Resolving neuronal population code and coordination with gradient boosted trees

Guillaume Viejo*[1], Thomas Cortier[1,2] and Adrien Peyrache†[1]

[1]Montreal Neurological Institute, McGill University
[2]École Normale Supérieure, Paris

June 13, 2017

Abstract

Understanding how neurons cooperate to integrate sensory inputs and guide behavior is a fundamental problem of neuroscience. A large body of methods have been developed to study neuronal firing at the single cell and population levels, generally seeking interpretability as well as predictivity. However, these methods are usually confronted with the lack of ground-truth necessary to validate the approach. Here, using neuronal data from the head-direction system, we present evidence how gradient boosted trees, a non-linear and supervised machine learning tool, learns the relationship between behavioral parameters and neuronal responses with high accuracy by optimizing the information rate. Interestingly, and unlike other classes of Machine Learning methods, the intrinsic structure of the trees can be interpreted in relation to behavior (e.g. to recover the tuning curves) or to study how neurons cooperate with their peers in the network. As an example, we show how the method reveals a temporally shifted coordination in a thalamo-cortical circuit during wakefulness and sleep, indicating a brain-state independent feed-forward circuit. Machine learning tools thus open new avenues for benchmarking model-based characterization of spike trains.

Introduction

Investigating how the brain operates at the neuronal level is usually addressed by the specification of neuronal responses to an experimentally measurable variable or by the quantification of the temporal coordination of neuronal ensembles [Harris, 2005, Rieke, 1999]. Using various methods, the responses of single neurons can be characterized by the tuning curves based on a single measurement (i.e. average firing rate as function of the observed value) [Hubel and Wiesel, 1962, O'keefe and Nadel, 1978, Taube et al., 1990], generalized linear models accounting for the coding of multiple variables [Harris et al., 2003, Truccolo et al., 2005], biophysical models of spike train generation [Pillow et al., 2005] or information measures and reverse reconstruction [Borst and Theunissen, 1999, Rieke, 1999].

The coding of information in the brain relies on the coordinated firing of neuronal population [Buzsáki, 2010, Harris, 2005, Pouget et al., 2013, Yuste, 2015]. The development of dense electrode arrays [Buzsáki, 2004, Jun et al., 2017] and imaging techniques [Chen et al., 2013, Dombeck et al., 2007] in awake animals now allows us to monitor large ensembles of neuronal activity and to address fundamental questions about neuronal network coordination. Neuronal interactions, in relation to behavior or internal parameters (e.g. brain states) are evaluated by the statistical dependencies of spike trains, the most widely used being linear cross-correlations [Perkel et al., 1967]. These linear measures can be generalized to population correlation with tools such as Principal Component Analysis [Chapin and Nicolelis, 1999, Peyrache et al., 2010] and Independent Component Analysis [Lopes-dos Santos et al., 2013]. Generalized linear models were used to build prediction of single spike trains as a func-

*guillaume.viejo@mcgill.ca
†adrien.peyrache@mcgill.ca

tion of the peer network activity [Harris et al., 2003] and to provide a full statistical description of spatio-temporal neuronal responses and correlations [Pillow et al., 2008]. Methods from graph theory offer ways to compare interactions at the network level across experimental conditions [Humphries, 2017]. Finally, among the large body of available tools, evaluating neuronal coupling by fitting spiking activity to Ising models has provided key insights into the nature of neuronal coordination in a population [Cocco et al., 2009, Schneidman et al., 2006].

The majority of the methods enumerated above rely on a set of assumptions regarding the statistics of the data or the biophysics of neuronal spiking, among others, while seeking explanatory power. To assert the validity of a particular approach, the usual procedure is to divide the data set into a *training set*, used to fit the model parameters, and a *test set*, on which the likelihood of the model is evaluated. However, this method, called cross-validation, does not rule out the possibility that a particular fit of the model parameters, even when leading to high likelihood, corresponds to the wrong model. For example, the omission of a key variable in the model may attribute erroneous contribution to the set of other variables. These limitations arise from the lack of ground-truth data that in the most complex (and, therefore, interesting) cases represent an unreachable goal.

This lack of ground-truth data when performing data analysis is particularly unavoidable in neuroscience [Harris et al., 2016]. It has thus become necessary to establish standard, model-free methods that, even if they do not contribute to our understanding of the data, set levels of performance that may be used to benchmark model-based approaches [Benjamin et al., 2017, Truccolo and Donoghue, 2007]. Machine learning provides a large array of techniques to classify dataset that have demonstrated high level of performance in fields ranging from image processing to astrophysics [LeCun et al., 2015]. Our approach is to apply a supervised classifier, the so-called *gradient boosting*, to determine an upper bound of performance for model-based, interpretable methods in neuronal data analysis [Benjamin et al., 2017, Truccolo and Donoghue, 2007]. We tested the validity of the approach on data of the head-direction system [Peyrache et al., 2015, Taube, 2007, Taube et al., 1990], a sensory pathway whose member neurons, the so-called *head-direction cells*, emit spike trains that can be explained with high accuracy simply by the direction of the head of the animal in the horizontal plane. Decision trees maximized their branching in input ranges where Fisher Information was maximal. We then determined the optimal parameters of the method for our data set. Finally, we applied this method to simultaneously recorded neurons in the thalamocortical network of the head-direction system, namely in the antero-dorsal nucleus of the thalamus (ADn) and the Post-subiculum (PoSub), and demonstrated that thalamic neurons lead cortical neurons in all brain states.

Methods

Gradient boosted trees

Machine learning literature defines boosting as the combination of many weak classifiers with limited prediction performances in order to build a stronger classifier. The first boosting algorithm is AdaBoost (Adaptive Boosting) [Freund and Schapire, 1995] which trains weak learners using a distribution of weight over the training set. This distribution of weight is updated after the convergence of a weak learner in order for the next weak learner to focus on the difficult examples i.e. the points that are hard to classify.

Boosting algorithms come in different flavors for the type of learners or the updating of the weights [Ferreira and Figueiredo, 2012, Schapire, 2003]. Here we focused on the boosting using the decision tree model as the weak learner. The goal of the gradient boosted trees algorithm is to determine the optimal successive partition of features space in order to assign a weight or a label to a subset of the train-

ing examples.

Focusing on supervised learning, we thus defined the training set $[(x_1, y_1), \ldots, (x_m, y_m)]$ where $x_i \in R^d$ is the i-th training example with d different features and y_i is the target value. The goal of the learner reduces to: how to make an accurate prediction $\hat{y}_i$ given x_i and the correct value y_i. A target value y_i for a given training example x_i is a spike count over a finite time bin for one neuron. Using Poisson distribution, we thus defined the prediction of the model as :

$$p(y_i = k|x_i;\theta) = \frac{\lambda^k}{k!}\exp^{-\lambda} \tag{1}$$

for a given set of parameters θ. We defined $\hat{y}_i$ for each training example as the prediction of the learning algorithm. This value corresponds to the mean of the predicted Poisson distribution.

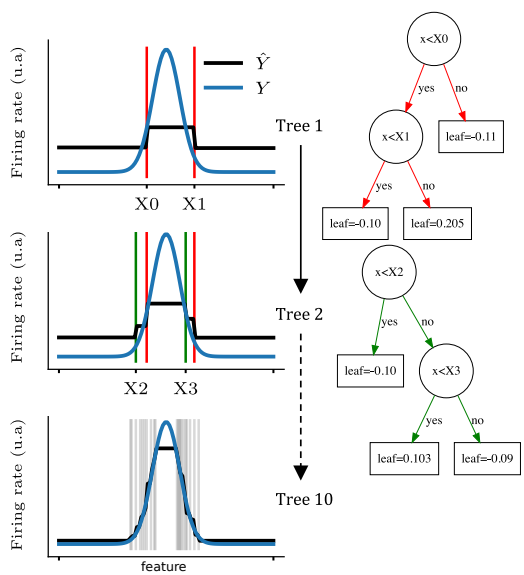

Figure 1: *Predicting the firing rate of a cell with gradient boosted trees. Each row corresponds to the learning of one tree by the algorithm. The tuning curve is sequentially split as shown on the left figures (vertical lines; blue line displays the actual tuning curve and black lines correspond to the prediction). Thus, intervals between each pair of splits are assigned a different target value. The first two trees are shown on the right and the exact values of each leaf are indicated in the square boxes.*

The measure of the performance of the model is made through an objective function $O(\theta) = L(\theta) + \Omega(\theta)$ that sums the training loss L and the regularization term (penalty for complexity) Ω. The training loss to be minimized is then defined as the negative log-likelihood over the full set :

$$-\mathcal{L}(\theta) = -\sum_i [y_i log(\hat{y}_i) - \hat{y}_i] \tag{2}$$

also known as the Poisson loss.

