPN, i.e. $nSPN^b$, is a super-set of $nSPN^1$. This is not true for deep architectures though. As a result, especially for the purpose of measuring population correlations, a mixture of deep and shallow architectures should be used, e.g. a model constructed with a sum node as the root connected to a shallow structure such as $nSPN^1$ and a deep one such as $nSPN^b_d$. We call this structure mixture-of-$nSPN$s.

4 RESULTS

We start with a simulation of population activity in which we can control the presence of fluctuations. We use a network of homogeneous exponential integrate-and-fire (EIF) neurons with common fluctuating input where each neuron's membrane voltage, $V_i (1 \leq i \leq N)$ evolves as follows:

$$\begin{aligned} \tau_m V_i' &= -V_i + \Delta_T e^{\frac{V_i - V_S}{\Delta_T}} + I_i(t), \\ I_i(t) &= \gamma + \sqrt{\sigma^2 \tau_m}\big[\sqrt{1-\lambda}\xi_i(t) + \sqrt{\lambda}\xi_c(t)\big]. \end{aligned} \tag{3}$$

Notably, it has been shown that pairwise maximum entropy (PME) models are incapable of estimating this network very well especially when the firing rate of neurons is low, neural correlations are high, or the population size is large Amari et al. (2003); Macke et al. (2011). Moreover, experimental studies support the existence of networks producing extensive high order correlations in the brain Binder & Powers (2001); Khuc-Trong & Rieke (2008). Therefore, the network above has been used as a tool to assess the performance of methods for capturing beyond pairwise correlations Macke et al. (2011); Leen & Shea-Brown (2015). Similarly, we generate multiple data sets using varying parameters to get different values for the mean activity μ and (pairwise) correlation ρ and compare the goodness of fit between $nSPN^N$ and PME models for population size of N (all parameter values in Supplementary Material). Note that $nSPN^N$ is chosen to have the same number number of parameters as the PME, N^2 ($nSPN^N$ has $N^2 - 1$ free parameters to be more precise).

Pairwise Maximum Entropy models have minimal assumptions about the spiking pattern of the population beyond the firing rate of each neuron and the relationship between each two neurons. In full generality, a PME model is complex with many parameters, and is computationally expensive in general. In fact, as mentioned before, an exact fit of PME is NP-hard. Therefore, instead of the whole joint probability distribution we used the *(total) population activity* as the measure of goodness of fit. According to the PME, the probability of observing overall k spikes in a population of size N is:

$$P_{\text{PME}}(k) \propto \binom{N}{k} \exp\left(\alpha k + \beta k^2\right) \tag{4}$$

where α and β are free parameters, obtained from fitting the model to population activity. Importantly, instead of fitting the PME to the whole joint distribution we obtained α and β by directly optimizing the above equation. This measure reflects the maximum possible power of any PME model without any need to actually obtain all variables (it essentially provides a bound on the performance of the full model). Note that, the distribution of average population activity fully reflects the goodness of fit in fully homogeneous populations Roudi et al. (2009a). However, there is no guarantee that existing approximation methods for PMEs achieve such fit.

As opposed to the PME, we fitted an $nSPN^N$ to the whole joint distribution (200 iterations of Expectation Maximization (EM) implemented in SPFlow library Molina et al. (2019)), giving the PME a huge advantage . Specifically, the difference between the model's prediction and the EIF (total) population spike distribution is quantified with the Jenson-Shannon (JS)-divergence of $\frac{1}{2}D\big(P_{model}(k) \| M(k)\big) + \frac{1}{2}D\big(P_{EIF}(k) \| M(k)\big)$ where M is the averaged distribution, i.e. $M(k) = \frac{1}{2}P_{model}(k) + \frac{1}{2}P_{EIF}(k)$ and $D(.||.)$ is the Kullback–Leibler (KL)-divergence. Since JS-div scales with $\log(N)$, we report the normalized value, i.e. JS-div$/log(N)$. We set $\lambda = .30$ and $.59$ to make the pairwise-correlation ρ equal to $.1$ and $.25$ respectively, keeping other parameters the same as above. In both cases, the mean activity rate μ is $.1$. The number of neurons in the network is $N \in [8, 16, 32, 64, 128]$, and each data point is obtained from 100000 samples. As shown in the left

