Synthetic Data Resource and Benchmarks for Time Cell Analysis and Detection Algorithms

AUTHORS: K. G. Ananthamurthy[1], Upinder S. Bhalla[1]*

1. National Centre for Biological Sciences, Tata Institute of Fundamental Research (NCBS-TIFR), GKVK Campus, Bellary Road, Bangalore – 560065, Karnataka, India
*To whom correspondence should be addressed. E-mail: bhalla@ncbs.res.in

Summary

Hippocampal CA1 cells take part in reliable, time-locked activity sequences in tasks that involve an association between stimuli, in a manner that tiles the interval between the stimuli. Such cells have been termed time cells. Here we adopt a first-principles approach to comparing diverse analysis and detection algorithms for identifying time cells. We developed a resource for generating synthetic activity datasets using calcium signals recorded *in vivo* from mouse hippocampus using 2-photon imaging, for template response waveforms. We assigned known, ground truth values for properties of time cells in this synthetic dataset, including noise, timing imprecision, hit-trial ratio and calcium event width. These datasets were the input to a pipeline for testing multiple algorithms for time cell detection to determine the conditions for which they were best suited, and evaluate their effective operating ranges. We find that most algorithms are sensitive to noise. Only a few methods benefit from larger event widths. Reassuringly, most methods are insensitive to timing imprecision, and exhibit successful time cell detection even at low hit trial ratios. Importantly, all methods show good concordance in identifying cells as time cells.

Introduction

The mammalian hippocampus is important for the formation of several kinds of memory, one of which is the association between stimuli occurring separately in time. Time cells were originally described using tuning curves from single-unit recordings of cellular activity when rats ran on a running wheel in between behavioural decisions (Pastalkova et al. 2008). These cells exhibited time tuning of the order of seconds. Several further studies have shown that small populations of hippocampal CA1 cells fire in time-locked sequences, "bridging" the time gap between stimulus and response in temporal delay tasks lasting several seconds (Kraus et al. 2013; MacDonald et al. 2013; MacDonald et al. 2011; Pastalkova et al. 2008). Cellular calcium imaging studies have also been used to report time cells,

albeit at slower sampling rate (Mau et al. 2018; Modi, Dhawale, and Bhalla 2014). For example, similar interval tiling properties of hippocampal CA1 neurons were observed on much shorter, 500 ms time-scales in a Trace Eye-Blink Conditioning (TEC) task (Modi et al. 2014). Spontaneous sequential activity has also been reported in free-running animals (Villette et al. 2015). Such cells with a well-defined temporal firing field are commonly termed time cells (Eichenbaum 2017; MacDonald et al. 2011). However, there is a wide diversity of methods used to detect and characterise time cells, and it is not clear how consistent these methods are in classifying cells as time cells. It is also unclear how sensitive each method may be to a range of physiological sources of variability and noise. A consistent set of benchmarks of classification performance is necessary to draw accurate and comparable conclusions from real physiology data across different methods and different laboratories. Our approach in the current study is not prescriptive, but pragmatic: we ask how existing methods work when we already know exactly which cells are time cells, and we determine how well each method deals with imperfect data.

The major approaches used to identifying time cells are tuning curves (peristimulus time histograms), temporal information, principal component analysis with time offset, support vector machines, and bootstrap analysis of activity peaks. Several studies have used a temporal delay task lasting several seconds, in which a rat runs on a treadmill during the delay period. A Temporal Information metric (Mau et al. 2018) has been used to find individual time cells in such tasks. A distinct task involves monitoring recurrent sequences of activity during free-running treadmill recordings. Such datasets have been analysed using Offset Principal Component Analysis (Kaifosh et al. 2013; Malvache et al. 2016; Villette et al. 2015), to first denoise 2P data, establish correlation coefficients, and detect hippocampal CA1 sequences. Time cells have also been reported for much shorter duration tasks (~500 ms) such as hippocampus-dependent trace conditioning (Modi et al. 2014; Tseng et al. 2004). Time cells in these 2P datasets were identified using yet another method, in which bootstrapping was used to determine if peak activity at a given time was different from chance. This method was termed Ratio of Ridge/Background (Modi et al, 2014). Yet other methods have utilised support vector machines to categorise time cells (Ahmed et al., 2020)

Time-cell detection is closely related to sequence detection, which has been fraught with statistical challenges. For example, detection of synfire chains has been the subject of some debate (Ikegaya et al. 2004; Lee and Wilson 2004; Mokeichev et al. 2007; Schrader et al. 2008). Time cell detection is usually easier, in that in most experiments there is a well-defined initiating stimulus and a known delay or trace phase (however, see Villette et al., 2015). For any cell identified as a time cell, it is desirable to define a score to measure quality or reliability along with decodable time. Hence it is also valuable to be able to compare the score of a time cell across recordings and even between groups, using well defined, analog measures. Each algorithm currently used in the literature implements a different scoring method and it is as yet unclear if comparable results would be observed with other metrics.

In the current study we compare these diverse methods by estimating their performance on synthetic test datasets where we controlled all features of the data, including the identity and timing of each time cell. The development of a synthetic dataset serves two purposes. First, it facilitates principled comparison of different methods, since the ground truth is known. Second, it facilitates an analysis over many dimensions of input variance, corresponding to very different experimental and circuit contexts. Richness in variety of input data allows for better sampling of the performance of the analyses under many potential conditions. We have explored variance along the key dimensions of Noise, Timing Imprecision, Signal Widths, Frequency of Occurrence, as well as several others. To strengthen the applicability of this synthetic data resource to real data, our generated output uses sampled experimental data.
Our experimental data, synthetic dataset, and code base are intended to be a resource for algorithm testing and optimization. All code is available and maintained at https://github.com/BhallaLab/rho-matlab .

Methods

Animals, Chronic Implants, and Behavioural Training
All experimental procedures were approved by the National Centre for Biological Sciences Institutional Animal Ethics Committee (Project ID: NCBS-IAE-2016/20(M)), in accordance with the guidelines of the Government of India (Animal Facility CPCSEA registration number 109/1999/CPCSEA) and equivalent guidelines of the Society for Neuroscience.

To be able to chronically monitor the activity of the same population of Hippocampal CA1 cells, we implanted 2-4 month old GCaMP6f mice with an optical window and head-bar using a protocol adapted from previously published methods (Dombeck et al. 2010). Briefly, anaesthesia was induced with 2-3% Isoflurane in a chamber, and subsequently maintained (breathing rate of ~1 Hz) with 1-2% Isoflurane, directly to the mouse's nose using an inverted pipette tip. Surgery was performed on a temperature-controlled table, maintained at 36.5 °C, while the anaesthetised animal was cheek-clamped. After a haircut, a ~5 cm piece of scalp was cut open to reveal the skull. A ~3mm circular craniotomy was then performed at a position 2 mm caudal and ~1.5 mm lateral to Bregma, on the left hemisphere. After gently tearing off the Dura, the underlying cortex was carefully aspirated till the Corpus Callosum (CC) layer, clearing out any blood using repeated washes of Cortex Buffer (Modi et al. 2014). A small thickness of corpus callosum fibres were then carefully aspirated till horizontal CC fibres were sparse but visible. The Cortex Buffer was then carefully suctioned out to dry the exposure till tacky. The exposure was then quickly sealed using a thin layer of Kwik-Sil and a coverslip attached to the bottom of a 3mm steel cannula. This preparation left the CA1 cell body layer ~200 µm below the most exposed tissue. Finally, an imaging head-bar was surgically implanted and fixed to the scalp, using dental cement and skull screws, before the animal was brought out of anaesthesia.

After being implanted with imaging head-bars, the animals were allowed to recover for 1-5 days, with a further 3-4 day habituation to the rig. Following this imaging commenced.

Two-Photon Imaging

To record calcium activity from 100-150 Hippocampal CA1 cell bodies, in vivo, at cellular resolution, we used a custom-built two photon laser-scanning microscope[5] and performed galvo-scans through the imaging window, over a field of view of ~100 μm^2, at 14.5 Hz, during TEC (Fig 1A). An Arduino microcontroller was used to control the behaviour routines, and it additionally sent a TTL trigger to initiate the imaging trials. The behaviour and imaging were conducted simultaneously to record calcium activity when the animal was learning the task

Time-series fluorescence data for various cells was extracted using Suite2P (Pachitariu et al. 2017), while further analysis was performed using custom written scripts on MATLAB R2017b. The average of the fluorescence values for cell specific pixels is then converted into the fold change relative to the baseline (dF/F_o; F_o as 10th percentile), for every marked cell, in every trial (Fig 1B). These dF/F traces were used for the rest of the analysis.

Curating a library of calcium events

We mapped our imaging dataset into a matrix of dF/F values for all cells, trials, and frames. We then identified calcium events as signal deviations that were above a threshold (mean + 2*standard deviation) for more than 4 consecutive frames (frame rate: 14.5 Hz or ~70 ms per frame). Once identified, we curated a library for each event by a cell, and saved the respective start indices and widths (Fig 1C). Using this library we generated synthetic data by inserting experimental calcium events into the time-series for each simulated cell. This approach just uses a time-series of signal bins and amplitudes, hence is signal-agnostic and could be applied to other imaging and recording modalities. In the interests of data integrity, our synthetic datasets were watermarked to be distinguishable from real physiology datasets.