For the regularization term Ω, the complexity of the tree set was defined as

$$\Omega(f) = \gamma T + \frac{1}{2}\bar{\lambda}\sum_{j=1}^{|Leaves|} w_j^2 \tag{3}$$

where w_j is the vector of scores on leaves and $\gamma, \bar{\lambda}$ are free parameters.

To minimize the objective function, the learning algorithm must find the optimal set of split values and the optimal set of leaf values for each tree. An efficient strategy is thus to optimize trees sequentially i.e. the input of a tree is the output of the previous tree. After optimizing $t-1$ trees, the prediction at tree t is $\hat{y}_i^t = \hat{y}_i^{t-1} + f_t(x_i)$.

By taking advantage of the fact that the same score is assigned to all the points that fall into the same leaf, the objective function can be transformed from a sum over the training set to a sum over the leaves set:

$$O^t \approx \sum_{j=1}^{|Leaves|} [(\sum_{i \in I_j} g_i)w_j + \frac{1}{2}(\sum_{i \in I_j} h_i + \lambda)w_j^2] + \gamma|Leaves| \tag{4}$$

The index function $I_j = \{i|f(x_i) = w_j\}$ maps each training point x_i to the corresponding leaf j while g_i and h_i are respectively the first order and second order derivatives of the loss function. In the case of Poisson regression, the g_i and h_i are defined as :

$$g_i = e^{\hat{y}_i} - y_i \tag{5}$$

$$h_i = e^{\hat{y}_i} \tag{6}$$

Finally, the sum of w_j and w_j^2 in equation 4 is quadratic which allows us to compute the optimal w_j^* and the corresponding best objective

value

$$O^*(w_j^*) = -\frac{1}{2}\sum_{j=1}^{|Leaves|}\frac{G_j^2}{H_j+\lambda} + \gamma|Leaves| \quad (7)$$

which is only a measure of how good a tree structure is.

The best tree structure is then found by sequentially splitting the features space, with each splitting position corresponding to the maximum gain:

$$Gain = \frac{1}{2}\left(\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{G_R^2+G_L^2}{H_R+H_L\lambda}\right) - \gamma \quad (8)$$

with $G_j = \sum_{i\in I_j} g_i$ and $H_j = \sum_{i\in I_j} h_i$. The gain for one split is a measure of fit improvement. It is the difference between the scores of the new leaves (right leaf and left leaf) after the split and the score of the previous leaf. Details of the derivative steps and full explanations of the algorithm can be found in Chen and Guestrin [2016].

An example of the gradient boosted trees algorithm is shown in Fig. 1 for a non-linear tuning curve (blue curves Y). For each tree sequentially optimized (1,2 and 10 shown), the algorithm splits the tuning curve at different positions (X0, X1, X2, X3, ...) and assigns a leaf score between each splits. When iterating this procedure, the predicted firing rate (black curves $\hat{Y}$) will progressively converge to the actual firing rate.

Scoring function

To estimate the quality of a model, we used the pseudo-R^2 score :

$$pR^2 = 1 - \frac{(y\log y - y) - (y\log\hat{y} - \hat{y})}{(y\log y - y) - (y\log\bar{y} - \bar{y})} \quad (9)$$

with y the target firing rate, $\hat{y}$ the prediction, $\bar{y}$ the mean firing rate [Cameron and Windmeijer, 1997]. A value of 1 indicates a perfect model that reproduces entirely the dataset while a value of 0 indicates a model that is no better than the average value of the training set.

To compute the pR^2 score, the data set was divided into a training set and a test set, a procedure known as cross-validation that prevents the model from over-fitting the training set. For all the predictions of firing rates, we used a 8-fold cross-validation, i.e the training set was divided into 8 discontinuous partitions with each one serving successively as the testing set. For each spiking activity predicted for one neuron, this procedure yields eight pR^2 that were averaged. This mean pR^2 served as a measure of performance of the gradient boosted trees technique.

Model comparison

To benchmark the performance, we compared the mean pR^2 from mean pR^2 obtained from respectively the model-based tuning curve of the neuron and a linear regression model. In both cases, the same 8-fold cross-validation procedure was used. For the model-based algorithm, the tuning curve of one head-direction neuron was constructed over 60 angular bins between 0 and 2π using the training set. This tuning curve was then used to predict the firing rate of the test set.

Fisher Information

Fisher Information is directly related to the variance of the most optimal decoder and can be computed, under the assumption of a Poisson Process, directly from the tuning curve [Brunel and Nadal, 1998]. For recall, $FI(x) = (df/dx)^2/f(x)$ with f(x) the firing rate at position x of the input feature. In practice the Fisher Information was reduced to the squared slope of the line fitted between three successive bins of the tuning curve divided by the firing rate of the middle bin.

Dataset

Neuronal recordings that are analyzed in this report were described in a previously published paper [Peyrache et al., 2015] and are available for download (https://crcns.org/data-sets/thalamus/th1). Briefly, multi-site silicon probes (Buzsaki32 and Buzsaki64 fom Neuronexus) were inserted over the antero-dorsal nucleus (ADn) of

the thalamus in 7 mice. In three of these animals, a second probe was lowered to the post-subiculum (PoSub).

During the recording session, neurophysiological signals were acquired continuously at 20 kHz on a 256-channel Amplipex system (Szeged; 16-bit resolution, analog multiplexing). The wide-band signal was downsampled to 1.25 kHz and used as the local-field potential signal. To track the position of the animals in the open maze and in their home cage during rest epochs, two small light-emitting diodes (LEDs; 5-cm separation), mounted above the headstage, were recorded by a digital video camera at 30 frames per second. The LED locations were detected online and resampled at 39 Hz by the acquisition system. Spike sorting was performed semi-automatically, using KlustaKwik (http://klustakwik.sourceforge.net/). This was followed by manual adjustment of the waveform clusters using the software Klusters.

In animals implanted over the antero-dorsal nucleus, the thalamic probe was lowered until the first thalamic units could be detected on at least 2-3 shanks. The thalamic probe was then lowered by 70-140 μm at the end of each session. In the animals implanted in both the thalamus and in the post-subiculum, the subicular probe was moved everyday once large head-direction cell ensembles were recorded from the thalamus. Thereafter, the thalamic probes were left at the same position for as long as the quality of the recordings remained high. They were subsequently adjusted to optimize the yield of head-direction cells. To prevent statistical bias of neuron sampling, we discarded sessions from analysis that were separated by less than 3 days during which the thalamic probe was not moved.

Data analysis

For all analysis, spike trains were binned for all analyses in 25ms bins and smoothed with a 125 ms kernel, unless stated otherwise.

Code availability

The analyses presented in this report were run on Matlab (Mathworks, 2017) and Python. Code is available online in a raw form and as a Jupyter notebook to present some of the analyses (www.github.com / PeyracheLab / NeuroBoostedTrees). Gradient boosting was implemented with the XGBoost toolbox [Chen and Guestrin, 2016].