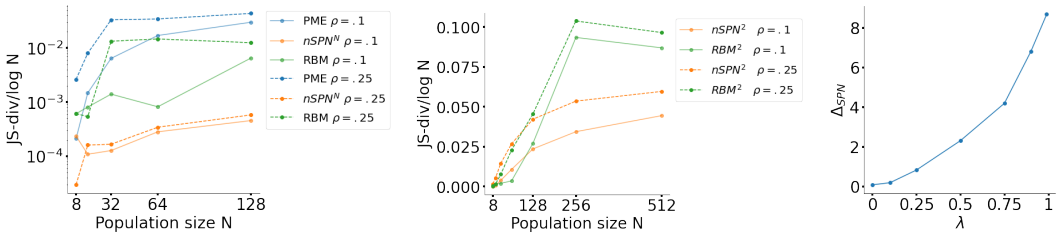

Figure 3: **Simulation of neural population activity confirms the practicality of our SPN-based framework to measure correlations among neurons.** Left: Normalized JS-divergence versus population size for two conditions with similar mean activity rate ($\mu = .10$) but different pairwise correlation. $nSPN^N$ outperformed both RBM (with the same number of free parameters) as well as the best possible PME. Middle: Similar to the left plot, with SPN and RBMs trained with only $2n$ free parameters. While the RBM outperformed the SPN (here $nSPN^2$) in small populations, the SPN became superior with larger population size. Right: Difference in log-likelihoods between $nSPN^N$ and $nSPN^1$ vs. λ (the main parameter for controlling correlations in the EIF model).

panel of figure 3, nSPN models the EIF network significantly better than PME (the JS-divergence is orders of magnitude lower) especially when the number of neurons are high.

We also compared our method to other approaches for capturing joint probability distributions of spiking activity. We trained a Restricted Boltzmann Machine (RBM) with N visible and N hidden units (N^2 free parameters). The JS-div of the RBM was an order of magnitude higher than the SPN (figure 3, left plot). Importantly, the RBM is the best known latent-variable model in capturing population correlations Köster et al. (2014). Moreover, adding energy-based component to a PME (called sRBM) does not provide a better fit to the neural population despite adding a huge amount of computation to the method Köster et al. (2014).

To further demonstrate that our model's main target is to capture the largest correlations with its available resources (free parameters), we compared the shallow SPN with 2 product nodes ($nSPN^2$) with an RBM with two hidden nodes. The RBM performs better with smile size populations. However, SPN's relative performance to RBM increases with population size as it eventually becomes superior for population size large than 128 (Figure 3, middle plot).

Finally, we simulated $N = 30$ neurons with the original EIF dynamics and parameters, with different values of $\lambda = [0, .1, .25, .5, .75, .90, .99]$. Fitted on $nSPN^N$ and $nSPN^1$ (Naive Bayes) models, their average log-likelihood difference ($\Delta ll_{SPN} = ll_{nSPN^N} - ll_{nSPN^1}$) increases with γ, confirming that our measurement is reflective of correlations (figure 3, right panel).

Experimental neural recordings

To assess our method on experimental neural recordings we modelled the neural data from the Neuropixels Visual Coding data set of the Allen Brain Observatory (https://observatory.brain-map.org). In this data set, different stimuli (Gabors, flashes, drifting gratings, etc) were shown to 26 mice while the neural activity of their visual cortical and sub-cortical regions were recorded simultaneously with multiple Neuropixels probes. Mice were head-fixed and on a wheel, free to run. Importantly, stimuli in this experiment were passively viewed, meaning that it did not involve any choice or reward.