Generating Synthetic Data

We generated synthetic activity by recycling samples of recorded calcium transients inserted into the synthetic datasets at frames (time) determined by the synthesis algorithm. The input parameters to this algorithm included timing, noise, imprecision, event width selection, hit-trial ratio, and several others. We aimed to cover the most likely conditions to affect timing and other experiment design properties. In more detail, we generated synthetic datasets using the following control parameters:

- Time Cell Percent

Value: Number between 0 and 100. This sets the number of cells that are assigned tuned calcium activity as a percentage of total cells, and controls the number of Positive and Negative Class cells in the dataset.

- Cell Order

Value: ‘basic’ or ‘random’. In ‘basic’ mode, Time Cells are indexed lower than Other Cells. In ‘random’ mode the indices of Time Cells and Other cells are randomly selected. This should have no impact on algorithm detection but is useful for visualisation.

- Max Hit Trial Percent

Value: Number between 0 and 100. This sets the maximum possible fraction of total trials, during which a Time Cell will exhibit tuned calcium activity.

- Hit Trial Percent Assignment

Value: ‘fixed’ or “random”. In ‘fixed’ mode, the number of Hit Trials is set as defined by Max Hit Trial Percent. In ‘random’ mode, the number of Hit Trials is calculated by randomly picking a value from a range (½*Max Hit Trials, Max Hit Trials).

- Trial Order

Value: ‘basic’ or ‘random’. In ‘basic’ mode the Hit Trials are indexed lower than Miss Trials. In ‘random’ mode the indices of Hit and Miss Trials are randomly selected. Specific patterns of Hit and Miss Trials for a session have not been reported in physiology, so this feature is not implemented.

- Event Width

Value: {0-100 percentile value, Integer N}. For each cell, this defines the selection of events based on width in frames. The percentile value is estimated from the histogram of all event widths. The variance of this selection is set by “N”, which adds N*Std. Dev. to the selection. All synthetic cells exhibit a range of different calcium events. This is considered an important parameter.

- Event Amplification Factor

Value: Number from 0 to +∞. This allows additional control to multiplicatively amplify any chosen calcium event, before incorporation. Our library was curated from physiologically recorded signals. The default value is 1.

- Event Timing

Value: ‘sequential’ or ‘random’. In ‘sequential’ mode the time of peak calcium activity is reflected by the indexing of the Time Cells. In ‘random’ mode the time of peak calcium activity is randomly dispersed over the trial frame-points.

- Start Frame

Value: Number from 0 to total number of frames. This sets the timing of the first cell in a Time Cell Sequence.

- End Frame

Value: Number from 0 to total number of frames. This sets the timing of the last cell in a Time Cell Sequence.

- Imprecision FWHM

Value: Number from 0 to total number of frames. This sets the lower and upper bounds for the difference in timing of calcium activity across trial pairs

for a Time Cell. We use this parameter to model trial to trial variability and is considered an important parameter to test.

- Imprecision Type

Value: ‘none’, ‘uniform’ or ‘normal’. In ‘uniform’ and ‘normal’ modes, the trial pair Imprecision is picked from a Normal and Uniform Distribution, respectively. In ‘none’ mode, the trial pair Imprecision defaults to 0.

- Noise

Value: ‘gaussian’ or ‘none’. In ‘gaussian’ mode the noise is sampled as a time-series vector with points from a Gaussian distribution. In ‘none’ mode, the Noise Percent defaults to 0.

- Noise Percent

Value: Number from 0 to 100. This allows scaling for any sample noise point, based on the max signal value for any cell.

Implementation of a Reference Quality measure, Q

In order to compare the readouts from the various time-cell detection methods, we implemented a reference measure of Quality (Q) of synthetic time cells that utilized the known inputs to the generation algorithm. Based on preliminary analysis, we selected following four parameters as the most likely to affect the behaviour and detection of Time Cells:

1. Noise
2. Event Width
3. Imprecision
4. Hit Trial Ratio

Accordingly, we were able to calculate a Reference Quality measure, using the following equation:

$$RefQ = HTR \times e^{-\left\{\frac{\alpha \times MNP}{100 \times EAF} + \frac{\beta \times std.\ dev.\ EW}{mean\ EW} + \frac{\gamma \times std.\ dev.\ Imp}{Stim\ Win}\right\}}$$

… (Eq. 1)

… where HTR: Hit Trial Ratio
MNP: Max Noise Percent (%)
EAF: Event Amplification Factor
EW: Event Widths (frames)
Imp: Imprecision (frames)
Stim Win: Stimulus Window (frames)
α: 1
β: 1
γ: 10

The values of α, β, and γ, were set to have comparable effects of each of the terms inside the exponent. This Reference Q was extremely useful for debugging code. A representative schematic for ‘low’ and ‘high’ values with each of these four parameters is shown (Fig. 1).

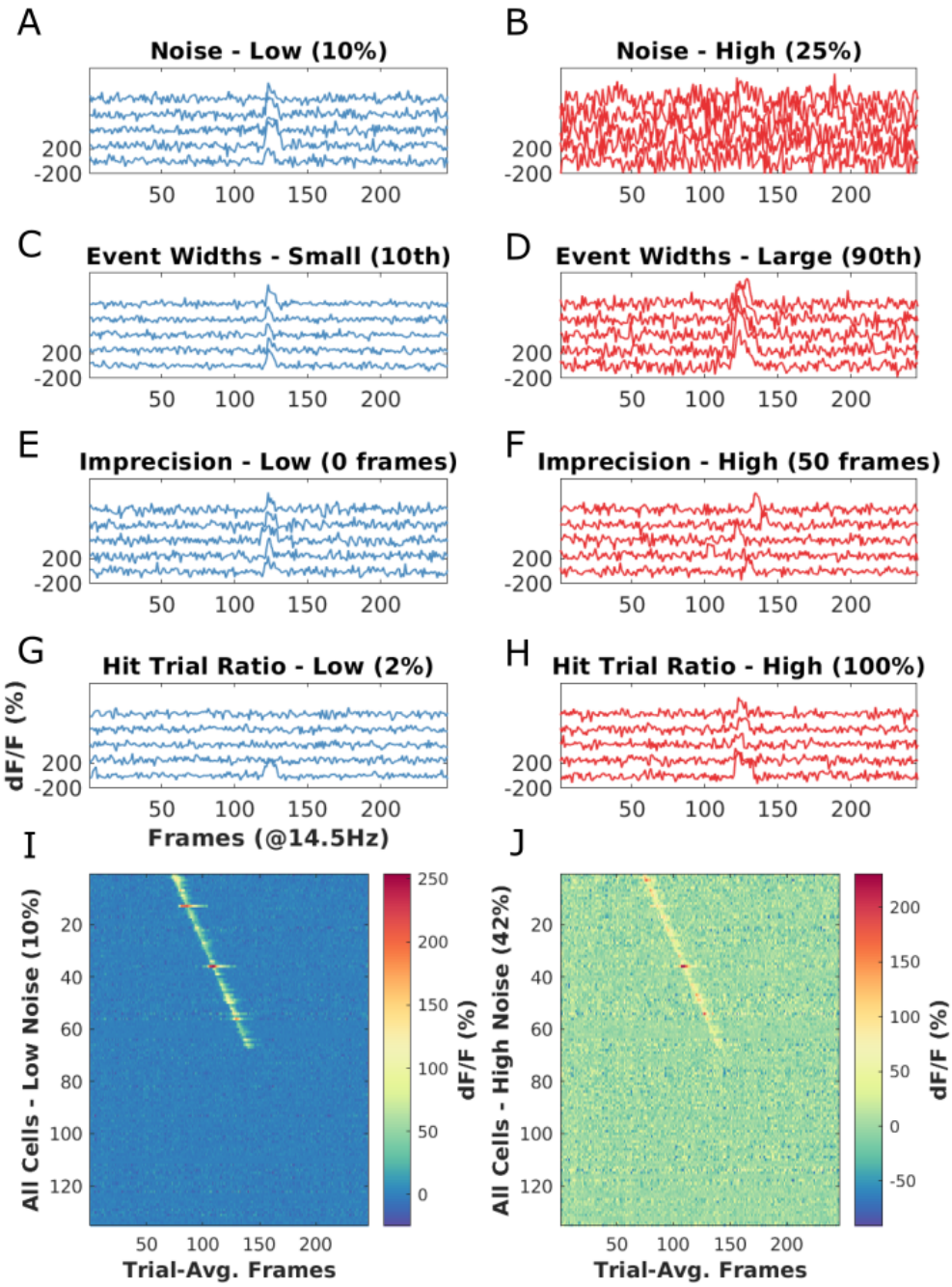

Fig. 1: Key features of synthetic datasets. Left, blue panels: low range of features. Right, red panels: High range of features. A: Noise = 10% B: Noise = 25%, C: Event Width: 10th percentile +/- 1 std. dev., D: Event Width 90th percentile +/ 1 std. dev. E: Imprecision, 0 frames FWHM F: Imprecision: Right: 50 frames FWHM, G: upto 2% Hit Trial Ratio from 0 to 2%. H: Hit Trial ratio from 0 to 100%, I: Example synthetic dataset of 135 neurons, displayed as heatmap sorted by time of peak Ca signal. I: Baseline Physiology synthetic data trial-average with 10% Noise (Low), and J: Same as I with 42% Noise (High). In both cases 50% of the cells (top 67) are time cells and the remainder are not.

Synthetic Datasets generated and analysed in batch mode

We generated datasets pertaining to Parameter Sensitivity Analysis by modulating one of the four main parameters and setting the others to non-interference levels. In this manner we devised 99 cases to study in which one of the main parameters was varied. Note that in these cases the resultant activity was in an un-physiological regime because other sources of variation were kept to low levels so as not to interfere with the parameter of interest. With 3 randomised shuffles, we generated 297 unique datasets. Additionally, we wanted to use more realistic datasets, where we would modulate one of the four parameters while keeping the others to ranges typical of physiological data. We devised 12 physiological cases. With 10 randomised shuffles, we generated 120 additional unique datasets.

