TRAKR - A RESERVOIR-BASED TOOL FOR FAST AND ACCURATE CLASSIFICATION OF NEURAL TIME-SERIES PATTERNS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neuroscience has seen a dramatic increase in the types of recording modalities and complexity of neural time-series data collected from them. The brain is a highly recurrent system producing rich, complex dynamics that result in different behaviors. Correctly distinguishing such nonlinear neural time series in real-time, especially those with non-obvious links to behavior, could be useful for a wide variety of applications. These include detecting anomalous clinical events such as seizures in epilepsy, and identifying optimal control spaces for brain machine interfaces. It remains challenging to correctly distinguish nonlinear time-series patterns because of the high intrinsic dimensionality of such data, making accurate inference of state changes (for intervention or control) difficult. Simple distance metrics, which can be computed quickly do not yield accurate classifications. On the other end of the spectrum of classification methods, ensembles of classifiers or deep supervised tools offer higher accuracy but are slow, data-intensive, and computationally expensive. We introduce a reservoir-based tool, state tracker (TRAKR), which offers the high accuracy of ensembles or deep supervised methods while preserving the computational benefits of simple distance metrics. After one-shot training, TRAKR can accurately, and in real time, detect deviations in test patterns. By forcing the weighted dynamics of the reservoir to fit a desired pattern directly, we avoid many rounds of expensive optimization. Then, keeping the output weights frozen, we use the error signal generated by the reservoir in response to a particular test pattern as a classification boundary. We show that, using this approach, TRAKR accurately detects changes in synthetic time series. We then compare our tool to several others, showing that it achieves highest classification performance on a benchmark dataset–sequential MNIST–even when corrupted by noise. Additionally, we apply TRAKR to electrocorticography (ECoG) data from the macaque orbitofrontal cortex (OFC), a higher-order brain region involved in encoding the value of expected outcomes. We show that TRAKR can classify different behaviorally relevant epochs in the neural time series more accurately and efficiently than conventional approaches. Therefore, TRAKR can be used as a fast and accurate tool to distinguish patterns in complex nonlinear time-series data, such as neural recordings.

1 INTRODUCTION

The size and complexity of neural data collected has increased greatly (Marblestone et al. (2013)). Neural data display rich dynamics in the firing patterns of neurons across time, resulting from the recurrently connected circuitry in the brain. As our insight into these dynamics increases through new recording modalities, so does the desire to understand how dynamical patterns change across time and, ultimately, give rise to different behaviors.

A lot of work in computational neuroscience over the past decade has focused on modeling the collective dynamics of a population of neurons in order to gain insight into how firing patterns are related to task variables (Márton et al. (2020); Richards et al. (2019); Yang et al. (2018); Remington et al. (2018); Kell et al. (2018); Zeng et al. (2018); Pandarinath et al. (2018); Durstewitz (2017); Chaisangmongkon et al. (2017); Rajan et al. (2016); Sussillo et al. (2015); Mante et al. (2013); Sus-

sillo & Barak (2013); Barak et al. (2013); Sussillo & Abbott (2009)). These approaches, however, rely on fitting the whole dynamical system through many rounds of optimization, either indirectly by modeling the task inputs and outputs (Márton et al. (2020); Kell et al. (2018); Chaisangmongkon et al. (2017); Sussillo et al. (2015); Mante et al. (2013); Sussillo & Barak (2013), or directly by fitting the weights of a neural network to recorded firing patterns (Pandarinath et al. (2018); Durstewitz (2017)). Thus, these approaches can be too time- and computation-intensive for certain applications, e.g. in clinical settings where decisions need to be taken based on recordings in real-time. In these settings, neural time-series patterns need to be accurately distinguished in order to, say, detect the onset of seizures, or distinguish different mental states.

Previous approaches to classifying time series lie on a spectrum from simple distance metrics (e.g., Euclidean) to more computationally intensive approaches such as dynamic time warping (Xing et al. (2010)), ensembles of classifiers (Bagnall et al.) or deep supervised learning (Jeong (2020); Fawaz et al. (2019)). Computing simple distance metrics is fast and straightforward, but does not always yield high accuracy results because the patterns may not be perfectly aligned in time. On the other end of the spectrum, ensembles of classifiers and deep learning-based approaches (Bagnall et al.; Jeong (2020); Fawaz et al. (2019)) have been developed that can offer high accuracy results, but at high computational cost. Dynamic time warping (DTW) has been consistently found to offer good results in practice relative to computational cost (Fawaz et al. (2019); Bagnall et al. (2016); Serrà & Arcos (2014)) and is currently routinely used to measure the similarity of time-series patterns.

Previous work in reservoir computing has shown that networks of neurons can be used as reservoirs of useful dynamics, so called echo-state networks (ESNs), without the need to train recurrent weights through successive rounds of expensive optimization (Vlachas et al. (2020); Pathak et al. (2018); Vincent-Lamarre et al. (2016); Buonomano & Maass (2009); Jaeger & Haas (2004); Jaeger (a;b); Maass et al. (2002)). This suggests reservoir networks could offer a computationally cheaper alternative to deep supervised approaches in the classification of neural time-series data. However, the training of reservoir networks has been found to be more unstable compared to methods that also adjust the recurrent connections (e.g., via backpropagation through time, BPTT) in the case of reduced-order data (Vlachas et al. (2020)). Even though ESNs have been shown to yield good results when fine-tuned (Tanisaro & Heidemann (2016); Aswolinskiy et al. (2016)), convergence represents a significant problem when training ESNs end-to-end to perform classification on complex time-series datasets, and is a hurdle to their wider adoption.

Here, we propose fitting the reservoir output weights to a single time series - thus avoiding many rounds of training that increase training time and could potentially cause instabilities. We use the error generated through the output unit in response to a particular test pattern as input to a classifier. We show that using this approach, we obtain high accuracy results on a benchmark dataset - sequential MNIST - outperforming other approaches such as simple distance metrics (e.g., based on Euclidean distance) and more compute-heavy approaches such as DTW and model-based approaches (e.g., naive Bayes). Importantly, while yielding high accuracy results, even when the data are corrupted by noise, our approach is computationally less time-intensive than DTW.

