Inferring plasticity rules from single-neuron spike trains using deep learning methods

Xingyun Wang
Department of Cellular and Molecular Medicine
University of Ottawa
Ottawa, ON, Canada
xwang671@uottawa.ca

Jean-Claude Béïque
Department of Cellular and Molecular Medicine
University of Ottawa
Ottawa, ON, Canada
jbeique@uottawa.ca

Richard Naud*
Department of Cellular and Molecular Medicine
Department of Physics
University of Ottawa
Ottawa, ON, Canada
rnaud@uottawa.ca

Abstract

Synaptic plasticity is a core basis for learning and adaptation. Determining how synapses are altered by local signals – the learning rules – is the hinge about which brain activity pivots. A large number of in vitro characterizations have focused on restricted sets of core properties [e.g., spike-timing-dependent plasticity (STDP), burst-dependent plasticity (BDP)], but it remains to be established which learning rule is most consistent with the changes in activity patterns observed while the network learns to perform a task. To address this question, we hypothesize that correlations between features of single post-synaptic neural activity and subsequent plasticity of the representations could distinguish between learning rules. Because this correlation was expected to be diluted by the notoriously large variability of brain activity, we propose to infer the learning rules from passive observations using artificial neural networks. Using surrogate data to test the validity of our approach, we found that transformers and temporal convolutional networks could classify learning rules far above the chance level (20%), with transformers achieving a better overall accuracy (78%) than the convolutional models (73%). This performance is achieved despite the presence of noise and representational drift. Although not as accurate as the complex transformer-based classifier, the convolutional model can provide insights into the interpretable features used by the algorithm to perform the classification. Our work establishes support for using deep learning methods to infer the category of learning rules given spiking data.

*Corresponding author

Preprint. Under review.

1 Introduction

Surviving and thriving are no easy tasks, but animals learn to adapt themselves to complex environments by reforming their neuronal networks. This ability is thought to arise from the interplay between learning rules and the natural reverberation of neuronal activity immersed in adaptive behavior [1]. A synaptic learning rule is a description of changes in synaptic efficacy from a restricted set of features of neuronal activity. A number of learning rules have been identified either based on recordings from neurons in brain slices or proposed from a theoretical standpoint. Some are expressed in terms of the firing rate on both sides of the synapse: Bienenstock-Cooper-Monro (BCM) [2] or correlation-based plasticity (CDP) [3, 4]. Others explicitly take into account spike timing: spike-timing-dependent plasticity (STDP) describes plasticity resulting from the relative coincidence between pre- and post-synaptic spikes [5, 6], while burst-dependent plasticity (BDP) focuses on the role of pre-and post-synaptic firing patterns on regulating plasticity [7, 8, 9, 6, 10]. Other examples include neuromodulated-STDP [11, 12, 13] and oscillation-modulated plasticity [14]. Given that synapses are exceedingly complex, learning rules are a simplified picture of their time-evolution: frameworks whose power can be established by learning a task [15, 16, 17, 18, 19] and whose validity can be verified by observational [20, 21, 22] as well as perturbation [23, 24, 25, 26, 9, 27] studies in vivo. Such inference of plasticity rules in vivo is essential to validate the hypotheses obtained from in vitro recordings. For example, even for the widely popular STDP learning rule, multiple factors could make this learning rule irrelevant in vivo [28, 29, 6]. To assess the validity of a learning rule using observational data, its manifestation during learning in the behaving animal must be distinguishable from other possibilities. In this paper, we ask whether artificial neural networks can be used to classify distinct learning rules from the evolving activity of recorded single neuron. We built a biological spiking neural network (SNN) and let the spike train evolve according to learning rules found in the slice physiology literature (STDP and BDP) or used in the machine learning field (i.e., backpropagation through time, BPTT). The spike trains during training were concatenated to generate learning trajectories. We used temporal convolutional neural network (TCNN)[30] and the temporal transformer encoder (TTE)[31] to classify learning rules from the surrogate data. After training, these classifiers should be capable of inferring the learning rule from a the spike train of a single neuron. Two distinct scenarios of operations were considered, one where learning rules were set to increase adequation between the end-representation and its power to solve MNIST. The other was where the end-representaiton was fixed across the different learning rules. In both scenarios, the TTE showed significantly better classification performance than the TCNN. The interpretation of the TCNN showed that the classifier had learned the inner difference between distinct plasticity rules. Prediction accuracy was shown to be robust to perturbations of the burst fraction baseline, as well as the presence of representational drift [32]. Together, our work indicates the practicability of artificial neural network approaches to distinguishing learning rules from single post-synaptic spike train.