Altogether, we had 417 unique datasets for our tests, each with 135 cells (total: 56,295 cells), 60 trials, and 246 frames/trial. Except when the percent time cells were modulated, all datasets featured 50% time cells.

We next implemented an analysis pipeline to run all the datasets through the time-cell detection algorithms, yielding scores and predictions for each case. Finally all the scores and predictions were collated for comparison and benchmarks as shown in the schematic (Fig. 2).

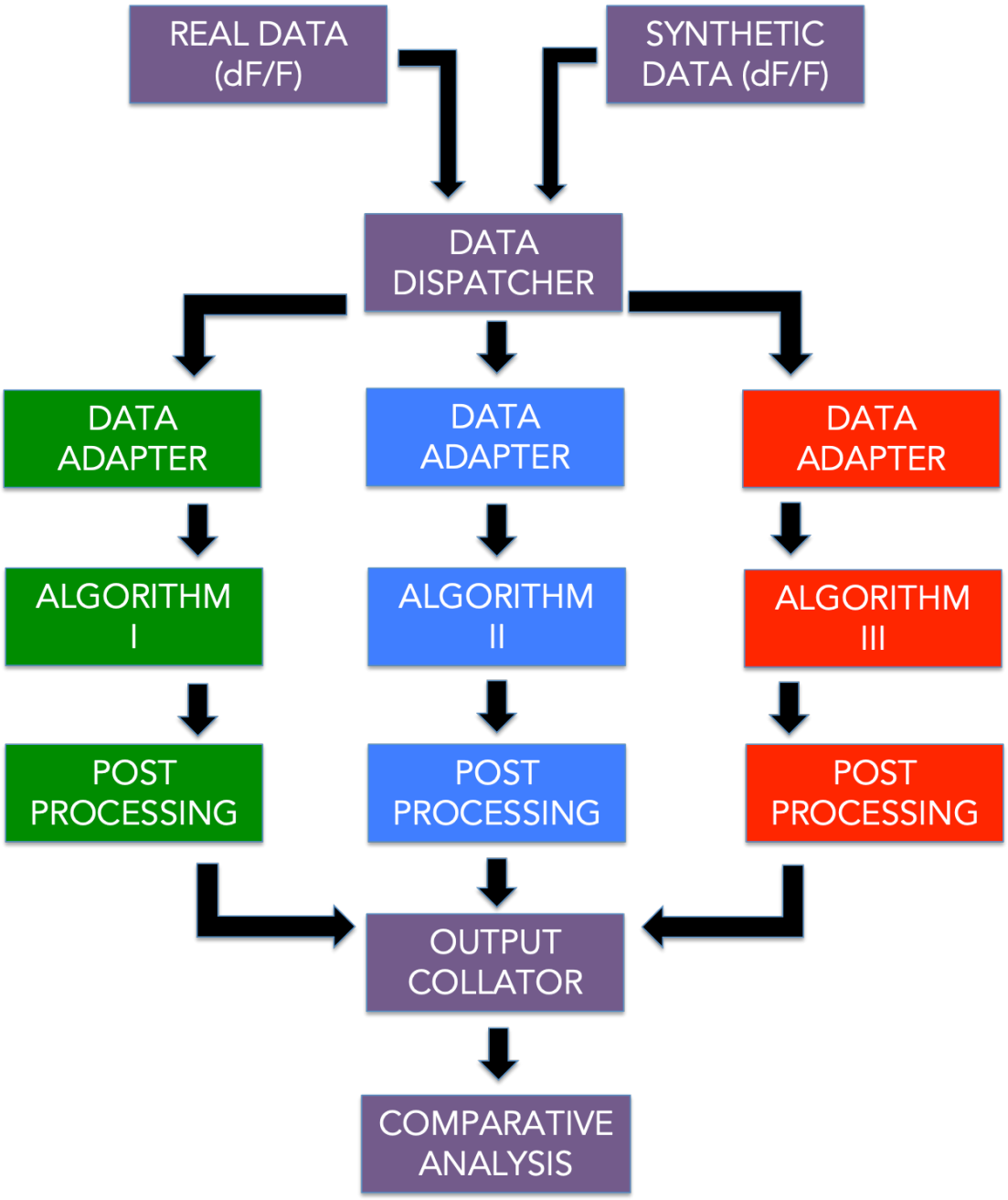

Fig. 2: A schematic representation of the analysis pipeline. Physiology data as well as synthetic data was analysed by 8 different implemented algorithms and the output was collated for comparative benchmarks.

All code development was done on MATLAB R2017b and batch analysis runs were performed on MATLAB R2021a.

Metrics for time cell Classification Performance

Recall is inversely proportional to the number of False Negatives (Type II error) and is the fraction of True Positive Class predictions over all Positive Class ground labels.

$$\text{Recall} = \text{TPR} / (\text{TPR} + \text{FNR}) \quad \text{… (Eq. 2)}$$

Precision is inversely proportional to the number of False Positives (Type I error) and is the fraction of True Positive Class predictions over all Positive Class predictions.

$$\text{Precision} = \text{TPR} / (\text{TPR} + \text{FPR}) \qquad \text{… (Eq. 3)}$$

F1 Score is the harmonic mean of Recall and Precision, and represents the lower of the two more.

$$\text{F1 Score} = 2*\text{Precision}*\text{Recall} / (\text{Precision} + \text{Recall}) \qquad \text{… (Eq. 4)}$$

… where,

TPR: True Positive Rate
FNR: False Negative Rate
FPR: False Positive Rate

Here are the definitions for Confusion Matrices (Tbl. 1),

GROUND TRUTH	PREDICTION/CLASSIFICATION	REMARK
0/False/Other cell	False/0	True Negative (TN)
0/False/Other cell	True/1	False Positive (FP)
1/True/Time cell	False/0	False Negative (FN)
1/True/Time cell	True/1	True Positive (TP)

Table 1: A tabular guide to Confusion Matrices for predictive/classification performance evaluation. For each detection algorithm, the classification results were compared to known ground truth values to get the total number of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) cases.

Results

We developed a pipeline (Fig. 2) with 8 different algorithm implementations for Time Cell detection, which involve scoring and then classifying cells. Here are the abbreviations for all the methods (Tbl. 2),

TIME CELL SCORING METHOD	ABBREVIATION
Ratio of Ridge to Background Ratios	rR2B
Temporal Information	TI
Peak Area Under Curve	pAUC
Offset PCA and Correlations	offPCA
Support Vector Machine	SVM
Parametric Equation	Param

Table 2: Abbreviations used for each of the detection algorithms.

Next, we describe the implementation of each of the methods.

Time Cell Scoring Methods and Classification

Ratio of Ridge/Background with Bootstrapping – "rR2B-Bo" (Modi et al., 2014)

The time of peak response for each cell was identified in averaged, non-overlapping trials' worth of ΔF/F traces, in the CS-onset to US-onset period, or as specified. The rest of the trials were averaged and the summed area under the time of peak was estimated. The ridge was then defined to be a 200 ms window centered at the peak. Next, we calculated the summed area in the ridge window as well as the background (non-ridge frames) to get the ridge to background ratio. As a control condition, these traces were given random time-offsets and then averaged. An independent time of peak was identified for each random-offset, averaged trace and ridge to background ratio calculated for it. This bootstrapping was repeated 5000 times for each cell's data and averaged. The reliability score was then calculated individually, for each cell, as the ratio of the ridge to background ratio for aligned traces to that of random-offset traces.

Quality (Q) = R2B/Mean R2B for randomised data,

… where, R2B stands for the Ridge/Background

This method can handle real or synthetic data. The code was adapted from Modi et al., 2014.

Temporal Information with Bootstrapping – "TI-Bo" (Mau et al., 2018)

Here we used the algorithm from Mau et al, as follows. There was an initial criterion of cells to have activity in at least 25% of trials. Their activity was summed into event time histograms with a bin size of 3 frames. The Temporal Information (TI) was estimated using Eq 5,

$$TI = {}^{1}/_{\lambda} \times \sum \lambda j \times log2\left({}^{\lambda j}/_{\lambda}\right) \times Pj$$

… (Eq. 5)

… where, λ is the Average Transient rate for each cell

λ_j is the Average transient rate for each cell in bin "j"

P_j is the Probability that the mouse was in time bin "j"

Quality (Q) = Temporal Information (TI)

Bootstrapping was used to determine if each trial had a TI greater than chance. We randomly shuffled the frame points to develop a random activity model (1000 iterations) and classified cells as time cells if $\lambda > \lambda_{rand}$ in greater than 99% of the models for at least 2 consecutive bins. This method can handle labelled and unlabeled data.

<u>Peak AUC of Signal with Bootstrapping and Otsu's Threhsold – "pAUC-Bo" and "pAUC-Ot" (in-house)</u>

Event Time Histograms (ETHs) were obtained for every cell, with 3 frames/bin. The bin with the peak Area Under Curve (AUC) was identified from the Event Time Histograms (ETH). With this, we developed a measure of Quality, described as,

Quality (Q) = Peak Time Bin AUC/Std. Dev. of the signal in the peak time bin of ETH

To classify cells for this method, we used Bootstrapping (as described above), as well as Otsu's threshold. This method can handle labelled and unlabeled data.