We also apply our tool, TRAKR, to neural data from the macaque orbitofrontal cortex (OFC), a higher-order brain region involved in encoding expectations, and inducing changes in behavior during unexpected outcomes (Rich & Wallis (2016); Rudebeck & Rich (2018); Jones et al. (2012); Wallis (2012); Schoenbaum (2009); Burke et al. (2009); Wallis & Miller (2003); Schoenbaum et al. (1998)). 128-channel electrocorticography (micro-ECOG) recordings were obtained from the macaque OFC, including anterior and posterior areas 11 and 13, during a reward expectation task. The task was designed to understand how expectations encoded in OFC are updated by unexpected outcomes. We show that TRAKR can be used to distinguish three different behaviorally relevant epochs based on the neural time-series with higher accuracy than conventional approaches.

Taken together, we show that TRAKR is a useful tool for the fast and accurate classification of time-series data. It can be applied to distinguish complex patterns in high-dimensional neural recordings.

2 METHODS

2.1 MODEL DETAILS

TRAKR (Figure 1A) is a reservoir-based recurrent neural network (RNN) with N recurrently connected neurons. Recurrent weights, J, are initialized randomly and remain aplastic over time (Buonomano & Maass (2009); Jaeger (b); Maass et al. (2002)). The readout unit, z_{out}, is connected to the reservoir through a set of output weights, w_{out}, which are plastic and are adjusted during training. The reservoir also receives an input signal, $I(t)$, through an aplastic set of weights w_{in}.

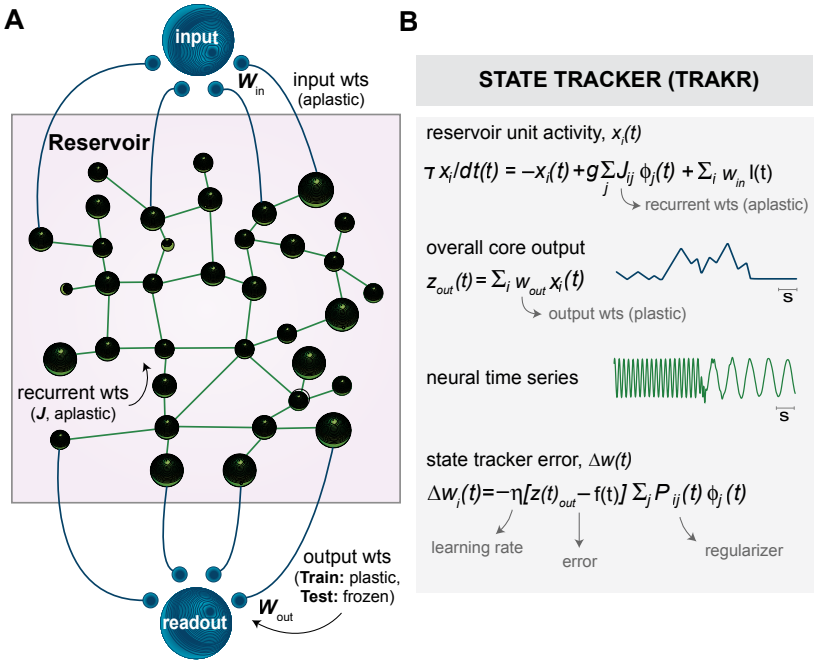

Figure 1: A) TRAKR setup overview. TRAKR consist of a reservoir connected to input and readout units via dedicated weights. Recurrent weights J and input weights w_{in} are aplastic. Only the output weights w_{out} are subject to training. B) TRAKR equations for single unit activity, readout unit activity and error term.

The network is governed by the following equations:

$$\tau \frac{dx_i(t)}{dt} = -x_i(t) + g \sum_{j=1}^{N} J_{ij}\phi_j(t) + w_{i,in} I(t) \tag{1}$$

$$z_{out}(t) = \sum_i w_{out,i}(t) x_i(t) \tag{2}$$

Here, $x_i(t)$ is the activity of a single neuron in the reservoir, τ is the integration time constant, g is the gain setting the scale for the recurrent weights, and J is the recurrent weight matrix of the reservoir. The term $g \sum_{j=1}^{N} J_{ij}\phi_j(t)$ denotes the strength of input to a particular neuron from other neurons in the reservoir and $I(t)$ is the input signal (Equation 1). $z_{out}(t)$ denotes the activity of the readout unit together with the output weights, w_{out} (Equation 2). In our notation, w_{ij} denote the weight from neuron j to i, and so $w_{out,i}$ means the weight from i^{th} unit in the reservoir to the readout unit. ϕ is the activation function given by:

$$\phi_i(t) = tanh(x_i(t)) \tag{3}$$

We use recursive least squares (RLS) to adjust the output weights, w_{out} during training (Haykin, Simon S. (1996)). The algorithm and the update rules are given by:

$$\Delta w_{out,i}(t) = -\eta(z_{out}(t) - f(t)) \sum_j P_{ij}(t)\phi_j(t) \tag{4}$$

$$w_{out,i}(t) = w_{out,i}(t-1) + \Delta w_{out,i}(t) \tag{5}$$

Here, η is the learning rate, $f(t)$ is the target function, and the term $\sum_j P_{ij}(t)\phi_j(t)$ acts as a regularizer where P is the inverse cross-correlation matrix of the network firing rates. For details on setting hyperparameters, see Appendix A.

2.2 Adjusting reservoir dynamics

During training, the output weights, w_{out}, are optimized using RLS based on the instantaneous difference between the output, $z_{out}(t)$, and the target function, $f(t)$. This optimization is performed in one shot (without the need for multiple optimization rounds). Here, we use the reservoir to autoencode the input signal, thus $f(t) = I(t)$. The instantaneous difference gives rise to an error term, $E(t)$, calculated as:

$$E(t) = \sum_{i=1}^{N} \Delta w_{out,i}(t) \tag{6}$$

2.3 Obtaining the error signal

After training, the output weights, w_{out} are frozen. The test pattern is fed to the network via the input, $I(t)$, and the network is iterated to obtain the error, $E(t)$ over the duration of the test signal. The error, $E(t)$ is computed as the difference between the test signal and the network output (Equation 6). The error may vary depending on the similarity of a given test signal to the learned time series. The error is used as input to a classifier.

2.4 Classification of the error signal

The error, $E(t)$, is used as input to a support vector machine (SVM) classifier with a Gaussian radial basis function (rbf) kernel. The classifier is trained using leave-one-out cross-validation. The same classifier and training procedure was used in comparing the different approaches. Accuracy and area under the curve (AUC) are computed as a measure of classification performance.