Results

Gradient boosted trees predict firing rates with raw features

We applied gradient boosted trees (XGB) to the prediction of spike counts from head-direction neurons recorded in ADn and PoSub (see Fig. 2.A for a full display of the training process). Since the head-direction signal is a well characterized signal relative to the angular direction of the animal's head, we compared the prediction of XGB with the output of the model-based (MB) tuning curve (that is, the firing rate expected from the head-direction of the animal knowing the tuning curve; see Fig. 2.B). The comparison shows that XGB reaches the same level of performance than MB for both ADn and PoSub. We then added a generalized linear regression model with raw head-direction values or a 6th order kernel. In the first case, the model learns only from the angular features θ ranging from 0 to 2π. In the second case, the model learns with all the k harmonics $(cos\theta, sin\theta, \ldots, cosk\theta, sink\theta)$. Not surprisingly, the simple linear model showed negative or null performances for both anatomical structures (Fig. 2.B). Preprocessing of the angular feature (with the 6th order kernel) increased the performance to the same levels as XGB and MB.

The comparison of our models indicates that XGB is the best model to learn raw data such as the angular head-direction signal. As expected, the linear regression model needs a kernel in order to linearize and increase the dimensionality of the feature space. The data we used contain only head-direction neurons and

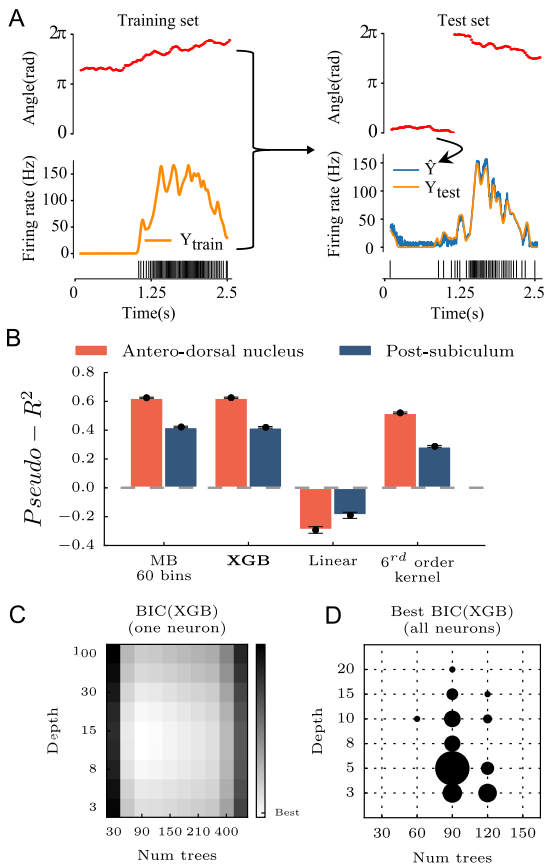

Figure 2: *Comparing gradient boosted trees (XGB) with classical methods.* ***A*** *Using the angle as the input feature (red), the machine learning algorithm is trained to minimize the error in predicting the firing rate of one head-direction neuron over time (yellow, spiking activity below) during the training phase. For each angular position in the test set, the algorithm predicts a firing rate (blue curve). The score of the algorithm measure how close is the prediction to the real value.* ***B*** *Using an 8-fold cross-validation, XGB was compared to model-based tuning curves (MB) with 60 bins, a linear regression model and a linear regression model with preprocessing of the features i.e the first six harmonics of the angular direction of the head were used instead of the raw angle. Recordings from ADn and PoSub were used to benchmark each model.* ***B*** *To find the optimal number of trees and the optimal depth of XGB, a grid-search was performed for each neuron using the Bayesian Information Criterion (BIC).* ***C*** *Distribution of the set of optimal parameters for all neurons. Overall, a maximum number of 100 trees with a depth of 5 was used to learn and predict spiking activity as in* ***A****.*

the relation between neural firing rates and head angular position was perfectly captured by XGB when compared to MB.

In comparison with XGB, linear models and MB are straightforward models in terms of numbers of free parameters compared to XGB. In order to validate our approach of predicting spike trains using XGB, we performed a grid-search to find the optimal number of trees and the optimal depth of each tree. A Bayesian Information Criterion (BIC) score (Fig. 2.C) was used to compare grid points. The BIC score was defined as $BIC(|Trees|, Depth) = (|Trees| + Depth)log(n) - 2log(L)$ with n the number of time steps in the data training set and L the likelihood of the model. By penalizing more complex models using this approach, we found that 100 trees with a maximal depth of 5 were sufficient to predict spike trains for all neurons (Fig. 2.C,D).

Information content of the feature space is revealed by data splitting

Gradient boosted trees, as most Machine Learning tools, can be considered as a black box that achieves high levels of performance with no possibility to interpret how learning was performed. But, unlike e.g. deep networks, it is possible to retrieve the thresholds at which trees split the data to predict the target output (as shown in Fig. 1). In the case of head-direction cells, predicted from the head-direction of the animals, the positions of splits in the interval $[0, 2\pi]$ were retrieved from the gradient boosted trees (see examples of figure 3.A). Splits concentrated for head-direction values where the tuning curves was the steepest. In fact, the density of splits was very strongly correlated to the Fisher Information (Fig. 3.A and insets in B), a measure related, but not equal, to tuning curve steepness that estimates the variance of an optimal decoder [Brunel and Nadal, 1998].

Many neurons of the brain's navigation system exhibit correlates to more than one behavioral parameters, for example head-direction and place [Cacucci et al., 2004, Peyrache et al.,

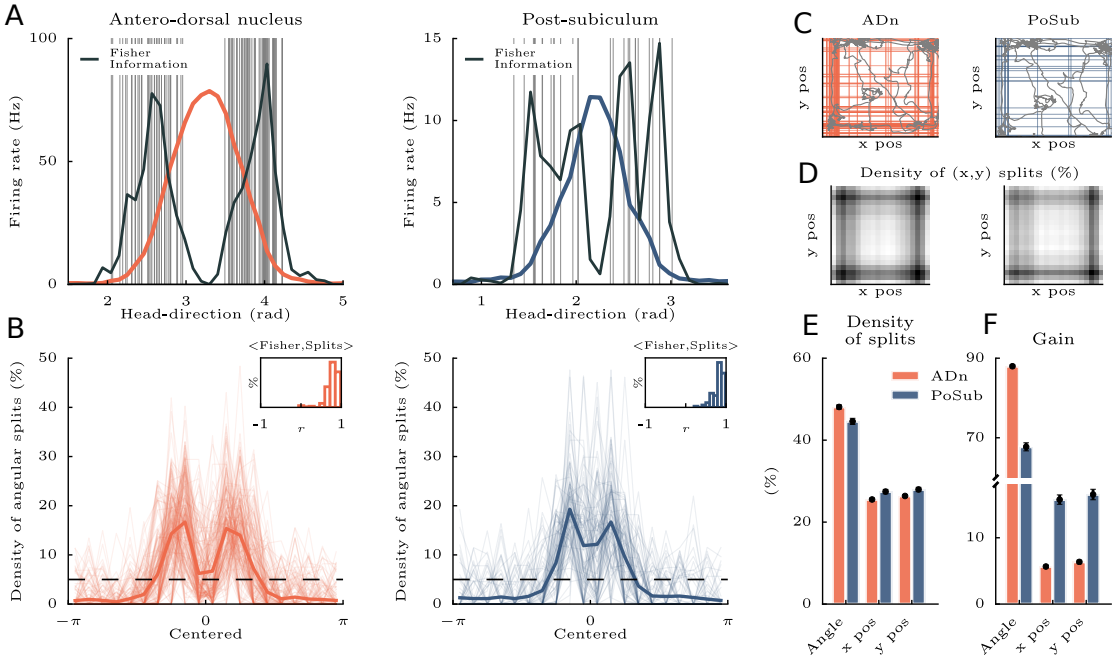

Figure 3: *Segmentation of behavioral features to predict neuronal spiking.* ***A*** *Tuning-curve splitting for one neuron of the antero-dorsal nucleus (ADn) and one neuron of the post-subiculum (PoSub). Each vertical gray line is a split from the gradient boosted trees used to predict firing rate. Black lines indicate Fisher Information (computed from the tuning curves).* ***B*** *Density of angular splits for ADn and PoSub for all the neurons, and average (thick line). Splits positions were realigned relative to the peak of the tuning curve. Horizontal dashed lines display chance levels. Insets show the distribution of correlation coefficients between Fisher Information and density of splits.* ***C*** *Using x and y coordinates of the animal in the environment as additional input features of the algorithm. Colored lines indicate spatial positions of splits along x and y. Gray lines indicate a short segment of the trajectory of the animal during the example session.* ***D*** *Density of splits for x and y position features for all neurons. The highest density is shown in black.* ***E*** *Proportion of splits for the three input features (head direction, x position and y position for AD and PoSub.* ***F*** *Mean gain value for the three input features (head direction, x position and y position for ADn and PoSub)*