<u>PCA based Analysis with Otsu's Threshold – "offPCA-Ot" (Villette et al., 2015)</u>

We performed Offset PCA (Kaifosh et al., 2013) on the Dataset to identify the Principal Components that best explain the variance in the data. In previous reports, a manual selection of the appropriate Principal Component (PC) was performed to find which component best captured the aspects of the data that one is interested in. To be able to automate this method pipeline for batch mode, we set the selected Principal Component to the 1st (pre-sorted). Next, we took the first derivative of this component, and estimated the correlation coefficient between a cell's activity and the first derivative of the PC1.

Quality (Q) = corrcoef(cell activity, d(PC1)/dt)

The classification uses Otsu's threshold and is applicable to labelled or unlabeled datasets.

Implemented based on Villette et al., 2015.

<u>Support Vector Machine with Otsu's Threshold – "SVM-Ot" (in house)</u>

We used half the trials of the test dataset to train a Support Vector Machine (SVM) model and identify the decision boundary. Using the remaining trials we classified cells based on the direction and distance from the decision boundary. We defined Quality (Q) = Euclidean distance from hyperplane

Given our variety of synthetic datasets, we found many cases where a decision boundary could not be estimated, leading to one-class learning scores, instead of Positive and Negative Class scores. Further, this method is limited because it requires labelled datasets for training.

Parametric Equations with Bootstrapping and Otsu's Threhsold – "Param-Bo" and "Param-Ot" (in-house)

We developed this method to score cells in a manner similar to the Reference Quality, which uses the known ground truth of the input parameters given to the generator functions for the synthetic dataset. Rather than using the known inputs, this method computes the corresponding parameters read out or estimated from the dataset, whether synthetic or real. It is applicable to labelled or unlabeled datasets. It is defined as:

$$Q = HTR \times e^{-\{\frac{\alpha \times N}{S} + \frac{\beta \times std.\ dev.\ EW}{mean\ EW} + \frac{\gamma \times std.\ dev.\ Imp}{Stim\ Win}\}}$$

… (Eq. 6)

… where HTR: Hit Trial Ratio

N/S: Estimated Noise/Signal

EW: Read out Event Widths (frames)

Imp: Estimated Imprecision (frames)

Stim Wind: Stimulus Window (frames)

α: 10

β: 1

γ: 10

While 10xα was required, β, and γ, were inspired by the same used for Reference Q. Classification was then performed using Bootstrapping (as described above) as well as Otsu's Threshold. This method can handle labelled and unlabeled data.

A schematic to describe the steps involved in each algorithm is shown (Fig. 3). We were then able to run all our synthetically generated datasets through each of the 8 implemented algorithms and perform comparative benchmarks.

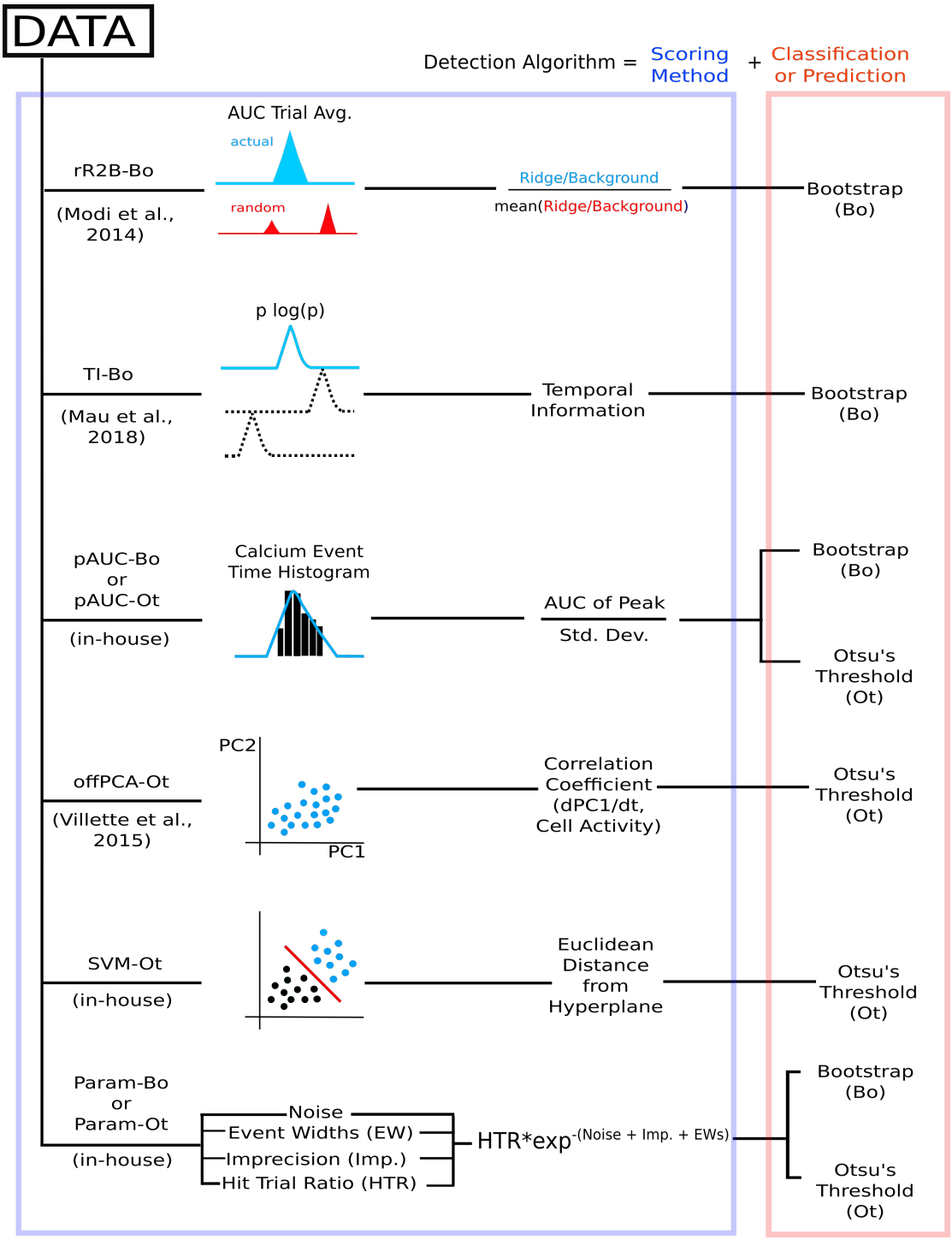

Fig. 3: Schematic representation of the implemented algorithms, involving 6 different scoring methods followed by a classification step (Bootstrapping or Otsu's automatic threshold) to have 8 complete time cell detection algorithms.

Good predictive power in time cell quality scores despite weak correlation

We ran each of the analysis methods on our synthetic datasets to assess how they scored the (known) time cells. Different methods yielded different scores, as described in histograms for the time cells and control cells, for each method (Fig.4A to 4F). In these synthetic experiments, time cells were generated with a single calcium event per Hit Trial. Event insertions into the synthetic datasets were subject to Noise, variable selection o f event widths,

trial pair Imprecision, and Hit Trial Ratio. We generated 99 unique un-physiological combinations (3x randomised shuffles) and 12 unique physiological regime combinations (10x randomised shuffles; total: 417 datasets, 135 cells, 60 trials/session, and 246 frames/trial at 14.5 Hz). Generalised Linear Regression (GLM) models were generated to look for the ideal thresholding value for the best classification predictions by each method. We obtained good predictive power for most methods in scoring for time cells by going over the full range of thresholds for the range of scores for each method (ROC Curves, Figure 4G). With offPCA, we found that the full list of time cells was not captured in any singular principal component but rather split across many principal components, for any dataset. This meant that the threshold values for high True Positive Rates (>0.5) could only be reached with concomitantly high False Positive Rates (Figure 4G). We also observed that there is at best weak pairwise correlation across all scores (Figure 4H). In summary, all the methods yielded good time-cell classification but there were notable differences in how they handled specific features of input variability.