We modelled the neural population structure during the viewing of drifting gratings in 4 directions ($[0°, 45°, 90°, 135°]$) with the contrast level of .9. Each of these 4 stimuli was shown to 26 mice 75 times. We analyzed the neural activity (spikes) of six regions in the visual cortex: VISp, VISl, VISal, VISpm, VISrl, VISam. Following other studies analyzing neural correlations, the time bin was set to $\Delta t = 20$ msc, balancing for minimizing spike loss due to binning, and accounting for activity delays between neurons. We analyzed each of these 6 regions separately.

Each stimulus duration was 2 seconds. We analyze the response from $400ms$ after stimulus onset to $800ms$ after stimulus onset. The delay is to make sure the corresponding signal has reached the neurons in all areas. Moreover, individual neurons' change upon exposure to the same sensory stimuli (known as adaptation), hence the short period. Since $400ms$ results in only 20 data points,

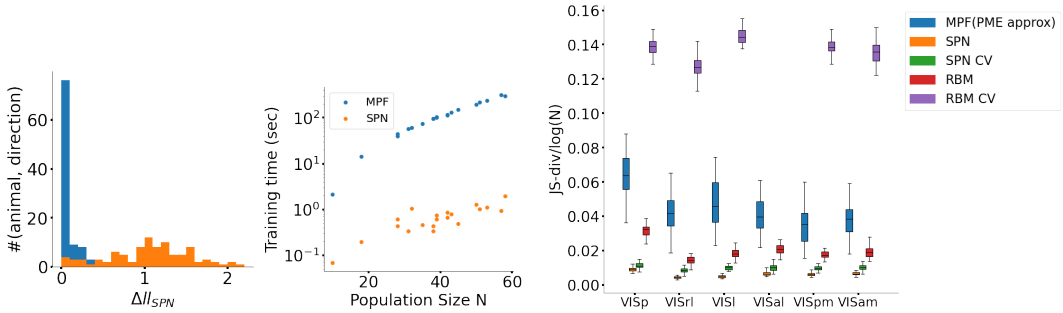

Figure 4: **Modeling high order correlations in neural recordings.** Tested on large-scale Neuropixel recordings from mouse visual cortex, mixture-of-$nSPN$s explains neural population structure and correlations in different regions of mouse brain significantly better than pairwise entropy model estimation (MPF) and RBM. (left) Distributions of Δll_{SPN} by experiment for the unshuffled (orange) vs shuffled (blue) spike distributions. (middle) Training time for MPF takes two orders of magnitude longer than training the SPN. (right) Box plots of the distribution of normalized JS-divergence over experimental sessions by visual cortical area.

we combined the data of 5 consequent presentations with the same direction. The order of stimulus condition (direction) of the stimulus type/block (drifting gratings) is random in the experiment. We chose only 5 consequent presentations to keep the state of the animal (such as running speed or arousal level) as unchanged as possible.

The described data processing results in 15 blocks of population activity per each tuple (mouse, direction, area), each of which we modelled with a mixture-of-$nSPN$s (mixture of $nSPN^1$ and $nSPN_1^2$; details supplementary material). We also fit our simplest structure $nSPN^1$ (Naive Bayes) in order to have a correlation-free baseline and compute the difference, Δll_{SPN}. We fit both models on a modified version of data in which we shuffle the spikes of each neuron in the $400ms$ of single stimulus showing for the primary visual cortical population (as the most sensitive area to events). We observed a large difference between Δll_{SPN} in the original and shuffled data. This difference confirmed our assumption about Δ_{SPN} in the original data is the sole product of neural correlations (Figure 4 left). For an extra measure of caution we removed (mouse, direction) pairs with Δll_{SPN} of more than .1 in the primary visual area for the shuffled data.