2.5 Neural Recordings

2.5.1 Task Design

Neural recordings were obtained from the macaque OFC using a custom designed 128-channel micro-ECOG array (NeuroNexus), with coverage including anterior and posterior subregions (areas 11/13). During preliminary training, the monkey learned to associate unique stimuli (natural images) with rewards of different values. Rewards were small volumes of sucrose or quinine solutions, and values were manipulated by varying their respective concentrations.

The behavioral task design is shown in (Figure 4A). During the task, the monkey initiated a trial by contacting a touch-sensitive bar and holding gaze on a central location. On each trial, either one or two images were presented, and the monkey selected one by shifting gaze to it and releasing the bar. At this point, a small amount of fluid was delivered, and then a neutral cue appeared (identical across all trials) indicating the start of a 5s response period where the macaque could touch the bar to briefly activate the fluid pump. By generating repeated responses, it could collect as much of the available reward as desired. There were two types of these trials. Match (mismatch) trials were those where the initial image accurately (did not accurately) signal the type of reward delivered on that trial. Behavioral performance and neural time series were recorded in 11 task sessions across 35 days. Each trial was approximately 6.5s long, including different behaviorally relevant epochs and cues.

The macaque performed approximately 550 trials within each task session ($mean \pm sd : 562 \pm 72$). Of note, 80% of the trials were match trials within each task session.

2.5.2 DATA PRE-PROCESSING

ECoG data were acquired by a neural processing system (Ripple) at $30kHz$ and then resampled at $1kHz$. The 128-channel data were first z-score normalized. Second-order butterworth bandstop IIR filters were used to remove 60Hz line noise and harmonics from the signal. We also used second-order Savitzky-Golay filters of window length 99 to smooth the data and remove very high frequency juice pump artifacts ($> 150Hz$). For most of the analysis here, we used the average of the 128-channel time series as an input to TRAKR.

3 RESULTS

3.1 DETECTING PATTERN CHANGES IN SYNTHETIC TIME SERIES

First, we trained TRAKR on idealized, synthetic signals using sin-functions of two different frequencies (Figure 2). Reservoir output weights were fitted to the signal using recursive least squares (RLS; see subsection 2.1). In Figure 2A, the network was trained on the first half of the signal (blue) while the output weights, w_{out}, remained frozen during the second half. Then with w_{out} frozen, a test signal (yellow) was fed to the reservoir. The network output, $z_{out}(t)$, in red and the error signal, $E(t)$, in green are depicted in Figure 2. The network correctly detects the deviation of the test pattern (yellow, 1st half of the signal) from the learned pattern (blue, 1st half of the signal), which results in an increase in the error signal (green, 1st half of the signal, Figure 2A). The second half of the test signal (yellow) aligns with the trained signal (blue, 1st half) and thus yields no error (green, 2nd half). In Figure 2B, the order of the training procedure was reversed in that output weights remained frozen for the first half of the signal (blue) and were plastic during the second half. As expected, the increase in the error signal (green) now occurs during the second half of the test signal (yellow). Thus, TRAKR correctly detects, via the error signal $E(t)$, when a new frequency pattern occurs in the test signal that deviates from the trained pattern.

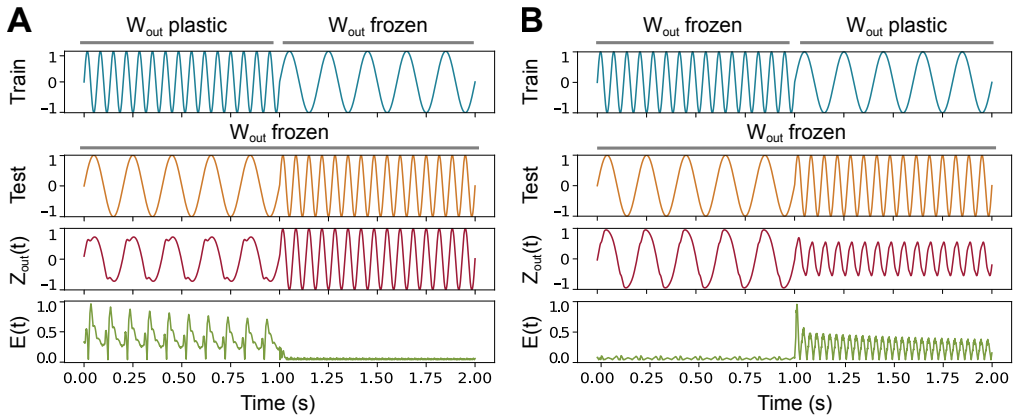

Figure 2: A) (Blue) w_{out} plastic for a 15 Hz sin-function, and frozen for a 5 Hz rhythm. (Yellow) Test pattern with the same frequencies but the signal order reversed. (Red) TRAKR output. (Green) The error signal, $E(t)$, is showing an increase for the part of the test pattern which was not learned during training. B) Similar to A but w_{out} were plastic during the second half of the training signal (5Hz rhythm).

3.2 CLASSIFYING DIGITS - SEQUENTIAL MNIST

We applied TRAKR to the problem of classifying the ten digits from sequential MNIST, a benchmark dataset for time-series problems (Le et al. (2015); Kerg et al. (2019)).

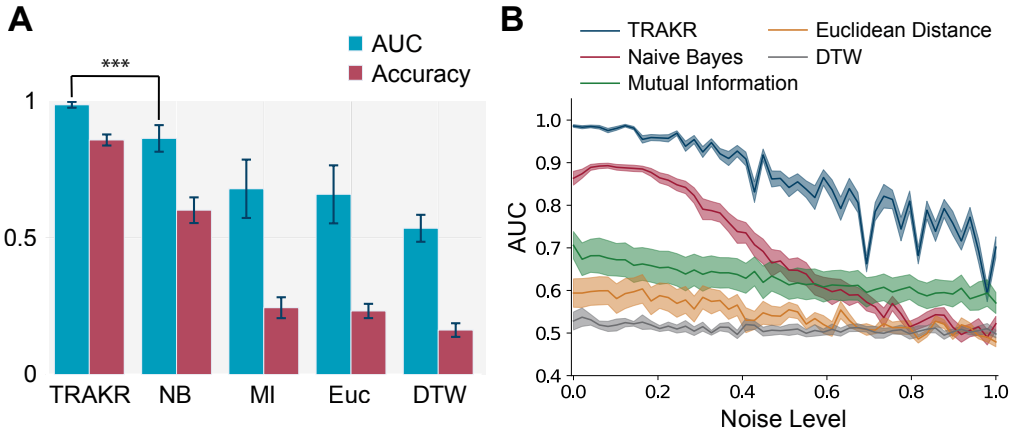

Figure 3: Classification performance on the sequential MNIST dataset. A) TRAKR outperforms all other methods ($99\% AUC$; $***: p < 0.001$, Bonferroni-corrected). NB: naive Bayes; MI: mutual information; Euc: Euclidean distance; DTW: dynamic time warping. B) Classification performance with increasing amount of noise added to the digits. TRAKR performance declines smoothly with noise level, while still outperforming other approaches in classifying noisy digits. Chance level is at 10%.