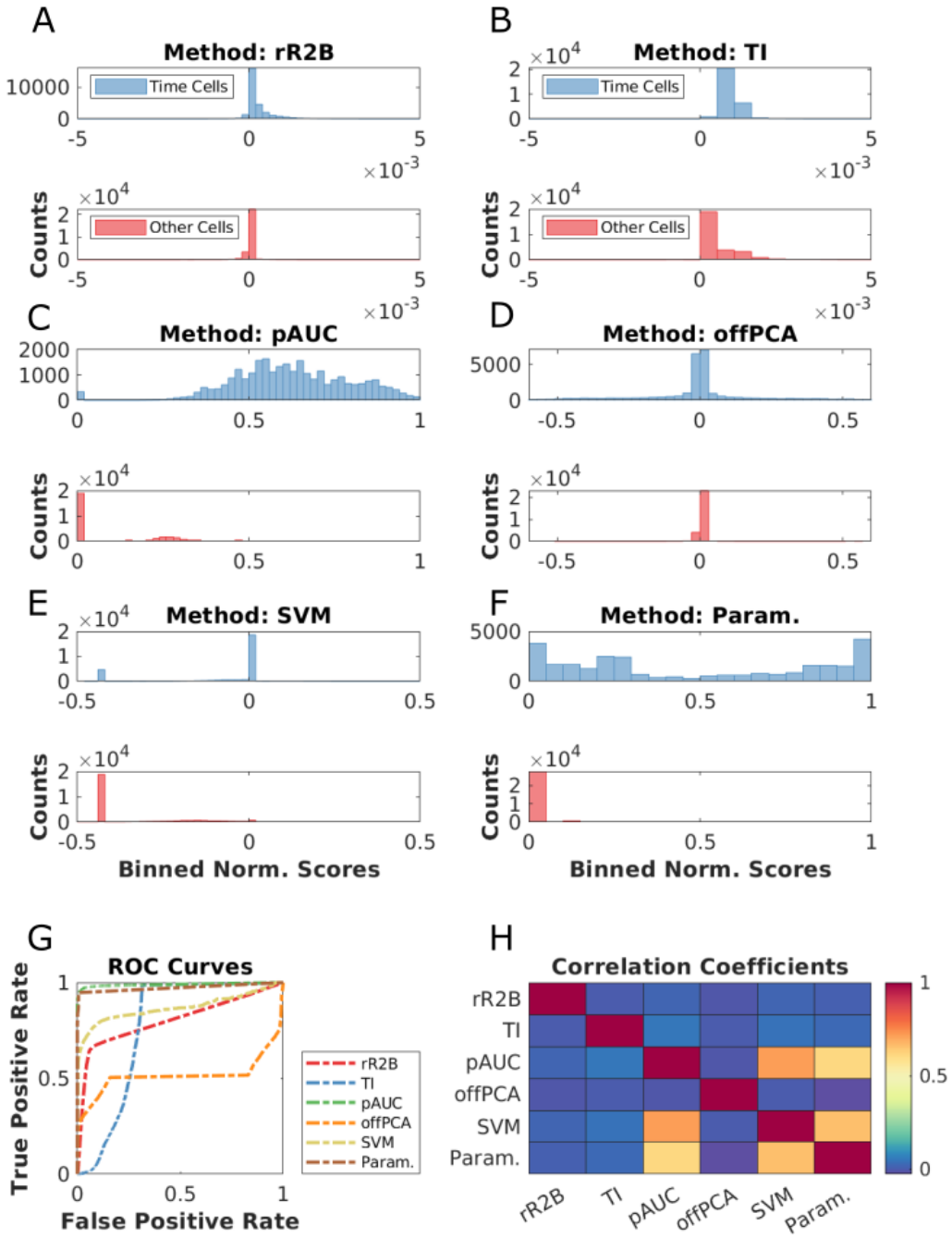

Fig. 4: Different distributions for the scores by each method, but with good predictive power. Scores for top (blue): time cells; bottom (red): other cells, across A: Method A, B: Method B, C: Method C, D: Method D, E: Method E, F: Method F. G: Generalised Linear Regression Models for all the method scores (ROC curves). H: Weak if any pairwise correlation between method scores.

All algorithms exhibit near perfect Precision with good Recall

Next, we used the scores to classify the cells in our synthetic datasets (Fig.3), compared the predictions to ground truth, and established summaries for True and False cases. Confusion matrices were estimated to compare the predictions (classifications) for each algorithm, with reference to ground truth,

521 and are shown (Fig. 5A to 5H). All methods exhibit very good Precision,
522 suggesting low False Positive Rates (Type I error, Fig. 5I). Most algorithms
523 also generate good values for Recall, except for Offset PCA (OffPCA-Ot) and
524 SVM (SVM-Ot).
525 With OffPCA-Ot, we had attempted to automate the selection of the Principal
526 Component (PC) after Offset PCA (Kaifosh et al., 2013), by choosing the PC
527 that explained the most variance in the data, viz., PC1. However, in our
528 analysis we found that PC1 typically only represented time cells with large
529 event widths. We inspected several (N >10) classifications, and found that no
530 one Principal Component captured all the activity from Time Cells. This meant
531 that a naive implementation of the Principal Component analysis was likely to
532 miss many time cells. Specifically (Fig. 5I), despite good Precision, we also
533 observed low Recall, which suggests a large number of False Negatives. With
534 SVM-Ot we again observed low Recall with a large number of False
535 Negatives. Also, Parametric Equations with Otsu's Threshold (Param-Ot)
536 seems to have a better Recall than the same with Bootstrapping (Param-Bo).
537 We observed F1 Scores greater than 0.75, all the way to 1 (perfect score), for
538 most of the algorithms, as shown (Fig. 5I), suggesting overall usability. We
539 also noticed that some algorithms were able to uniquely identify time cells that
540 were rejected by the other algorithms. Ratio of Ridge/Background with
541 Bootstrapping (rR2B-Bo), Temporal Information with Bootstrapping (TI-Bo),
542 and Peak AUC with Otsu's Threshold (pAUC-Ot) were more likely to catch
543 such time cells uniquely (Fig. 5J), but with some cost to Precision (Type I
544 error; Fig. 5I).
545
546 *Algorithms differ in memory use and speed*
547 Hardware and runtime requirements are a secondary, but practical concern
548 when designing analysis of large datasets, and are specially relevant for
549 experiment designs which need on-line analysis. We therefore looked at how
550 memory use and runtime scaled across the different methods. We have
551 already implemented some optimizations in the code, such as preallocation of
552 analysis variables to help memory management during analysis runs. This
553 leads to a large initial memory requirement. Importantly, this only rises in a
554 shallow sublinear manner for larger numbers of datasets. While ~200 MB of
555 workspace memory is required for the generation and analysis of 1 dataset
556 (135 cells) in a stepwise manner, this memory requirement only rises to ~500
557 MB for 10 datasets (1350 cells) (Fig. 5K).
558 We also analysed runtime, using a a 6 core AMD Ryzen 5 3600 with 32 GB
559 RAM (3200 MHz) running MATLAB R2021a on Ubuntu 20.04. Note, however,
560 that we have implemented all the time-cell algorithms in serial and these do
561 not use the additional cores. We found that most algorithms ran to completion
562 at a rate of ~1-10 min./dataset. The runtime scaling was roughly linear for all
563 methods (Fig. 5L). We were able to complete the synthesis and analysis of 10
564 datasets (all algorithms) within ~500 mins. or ~8 hrs. (Fig. 5L). The effect of
565 the dataset size (as the number of cells per dataset) was not explored in our
566 experiments. However, we note that the analysis algorithm works
567 independently for each cell. Thus in principle the analysis could be run in an
568 embarrassingly parallel manner and should scale well on multi-CPU
569 architectures.
570

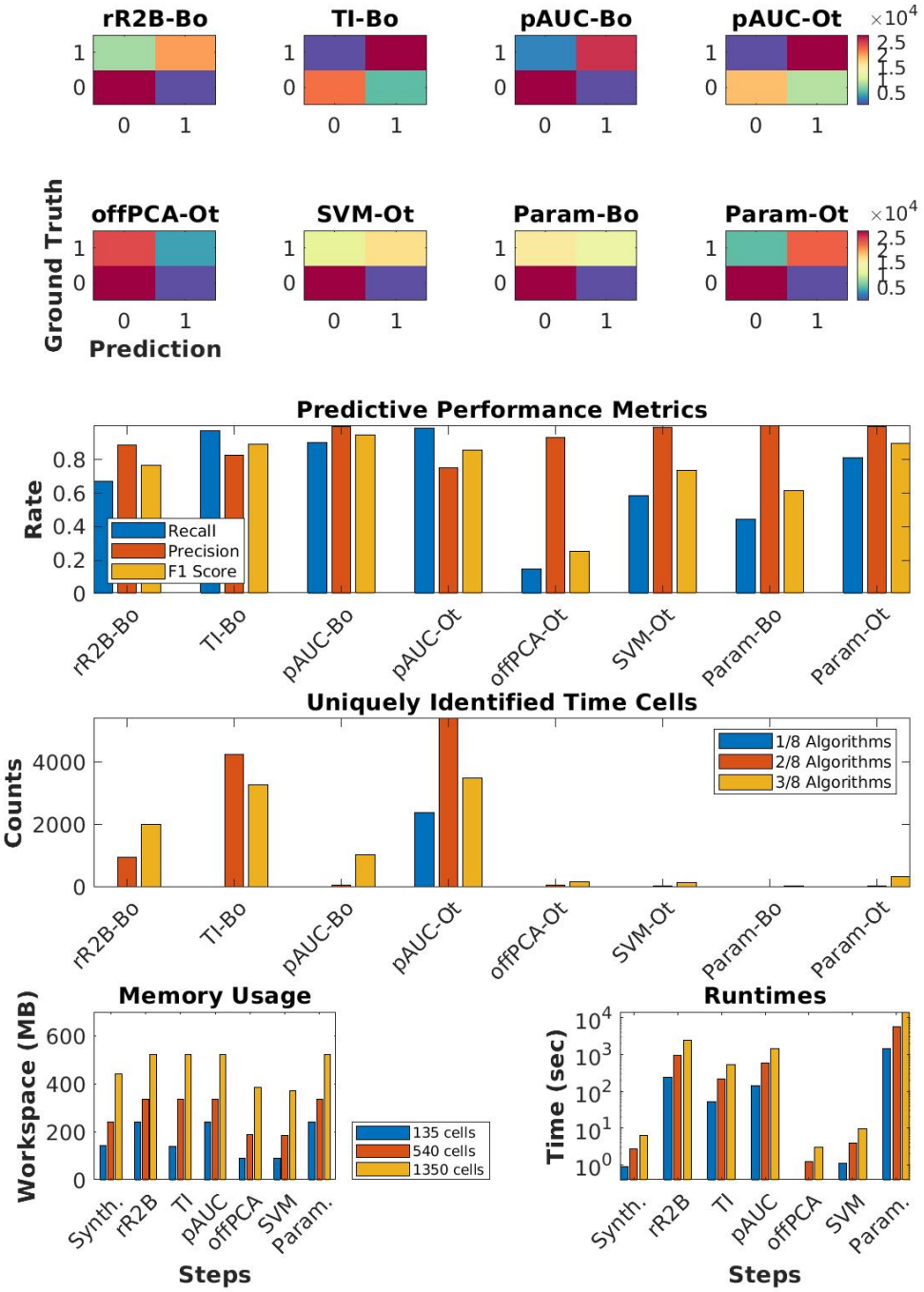

Fig. 5: Good predictive performance by all algorithms. Confusion Matrices were plotted to study classification performance in terms of False Negative (FN; top-left), True Positive (TP; top-right), True Negative (TN; bottom-left), and False Positive (FP; bottom-right) for all implemented algorithms, A: RR2B-Bo, B: TI-Bo, C: pAUC-Bo, D: pAUC-Ot, E: OffPCA-Ot, F: SVM-Ot, G: Param-Bo, H: Param-Ot, I: Predictive Performance Metrics exhibit good Precision across all algorithms, J: Uniquely identified Time Cells for 1, 2 and 3 Algorithms, K: Total workspace memory usage, and L: Runtimes, for 1 dataset (135 cells, blue), 4 datasets (540 cells, red), and 10 datasets (1350 cells, yellow)(bars: mean of 3 independent runs), for all steps in the full experiment loop.