Since the population is not homogeneous, this time we used Minimum Flow Probability (MFP) Sohl-Dickstein et al. (2011) that approximates pairwise maximum entropy variables through a direct fit to the data. To the best of our knowledge this is the most accurate methods of pairwise approximation. First, to further demonstrate the computational efficiency of our method, we compared the training time of SPN and MFP approaches for 16 example data blocks of different sessions (area $VISp$) on the same machine. As demonstrated in the middle plot of figure 4, training time for the MFP is two orders of magnitude longer. Importantly, for MFP we used an efficient package that takes advantage of a C++ implementation in Python Lee & Daniels (2019), while the SPN was fully implemented in Python Molina et al. (2019).

The (total) population activity is not an ideal measure for performing model comparison. However, given the number of samples, i.e. 100, more desirable measures are prohibitive. We also fitted an RBM to the data. The number of hidden units in the RBM, and the depth added to the mixture-of$nSPN$s ($nSPN^1$ and $nsPN_1^2$) were both determined through 10-fold cross-validation within the 100 samples.

Compared to other models, the mixture-of-$nSPN$s produced significantly lower normalized JS-div for all of the 6 regions (Figure 4, right plot). We also performed a Leave One Out Cross Validation (LOOCV) test, in which the samples of each of the 15 blocks is generated based on the other 14 for each animal and direction (the validation set is outside of the fitted 100 samples and therefore unseen by the models). For the SPN, the JS-div was also significantly lower than the MPF, in all 6 regions. However, the cross validated JS-div for the RBM was extremely high, showing over-fitting to the data. Importantly, the non-cross validated result shows that even when over-fitted the RBM performed worse than the SPN.

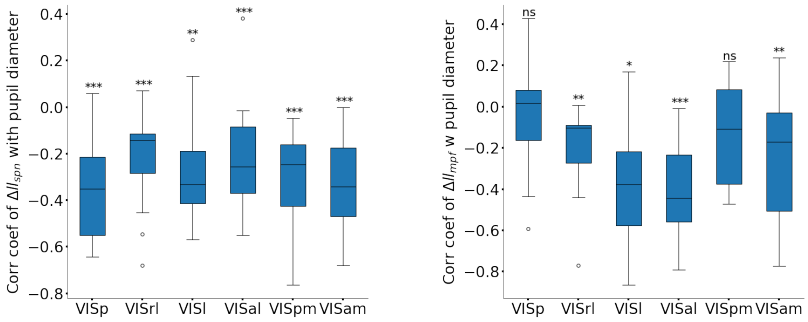

Figure 5: **Anti-correlation between arousal and neural population correlation (left)**. The Pearson correlation between Δll_{spn} and pupil diameter over experiments is significantly negative in visual cortical areas. (right) Same approach applied on MPF, Δll_{mpf} can not fully capture this phenomenon.

USING *nSPN* MODELS AS A MEASURE OF NEURAL CORRELATIONS

Many experiments have suggested that arousal has the effect of reducing neural correlations, or decreasing the level of synchrony Gandal et al. (2012); Uhlhaas et al. (2009). These experiments are mostly based on very noisy data such as the Local Field Potential (LFP) and rough estimates of signal frequency Gandal et al. (2012); Pfeffer et al. (2022); Vinck et al. (2015). Having a quantitatively reliable measure based on spiking data, i.e. Δll_{SPN}, we examined the effect of arousal in the Allen Brain Observatory. Specifically, in each animal we looked at the (ratio to minimum) pupil diameter (a common measure of arousal in mice) changes at each of the chunks described above for all directions that have passed the shuffle test combined. The mean distribution of Pearson correlation coefficient between pupil diameter and Δll_{SPN} across all animals was significantly below zero for all regions as shown in Fig. 5 (left). To the best of our knowledge, even with LFP data, there is no work demonstrating the effect of arousal on higher visual areas during an experiment (See Vinck et al. (2015) for VISp).