For training, we curated a dataset of 1000 sequential MNIST digits including 100 samples for each digit (0-9). We fed each sequential digit (28 x 28 pixel image flattened into a vector of length 784) as a one-shot training signal to TRAKR. Reservoir output weights were again fitted to the signal using recursive least squares (RLS; see subsection 2.1). After fitting TRAKR to one time series corresponding to one of the samples of a particular digit, we froze the output weights and fed all the other digits as test samples to TRAKR. We obtained an error signal, $E(t)$, from every test sample, with the magnitude of the error varying depending on the similarity with the learned digit. The error signal was then fed into a classifier which was trained to differentiate the digits based on the error terms (see subsection 2.4 for more details). We repeated this procedure for all digits and samples in the dataset to obtain the averaged classification performance for TRAKR (Figure 3A).

We obtained a high performance for TRAKR, with an $AUC = 99\%$ (Figure 3A, leftmost entry). We compared the performance of our approach against other common ways of classifying time series, again using the same classifier as before (subsection 2.4). We compared our results against other distance metrics (Euclidean distance, DTW, mutual information) and a generative model (naive Bayes). For DTW, an implementation of the FastDTW algorithm was used (Salvador & Chan (2007)). All other approaches performed significantly worse than TRAKR ($p < 0.001$), with naive Bayes derived metric performing the best among the compared approaches ($AUC = 86\%$).

We also tested the performance of TRAKR under different noise conditions (Figure 3B). For this purpose, we added random independent Gaussian noise to the training digits with $\mu = 0$ and varying standard deviation (σ). The actual noise that was added (noise levels as depicted in Figure 3B) can be calculated as $\sigma * 255$, with $\sigma \in [0, 1]$. The number 255 represents the maximal pixel value in the sequential digits. We again compared against all the other approaches, as above. We found TRAKR to perform the best, also at high noise levels: performance decays gradually as the noise is increased and is at $AUC = 70\%$ even at higher noise levels ($\sigma = 1$).

We also measured the time it takes to obtain the error signal using TRAKR (Table 1). While it does require upfront fitting, our approach has the advantage that it does not require multiple rounds of optimization because the signal is fit in one shot (see subsection 2.2 for details). After fitting, TRAKR can detect deviations from the learned signal in real-time. While TRAKR is not as fast as obtaining a simple distance metric, we found that it performs relatively faster than DTW, a commonly used approach to differentiate time-series signals (Table 1). We actually found DTW to yield the lowest accuracy (Figure 3A) of all the approaches. Deep supervised approaches and ensemble methods are computationally even more intensive than DTW (Fawaz et al. (2019)). Altogether, this

Table 1: Computational cost compared

Method	Time taken to classify test pattern
Dynamic time warping	188 ms ± 11.1 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
TRAKR	32.8 ms ± 1.52 ms per loop (mean ± std. dev. of 7 runs, 10 loops each)
Naive Bayes	2.14 ms ± 119 µs per loop (mean ± std. dev. of 7 runs, 100 loops each)
Mutual information	114 µs ± 711 ns per loop (mean ± std. dev. of 7 runs, 10000 loops each)
Euclidean distance	7.19 µs ± 403 ns per loop (mean ± std. dev. of 7 runs, 100000 loops each)

shows TRAKR yields good performance at relatively high computational speed, which is beneficial for real-time applications.

3.3 Performance on Neural Time Series Recorded from the Macaque OFC

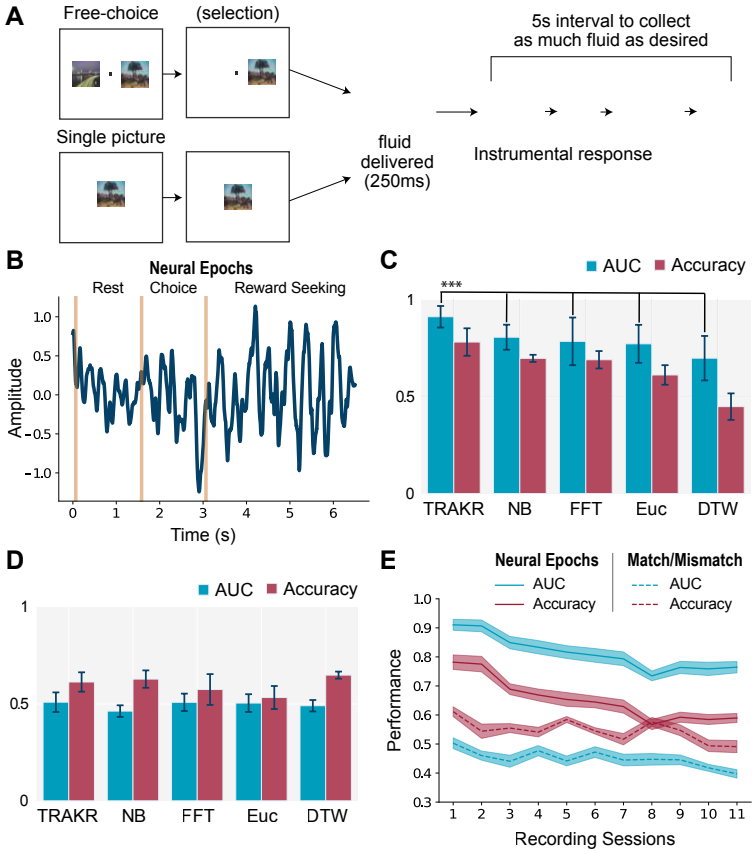

Figure 4: A) Neural task design (see subsubsection 2.5.1 for detailed description). B) Example neural time series from a single trial, with three behaviorally relevant epochs (rest, choice and instrumental reward seeking period). C) TRAKR outperforms all other methods in classifying the 3 neural epochs ($***: p < 0.001$, Bonferroni-corrected; chance-level at 33%) NB: naive Bayes; FFT: fast Fourier transform; Euc: Euclidean distance; DTW: Dynamic time warping. AUC in blue, $Accuracy$ in red. D) TRAKR and other methods show a chance performance ($50\% AUC$) in classifying the neural time-series patterns as belonging to match/mismatch trials. E) Classification performance (TRAKR) in distinguishing neural epochs decreases over 11 recording sessions (35 days).