Most methods are sensitive to Noise but not to other features of the dataset. Given our categorical labels on the synthetic data, we were able to split the datasets to look for the effects of the four main parameters, Noise, Event Widths, Imprecision, and Hit Trial Ratio. We first computed the baseline physiology readouts keeping Noise to 10%, Event Widths to the 60th percentile (+/- 1 std. dev.), Imprecision to 0 frames, and Hit Trial Ratios to a range of 33-66% for Time Cells (50%, N = 40 datasets, 135 cells, 60 trials). Next, we were able to establish dependence profiles for each of the algorithms, based on their predictions (N = 10 randomised shuffles for each case, Fig. 6B to 6E).

Most methods exhibited a negative dependence of Noise (range: 10% to 70%) on prediction F1 Score (Fig. 6B). Although many methods are designed with some form of denoising strategy (trial-averaging, PCA, etc.), as expected all algorithms ran into classification difficulties at higher Noise levels. This reinforces the value of relatively high signal to noise recordings.
We only observed Offset PCA (OffPCA-Ot), SVM (SVM-Ot), and Parametric Equations with Bootstrapping (Param-Bo) to perform better with larger Event Widths (range: 33rd to 99th percentile; Fig. 6C). None of the algorithms exhibit a clear trend with Normal/Gaussian Imprecision (range: 0-66 frame FWHM; Fig. 6D). The insensitivity to signal or event widths as well as Imprecision is potentially useful for Calcium Imaging datasets since a wider signal tends to describe wider detected tuning curves. However, this criterion may require stricter definitions to precisely identify fine differences in cell responses. We observed OffPCA-Ot to be biased to cells with large event widths, in our automated implementation, mandating manual intervention, as previously reported. While Peak AUC with Bootstrapping (pAUC-Bo) is clearly more sensitive to Imprecision, it is susceptible to noise, which may limit its applicability for certain datasets.
We also report that only SVM-Ot and Param-Bo exhibited a higher F1 score when the Hit Trial Ratio was increased (range: 33% to 99%; Fig. 6E). This suggests that most methods are insensitive to how frequently time cells are active across trials in a session. This is possibly the reason for a lack of such description in the time cell physiology in the literature. Our results suggest that SVM-Ot and Param-Bo may be the best options to delineate Hit Trial Ratio based distinctions amongst identified Time Cells.

We also looked to test the predictive power of a Concordance based algorithm, which considers time cells based on consensus amongst the predictions from all the eight implemented algorithms. We were also able to identify differences in the classification performance, across the full range of concordance thresholds. With lower threshold values (3 and below), we notice a slight drop in the Precision, indicating an increase in False Positive Rate (Type I error). On the other hand, with larger threshold values (5 and above), it is the Recall that drops, suggesting a higher False Negative Rate (Type II error). Overall, we find that a Concordance threshold of >=4 achieves the best Recall, Precision, and F1 Scores, for Time Cell prediction (Fig. 6F). The utility of this approach is subject to the availability of resources to apply multiple methods to each dataset.

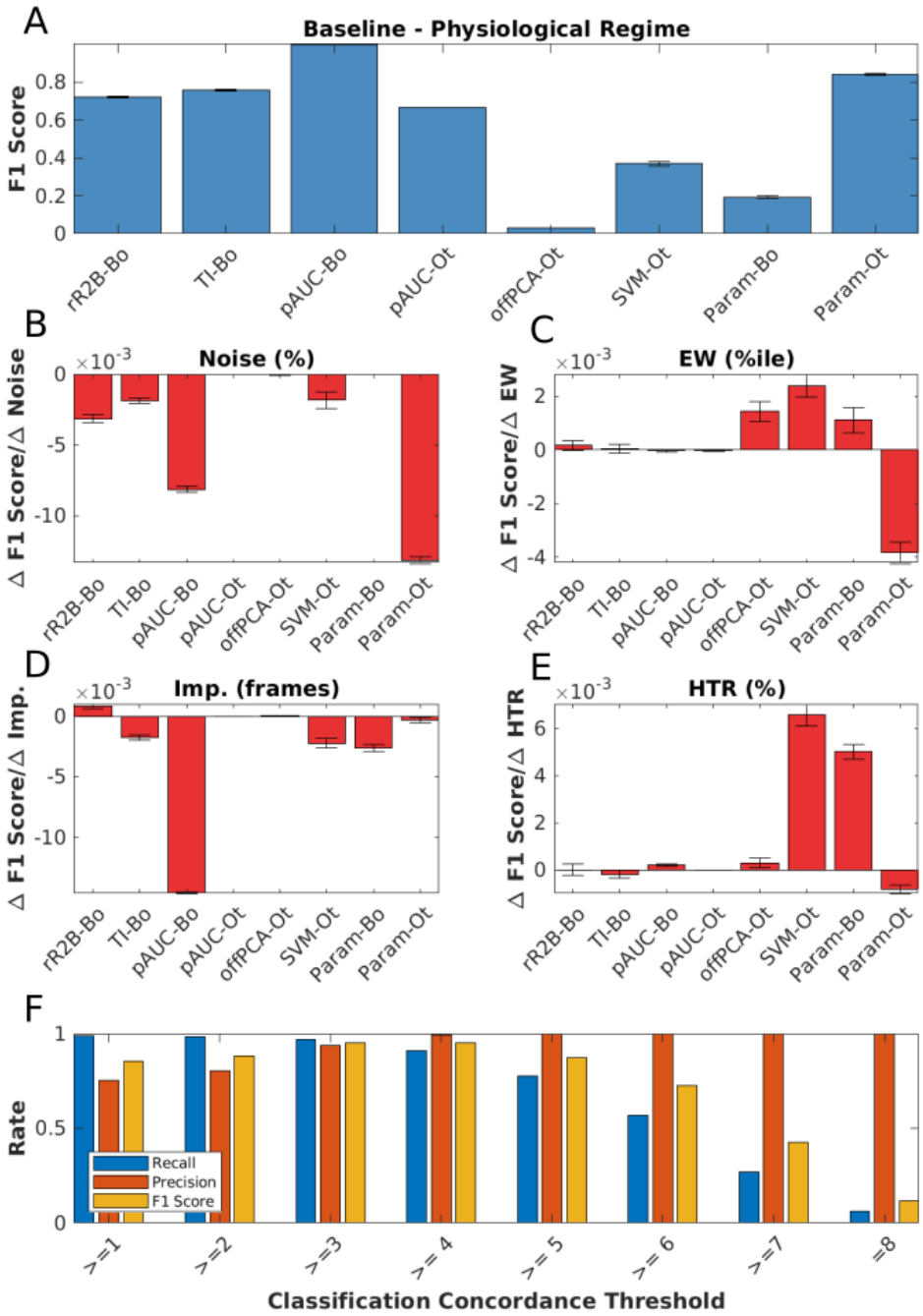

Fig. 6: Physiological Sensitivity Analysis and Concordance. A: Classification performance scores for all algorithms with the baseline Physiology synthetic datasets (N=40). Parameters were then systematically modulated with respect to baseline and all algorithms were studied for B: Dependence on Noise % (N=10), C: Dependence on Event Width percentiles (N=10), D: Dependence on Imprecision frames (N=10), E: Dependence on Hit Trial Ratio (HTR) % (N=10). Error Bars: Standard Error of the Mean. F: Classification performance using Concordance for a range of classification thresholds.

Discussion

We have developed a full pipeline for comparing time-cell detection algorithms. This starts with synthetic datasets for benchmarking, in which we program in the ground truth of cell identity and timed activity, and a range of experimental elaborations including noise, event widths, imprecision and hit trial ratio, to emphasise cell-wise independent tuning curves. This resource is, in itself, a key outcome of the current study, and though it is designed for 2P Ca imaging data it can be extended to rate-averaged single-unit recordings. We built a pipeline for running and comparing the outcome from six different methods applied to the same dataset. Most methods agreed on their time-cell classification, but accuracy and computational efficiency were particularly high for a couple of methods: Temporal Information (Mau et al., 2018) and pAUC-Bo. Out of these TI-Bo exhibited slightly more Type I errors (higher False Negatives), described by the slightly lower Precision values, as compared to pAUC-Bo.