We repeated the same process for MPF by using the difference between log-likelihood of the full (first and second order) model with the first-order only version (second order variables set to zero). The result was significantly different (Fig. 5, right). Importantly and as mentioned, previous literature results (especially on VISp) are strongly in favour of the SPN results. Moreover, this result further demonstrates the inability of the PME in modelling the full joint population activity (as opposed to the total population activity).

5 DISCUSSION

Correlations are a potentially important channel of information in neural activity, with strong consequences for coding properties. Using Sum Product Networks, we have constructed a computationally efficient approach to modeling structure and correlations in populations of spiking neurons. Tested on simulated and experimental data, our approach outperformed both energy-based and latent variable methods.

Due to the nature of our scientific question in this paper, i.e. population correlations, we focused on short time spans in which the external variables and animal state remain constant. Our framework, however, has the capacity to be applied more broadly in computational neuroscience, for example to capture population dynamics in different time scales or as a de-mixing tool for coded features in individual or population of neurons. We believe, in these promising future directions, deep and mixed architectures play a larger role.

Reproducibility: The data set used in this paper, Allen Brain Observatory, is publicly available online. All parameters of the simulated data has been specified in the result section and/or the supplementary material. Similarly, all parameters related to the methods are mentioned in the results section and/or the supplementary material. Finally, all the code is based on open-source repositories, cited in the results section. Finally, the code for data analysis will be publicly available when the paper gets published.

REFERENCES

L. F. Abbott and Peter Dayan. The Effect of Correlated Variability on the Accuracy of a Population Code. *Neural Computation*, 11(1):91–101, January 1999. ISSN 0899-7667. doi: 10.1162/089976699300016827. URL https://doi.org/10.1162/089976699300016827.

Shun-ichi Amari, Hiroyuki Nakahara, Si Wu, and Yutaka Sakai. Synchronous Firing and Higher-Order Interactions in Neuron Pool. *Neural Computation*, 15(1):127–142, January 2003. ISSN 0899-7667. doi: 10.1162/089976603321043720. URL https://doi.org/10.1162/089976603321043720.

Erik Aurell and Magnus Ekeberg. Inverse Ising Inference Using All the Data. *Physical Review Letters*, 108(9):090201, March 2012. doi: 10.1103/PhysRevLett.108.090201. URL https://link.aps.org/doi/10.1103/PhysRevLett.108.090201. Publisher: American Physical Society.

Bruno B. Averbeck, Peter E. Latham, and Alexandre Pouget. Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, 7(5):358–366, May 2006. ISSN 1471-0048. doi: 10.1038/nrn1888. URL https://www.nature.com/articles/nrn1888. Number: 5 Publisher: Nature Publishing Group.

Rava Azeredo da Silveira and Fred Rieke. The Geometry of Information Coding in Correlated Neural Populations. *Annual Review of Neuroscience*, 44(1):403–424, 2021. doi: 10.1146/annurev-neuro-120320-082744. URL https://doi.org/10.1146/annurev-neuro-120320-082744. _eprint: https://doi.org/10.1146/annurev-neuro-120320-082744.

M. D. Binder and R. K. Powers. Relationship between simulated common synaptic input and discharge synchrony in cat spinal motoneurons. *Journal of Neurophysiology*, 86(5):2266–2275, November 2001. ISSN 0022-3077. doi: 10.1152/jn.2001.86.5.2266.

Craig Boutilier, Nir Friedman, Moises Goldszmidt, and Daphne Koller. Context-specific independence in Bayesian networks. In *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, UAI'96, pp. 115–123, San Francisco, CA, USA, August 1996. Morgan Kaufmann Publishers Inc. ISBN 978-1-55860-412-4.

Gregory F. Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial Intelligence*, 42(2):393–405, March 1990. ISSN 0004-3702. doi: 10.1016/0004-3702(90)90060-D. URL https://www.sciencedirect.com/science/article/pii/000437029090060D.