The OFC is involved in encoding and updating affective expectations. We used a behavioral task designed to study the neural mechanisms of how such expectations are encoded in the OFC of primates and how they may guide behavior under different conditions. The goal here was to determine whether TRAKR could classify behaviorally relevant epochs from neural data, and whether it could further distinguish different task conditions (Figure 4A; see also subsubsection 2.5.1 for more details).

A sample of the different neural epochs is shown for a single trial from a particular recording electrode in Figure 4B. The three neural epochs are behaviorally meaningful in that they correspond to rest, choice and instrumental reward seeking. We used TRAKR to classify the neural time series recorded from different trials into these three epochs (see section 2 for more details).

We trained TRAKR on the neural time series corresponding to rest from a particular trial, and used the other complete trials as test signals to obtain the error, $E(t)$ as before. The error signal was used as input to a classifier. We repeated this procedure for all trials in the dataset to obtain the averaged classification performance. We also compared against other conventional approaches, as before. In addition, we also calculated the Fast Fourier transform (FFT) of the signals and obtained the magnitude (power) in the α $(0 - 12Hz)$, β $(13 - 35Hz)$, and γ $(36 - 80Hz)$ bands within the 3 epochs. We found that TRAKR outperformed all the other methods (Figure 4C), accurately classifying the neural time-series patterns as belonging to either rest, choice, or instrumental reward period ($AUC = 91\%$; $p < 0.001$).

Additionally, we determined whether TRAKR was able to distinguish the neural time-series patterns as belonging to either match or mismatch trials (as described in further detail in section 2). For this purpose, we trained TRAKR on the neural time series corresponding to choice period from a particular trial, and used the other complete trials as test signals to obtain the error, $E(t)$ as before. TRAKR, along with all the other methods, was not able to accurately classify the neural time-series patterns as belonging to either match or mismatch trials (Figure 4D). Further investigations of signal from individual electrodes or in specific frequency bands may be needed to detect such trial-wise differences.

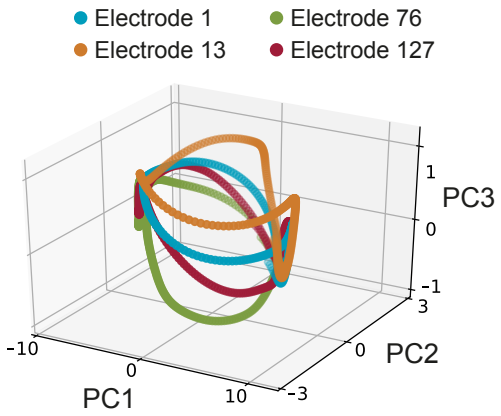

Figure 5: Single electrode recordings projected into the space spanned by the first three principal components of reservoir activations. The four electrodes trace out different trajectories in reservoir space, suggesting they capture potentially different neural dynamics.

We then used TRAKR to measure the classification performance over recording sessions (Figure 4E), for both classifying behaviorally relevant epochs in the neural signal (Figure 4C) and for classifying trials into either match or mismatch (Figure 4D). We found that classification performance for the behaviorally relevant epochs degrades over days (Figure 4E; blue & red solid lines), while that for match/mismatch trials consistently performs around chance-level (Figure 4E; blue & red dotted lines).

Lastly, we wanted to see if the activations of the units in the reservoir could be used to re-group the electrodes (128-channel recordings) into functionally meaningful groups. For this purpose, we

fitted the reservoir to the time series obtained from a particular electrode, froze the output weights, and used the signal from the other electrodes as test inputs to obtain the error terms. In order to visualize the recordings from the different electrodes in the reservoir space, we performed principal component analysis (PCA) on the tensor of reservoir activities obtained from all the test electrodes. We then projected the signal from every electrode onto the first three principal components of the reservoir space in order to examine if electrodes traced out similar trajectories in this space. Figure 5 shows the result of visualizing four different electrodes in this reservoir space. The four electrodes trace out different paths. Thus, in principle, TRAKR can be used to cluster the neural time series obtained from different electrodes into functionally meaningful groupings, which may represent coherent regions in the brain or interdigitated modules within single regions.

4 Discussion

We have shown that TRAKR can accurately detect deviations from learned signals. TRAKR outperforms other approaches in classifying time-series data on a benchmark dataset, sequential MNIST, and on differentiating behaviorally meaningful epochs in neural data from macaque OFC.

While TRAKR could accurately classify neural epochs, it could not classify neural time-series patterns into either match or mismatch trials. It is possible that receiving a better or worse reward than expected affects the neural signal in distinct/opposite ways, such that the effect is cancelled out on average. It is also possible that the difference in neural time-series patterns is only discernible if the reward is maximally different (better or worse) than expected. In the current task design, there were 4 different levels of reward (flavors) that the macaque associated with different pictures (sub-subsection 2.5.1). The number of trials in which the obtained reward was maximally different from the expected was low and possibly not sufficient for accurate classification. Another possibility, corroborated by several studies (Stalnaker et al. (2018); McDannald et al. (2014); Takahashi et al. (2013); Kennerley et al. (2011)), is that OFC neural activity may signal reward *values* but not reward *prediction errors*, which instead are mediated through the ventral tegmental area (VTA) in the midbrain.

We found that the classification performance decreased over recording sessions. This could mean that the difference between task epochs being classified decreased because of increased familiarity with the task. That is less likely, however, because the subject was well-trained prior to recordings. Instead, since the signal was recorded over a period of 35 days, the decrease in the classification performance could be a result of degrading signal quality, perhaps due to electrode impedance issues (Kozai et al. (2015a;b); Holson et al. (1998); Robinson & Camp (1991)).