2 Related Work

In the hippocampus, a learning rule was inferred from in vivo observations during the emergence of place selectivity [33, 22]. Two unique features of the hippocampus enable this discovery: a well-characterized structure of selectivity and a connectivity structure with weak recurrence. In the cortex, however, representations are more distributed, dynamic [34, 35], and the presence of recurrent connections makes the relationship between changes in activity patterns and the learning rules particularly intricate. Therefore, a methodology able to discriminate different types of learning rules in a spiking, recurrent architecture is needed, particularly one applicable in the presence of representational drift, noise, and other confounding factors.

In the cortex, Lim et al. [20] revealed a signature of a learning rule by comparing the post-synaptic firing rate distribution between two learning sessions. This method is capable of unveiling a Hebbian-type learning rule during learning whose specific dependence on pre-and post-firing rate is inferred. Simulation using the inferred plasticity rule can reproduce experimental firing rate distributions of novel and familiar sessions. Our approach is distinct in that even though we cannot infer the structure of the learning rule, we can still classify distinct learning rules. Importantly, Lim et al. could not discriminate which features of the pre-and post-synaptic spiking activity (spike time, bursts, etc.) were most consistent with the changes in firing rate distribution. Furthermore, this approach is insensitive to the trajectory of neuronal responses taking place between learned and naive states.

While many studies have established that some learning rules are consistent with a restricted set of features of neuronal activity [21, 36, 37, 38], our work aims at finding which out of many is more consistent with a large amount of neuronal activity and thus with multiple features of the activity. Asking a similar question entirely within artificial neural networks, Nayebi et al. [39] investigated machine learning methods that could successfully identify distinct optmization algorithms used to solve image classification tasks. In this work, the activation patterns predicted the learning rules with very high accuracy. This result is consistent with Portes et al. [40] which found a neural activity metric to distinguish supervised from reinforced learning rule in a recurrent neural network. Notably, Nayebi et al. [39] utilized a population of post-synaptic neurons to classify the learning rules while we focus on classification from single-neuron spiking activity.