2016, Sargolini et al., 2006]. To determine the relative correlation of a neuron with these different behavioral parameters, we regressed spike trains against x and y positions of the animal randomly foraging in the environment, in addition to the head-direction. We thus increased the feature space and dissected the resulting splitting distribution of the gradient boosted trees (Fig. 3.C). Averaging over all neurons, the density of splits was the highest in the corner of the environment (Fig. 3.D) where the animal spend naturally most of its time. When averaging the distribution of splits between the head-direction and the two-dimensional position (Fig. 3.E), the angular feature was more segmented for both ADn and PoSub. Nevertheless, we observed that the proportion of angular splits to position splits was slightly lower for PoSub when compared to ADn.

One potential issue with this approach is that training a large number of trees overfit the learning procedure: it is optimal for decoding performance, not necessarily for the interpretability of the tree structure. To best explained the contribution of various features to the spiking activity, it is sometimes more suited to concentrate on the structure of a smaller number of trees, and examine the 'gain' of each

feature when training the first trees. In fact, the average gain (see equation 8) for each feature decreases exponentially as the number of trees increases (as shown in Fig. S1 when predicting firing rate only with the angular direction). Besides, we found that random features were also more split as the number of trees increased (see Fig. S2). For all those reasons, analysis of the algorithm were performed with the first 30 trees with a depth of 2 as it corresponds to the most meaningful split density and average gain. As a result, the gain of spatial features (x and y position) was approximatively three times higher for PoSub neurons compared to ADn neurons (Fig. 3.E and F), in agreement with previous studies that employed model-based methods (i.e. that assumed various properties for spike trains and sampling of the feature space) [Cacucci et al., 2004, Peyrache et al., 2016]. Thus, we concluded that XGB, when used appropriately, is an efficient method for determining the relative contribution of various features to a series of spike trains.

Peer-prediction reveals the directionality of information flow across brain structures

We then applied XGB to neuronal peer-prediction, that is learning to estimate the spiking activity of one neuron as a function of the activity of a population of other neurons (Harris et al. [2003], Peyrache et al. [2015], Pillow et al. [2008]). For each session that contained at least 7 neurons in both ADn and PoSub, the model learned all possible group combinations (ADn->ADn, PoSub->ADn, PoSub->PoSub, ADn->PoSub). For intra-group prediction, the target neuron was removed from the pool of feature neurons. Tested during wake, REM and slow wave sleep (referred also to as 'non-REM sleep'), we found that peer-prediction had the highest prediction score between ADn neurons and the lowest score between PoSub neurons (Fig. 4.A). Inter-group predictions were similar. In all cases, scores during slow wave sleep were systematically lower than during wakefulness and REM, in agreement with previous analysis of peer-prediction in thalamo-cortical assemblies Peyrache et al. [2015].

Uneven number of feature neurons is a potential confound in peer-prediction analysis. The prediction process was thus repeated by equalizing the number of neurons in both structures and it yielded similar scores as the original analysis (Fig. 4.B). The activity within ADn is therefore more predictable than in the PoSub.

To best capture the statistical dependencies between spikes trains, we focused on a gain analysis (i.e. from the branching structure resulting from learning on only 30 trees with a depth of 2) and we found that the angular distance was a weak predictor of the split density for both ADn and PoSub (Fig. 4.C). In others words, gradient boosted trees tend to split preferentially, yet mildly, feature neurons that have a preferred direction closer to the target neuron. More surprisingly, we found no correlation between the mean firing rate of neurons and the density of splits (Fig. 4.D). Feature data from neurons with high firing rates are characterized by a wider range of values to be split, yet, this does not lead to increased splitting. Thus, all neurons contributed to the prediction of the activity of another neuron despite each idiosyncratic spiking activity.

Can XGB reveal the temporal component of neuronal communication across brain areas? To investigate this question, XBG was run for peer-prediction of individual PoSub neurons from multiple copies of ADn population activity at various time-lags. In other words, the model learned the relationship between the firing rate of feature neurons from time $t - T$ to $t + T$ (in Fig. 4.A, the model had access only to time t). A graphical explanation of this procedure is shown in Fig. S3. Using only raw, unsmoothed spike counts, we found that the gain (the number of splits multiplied by the mean gain) was maximal at -25 ms when predicting PoSub firing rate with ADn activity (Fig. 4.E). This delay of transmission between structures was held constant between wake, REM sleep and slow wave sleep, suggesting a hard-wired,