The value of synthetic data in experimental science

Synthetic neural activity datasets are valuable in at least two main ways: evaluating algorithms for detection of important activity features, and for delivering stimuli to in-vitro and simulated neurons, so as to provide a more physiological context in which to study input-output properties (Abbasi, Maran, and Jaeger 2020). While we have deployed our synthetic dataset for the specific purpose of comparing time-cell detection algorithms, we suggest that it could also be useful for evaluating sequence analysis algorithms (Foster and Wilson 2006; Ikegaya et al. 2004; Villette et al. 2015). Beyond the domain of neuronal data analysis, such synthetic datasets act as a test-bed for critique and development of analysis algorithms meant for deployment on real-world or typical use case data. They have been used previously to benchmark unsupervised outlier detection (Steinbuss and Bohm 2020), explainable Machine Learning (Liu et al. 2021), Intrusion Detection Systems (Iannucci and Kholidy 2017), 3D Reconstruction Algorithms (Koch et al. 2021), among several others. We report the first use of synthetic data pertaining to cellular physiology in the context of identifying time cells from network recordings. Moreover, our experiments study important operational differences across several previously published and new detection algorithms. Our dataset may also be valuable for the second use case, stimulus delivery. There is a growing body of work on network detection of sequences (Csicsvari et al. 2007; Foster and Wilson 2006; Ikegaya et al. 2004; Jadhav et al. 2012; Malvache et al. 2016; Villette et al. 2015) or even single-neuron sequence selectivity (Bhalla 2017; Branco, Clark, and Häusser 2019). More realistic input activity patterns with a range of physiological perturbations may be useful probes for such experimental and theoretical studies. Further, experimenter-defined neural activity inputs through optogenetic stimulation has already begun to utilize more complex temporal patterns than static or periodic illumination (Bhatia, Moza, and Bhalla 2021; Dhawale et al. 2010; Schrader et al. 2008). Our approaches to synthetic sequential neuronal activity generation may be useful to add more physiological dimensions to the sequential activity employed in such studies.

Further dimensions of time-cell modulation

Our experiments allowed us to probe for parametric dependence systematically across published and new algorithms. We observed little or no sensitivity to Event Widths, Imprecision, and Hit Trial Ratios for most algorithms. Only SVM-Ot and Param-Bo were found to be somewhat dependent on Hit Trial Ratio and only pAUC-Bo was sensitive to Imprecision. On the one hand, this is a useful outcome in that different methods yield similar time-cell classification. It is a limitation, however, if the network uses such response features for coding, since it means that these methods are insensitive to relevant response changes. Further potential coding dimensions were not explored. Thus several potential behavioural correlates of tuned cells, an important direction for physiology (Ranck 1973), could not be studied in our experiments. Such correlates include but are not limited to measurements of spatial navigation (O'Keefe and Dostrovsky 1971; O'Keefe and Nadel 1978; Wilson and McNaughton 1993) and decision making (Csicsvari et al. 2007; Davidson et al. 2009; Diba and Buzsaki 2007; Foster and Wilson 2006; Gupta et al. 2010; Karlsson and Frank 2009; MacDonald et al. 2013; Villette et al. 2015), as well as navigation across tone frequencies (Aronov and Tank 2014). While each of these further inputs would be interesting to incorporate into synthetic datasets, this requires that the time-cell generation algorithm itself incorporate some form of simulation of the neural context. This is beyond the scope of the current study.

A specific limitation of our dataset is that it assumes that time is encoded by individual neurons. This leaves out population encoding schemes in which no one cell responds with the level of precision or consistency that would clear the criteria we use. For example, many of the same studies that utilise the methods tested here also use neural network decoders to report time (Mau et al. 2018). Such decoders might detect time encoding without time cells. A similar situation of individual vs. network coding appears for the closely related problem of sequence representation. Place cell replay sequences have been shown to be modulated by the prevalence of location specific aversive (Wu et al. 2017) as well as appetitive stimuli (Bhattarai, Won, and Whan 2019). Such physiological findings have been the subject of theoretical models of behaviour planning (Foster 2017; Mattar and Daw 2019), and have been reported to improve performance on multiple Atari games by artificial neural networks (Mnih et al. 2015) featuring salience detection and experience mapping. We suggest that synthetic data for such higher-order encoding schemes might be a useful tool, and could draw upon the approaches in the current study.

Comparative Analysis Benchmarks and Concordance

A particularly challenging time-cell classification problem is when the same cells may play different timing roles, such as forward and reverse replay. This is made more difficulty because of the relative rarity of forward replay sequences over the more typical reverse replay (Diba and Buzsaki 2007; Foster 2017). Pre-play is also a topic of some debate (Dragoi and Tonegawa 2013; Foster 2017). At least one possible problem in such debates is the degree of consistency between time-cell or sequence classifiers. Our pipeline allows for i) error correction in case of non-concordant classifications, ii) suggest candidate algorithms with a dependence on dataset features like Event Widths, Imprecision, and Hit Trial Ratio, as well as iii) the possibility to

expand the detection regime in more realistic physiological datasets using Concordance.

Which algorithms to use?

We did not set out to rank algorithms, but our analysis does yield suggestions for possible use domains. In cases where runtime and compute resource use is a concern, we recommend using the Temporal Information method with Bootstrap (TI-Bo). However, a unique set of time cells may be identified by pAUC-Ot (although with some loss to Precision; increased Type I error). This second analysis may be useful where there are rare and potentially multimodally tuned time cells (Pastalkova et al. 2008; Villette et al. 2015). Ultimately, all algorithms have very good Precision, and classify with very few false positives (low Type I error). Many methods are susceptible to classification errors if the dataset is noisy, and the pAUC-Ot in particular should be avoided in such cases..

OffPCA-Ot is currently best handled with manual selection of Principal Components, and is as yet automatable only to a limited extent, but we report a benefit to classification with larger Event Widths. The F1 Score for SVM-Ot benefits from increased Hit Trial Ratios, although most algorithms perform well even at a low range of values (16-33%). In real physiology recordings it may be more important to be selective for Hit Trial Ratio due to the occurrence of background calcium events.

Sequence detection in large-scale recordings and scaling of analysis runs.

The discovery of replay over the past two decades, has benefitted from the technological advances made in increasing the cellular yield of network recordings and has been reviewed previously (Foster 2017). Further advances such as with the large scale recordings of ~10^3 by electrical recording using Neuropixels (Jun et al. 2017), fast volumetric fluorescence scanning with upto ~10^4 cells using Resonant Electro-Optic Imaging (Bowman and Kasevich 2021; Pachitariu et al. 2017; Poort et al. 2015), ~10^3 Mesoscopes (Sofroniew et al. 2016), as well as advances in automated cell ROI detection, denoising, and neuropill subtraction (Pnevmatikakis et al., 2014; Pachitariu et al., 2016) only increase the scale and size of datasets, likely leading to longer analysis runtimes. In addition to our recommendations above for the Temporal Information/Boot method for scalable time-cell analysis, our implementations may also be useful in further optimising these methods. Our implementations allow for relatively fast analysis of the same datasets with multiple algorithms.

Acknowledgements

KGA and the experiments received support from Department of Biotechnology, BT/PR12255/MED/122/8/2016, to USB. NCBS-TIFR provided resources for analysis equipment and support to USB through intramural funds supported by the Department of Atomic Energy, Government of India, under project identification No. RTI 4006.

We thank Daniel Dombeck for help with the chronic preparation for physiological 2P calcium recording of Hippocampal CA1 cells, and

William Mau for his Temporal Information calculation code. We thank Vinu Varghese and Bhanu Priya Somashekar for helpful comments on the manuscript.

Code and Resource Availability

All codes for Synthetic Dataset generation along with Time Cell Analysis may be found at https://github.com/BhallaLab/rho-matlab.

Bibliography

Abbasi, Samira, Selva Maran, and Dieter Jaeger. 2020. "A General Method to Generate Artificial Spike Train Populations Matching Recorded Neurons." *Journal of Computational Neuroscience* 47–63.

Aronov, Dmitriy and David W. Tank. 2014. "Engagement of Neural Circuits Underlying 2D Spatial Navigation in a Rodent Virtual Reality System." *Neuron* 84(2).

Bhalla, Upinder Singh. 2017. "Synaptic Input Sequence Discrimination on Behavioral Timescales Mediated by Reaction-Diffusion Chemistry in Dendrites." *ELife* 1–24.

Bhatia, Aanchal, Sahil Moza, and Upinder S. Bhalla. 2021. "Patterned Optogenetic Stimulation Using a DMD Projector." *Methods in Molecular Biology (Clifton, N.J.)* 2191:173–88.

Bhattarai, Baburam, Jong Won, and Min Whan. 2019. "Distinct Effects of Reward and Navigation History on Hippocampal Forward and Reverse Replays." *PNAS*.

Bowman, Adam J. and Mark A. Kasevich. 2021. "Resonant Electro-Optic Imaging for Microscopy at Nanosecond Resolution." *ArXiv*.

Branco, Tiago, Beverley A. Clark, and Michael Häusser. 2019. "Dendritic Discrimination of Temporal Input Sequences in Cortical Neurons." 329(5999):1671–75.

Csicsvari, Jozsef, Joseph O. Neill, Kevin Allen, and Timothy Senior. 2007. "Place-Selective Firing Contributes to the Reverse-Order Reactivation of CA1 Pyramidal Cells during Sharp Waves in Open-Field Exploration." *European Journal of Neuroscience* 26:704–16.