Adnan Darwiche. A differential approach to inference in Bayesian networks. *Journal of the ACM*, 50(3):280–305, May 2003. ISSN 0004-5411. doi: 10.1145/765568.765570. URL https://doi.org/10.1145/765568.765570.

Olivier Delalleau and Yoshua Bengio. Shallow vs. Deep Sum-Product Networks. In *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper/2011/hash/8e6b42f1644ecb1327dc03ab345e618b-Abstract.html.

Brent Doiron, Ashok Litwin-Kumar, Robert Rosenbaum, Gabriel K. Ocker, and Krešimir Josić. The mechanics of state-dependent neural correlations. *Nature Neuroscience*, 19(3):383–393, March 2016. ISSN 1546-1726. doi: 10.1038/nn.4242. URL https://www.nature.com/articles/nn.4242. Number: 3 Publisher: Nature Publishing Group.

Benjamin Dunn, Maria Mørreaunet, and Yasser Roudi. Correlations and functional connections in a population of grid cells. *PLoS computational biology*, 11(2):e1004052, February 2015. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1004052.

Michael J. Gandal, J. Christopher Edgar, Kerstin Klook, and Steven J. Siegel. Gamma synchrony: towards a translational biomarker for the treatment resistant symptoms of schizophrenia. *Neuropharmacology*, 62(3):1504–1518, March 2012. ISSN 0028-3908. doi: 10.1016/j.neuropharm.2011.02.007. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3264822/.

Elad Ganmor, Ronen Segev, and Elad Schneidman. Sparse low-order interaction network underlies a highly correlated and learnable neural population code. *Proceedings of the National Academy of Sciences*, 108(23):9679–9684, June 2011. doi: 10.1073/pnas.1019641108. URL https://www.pnas.org/doi/10.1073/pnas.1019641108. Publisher: Proceedings of the National Academy of Sciences.

Robert Gens and Domingos Pedro. Learning the Structure of Sum-Product Networks. In Sanjoy Dasgupta and David McAllester (eds.), *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 873–880, Atlanta, Georgia, USA, June 2013. PMLR. URL https://proceedings.mlr.press/v28/gens13.html. Issue: 3.

Shohei Hidaka and Masafumi Oizumi. Fast and exact search for the partition with minimal information loss. *PLOS ONE*, 13(9):e0201126, September 2018. ISSN 1932-6203. doi: 10.1371/journal.pone.0201126. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0201126. Publisher: Public Library of Science.

Philipp Khuc-Trong and Fred Rieke. Origin of correlated activity between parasol retinal ganglion cells. *Nature neuroscience*, 11(11):1343–1351, November 2008. ISSN 1097-6256. doi: 10.1038/nn.2199. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2575139/.

Adam Kohn, Ruben Coen-Cagli, Ingmar Kanitscheider, and Alexandre Pouget. Correlations and Neuronal Population Information. *Annual review of neuroscience*, 39:237–256, July 2016. ISSN 0147-006X. doi: 10.1146/annurev-neuro-070815-013851. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5137197/.

Urs Köster, Jascha Sohl-Dickstein, Charles M. Gray, and Bruno A. Olshausen. Modeling Higher-Order Correlations within Cortical Microcolumns. *PLOS Computational Biology*, 10(7):e1003684, July 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003684. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003684. Publisher: Public Library of Science.

Edward D. Lee and Bryan C. Daniels. Convenient Interface to Inverse Ising (ConIII): A Python 3 Package for Solving Ising-Type Maximum Entropy Models. *Journal of Open Research Software*, 7(1):3, March 2019. ISSN 2049-9647. doi: 10.5334/jors.217. URL https://openresearchsoftware.metajnl.com/article/10.5334/jors.217/.

David A. Leen and Eric Shea-Brown. A Simple Mechanism for Beyond-Pairwise Correlations in Integrate-and-Fire Neurons. *Journal of Mathematical Neuroscience*, 5(1):30, December 2015. ISSN 2190-8567. doi: 10.1186/s13408-015-0030-9.