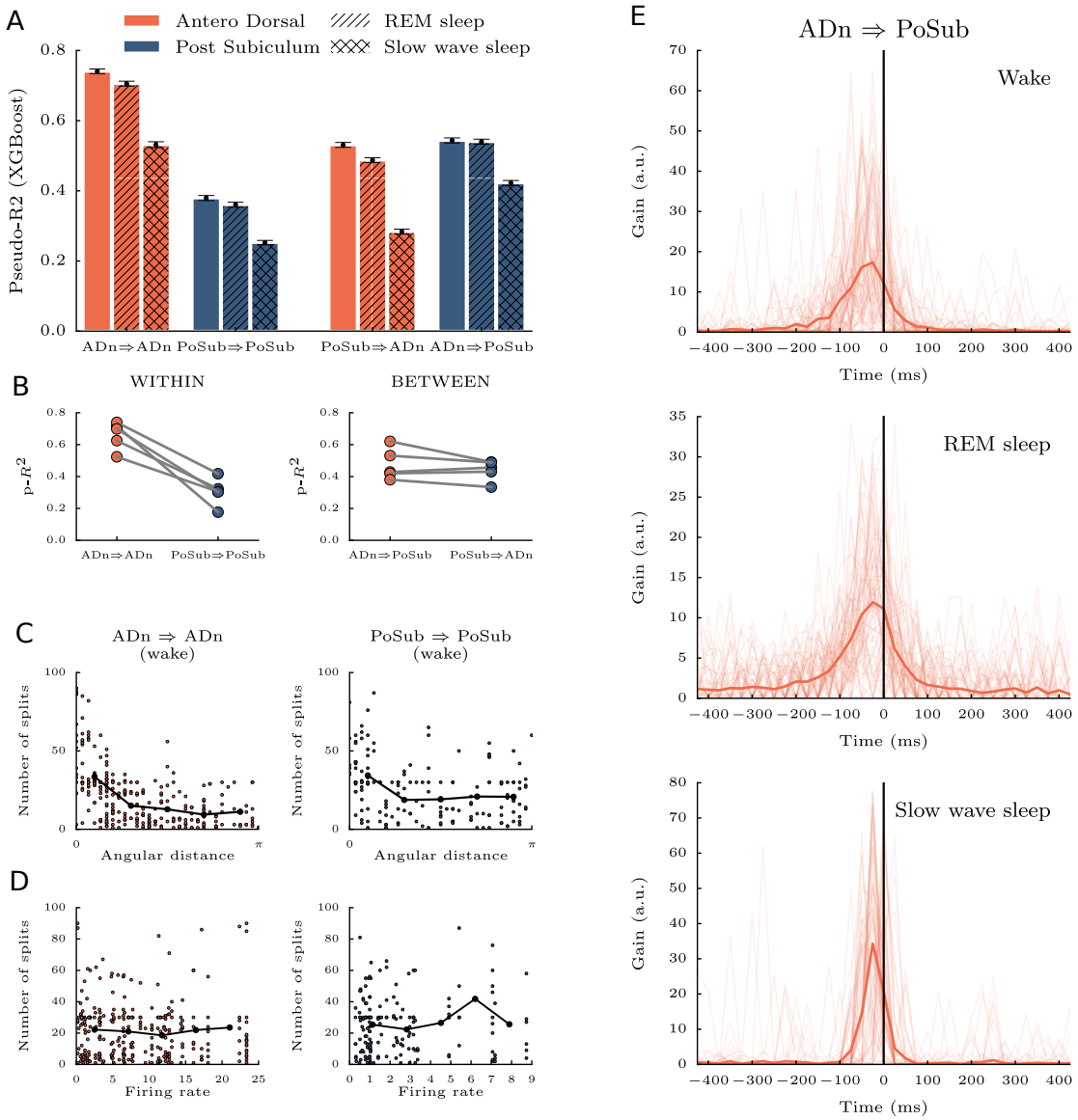

Figure 4: *Peer-prediction between ADn and PoSub.* ***A*** *Two conditions were tested: prediction between neurons of the same population (ADn⇒ADn and PoSub⇒PoSub) and prediction using neurons of the other population (PoSub⇒ADn and ADn⇒PoSub). Only sessions with at least 7 neurons in each population were included (2 animals). Peer-prediction was then tested during wake (plain bars), REM sleep (dashed bars) and slow wave sleep (crossed bars) episodes.* ***B*** *To rule out the possibility that the difference in scores resulted from uneven number of recorded neurons, the score were recomputed using an equal number of neurons in each population (i.e by randomly selecting neurons within the largest group).* ***C*** *Number of splits of one feature neuron given its angular distance with the target neuron.* ***D*** *Number of splits given the mean firing rate of the feature neuron. Despite firing rate differences, all features neurons contributed.* ***E*** *For the ADn⇒PoSub condition, the activity of ADn neurons at successive past to future time steps was used as feature space. The number of splits was maximal around 25 ms before the current time (vertical black lines) for wake and sleep.*

internally organized circuit [Peyrache et al., 2015].

Decoding signals from neuronal populations

XGB is a powerful tool to decode information, and we thus tested its performance on the decoding of the head-direction signal distributed over population of head-direction cells. To this end, unlike previous analysis, spiking activity was binned in 200ms windows and XGB was trained and compared to a Bayesian decoding method, a technique widely used for such tasks [Peyrache et al., 2015, Zhang et al., 1998], that predicts the probability of being in a particular head-direction at each time step based on the instantaneous spike count in the population. For both algorithms, 60 angular bins were used to predict the head-direction. We parametrized the gradient boosted trees to use the multi-class log-loss that outputs a probability of being in a certain class or not.

Using sessions that contained more than 7 neurons in both ADn and PoSub (n=5, two animals), we showed that gradient boosted trees and Bayesian decoding show similar performance when using ADn activity as a feature while gradient boosted trees slightly outperforms Bayesian decoding for PoSub activity(Fig. 5.A). An example of decoding from gradient boosted trees is shown in Fig. 5.B.

Discussion

The present methodology offers a new perspective for the analysis of neuronal dynamics and coding. Unlike classical tools that aim to provide interpretation of the data by investigating the predictability of a particular model of neuronal function, we show that gradient boosted trees [Chen and Guestrin, 2016, Friedman, 2001], a supervised learning technique commonly used in various fields of data mining, equals, if not outperforms other classes of machine learning models [Burges, 2010, Li, 2012]. This performance was achieved by a direct fit of raw data to the targeted spike trains, with no explicit prior on cell's response (e.g. a tuning curve, or a model of mixed-selectivity to a set of variables). We report optimal parameters and detailed methods to study neuronal response and dynamics as a function of behavior or endogeneous processes (e.g. the neuronal peer network). Furthermore, we show that the resulting tree structure, after learning of the data, can be itself analyzed to reveal important properties of the neuronal networks.

Learning neuronal firing in relation to behavioral data: performance and optimal parameters

We first sought to validate the approach of learning a predictive model of spike trains from behavioral data with a decision tree learning algorithm that does not include a predefined model of the training set. To this end, we analyzed a dataset of head-direction cells [Peyrache et al., 2015, Taube et al., 1990], whose firing in relation to behavior is among the best characterized signals in the mammalian nervous system. We have demonstrated first that gradient boosted trees predicts the firing of the neurons with high accuracy by establishing a direct correspondence between the raw behavioral data (in this case the head-direction angular value alone) and the instantaneous spiking of the neurons (Fig. 2). The performance was similar to a model-based approach (i.e. prediction of the firing rate on test data based on the tuning curve of the training set). This is not surprising for a class of neurons whose spiking activity is explained so well by an experimentally tractable signal. However, in general, tuning curves for even well defined neuronal responses explain only partially actual spike trains and XGB may well capture previously undetermined sources of variance.

Although XGB can be viewed as a model-free technique that does not assume any particular statistics or generative model of the input data, the procedure still depends on a limited set of free parameters that need to be tuned for optimal performance. To facilitate the use of this classifier for future studies and assure repro-

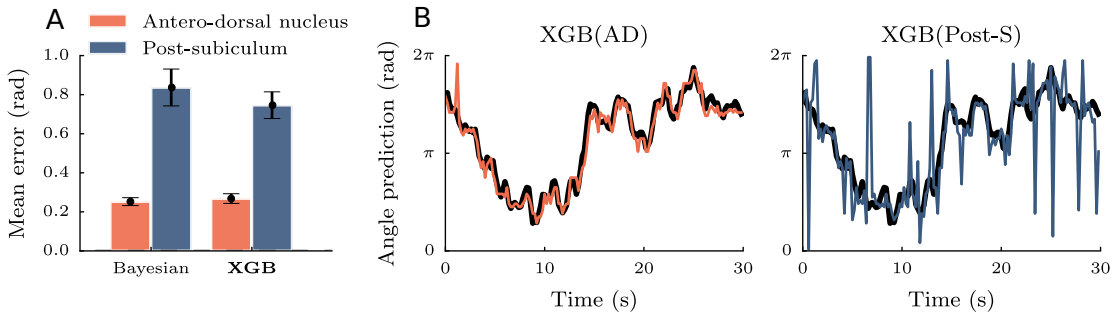

Figure 5: *Decoding of head-direction angle.* ***A*** *For sessions with large groups of neurons ($n \geq 7$) in ADn and PoSub, the head-direction of the animal was decoded based on spiking activity with the classical Bayesian decoding and gradient boosted trees (XGB) over 60 angular bins.* ***B*** *Example of decoding for XGB during 30 seconds of head rotation or both ADn and PoSub. The black line shows the real angular head-direction.*

ducibility of analyses across laboratories, we systematically explored the parameter space for depth and number of trees in the case of single neuron spike train prediction (in relation to head-direction). We found that minima were well localized, for all neurons, using the BIC score that penalizes over-complex models. In particular, we show how the use of multiple trees (approximatively 100), each limited in depth (typically five branching), was an optimal choice of parameters. The narrow distribution of parameters across neurons can stem from the fact that angular values are homogeneously distributed across sessions. Importantly, these optimal parameters did not seem to depend on neuron's intrinsic parameters (e.g. firing rates) and there was no obvious trade-off between tree depth and number of trees (the two optimal values were independently distributed across neurons).