Davidson, Thomas J., Fabian Kloosterman, Matthew A. Wilson, and Cognitive Sciences. 2009. "Hippocampal Replay of Extended Experience." *Neuron* 63(4):497–507.

Dhawale, Ashesh K., Akari Hagiwara, Upinder S. Bhalla, Venkatesh N. Murthy, and Dinu F. Albeanu. 2010. "Non-Redundant Odor Coding by Sister Mitral Cells Revealed by Light Addressable Glomeruli in the Mouse." *Nat Neurosci* 13(11).

Diba, Kamran and Gyorgy Buzsaki. 2007. "Forward and Reverse Hippocampal Place-Cell Sequences during Ripples." *Nat Neurosci* 10(10):1241–42.

Dombeck, Daniel A., Christopher D. Harvey, Lin Tian, Loren L. Looger, and David W. Tank. 2010. "Functional Imaging of Hippocampal Place Cells at Cellular Resolution during Virtual Navigation." *Nature Neuroscience* 13(11):1433–40.

Dragoi, George and Susumu Tonegawa. 2013. "Distinct Preplay of Multiple Novel

Spatial Experiences in the Rat." *PNAS* 110(22):9100–9105.
Eichenbaum, Howard. 2017. "On the Integration of Space, Time, and Memory." *Neuron* 95(5):1007–18.
Foster, David J. 2017. "Replay Comes of Age." *Annu. Rev. Neurosci.*
Foster, David J. and Matthew A. Wilson. 2006. "Reverse Replay of Behavioural Sequences in Hippocampal Place Cells during the Awake State." *Nature* 440(March):1–4.
Gupta, Anoopum S., Matthijs A. A. Van Der Meer, David S. Touretzky, and A. David Redish. 2010. "Article Hippocampal Replay Is Not a Simple Function of Experience." *Neuron* 65(5):695–705.
Iannucci, Stefano and Hisham A. Kholidy. 2017. "A Comparison of Graph-Based Synthetic Data Generators for Benchmarking Next-Generation Intrusion Detection Systems."
Ikegaya, Yuji, Gloster Aaron, Rosa Cossart, Dmitriy Aronov, Ilan Lampl, David Ferster, and Rafael Yuste. 2004. "Synfire Chains and Cortical Songs: Temporal Modules of Cortical Activity." *Science* 304(5670):559–64.
Jadhav, Shantanu P., Caleb Kemere, P. Walter German, and Loren M. Frank. 2012. "Awake Hippocampal Sharp-Wave Ripples Support Spatial Memory." *Science* 336(6087):1454–58.
Jun, James J., Nicholas A. Steinmetz, Joshua H. Siegle, Daniel J. Denman, Marius Bauza, Brian Barbarits, Albert K. Lee, Costas A. Anastassiou, Alexandru Andrei, Çağatay Aydın, Mladen Barbic, Timothy J. Blanche, Vincent Bonin, João Couto, Barundeb Dutta, Sergey L. Gratiy, Diego A. Gutnisky, Michael Häusser, Bill Karsh, Peter Ledochowitsch, Carolina Mora Lopez, Catalin Mitelut, Silke Musa, Michael Okun, Marius Pachitariu, Dylan Rich, Cyrille Rossant, Wei-lung Sun, Karel Svoboda, Matteo Carandini, Kenneth D. Harris, Jan Putzeys, Christof Koch, John O. Keefe, and Timothy D. Harris. 2017. "Fully Integrated Silicon Probes for High-Density Recording of Neural Activity." *Nature* 551(7679):232–36.
Kaifosh, Patrick, Matthew Lovett-Barron, Gergely F. Turi, Thomas R. Reardon, and Attila Losonczy. 2013. "Septo-Hippocampal GABAergic Signaling across Multiple Modalities in Awake Mice." *Nat Neurosci* 16(9):1182–84.
Karlsson, Mattias P. and Loren M. Frank. 2009. "Awake Replay of Remote Experiences in the Hippocampus." *Nat Neurosci* 12(7).
Koch, Sebastian, Markus Worchel, Claudio Silva, and Marc Alexa. 2021. "Hardware Design and Accurate Simulation of Structured-Light Scanning for Benchmarking of 3D Reconstruction Algorithms." 2021(NeurIPS 2021).
Kraus, BenjaminJ, RobertJ Robinson, JohnA White, Howard Eichenbaum, and MichaelE Hasselmo. 2013. "Hippocampal 'Time Cells': Time versus Path Integration." *Neuron* 78(6):1090–1101.
Lee, Albert K. and Matthew A. Wilson. 2004. "A Combinatorial Method for Analyzing Sequential Firing Patterns Involving an Arbitrary Number of Neurons Based on Relative Time Order." *J Neurophysiol* 2555–73.
Liu, Yang, Sujay Khandagale, Colin White, and Willie Neiswanger. 2021. "Synthetic Benchmarks for Scientific Research in Explainable Machine Learning." (NeurIPS):1–14.
MacDonald, C. J., S. Carrow, R. Place, and H. Eichenbaum. 2013. "Distinct Hippocampal Time Cell Sequences Represent Odor Memories in Immobilized Rats." *Journal of Neuroscience* 33(36):14607–16.

MacDonald, Christopher J., Kyle Q. Lepage, Uri T. Eden, and Howard Eichenbaum. 2011. "Hippocampal 'Time Cells' Bridge the Gap in Memory for Discontiguous Events." *Neuron* 71(4):737–49.

Malvache, Arnaud, Susanne Reichinnek, Vincent Villette, and Caroline Haimerl. 2016. "Awake Hippocampal Reactivations Project onto Orthogonal Neuronal Assemblies." *Science* 353(6305).

Mattar, Marcelo G. and Nathaniel D. Daw. 2019. "Prioritized Memory Access Explains Planning and Hippocampal Replay." *Nat Neurosci* 21(11):1609–17.

Mau, William, David W. Sullivan, Nathaniel R. Kinsky, Michael E. Hasselmo, Marc W. Howard, and Howard Eichenbaum. 2018. "The Same Hippocampal CA1 Population Simultaneously Codes Temporal Information over Multiple Timescales." *Current Biology* 28(10):1499-1508.e4.

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. 2015. "Human-Level Control through Deep Reinforcement Learning." *Nature*.

Modi, M. N., a. K. Dhawale, and U. S. Bhalla. 2014. "CA1 Cell Activity Sequences Emerge after Reorganization of Network Correlation Structure during Associative Learning." *ELife* 3:e01982–e01982.

Mokeichev, Alik, Michael Okun, Omri Barak, Yonatan Katz, Ohad Ben-shahar, and Ilan Lampl. 2007. "Stochastic Emergence of Repeating Cortical Motifs in Spontaneous Membrane Potential Fluctuations In Vivo." *Neuron* 413–25.

O'Keefe, J. and J. Dostrovsky. 1971. "The Hippocampus as a Spatial Map. Preliminary Evidence from Unit Activity in the Freely-Moving Rat." *Brain Research* 34(1):171–75.

O'Keefe, John and Lynn Nadel. 1978. *The Hippocampus as a Cognitive Map*.

Pachitariu, Marius, Carsen Stringer, Mario Dipoppa, Sylvia Schröder, and L. Federico Rossi. 2017. "Suite2p : Beyond 10 , 000 Neurons with Standard Two-Photon Microscopy." *BioRxiv* 1–30.

Pastalkova, E., V. Itskov, A. Amarasingham, and G. Buzsaki. 2008. "Internally Generated Cell Assembly Sequences in the Rat Hippocampus." *Science* 321(5894):1322–27.

Poort, Jasper, Adil G. Khan, Marius Pachitariu, Abdellatif Nemri, Ivana Orsolic, Julija Krupic, Marius Bauza, Maneesh Sahani, Georg B. Keller, Thomas D. Mrsic-Flogel, and Sonja B. Hofer. 2015. "Learning Enhances Sensory and Multiple Non-Sensory Representations in Primary Visual Cortex." *Neuron* 86(6):1478–90.

Schrader, Sven, Sonja Grün, Markus Diesmann, and George L. Gerstein. 2008. "Detecting Synfire Chain Activity Using Massively Parallel Spike Train Recording." *Journal of Neurophysiology* 100(4):2165–76.

Sofroniew, Nicholas James, Daniel Flickinger, Jonathan King, and Karel Svoboda. 2016. "A Large Field of View Two-Photon Mesoscope with Subcellular Resolution for in Vivo Imaging." *ELife* 1–20.

Steinbuss, Georg and Klemens Bohm. 2020. "Generating Artificial Outliers in the Absence of Genuine Ones — a Survey ArXiv : 2006 . 03646v1 [Cs . LG] 5 Jun 2020."

Tseng, W., R. Guan, J. F. Disterhoft, and Craig Weiss. 2004. "Trace Eyeblink

Conditioning Is Hippocampally Dependent in Mice." *Hippocampus* 14(1):58–65.

Villette, Vincent, Arnaud Malvache, Thomas Tressard, Nathalie Dupuy, and Rosa Cossart. 2015. "Internally Recurring Hippocampal Sequences as a Population Template of Spatiotemporal Information." *Neuron* 88(2):357–66.

Wilson, M. A. and B. L. McNaughton. 1993. "Dynamics of the Hippocampal Ensemble Code for Space." *Science (New York, N.Y.)* 261(5124):1055–58.

Wu, Chun-ting, Daniel Haggerty, Caleb Kemere, and Daoyun Ji. 2017. "Hippocampal Awake Replay in Fear Memory Retrieval." *Nat Neurosci* 20(4):571–80.