3 Results

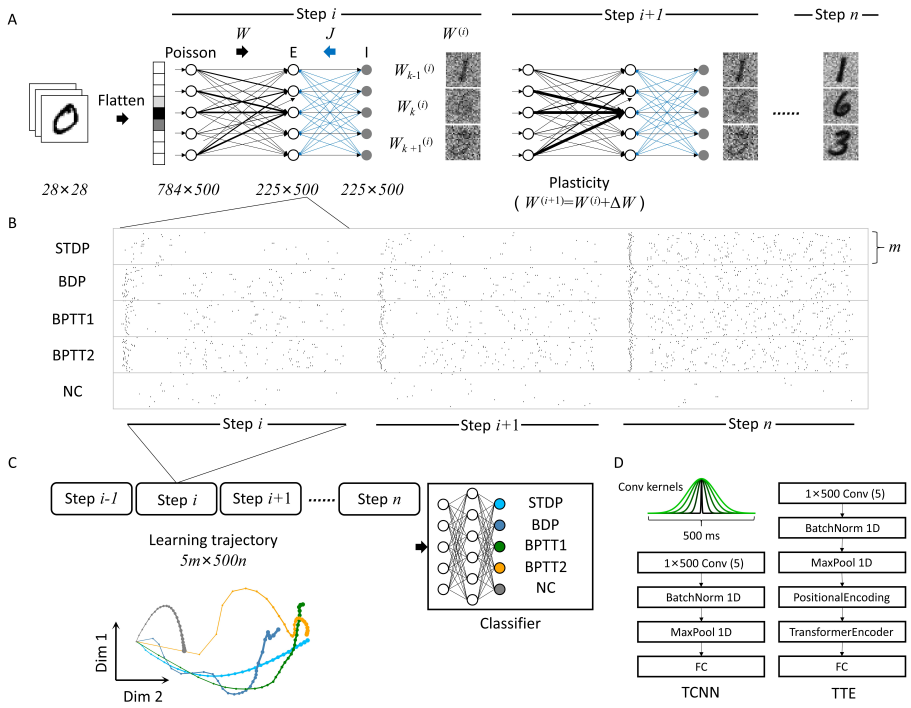

Figure 1: **Trajectory generation from spiking neural network simulations in the context of image classification.** **(A)** Schematic representation of the three layers of the SNN. Weights onto three exemplar neurons are shown as heatmaps, illustrating changes in weights between two learning steps. **(B)** During the learning process, for each of the 5 learning rules, m spike trains from a subset of responsive excitatory neurons are recorded for 500 ms and concatenated across n plasticity steps. **(C)** Concatenation of all learning steps of all responsive neurons form all the trajectories of the training set. Single spike train-learning rule pairs are fed into the classifier. Two-dimensional latent learning trajectories of distinct learning rules are also shown for visualization (using Gaussian process factor analysis, GPFA [41]). **(D)** Architectures of the two classifiers used.

3.1 An architecture for the simulation of learning in spiking neural networks

To estimate the feasibility of separating plasticity rules from observational spiking activity, we simulated synthetic spike trains of a functional spiking neural network (SNN) with distinct synaptic plasticity rules and trained an artificial neural network to classify the types of plasticity rules. We focused on a restricted yet representative set of learning rules. Namely, we considered two spike-based biological learning rules (STDP, BDP) and two variants of a machine learning rule called spike-based BPTT. The two variants of the back-propagation through time (BPTT) algorithms differed in their

cost functions: one using $L1$ loss (BPTT1) and the other using $L2$ loss (BPTT2). A fifth condition was added as a negative control (NC). We describe each learning rule in turn:

- **Spike-timing-dependent plasticity (STDP)** For STDP, the synaptic weight is determined by the relative spike co-occurrence between pre- and post-synaptic neurons. Our implementation follows Ref. [15] which models experimental observations at GABAergic and glutamatergic synapses [42, 43, 44], but is distinct from the common asymmetric forms of STDP [45]. The synaptic weight is updated at every post-synaptic spike time, t_{post}, according to the learning rule, $\Delta W(t_{post}) = \eta(x_{pre}(t_{post}) - x_{th})(W_{max} - W)$, where x_{pre} is an eligibility trace of pre-synaptic spikes (S_{pre}) with a timescale set by α^{-1} and dynamics following $dx_{pre}/dt = -\alpha x_{pre} + S_{pre}$. The parameter x_{th} is an eligibility trace threshold for eliciting long-term potentiation (LTP). The synaptic weights W are modeled as saturating at W_{max}.
- **Burst-dependent plasticity (BDP)** For BDP, the neuronal response is assumed to be made of singlets or bursts with different effects on the plasticity. Bursts are spikes preceded by maximum one interval less than 16 ms, while singlets are the spikes preceded by an interval more than 16 ms. For each post-synaptic neuron, we form a burst train, $B(t)$, and an event train, $E(t)$. The synaptic weight change follows $\Delta W = \eta(B(t) - b^f_{bl}E(t))Relu(x_{pre} - x_{th})$ [6, 9, 27, 46], where b^f_{bl} is a parameter regulating the strength of event-evoked long-term depression versus the strength of burst-evoked LTP. By controlling the burst fraction of the neurons, it is possible to control the plasticity [47]. Thus, when setting the burst fraction to the gradient of a loss, BDP can be used in a supervised setting. In what follows, we will use BDP to learn a particular representation $\mathbf{r}_{tar}$, $Loss_{BDP} = L2(\mathbf{r}_{tar}, \mathbf{r}_{BDP})$, where $\mathbf{r}_{BDP}$ is the representation observed in the simulations.
- **Spike-based surrogate gradient backpropagation through time (BPTT)** We used BPTT as a learning rule, with surrogate gradients to pass the gradient through the nondifferentiable spike generation step [48, 49]. For BPTT1, the SNN only uses the smooth $L1$ to learn target representation $\mathbf{r}_{tar}$ for two learning protocols (i.e., the fixed end-representation protocol and the task-oriented protocol). For BPTT2, the SNN exerts the $L2$ loss to learn the target representation in the fixed end-representation protocol and the $NLLLoss$ (Negative Log-Likelihood Loss) in the task-oriented protocol. In the fixed end-representation protocol, the target representation $\mathbf{r}_{tar}$ is first generated by the STDP, and then other plasticity rules are employed to learn this representation. For the task-oriented protocol, the SNN needs to find the target representation, which enables itself to independently complete the category classification task.
- **Weight updating fluctuation and representational drifts** In biological conditions, the synaptic weight updates are noisy, which might contribute to representation drift commonly observed [32, 50]. For comparison, all plasticity rules (both biological and non-biological rules) have a noisy weight-updating process. Here, we perturbed the deterministic weight update ΔW with a Gaussian noise, ξ_W, having standard deviation σ_W such that: $\Delta W' = \eta(\Delta W + \xi_W), \xi_W \sim \mathcal{N}(0, \sigma^2_W)$, where η is the learning rate. During all the learning steps, we set the fluctuation level to be half of the maximum value of the initially learned feedforward connection strength, $\sigma_W = 5e^{-4}$.
- **Negative control (NC)** For the negative control, the weight update of the SNN is only driven by weight fluctuations, $\Delta W' = \eta\xi_W, \xi_W \sim \mathcal{N}(0, \sigma^2_W)$.