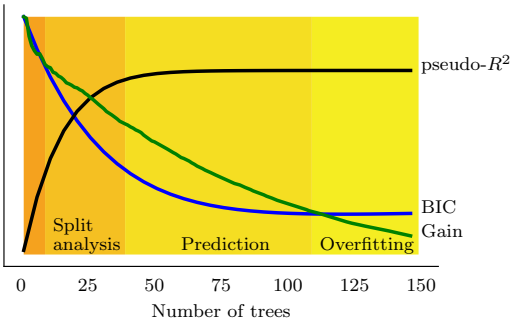

Figure 6: *Split analysis and optimal data prediction lies within different ranges of tree numbers (for a fixed tree depth). Thus, the use of gradient boosted trees requires a careful tuning of the parameters of the algorithm depending on the question (interpretability of the structure versus prediction and decoding of the signal).*

Interpreting the structure of the gradient boosted trees

While the structure of a multi-layered neural nets (or other forms of *deep* architecture) after learning the classification of a dataset is notoriously unanalyzable [Mikolov et al., 2013, Szegedy et al., 2013, Zeiler and Fergus, 2014], we show how the branching of the decision tree may be highly informative on how input data are matched to their output targets. The density of splits (or branching) across the series of trees was maximal in the range of inputs where firing rate changes the most. More precisely, data splitting was tightly correlated with the amount of Fisher Information, a measure of decoding performance. Indeed, there's a direct relationship between Fisher Information and the variance of the optimal estimator (the Cramer-Rao bound) Averbeck et al. [2006], Brunel and Nadal [1998], in one or several dimensions.

In the case of neuronal peer-prediction, ana-

lyzing the structure of gradient boosted trees presents the advantage that all kinds of neuronal interactions (positive, negative, linear or monotonically non-linear) yield comparable estimates (when quantifying split density or gain). This is in contrast with classical correlation analyses of individual spike trains relative to a population of peers that may be hard to interpret in certain cases where these interactions are both negative and positive ([Okun et al., 2015, Renart et al., 2010]. In addition, fitting spiking data to maximum entropy (i.e. Ising) models have revealed that linear correlations may not indicate the true coordination between spike trains [Cocco et al., 2009, Schneidman et al., 2006]. The analysis of tree branching provides an estimate of the statistical dependencies between spike trains, independent of the underlying type of interaction and without assuming a particular transfer function for the target neuron [Harris et al., 2003]. The nature of neuronal coordination as observed from spike trains is still debated, for example in the hippocampus [Chadwick et al., 2015], and unbiased, model-free methods may be highly informative on the nature of the actual statistical dependencies between neurons.

As summarized in Fig. 6, we also report an optimal range of tree number that should be used for split analysis. Using less trees allows to reveal the contribution of different features to the output target, at the expense of prediction and decoding performance. The structure of the branching is then best captured by the 'gain' of the learning. In contrast, fitting the data on too many trees lead to overfitting and should be avoided. Overall, the reader interested in using this technique should bear in mind that meaningful information about the dataset can sometimes be overshadowed by high split density. In such case, it is of best interest to reduce the number of trees and to control that the average gain for splits is large enough.

Measuring the contribution of multiple behavioral variables

A large class of neurons in the brain are modulated by several dimensions of incoming stimuli [Finkelstein et al., 2015, Hardcastle et al., 2017, Rigotti et al., 2013, Sargolini et al., 2006], a property refereed to as mixed-selectivity. Untangling the different contributions is sometimes challenging and gradient boosted trees offer a rapid and unequivocal approach to address this issue Benjamin et al. [2017], Truccolo and Donoghue [2007]. In fact, there is no intrinsic limit to the dimensionality of the inputs that can be learned. To further test this technique, we regressed spike trains of head-direction cells with position, as well as head-direction data. In line with previous reports [Cacucci et al., 2004, Peyrache et al., 2016], the head-direction cells of the PoSub correlated also with spatial factors while in the ADn, neurons coded mostly for the head-direction. XGB thus allows to rapidly explore the correlates of spike trains to measurements of external or internal variables of the system.

Prediction of feed-forward activation in a thalamo-cortical network in vivo

Investigation of neuronal dynamics does not always entail the regression of spiking data to variables of the experiments. Many studies have focused on the spatio-temporal coordination of neuronal networks *in vivo*, independent of any behaviorally-related processing ([Luczak et al., 2007, Okun et al., 2015, Peyrache et al., 2012]. In fact, the characterization of signal transmission between brain areas remains one of the most complex challenges of neuroscience as it requires first the recording of such data in vivo as well as the establishment of a proper model of interaction to determine the parameters of spike transmission (e.g. conduction delay and post-synaptic integration time).

Here we used data from the head-direction thalamo-cortical network [Peyrache et al., 2015] with simultaneous recording of PoSub and ADn. It allowed us to demonstrate a temporally-shifted relationship from ADn to

PoSub. More precisely, we used gradient boosted trees to predict PoSub head-direction cell firing activity based on the ensemble spike trains of the head-direction cells of the ADn, at various time-lags between the two series of spike trains. PoSub spiking was mostly dependent on ADn activity in preceding time bins (in average 25ms) thus indicating a likely feed-forward pathway. First, this replicates the findings that the head-direction signal of ADn neurons precedes the actual head-direction by about 25 ms [Blair and Sharp, 1995, Goodridge and Taube, 1997]. Second, the temporal asymmetry in the prediction of cortical spiking relative to thalamic activity was preserved during sleep, both during REM and non-REM, and therefore it indicates that this differential temporal coding likely emerges from intrinsic wiring and dynamics. This confirms antomical studies, as well as examination of putative synaptic interaction between neurons in this pathway Peyrache et al. [2015].

Gradient boosted trees match Bayesian decoding in performance

Neurons convey information about external parameters, and it should thus be possible to decode these signals from population activity. The best examples are the demonstration that position can be estimated from ensembles of hippocampal place cells during exploration and 'imagination' of future paths [Johnson and Redish, 2007, Pfeiffer and Foster, 2013, Wilson and McNaughton, 1993] as well as the head-direction signal during wakefulness and sleep Peyrache et al. [2015]. Decoding of neuronal signals has also been widely studied in the context of brain machine interface [Laubach et al., 2000].

Bayesian decoding is the tool of reference to estimate a signal from ensembles of neurons. In fact, it computes the probability distribution of a particular signal given the tuning curves of the neurons and the instantaneous spike counts in the neuronal population. This technique generally assumes that spike counts are drawn from Poisson processes and that neurons are independent from each other (Zhang et al. [1998]). Here we have compared the performance of Bayesian decoders with gradient boosted trees when decoding angular values, based on the activity of either ADn or PoSub. We found that gradient boosted trees matched Bayesian decoding when using ADn neurons but were better with PoSub activity. As emphasized in this report, PoSub activity does not encode only the head-direction. In case of mixed-selectivity signals, a model-free technique such as gradient boosted trees show decisive superiority in predicting an external variable.

Potential for neuroscience and future work

The potential of these methods to unravel dynamics of biological neuronal networks is tremendous and will be the scope of further studies. For instance, tracking synaptic transmission in pairwise spike trains [Barthó et al., 2004], uncoupling the phase-locking of neuronal spiking to concomitant and nested brain oscillations [Belluscio et al., 2012, Tort et al., 2009] and determining the nature of the coordination in neuronal populations in relation to behavior [Chadwick et al., 2015, Harris et al., 2003] are examples of the current challenges of data analysis in systems neuroscience. In addition, future improvements of brain-machine interface will require the development of reliable and robust tools to decode neuronal activity [Andersen et al., 2004, Lebedev and Nicolelis, 2006].

In summary, gradient boosted trees methods are potentially helpful tools to explore a dataset and make prediction on the underlying biological processes which, in turn, can be tested with more classical methods. They may also be used to decode signals for closed-loops experiments and brain-machine interface in animals or humans. Finally, these methods open avenues for the study of neuronal data in general as the branching of the tree structure can be analyzed as a 'proxy' of the biological system itself.