Jakob H. Macke, Manfred Opper, and Matthias Bethge. Common input explains higher-order correlations and entropy in a simple model of neural population activity. *Physical Review Letters*, 106(20):208102, May 2011. ISSN 1079-7114. doi: 10.1103/PhysRevLett.106.208102.

Alejandro Molina, Antonio Vergari, Karl Stelzner, Robert Peharz, Pranav Subramani, Nicola Di Mauro, Pascal Poupart, and Kristian Kersting. SPFlow: An Easy and Extensible Library for Deep Probabilistic Learning using Sum-Product Networks, 2019. _eprint: arXiv:1901.03704.

Rubén Moreno-Bote, Jeffrey Beck, Ingmar Kanitscheider, Xaq Pitkow, Peter Latham, and Alexandre Pouget. Information-limiting correlations. *Nature Neuroscience*, 17(10):1410–1417, October 2014. ISSN 1546-1726. doi: 10.1038/nn.3807. URL https://www.nature.com/articles/nn.3807. Number: 10 Publisher: Nature Publishing Group.

Mukund Narasimhan, Nebojsa Jojic, and Jeff A Bilmes. Q-Clustering. In Y. Weiss, B. Schölkopf, and J. Platt (eds.), *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005. URL https://proceedings.neurips.cc/paper/2005/file/b0bef4c9a6e50d43880191492d4fc827-Paper.pdf.

Thomas Pfeffer, Christian Keitel, Daniel S Kluger, Anne Keitel, Alena Russmann, Gregor Thut, Tobias H Donner, and Joachim Gross. Coupling of pupil- and neuronal population dynamics reveals diverse influences of arousal on cortical processing. *eLife*, 11:e71890, February 2022. ISSN 2050-084X. doi: 10.7554/eLife.71890. URL https://doi.org/10.7554/eLife.71890. Publisher: eLife Sciences Publications, Ltd.

Hoifung Poon and Pedro Domingos. Sum-product networks: A new deep architecture. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 689–690, November 2011. doi: 10.1109/ICCVW.2011.6130310.

Maurice Queyranne. Minimizing symmetric submodular functions. *Mathematical Programming*, 82 (1):3–12, June 1998. ISSN 1436-4646. doi: 10.1007/BF01585863. URL https://doi.org/10.1007/BF01585863.

Amirmohammad Rooshenas and Daniel Lowd. Learning Sum-Product Networks with Direct and Indirect Variable Interactions. In *Proceedings of the 31st International Conference on Machine Learning*, pp. 710–718. PMLR, January 2014. URL https://proceedings.mlr.press/v32/rooshenas14.html. ISSN: 1938-7228.

Yasser Roudi, Erik Aurell, and John Hertz. Statistical physics of pairwise probability models. *Frontiers in Computational Neuroscience*, 3, 2009a. ISSN 1662-5188. doi: 10.3389/neuro.10.022.2009. URL https://www.frontiersin.org/articles/10.3389/neuro.10.022.2009.

Yasser Roudi, Sheila Nirenberg, and Peter E. Latham. Pairwise Maximum Entropy Models for Studying Large Biological Systems: When They Can Work and When They Can't. *PLoS Computational Biology*, 5(5):e1000380, May 2009b. ISSN 1553-734X. doi: 10.1371/journal.pcbi.1000380. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2674569/.

Yasser Roudi, Joanna Tyrcha, and John Hertz. Ising model for neural data: Model quality and approximate methods for extracting functional connectivity. *Physical Review E*, 79(5):051915, May 2009c. doi: 10.1103/PhysRevE.79.051915. URL https://link.aps.org/doi/10.1103/PhysRevE.79.051915. Publisher: American Physical Society.

Raquel Sanchez-Cauce, Iago Paris, and Francisco Javier Diez. Sum-product networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, PP, February 2021. ISSN 1939-3539. doi: 10.1109/TPAMI.2021.3061898.