TRAKR offers high classification accuracy at relatively low computational cost, outperforming a commonly used approach such as dynamic time warping (DTW). While ensemble methods and deep supervised approaches may yield high accuracy, they are more time-intensive than DTW (Fawaz et al. (2019)). In particular, deep learning-based approaches, with a high number of parameters to tune, come with high upfront computational cost during training. TRAKR avoids expensive rounds of successive optimization during training by allowing only output weights to change and by fitting a given time series directly using recursive least squares. Moreover, avoiding the need of training on many samples, the error signal can be used directly to distinguish patterns in real-time. This suggests TRAKR can be particularly useful for many real-time applications where available training time is restricted and fast classification performance is desired when deployed.

5 Conclusion

There is a need for and renewed interest in tools for the analysis of time-series data (Bhatnagar et al. (2021)). We show that TRAKR is a fast and accurate tool for the classification of time-series patterns. It is suitable for real-time applications where fast classification of time-series patterns is needed, such as in clinical settings. TRAKR is particularly suited for differentiating complex nonlinear signals, such as those obtained from neural or behavioral data in neuroscience, which can shed light on how complex neural dynamics are related to behavior.

REFERENCES

Witali Aswolinskiy, René Felix Reinhart, and Jochen Steil. Time Series Classification in Reservoir- and Model-Space: A Comparison. In Friedhelm Schwenker, Hazem M. Abbas, Neamat El Gayar, and Edmondo Trentin (eds.), *Artificial Neural Networks in Pattern Recognition*, volume 9896, pp. 197–208. Springer International Publishing, Cham, 2016. ISBN 978-3-319-46181-6 978-3-319-46182-3. doi: 10.1007/978-3-319-46182-3_17. URL http://link.springer.com/10.1007/978-3-319-46182-3_17. Series Title: Lecture Notes in Computer Science.

Anthony Bagnall, Jason Lines, Jon Hills, and Aaron Bostrom. Time-Series Classification with COTE: The Collective of Transformation-Based Ensembles. pp. 2.

Anthony Bagnall, Aaron Bostrom, James Large, and Jason Lines. The Great Time Series Classification Bake Off: An Experimental Evaluation of Recently Proposed Algorithms. Extended Version. *arXiv:1602.01711 [cs]*, February 2016. URL http://arxiv.org/abs/1602.01711. arXiv: 1602.01711.

Omri Barak, David Sussillo, Ranulfo Romo, Misha Tsodyks, and L F Abbott. From fixed points to chaos: Three models of delayed discrimination. *Progress in Neurobiology*, 103:214–222, March 2013. doi: 10.1016/j.pneurobio.2013.02.002. URL http://dx.doi.org/10.1016/j.pneurobio.2013.02.002. Publisher: Elsevier Ltd.

Aadyot Bhatnagar, Paul Kassianik, Chenghao Liu, Tian Lan, Wenzhuo Yang, Rowan Cassius, Doyen Sahoo, Devansh Arpit, Sri Subramanian, Gerald Woo, Amrita Saha, Arun Kumar Jagota, Gokulakrishnan Gopalakrishnan, Manpreet Singh, K. C. Krithika, Sukumar Maddineni, Daeki Cho, Bo Zong, Yingbo Zhou, Caiming Xiong, Silvio Savarese, Steven Hoi, and Huan Wang. Merlion: A Machine Learning Library for Time Series. *arXiv:2109.09265 [cs, stat]*, September 2021. URL http://arxiv.org/abs/2109.09265. arXiv: 2109.09265.

Dean V Buonomano and Wolfgang Maass. State-dependent computations: spatiotemporal processing in cortical networks. *Nature Reviews Neuroscience*, 10(2):113–125, January 2009. doi: 10.1038/nrn2558. URL http://www.nature.com/articles/nrn2558. Publisher: Nature Publishing Group.

Kathryn A Burke, Theresa M Franz, Danielle N Miller, and Geoffrey Schoenbaum. The role of the orbitofrontal cortex in the pursuit of happiness and more specific rewards. pp. 10, 2009.

Warasinee Chaisangmongkon, Sruthi K Swaminathan, David J Freedman, and Xiao-Jing Wang. Computing by Robust Transience: How the Fronto- Parietal Network Performs Sequential, Category- Based Decisions. *Neuron*, 93(6):1504–1517.e4, March 2017. doi: 10.1016/j.neuron.2017.03.002. URL http://dx.doi.org/10.1016/j.neuron.2017.03.002. Publisher: Elsevier Inc.

Daniel Durstewitz. A state space approach for piecewise-linear recurrent neural networks for identifying computational dynamics from neural measurements. *PLOS Computational Biology*, 13(6): e1005542–33, June 2017. doi: 10.1371/journal.pcbi.1005542. URL http://dx.plos.org/10.1371/journal.pcbi.1005542. Publisher: Public Library of Science.

Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre Alain Muller. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4), 2019. ISSN 1573756X. doi: 10.1007/s10618-019-00619-1.

Haykin, Simon S. *Adaptive Filter Theory*. Prentice Hall, 3rd edition, 1996. ISBN 978-0-13-004052-7.

R.Robert Holson, Russell A Gazzara, and Bobby Gough. Declines in stimulated striatal dopamine release over the first 32 h following microdialysis probe insertion: generalization across releasing mechanisms. *Brain Research*, 808(2):182–189, October 1998. ISSN 00068993. doi: 10.1016/S0006-8993(98)00816-6. URL https://linkinghub.elsevier.com/retrieve/pii/S0006899398008166.

Herbert Jaeger. Adaptive Nonlinear System Identification with Echo State Networks. pp. 8, a.

Herbert Jaeger. The "echo state" approach to analysing and training recurrent neural networks – with an Erratum note. pp. 48, b.

Herbert Jaeger and Harald Haas. Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication. 304:3, 2004.

Taikyeong Jeong. Time-Series Data Classification and Analysis Associated With Machine Learning Algorithms for Cognitive Perception and Phenomenon. *IEEE Access*, 8:222417–222428, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3018477. URL https://ieeexplore.ieee.org/document/9173667/.

Joshua L Jones, Guillem R Esber, Michael A McDannald, Aaron J Gruber, Alex Hernandez, Aaron Mirenzi, and Geoffrey Schoenbaum. Orbitofrontal Cortex Supports Behavior and Learning Using Inferred But Not Cached Values. 338:5, 2012.

Alexander J.E. Kell, Daniel L.K. Yamins, Erica N. Shook, Sam V. Norman-Haignere, and Josh H. McDermott. A Task-Optimized Neural Network Replicates Human Auditory Behavior, Predicts Brain Responses, and Reveals a Cortical Processing Hierarchy. *Neuron*, 98(3):630–644.e16, May 2018. ISSN 08966273. doi: 10.1016/j.neuron.2018.03.044. URL https://linkinghub.elsevier.com/retrieve/pii/S0896627318302502.