References

R. A. Andersen, S. Musallam, and B. Pesaran. Selecting the signals for a brain–machine interface. *Current opinion in neurobiology*, 14(6): 720–726, 2004.

B. B. Averbeck, P. E. Latham, and A. Pouget. Neural correlations, population coding and computation. *Nature reviews neuroscience*, 7 (5):358–366, 2006.

P. Barthó, H. Hirase, L. Monconduit, M. Zugaro, K. D. Harris, and G. Buzsáki. Characterization of neocortical principal cells and interneurons by network interactions and extracellular features. *Journal of neurophysiology*, 92(1):600–608, 2004.

M. A. Belluscio, K. Mizuseki, R. Schmidt, R. Kempter, and G. Buzsáki. Cross-frequency phase–phase coupling between theta and gamma oscillations in the hippocampus. *Journal of Neuroscience*, 32(2):423–435, 2012.

A. S. Benjamin, H. L. Fernandes, T. Tomlinson, P. Ramkumar, C. VerSteeg, L. Miller, and K. P. Kording. Modern machine learning far outperforms glms at predicting spikes. *bioRxiv*, page 111450, 2017.

H. T. Blair and P. E. Sharp. Anticipatory head direction signals in anterior thalamus: evidence for a thalamocortical circuit that integrates angular head motion to compute head direction. *Journal of Neuroscience*, 15(9): 6260–6270, 1995.

A. Borst and F. E. Theunissen. Information theory and neural coding. *Nature neuroscience*, 2 (11):947–957, 1999.

N. Brunel and J.-P. Nadal. Mutual information, fisher information, and population coding. *Neural computation*, 10(7):1731–1757, 1998.

C. J. Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81, 2010.

G. Buzsáki. Large-scale recording of neuronal ensembles. *Nature neuroscience*, 7(5):446–451, 2004.

G. Buzsáki. Neural syntax: cell assemblies, synapsembles, and readers. *Neuron*, 68(3): 362–385, 2010.

F. Cacucci, C. Lever, T. J. Wills, N. Burgess, and J. O'Keefe. Theta-modulated place-by-direction cells in the hippocampal formation in the rat. *Journal of Neuroscience*, 24(38):8265–8277, 2004.

A. C. Cameron and F. A. Windmeijer. An r-squared measure of goodness of fit for some common nonlinear regression models. *Journal of Econometrics*, 77(2):329–342, 1997.

A. Chadwick, M. C. van Rossum, and M. F. Nolan. Independent theta phase coding accounts for ca1 population sequences and enables flexible remapping. *Elife*, 4:e03542, 2015.

J. K. Chapin and M. A. Nicolelis. Principal component analysis of neuronal ensemble activity reveals multidimensional somatosensory representations. *Journal of neuroscience methods*, 94(1):121–140, 1999.

T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM, 2016.

T.-W. Chen, T. J. Wardill, Y. Sun, S. R. Pulver, S. L. Renninger, A. Baohan, E. R. Schreiter, R. A. Kerr, M. B. Orger, V. Jayaraman, et al. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499(7458):295–300, 2013.

S. Cocco, S. Leibler, and R. Monasson. Neuronal couplings between retinal ganglion cells inferred by efficient inverse statistical physics methods. *Proceedings of the National Academy of Sciences*, 106(33):14058–14062, 2009.

D. A. Dombeck, A. N. Khabbaz, F. Collman, T. L. Adelman, and D. W. Tank. Imaging large-scale neural activity with cellular resolution in awake, mobile mice. *Neuron*, 56(1): 43–57, 2007.

A. J. Ferreira and M. A. Figueiredo. Boosting algorithms: A review of methods, theory, and applications. In *Ensemble Machine Learning*, pages 35–85. Springer, 2012.

A. Finkelstein, D. Derdikman, A. Rubin, J. N. Foerster, L. Las, and N. Ulanovsky. Three-dimensional head-direction coding in the bat brain. *Nature*, 517(7533):159–164, 2015.

Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*, pages 23–37. Springer, 1995.

J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

J. P. Goodridge and J. S. Taube. Interaction between the postsubiculum and anterior thalamus in the generation of head direction cell activity. *Journal of Neuroscience*, 17(23): 9315–9330, 1997.

K. Hardcastle, N. Maheswaranathan, S. Ganguli, and L. M. Giocomo. A multiplexed, heterogeneous, and adaptive code for navigation in medial entorhinal cortex. *Neuron*, 94(2):375–387, 2017.

K. D. Harris. Neural signatures of cell assembly organization. *Nature Reviews Neuroscience*, 6 (5):399–407, 2005.

K. D. Harris, J. Csicsvari, H. Hirase, G. Dragoi, and G. Buzsáki. Organization of cell assemblies in the hippocampus. *Nature*, 424(6948): 552–556, 2003.

K. D. Harris, R. Q. Quiroga, J. Freeman, and S. L. Smith. Improving data quality in neuronal population recordings. *Nature Neuroscience*, 19(9):1165–1174, 2016.

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

M. D. Humphries. Dynamical networks: finding, measuring, and tracking neural population activity using network science. *bioRxiv*, page 115485, 2017.

A. Johnson and A. D. Redish. Neural ensembles in ca3 transiently encode paths forward of the animal at a decision point. *Journal of Neuroscience*, 27(45):12176–12189, 2007.

J. J. Jun, C. Mitelut, C. Lai, S. Gratiy, C. Anastassiou, and T. D. Harris. Real-time spike sorting platform for high-density extracellular probes with ground-truth validation and drift correction. *bioRxiv*, page 101030, 2017.

M. Laubach, J. Wessberg, and M. A. Nicolelis. Cortical ensemble activity increasingly predicts behaviour outcomes during learning of a motor task. *Nature*, 405(6786):567–571, 2000.

M. A. Lebedev and M. A. Nicolelis. Brain–machine interfaces: past, present and future. *TRENDS in Neurosciences*, 29(9):536–546, 2006.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

P. Li. Robust logitboost and adaptive base class (abc) logitboost. *arXiv preprint arXiv:1203.3491*, 2012.

V. Lopes-dos Santos, S. Ribeiro, and A. B. Tort. Detecting cell assemblies in large neuronal populations. *Journal of neuroscience methods*, 220(2):149–166, 2013.

A. Luczak, P. Barthó, S. L. Marguet, G. Buzsáki, and K. D. Harris. Sequential structure of neocortical spontaneous activity in vivo. *Proceedings of the National Academy of Sciences*, 104(1):347–352, 2007.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

J. O'keefe and L. Nadel. *The hippocampus as a cognitive map*. Oxford: Clarendon Press, 1978.

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

D. H. Perkel, G. L. Gerstein, and G. P. Moore. Neuronal spike trains and stochastic point processes: Ii. simultaneous spike trains. *Biophysical journal*, 7(4):419–440, 1967.

A. Peyrache, K. Benchenane, M. Khamassi, S. I. Wiener, and F. P. Battaglia. Principal component analysis of ensemble recordings reveals cell assemblies at high temporal resolution. *Journal of computational neuroscience*, 29(1-2): 309–325, 2010.

A. Peyrache, N. Dehghani, E. N. Eskandar, J. R. Madsen, W. S. Anderson, J. A. Donoghue, L. R. Hochberg, E. Halgren, S. S. Cash, and A. Destexhe. Spatiotemporal dynamics of neocortical excitation and inhibition during human sleep. *Proceedings of the National Academy of Sciences*, 109(5):1731–1736, 2012.

A. Peyrache, M. M. Lacroix, P. C. Petersen, and G. Buzsáki. Internally organized mechanisms of the head direction sense. *Nature neuroscience*, 18(4):569–575, 2015.

A. Peyrache, L. Roux, N. Schieferstein, and G. Buzsaki. Transformation of head-direction signal into spatial code. *bioRxiv*, page 075986, 2016.