Elad Schneidman, Michael J. Berry, Ronen Segev, and William Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007–1012, April 2006. ISSN 1476-4687. doi: 10.1038/nature04701. URL https://www.nature.com/articles/nature04701. Number: 7087 Publisher: Nature Publishing Group.

Yujia Shen, Arthur Choi, and Adnan Darwiche. Tractable Operations for Arithmetic Circuits of Probabilistic Models. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/hash/5a7f963e5e0504740c3a6b10bb6d4fa5-Abstract.html.

Yujia Shen, Arthur Choi, and Adnan Darwiche. A New Perspective on Learning Context-Specific Independence. In *Proceedings of the 10th International Conference on Probabilistic Graphical Models*, pp. 425–436. PMLR, February 2020. URL https://proceedings.mlr.press/v138/shen20a.html. ISSN: 2640-3498.

Jascha Sohl-Dickstein, Peter B. Battaglino, and Michael R. DeWeese. New Method for Parameter Estimation in Probabilistic Models: Minimum Probability Flow. *Physical Review Letters*, 107(22): 220601, November 2011. doi: 10.1103/PhysRevLett.107.220601. URL https://link.aps.org/doi/10.1103/PhysRevLett.107.220601. Publisher: American Physical Society.

G. Tkačik, Elad Schneidman, Michael J. Berry, and W. Bialek. Ising models for networks of real neurons. 2006.

Gašper Tkačik, Olivier Marre, Dario Amodei, Elad Schneidman, William Bialek, and Michael J. Berry Ii. Searching for Collective Behavior in a Large Network of Sensory Neurons. *PLOS Computational Biology*, 10(1):e1003408, January 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003408. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003408. Publisher: Public Library of Science.

Peter Uhlhaas, Gordon Pipa, Bruss Lima, Lucia Melloni, Sergio Neuenschwander, Danko Nikolić, and Wolf Singer. Neural synchrony in cortical networks: history, concept and current status. *Frontiers in Integrative Neuroscience*, 3, 2009. ISSN 1662-5145. doi: 10.3389/neuro.07.017.2009. URL https://www.frontiersin.org/article/10.3389/neuro.07.017.2009.

Anne E. Urai, Brent Doiron, Andrew M. Leifer, and Anne K. Churchland. Large-scale neural recordings call for new insights to link brain and behavior. *Nature Neuroscience*, 25(1):11–19, January 2022. ISSN 1546-1726. doi: 10.1038/s41593-021-00980-9. URL https://www.nature.com/articles/s41593-021-00980-9. Number: 1 Publisher: Nature Publishing Group.

Antonio Vergari, Nicola Di Mauro, and Floriana Esposito. Simplifying, Regularizing and Strengthening Sum-Product Network Structure Learning. In Annalisa Appice, Pedro Pereira Rodrigues, Vítor Santos Costa, João Gama, Alípio Jorge, and Carlos Soares (eds.), *Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pp. 343–358, Cham, 2015. Springer International Publishing. ISBN 978-3-319-23525-7. doi: 10.1007/978-3-319-23525-7_21.

Martin Vinck, Renata Batista-Brito, Ulf Knoblich, and Jessica A. Cardin. Arousal and Locomotion Make Distinct Contributions to Cortical Activity Patterns and Visual Encoding. *Neuron*, 86(3): 740–754, May 2015. ISSN 0896-6273. doi: 10.1016/j.neuron.2015.03.028. URL https://www.cell.com/neuron/abstract/S0896-6273(15)00252-4. Publisher: Elsevier.

He Zhao, Piyush Rai, Lan Du, Wray Buntine, Dinh Phung, and Mingyuan Zhou. Variational Autoencoders for Sparse and Overdispersed Discrete Data. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pp. 1684–1694. PMLR, June 2020. URL https://proceedings.mlr.press/v108/zhao20c.html. ISSN: 2640-3498.