For the SNN, we built a three-layer SNN model following the architecture used by Diehl and Cook [15], as shown in Fig 1A. We chose this architecture because it has been shown to learn representations amenable to some degree of image classification with STDP [15]. BDP [47, 19], BPTT1 and BPTT2 can also be implemented in this architecture with good task performance; thus choosing this architecture allowed us to compare the different learning rules within a similar context.

The specific SNN [15] architecture we considered is made of three layers. The first layer consists of Poisson spike trains (784 neurons wide, 500 ms long) whose firing intensity is set by the pixel intensity of an input image (flattened). These spike trains project to the second layer using the weight matrix, **W**. This second layer is constituted only of excitatory neurons. Their activity affects neurons in the third layer, itself made only of inhibitory neurons matching the number of excitatory neurons. A single excitatory neuron activates a single inhibitory neuron (diagonal weight matrix). Each inhibitory

neuron projects back to all second layer neurons except for the one from which it receives excitation input (off-diagonal weight matrix). This connection, **J**, models lateral inhibition, essential to sparsify and decorrelate the representation. In this SNN, both excitatory neurons and inhibitory neurons are modeled as the leaky integrate-and-fire (LIF) model, $\tau_{mem} dV/dt = -(V - V_{rest}) + RI + \xi_V$, in which accumulated input currents I push the membrane potential V above a threshold, V_T, to generate a spike S. The dynamics evolve at the time scale set by the membrane, τ_{mem}. Additional parameters are V_{rest}, the resting potential, and R, the resistance. Each excitatory neuron received background noise, ξ_V. Only the connection between the first and second layer, **W**, is plastic and follows the learning rule. When the spike train $S(t)$ for one learning step is generated, we used the burst fraction b^f to generate the burst train $B(t)$ and the event trains $E(t)$. The burst fraction follows $b^f = Relu(b^f_{bl} + \beta b^f_{bl} Loss_{BDP})$, where β is used to scale the output b^f less than 1 and $Relu$ function constrains the output larger than 0.