Steven W Kennerley, Timothy E J Behrens, and Jonathan D Wallis. Double dissociation of value computations in orbitofrontal and anterior cingulate neurons. *Nat Neurosci*, 14(12):1581–1589, December 2011. ISSN 1097-6256, 1546-1726. doi: 10.1038/nn.2961. URL http://www.nature.com/articles/nn.2961.

Giancarlo Kerg, Kyle Goyette, Maximilian Puelma Touzel, Gauthier Gidel, Eugene Vorontsov, Yoshua Bengio, and Guillaume Lajoie. Non-normal Recurrent Neural Network (nnRNN): learning long time dependencies while improving expressivity with transient dynamics. *NeurIPS*, pp. 11, 2019.

Takashi D. Y. Kozai, Andrea S. Jaquins-Gerstl, Alberto L. Vazquez, Adrian C. Michael, and X. Tracy Cui. Brain Tissue Responses to Neural Implants Impact Signal Sensitivity and Intervention Strategies. *ACS Chem. Neurosci.*, 6(1):48–67, January 2015a. ISSN 1948-7193, 1948-7193. doi: 10.1021/cn500256e. URL https://pubs.acs.org/doi/10.1021/cn500256e.

Takashi D.Y. Kozai, Zhanhong Du, Zhannetta V. Gugel, Matthew A. Smith, Steven M. Chase, Lance M. Bodily, Ellen M. Caparosa, Robert M. Friedlander, and X. Tracy Cui. Comprehensive chronic laminar single-unit, multi-unit, and local field potential recording performance with planar single shank electrode arrays. *Journal of Neuroscience Methods*, 242:15–40, March 2015b. ISSN 01650270. doi: 10.1016/j.jneumeth.2014.12.010. URL https://linkinghub.elsevier.com/retrieve/pii/S0165027014004312.

Quoc V. Le, Navdeep Jaitly, and Geoffrey E. Hinton. A Simple Way to Initialize Recurrent Networks of Rectified Linear Units. *arXiv:1504.00941 [cs]*, April 2015. URL http://arxiv.org/abs/1504.00941. arXiv: 1504.00941.

Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Computation*, 14(11):2531–2560, November 2002. ISSN 0899-7667, 1530-888X. doi: 10.1162/089976602760407955. URL https://direct.mit.edu/neco/article/14/11/2531-2560/6650.

Valerio Mante, David Sussillo, Krishna V Shenoy, and William T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, November 2013. ISSN 0028-0836. doi: 10.1038/nature12742. URL http://www.nature.com/doifinder/10.1038/nature12742.

Adam H. Marblestone, Bradley M. Zamft, Yael G. Maguire, Mikhail G. Shapiro, Thaddeus R. Cybulski, Joshua I. Glaser, Dario Amodei, P. Benjamin Stranges, Reza Kalhor, David A. Dalrymple, Dongjin Seo, Elad Alon, Michel M. Maharbiz, Jose M. Carmena, Jan M. Rabaey, Edward S. Boyden, George M. Church, and Konrad P. Kording. Physical principles for scalable neural recording. *Frontiers in Computational Neuroscience*, (OCT), 2013. ISSN 16625188. doi: 10.3389/fncom.2013.00137.

Michael A McDannald, Guillem R Esber, Meredyth A Wegener, Heather M Wied, Tzu-Lan Liu, Thomas A Stalnaker, Joshua L Jones, Jason Trageser, and Geoffrey Schoenbaum. Orbitofrontal neurons acquire responses to 'valueless' Pavlovian cues during unblocking. *eLife*, 3:e02653, July 2014. ISSN 2050-084X. doi: 10.7554/eLife.02653. URL https://elifesciences.org/articles/02653.

Christian D. Márton, Simon R. Schultz, and Bruno B. Averbeck. Learning to select actions shapes recurrent dynamics in the corticostriatal system. *Neural Networks*, 132:375–393, December 2020. ISSN 08936080. doi: 10.1016/j.neunet.2020.09.008. URL https://linkinghub.elsevier.com/retrieve/pii/S0893608020303312.

Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, L. F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nat Methods*, 15(10):805–815, October 2018. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-018-0109-9. URL http://www.nature.com/articles/s41592-018-0109-9.

Jaideep Pathak, Brian Hunt, Michelle Girvan, Zhixin Lu, and Edward Ott. Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach. *Phys. Rev. Lett.*, 120(2):024102, January 2018. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.120.024102. URL https://link.aps.org/doi/10.1103/PhysRevLett.120.024102.

Kanaka Rajan, Christopher D Harvey, and David W Tank. Recurrent Network Models of Sequence Generation and Memory. *Neuron*, 90(1):128–142, April 2016. doi: 10.1016/j.neuron.2016.02.009. URL http://dx.doi.org/10.1016/j.neuron.2016.02.009. Publisher: Elsevier Inc.

Evan D. Remington, Seth W. Egger, Devika Narain, Jing Wang, and Mehrdad Jazayeri. A Dynamical Systems Perspective on Flexible Motor Timing. *Trends in Cognitive Sciences*, 22(10):938–952, October 2018. ISSN 13646613. doi: 10.1016/j.tics.2018.07.010. URL https://linkinghub.elsevier.com/retrieve/pii/S1364661318301724.

Erin L Rich and Jonathan D Wallis. Decoding subjective decisions from orbitofrontal cortex. pp. 27, 2016.

Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy Berker, Surya Ganguli, Colleen J Gillon, Danijar Hafner, Adam Kepecs, Nikolaus Kriegeskorte, Peter Latham, Grace W Lindsay, Kenneth D Miller, Richard Naud, Christopher C Pack, Panayiota Poirazi, Pieter Roelfsema, João Sacramento, Andrew Saxe, Benjamin Scellier, Anna C Schapiro, Walter Senn, Greg Wayne, Daniel Yamins, Friedemann Zenke, Joel Zylberberg, Denis Therien, and Konrad P Kording. A deep learning framework for neuroscience. *Nature Neuroscience*, 22(11):1–10, October 2019. doi: 10.1038/s41593-019-0520-2. URL http://dx.doi.org/10.1038/s41593-019-0520-2. Publisher: Springer US.