B. E. Pfeiffer and D. J. Foster. Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, 497(7447):74–79, 2013.

J. W. Pillow, L. Paninski, V. J. Uzzell, E. P. Simoncelli, and E. Chichilnisky. Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model. *Journal of Neuroscience*, 25(47):11003–11013, 2005.

J. W. Pillow, J. Shlens, L. Paninski, A. Sher, A. M. Litke, E. Chichilnisky, and E. P. Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

A. Pouget, J. M. Beck, W. J. Ma, and P. E. Latham. Probabilistic brains: knowns and unknowns. *Nature neuroscience*, 16(9):1170–1178, 2013.

A. Renart, J. De La Rocha, P. Bartho, L. Hollender, N. Parga, A. Reyes, and K. D. Harris. The asynchronous state in cortical circuits. *science*, 327(5965):587–590, 2010.

F. Rieke. *Spikes: exploring the neural code*. MIT press, 1999.

M. Rigotti, O. Barak, M. R. Warden, X.-J. Wang, N. D. Daw, E. K. Miller, and S. Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497(7451):585–590, 2013.

F. Sargolini, M. Fyhn, T. Hafting, B. L. McNaughton, M. P. Witter, M.-B. Moser, and E. I. Moser. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774):758–762, 2006.

R. E. Schapire. The boosting approach to machine learning: An overview. In *Nonlinear estimation and classification*, pages 149–171. Springer, 2003.

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

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

J. S. Taube. The head direction signal: origins and sensory-motor integration. *Annu. Rev. Neurosci.*, 30:181–207, 2007.

J. S. Taube, R. U. Muller, and J. B. Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435, 1990.

A. B. Tort, R. W. Komorowski, J. R. Manns, N. J. Kopell, and H. Eichenbaum. Theta–gamma coupling increases during the learning of item–context associations. *Proceedings of the National Academy of Sciences*, 106(49):20942–20947, 2009.

W. Truccolo and J. P. Donoghue. Nonparametric modeling of neural point processes via stochastic gradient boosting regression. *Neural computation*, 19(3):672–705, 2007.

W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005.

M. A. Wilson and B. L. McNaughton. Dynamics of the hippocampal ensemble code for space. *Science*, 261(5124):1055–1058, 1993.

R. Yuste. From the neuron doctrine to neural networks. *Nature Reviews Neuroscience*, 16(8): 487–497, 2015.

M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

K. Zhang, I. Ginzburg, B. L. McNaughton, and T. J. Sejnowski. Interpreting neuronal population activity by reconstruction: unified framework with application to hippocampal place cells. *Journal of neurophysiology*, 79(2): 1017–1044, 1998.

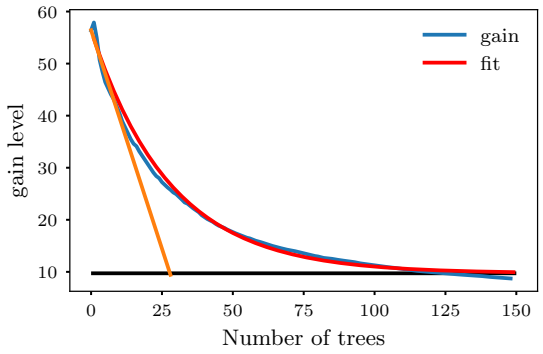

Figure S1: *Learning gain per tree decreases with the number of trees (blue line). This decay was well captured by an exponential fit (red line), from which an optimal number of trees of approximatively 30 trees is derived (interesect of the linear fit at origin with the x-axis). At this stage the mean gain per tree is approximately* $\frac{1}{3}$ *of its initial value and most of the learning has already occurred.*

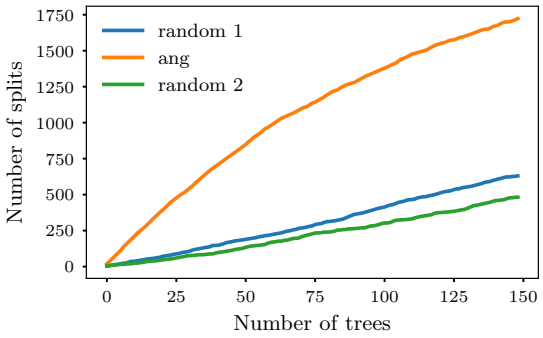

Figure S2: *Features carrying actual signal are preferentially split in the first trees, resulting in higher gain. The graph illustrates the evolution of split density when learning the spike train of a head-direction neuron as a function of the number of trees for three features: the actual head-direction and two random vectors. Split density increased linearly and similarly with the number of trees in the asymptotic regime for all features. However, the increase was much higher for the head-direction at low tree numbers, a difference well captured by gain analysis. Note that, as the order of features in the algorithm may impact which are split first, we showed how the feature data were organized (random 1, angle and random 2).*

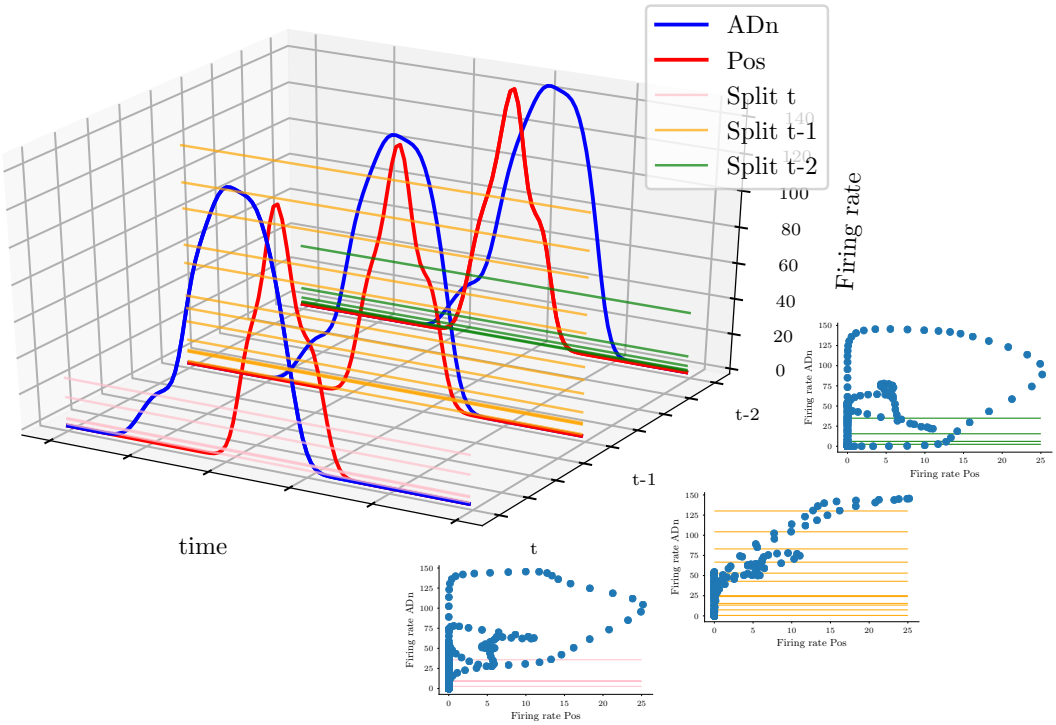

Figure S3: *Revealing temporal delay in peer-prediction. Feature space is composed of multiple copies of the activity of the feature neuron (in this case, in the ADn) at various time-lags (blue curves) to learn the target spike train (PoSub, red curves). The relationship between the two spike trains shows maximal dependence at t-1, resulting in a high number of splits by the algorithm (yellow horizontal lines). Splitting was less effective for more independent firing at t and t-2. In this example, the relationship at t-1 is trivial (linear and positively correlated). However, the quantification of these interactions give comparable values for a large variety of interactions (e.g. positive, negative or monotonically non linear).*