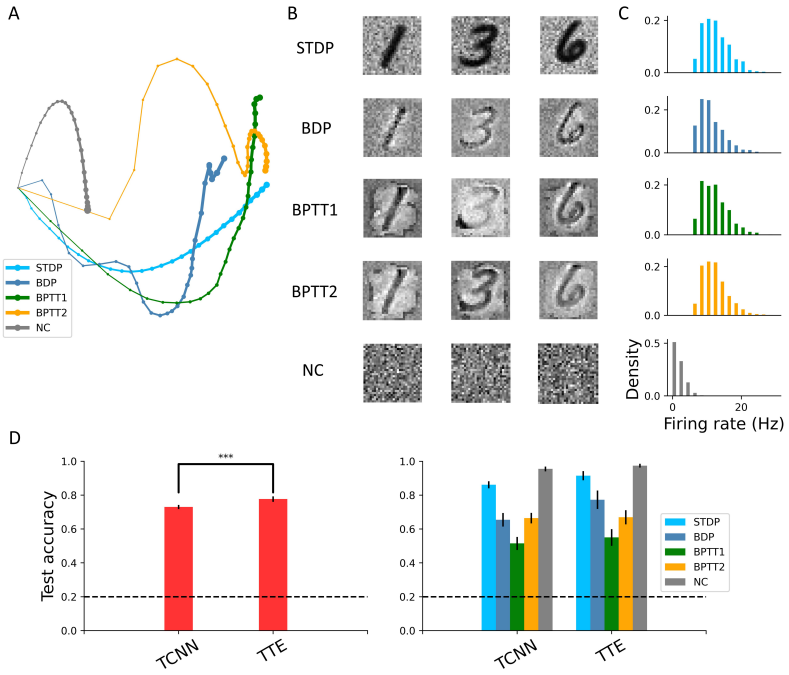

Figure 2: **Learning rule categorization in the fixed end-representation training protocol. (A)** Latent space (two GPFA components) trajectories of distinct plasticity rules in the fixed end-representation protocol. The marker size of each learning trajectory grows with learning steps. **(B)** Connection weights onto three exemplar excitatory neurons for each plasticity rule. **(C)** Mean firing rate distribution of responsive neurons for each learning rule (bin size: 2 Hz). **(D)** Left, overall accuracy of classifying five plasticity rules. Right, accuracy specific to each learning rule. Independent two-sample t-test, $^{***}P < 0.001$. Chance level is indicated with a dashed line in each graph.

For each learning rule, we built trajectories of recorded responses to be fed into a classifier. To this end, we recorded the responses of all neurons in the network upon presentation of a restricted set of 300 examples from the MNIST dataset [51]. While an example is being presented, the weights onto the excitatory neurons are updated according to one of the plasticity rules (Fig. 2A), such that the next presentation of the set of 300 training digits elicits slightly different responses. We repeated input presentation and plasticity for n plasticity steps and for each learning rule. In the last plasticity step, we selected the subset of neurons having shown significant responses (firing rate above 6 Hz) for at least one of the categories of digits. To build the trajectories of responses for learning rule classification, we associated each responsive cell to one (or more) digit category for which it responded after the last plasticity step. We then extracted and concatenated the responses to that digit for all training steps (Fig. 2B). If a responsive neuron was responding to more than one category, then this neuron provided more than one trajectory. In this way, although a single trajectory sees only one digit, different trajectories were made of responses to different digits. The task of the classifier is then to map each response trajectory onto the learning rule category (Fig. 2C). The classifier only

sees the response of one neuron to one particular digit at a time; different neurons or responses to different digits are treated as different examples.

3.2 Plasticity prediction in the fixed end-representation protocol

First, we investigated the feasibility of separating learning trajectories while enforcing that the representation at the end of training was the same for the different learning rules, a protocol labeled '*fixed end-representation*'. This was introduced to disambiguate the role of the end-representation from that of the intermediate dynamics. This constraint was later relaxed. Because all learning rules but STDP can be built to learn a particular representation, we have emulated the end-representation obtained under STDP plasticity using the other learning rules, setting $\mathbf{r}_{tar} = \mathbf{r}_{STDP}$. In this scenario, the networks do not optimize classification accuracy directly. Instead, learning is associated with task improvement since the representation obtained via STDP is better at the task than those obtained with random initializations. For the BDP, BPTT1, and BPTT2 learning rules, this protocol is not expected to be biological because it involves knowing the end-representation of another network. Nevertheless, it is an important test case for the possibility of separating the learning rules based on response trajectories alone.

We considered two classifiers to infer the plasticity types from the SNN trajectories:

- **Temporal convolutional neural network (TCNN)** [30] This classifier contains only one convolutional layer and one fully connected layer (Fig. 1D). The "temporal convolutional" means the TCNN convolves its one-dimensional kernels with each learning trajectory to extract temporal features. To improve the interpretability of the TCNN, we chose the shape of these kernels to be Gaussian with widths evenly tiled between 2 ms and 66 ms, as shown in Figure 1D left panel. These temporal kernels are fixed during training. We also made the bias in the convolutional layer and the fully connected layer 0, which let the weight in the fully connected layer represent the contribution of each temporal feature for some plasticity category. During the classifier training procedure, only the weight in the fully connected layer is adjusted by the Adam algorithm to minimize the classification loss.
- **Temporal transformer encoder (TTE)**[31, 52, 53] The learning trajectory is first temporally embedded using a temporal convolutional layer, as in the TCNN. As a more complex classifier, information is then processed by positional encoding followed by a transformer encoder, which is cascaded by a fully connected layer (Fig. 1D). Same as the TCNN, the Adam optimizer is used to train the TTE model to classify learning trajectories.