Terry E. Robinson and Dianne M. Camp. The effects of four days of continuous striatal microdialysis on indices of dopamine and serotonin neurotransmission in rats. *Journal of Neuroscience Methods*, 40(2-3):211–222, December 1991. ISSN 01650270. doi: 10.1016/0165-0270(91)90070-G. URL https://linkinghub.elsevier.com/retrieve/pii/016502709190070G.

Peter H. Rudebeck and Erin L. Rich. Orbitofrontal cortex. *Current Biology*, 28(18):R1083–R1088, September 2018. ISSN 09609822. doi: 10.1016/j.cub.2018.07.018. URL https://linkinghub.elsevier.com/retrieve/pii/S0960982218309175.

Stan Salvador and Philip Chan. FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space. pp. 11, 2007.

Geoffrey Schoenbaum. A new perspective on the role of the orbitofrontal cortex in adaptive behaviour. pp. 8, 2009.

Geoffrey Schoenbaum, Andrea A Chiba, and Michela Gallagher. Orbitofrontal cortex and basolateral amygdala encode expected outcomes during learning. *nature neuroscience*, 1(2):5, 1998.

Joan Serrà and Josep Lluis Arcos. An Empirical Evaluation of Similarity Measures for Time Series Classification. *Knowledge-Based Systems*, 67:305–314, September 2014. ISSN 09507051. doi: 10.1016/j.knosys.2014.04.035. URL http://arxiv.org/abs/1401.3973. arXiv: 1401.3973.

Thomas A. Stalnaker, Tzu-Lan Liu, Yuji K. Takahashi, and Geoffrey Schoenbaum. Orbitofrontal neurons signal reward predictions, not reward prediction errors. *Neurobiology of Learning and Memory*, 153:137–143, September 2018. ISSN 10747427. doi: 10.1016/j.nlm.2018.01.013. URL https://linkinghub.elsevier.com/retrieve/pii/S1074742718300133.

David Sussillo and L F Abbott. Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, 63(4):544–557, August 2009. doi: 10.1016/j.neuron.2009.07.018. URL http://dx.doi.org/10.1016/j.neuron.2009.07.018. Publisher: Elsevier Ltd.

David Sussillo and Omri Barak. Opening the Black Box: Low-Dimensional Dynamics in High-Dimensional Recurrent Neural Networks. *Neural Computation*, pp. 1–24, January 2013. doi: https://doi.org/10.1162/NECO_a_00409. URL https://www.mitpressjournals.org/doi/pdf/10.1162/NECO_a_00409.

David Sussillo, Mark M Churchland, Matthew T Kaufman, and Krishna V Shenoy. A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*, 18(7):1025–1033, June 2015. doi: 10.1038/nn.4042. URL http://www.nature.com/articles/nn.4042. Publisher: Nature Publishing Group.

Yuji K. Takahashi, Chun Yun Chang, Federica Lucantonio, Richard Z. Haney, Benjamin A. Berg, Hau-Jie Yau, Antonello Bonci, and Geoffrey Schoenbaum. Neural Estimates of Imagined Outcomes in the Orbitofrontal Cortex Drive Behavior and Learning. *Neuron*, 80(2):507–518, October 2013. ISSN 08966273. doi: 10.1016/j.neuron.2013.08.008. URL https://linkinghub.elsevier.com/retrieve/pii/S0896627313007198.

Pattreeya Tanisaro and Gunther Heidemann. Time Series Classification Using Time Warping Invariant Echo State Networks. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 831–836, Anaheim, CA, USA, December 2016. IEEE. ISBN 978-1-5090-6167-9. doi: 10.1109/ICMLA.2016.0149. URL http://ieeexplore.ieee.org/document/7838253/.

Philippe Vincent-Lamarre, Guillaume Lajoie, and Jean-Philippe Thivierge. Driving reservoir models with oscillations: a solution to the extreme structural sensitivity of chaotic networks. *J Comput Neurosci*, 41(3):305–322, December 2016. ISSN 0929-5313, 1573-6873. doi: 10.1007/s10827-016-0619-3. URL http://link.springer.com/10.1007/s10827-016-0619-3.

P.R. Vlachas, J. Pathak, B.R. Hunt, T.P. Sapsis, M. Girvan, E. Ott, and P. Koumoutsakos. Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics. *Neural Networks*, 126:191–217, June 2020. ISSN 08936080. doi: 10.1016/j.neunet.2020.02.016. URL https://linkinghub.elsevier.com/retrieve/pii/S0893608020300708.

Jonathan D Wallis. Cross-species studies of orbitofrontal cortex and value-based decision-making. *nature neuroscience*, 15(1):7, 2012.

Jonathan D Wallis and Earl K Miller. Neuronal activity in primate dorsolateral and orbital prefrontal cortex during performance of a reward preference task. *European Journal of Neuroscience*, pp. 13, 2003.

Zhengzheng Xing, Jian Pei, and Eamonn Keogh. A brief survey on sequence classification. *ACM SIGKDD Explorations Newsletter*, 12(1), 2010. ISSN 1931-0145. doi: 10.1145/1882471.1882478.

Guangyu Robert Yang, Madhura R Joglekar, H Francis Song, William T Newsome, and Xiao-Jing Wang. Task representations in neural networks trained to perform many cognitive tasks. *Nature Neuroscience*, 22(2):1–16, December 2018. doi: 10.1038/s41593-018-0310-2. URL http://dx.doi.org/10.1038/s41593-018-0310-2. Publisher: Springer US.

Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continuous Learning of Context-dependent Processing in Neural Networks. *arXiv:1810.01256 [cs]*, October 2018. URL http://arxiv.org/abs/1810.01256. arXiv: 1810.01256.

A TRAKR HYPERPARAMETERS

The recurrent weights J_{ij} are weights from unit j to i. The recurrent weights are initially chosen independently and randomly from a Gaussian distribution with mean of 0 and variance given by g^2/N. The input weights w_{in} are also chosen independently and randomly from the standard normal distribution.

An integration time constant $\tau = 1ms$ is used. We use gain $g = 1.2$ for all the networks.

The matrix P is not explicitly calculated but updated as follows:

$$P(t) = P(t-1) - \frac{P(t-1)\phi(t)\phi'(t)P(t-1)}{1 + \phi'(t)P(t-1)\phi(t)}$$

The learning rate η is given by $\dfrac{1}{1 + \phi'(t)P(t)\phi(t)}$.

The number of units used in the reservoir is generally $N = 30$.