Figure 2A shows latent trajectories describing the network activity evolution under different learning rules in GPFA space. These network-level trajectories from BDP, BPTT1, and BPTT2 end in close points around STDP, which is consistent with the fixed end-representation requirement. The changes in representation can be visualized further by inspecting the connection weights between the input neurons and individual excitatory neurons in the second layer (weights onto exemplar neurons are shown in Fig 2B). At the end of learning, STDP has developed a weight structure where shapes of particular digits can be visually recognized. Inspecting the end connection weights of corresponding neurons under BDP, BPTT1, and BPTT2 shows a similar weight structure. Similarly, the firing rate distribution cannot reveal the learning rule (Fig 2C). Although trained by different plasticity rules, these results show that the fixed end-representation protocol effectively confines these SNNs to learn similar weight structure, representation, and firing statistics at the end of learning.

We split the simulated trajectories into a training dataset and a test dataset ($2:1$ ratio). The classifier learns a relationship between the trajectory of a spiking response and the plasticity category from the training dataset and then fixes its parameters to perform predictions within the testing dataset. We repeated the learning and prediction process 50 times to obtain prediction accuracy statistics. We used two accuracy metrics to measure classifiers' performance: the first one is the overall accuracy, which is defined as the correct classified number divided by sample number; the second one is the individual accuracy, which is correct identified number for individual category divided by the sample number of this category.

We found that the overall prediction accuracy is clearly larger than the chance level (0.2) for both classifiers (Fig. 2D). The accuracy using TTE is 0.777, which is significantly higher than the accuracy using TCNN, 0.730 (independent two-sample t-test, $P < 0.001$). In the right panel of Fig 2D, the individual mean prediction accuracy specific to each plasticity rule is shown, with values given

Table 1: Prediction accuracy for different learning protocols

	Fixed end-representation				Task-oriented			
	Mean accuracy		Standard deviation		Mean accuracy		Standard deviation	
	TCNN	TTE	TCNN	TTE	TCNN	TTE	TCNN	TTE
STDP	0.861	0.915	0.021	0.027	0.771	0.786	0.028	0.043
BDP	0.654	0.773	0.040	0.054	N/A	N/A	N/A	N/A
BPTT1	0.515	0.550	0.039	0.050	0.553	0.572	0.045	0.066
BPTT2	0.664	0.669	0.031	0.042	0.465	0.504	0.035	0.055
NC	0.955	0.974	0.014	0.011	0.986	0.993	0.009	0.005
Overall	0.730	0.777	0.012	0.016	0.694	0.714	0.014	0.015

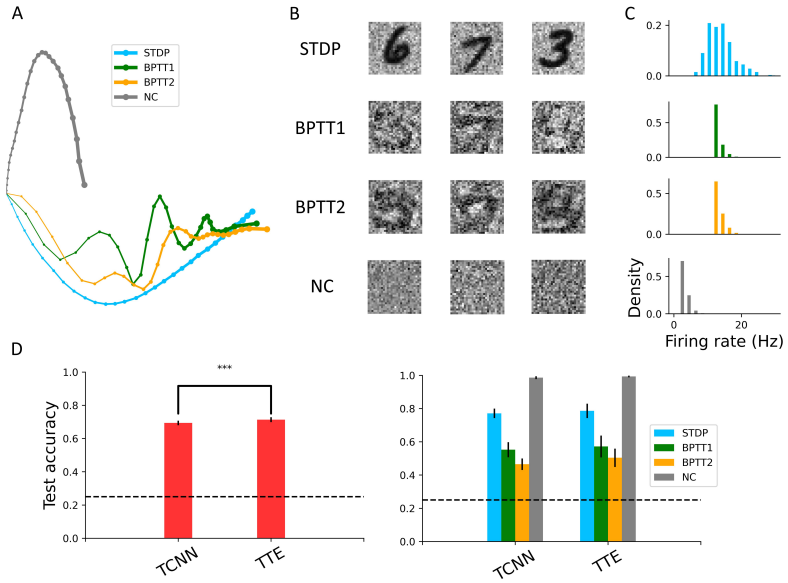

Figure 3: **Learning rule categorization in the task-oriented protocol.** As in Fig. 2, but for the task-oriented protocol.

in Table 1. According to the individual prediction accuracy, we found that the overall accuracy is pulled down by BPTT1 and BPTT2. This is likely reflecting that distinguishing cost function is more difficult than distinguishing the learning rule. Nevertheless, learning trajectories from the same learning rules with distinct loss functions can be separated (individual accuracy larger than chance level). Classification is easiest for the STDP and NC category. There are also classifier-specific aspects as BDP is best classified with TTE. Together, these results demonstrate that the learning rules can be inferred quite reliably from the response changes of individual neurons in the course of plasticity.

3.3 Plasticity prediction in the task-oriented protocol

We next investigated a more biological condition where each learning rule attempted to classify digits with an output layer. Because the readout layer is fixed and only feedforward weights in SNN undergo plasticity, we called this approach the task-oriented protocol. Here, SNNs trained with different plasticity rules must complete the MNIST task independently, with adaptations specific to each learning rule (except BDP which cannot be adapted to this protocol):

- **STDP** The SNN representation evolves without supervision. Previous work demonstrated that this change in representation is associated with improvement on MNIST when paired with a readout mechanism [15].

- **BPTT1 and BPTT2** We added a linear readout layer $\mathbf{W}_{lin}$ to map the excitatory neuron's activity $\mathbf{r}$ into the output category. The readout layers for both BPTT1 and BPTT2 are fixed in order to constrain the weight updates to only occur in the weights from the input layer. This requirement prevents different loss functions from affecting the readout weights, thus enforcing a more direct effect of the loss function onto the observed trajectories. BPTT1 uses smooth $L1$ loss to the one-hot representation $\mathbf{y}$ of categories, which is $Loss_{BPTT1} = smoothL1(\mathbf{W}_{lin}\mathbf{r}, \mathbf{y})$. BPTT2 uses classic $NLLLoss$ loss to learn the mapping from input to target categories, which is written as $Loss_{BPTT} = NLLLoss(Softmax(\mathbf{W}_{lin}\mathbf{r}), \mathbf{y})$.

Figure 3A shows different learning rules' low-dimensional learning trajectories in the task-oriented training paradigm, showcasing that population-level trajectories end in the same neighborhood. When looking at the weight structure, we found that the SNNs trained with distinct plasticity rules showed distinct connection weights onto exemplar excitatory neurons at the end of training (Fig 3B). The connectivity matrices resulting from BPTT1 and BPTT2 are qualitatively similar and do not align to a particular digit example as much as STDP; instead, they learn an abstract representation amenable to digit categorization. Similarly, the firing rate distribution of BPTT1 and BPTT2 appear distinct from that of STDP, but remain within a physiological range (Fig 3C). In sum, our surrogate data exemplify the changes in representations occurring while different plasticity rules are addressing the same learning task.

In this scenario, we found that the classifiers could again successfully predict the learning rule categories (Fig 3D). The overall accuracy is higher than the chance level (0.25) for both classifiers (Table 1). The accuracy to individual learning rule indicates that it is led by reliable classification of the negative control. BPTT1 and BPTT2 remain the most difficult classes to disambiguate.

3.4 Interpreting features underlying the TCNN classifier

Based on the above results, learning trajectories from different plasticity rules are tolerably separable. An important question for using this methodology for answering neurobiology questions, however, is: what features of the neuronal responses are being used by the classifiers? We addressed this question with the TCNN because its simpler architecture makes this question tractable.

TCNN relies on temporal convolutions applied to the input spike trains. Each of the five kernels for these convolutions is shown in Fig 4. In these kernels, the narrowest (2 ms kernel width, black curve in Fig 4A) kernel filters for individual spike times. The slightly larger kernels can be sensitive to bursts of spikes over 10-40 ms (green and dark blue curves in Fig 4A). The largest kernels communicate a quantity consistent with a time-average firing rate. Since we fixed the convolutional layer (kernels and biases) and set fully connected layers' biases to zero during learning, the weight of the fully connected layer indicates the relative contribution of each temporal kernel to each digit category. Investigating what parameters the learning algorithm has chosen can reveal what feature is relevant to the classification. Note that there is a set of weights for every time bin in the trajectory, so the relative relevance of each feature (or kernel) can change in the course of the learning trajectory.

Fig 4B displays the learned weights over the trajectory for each kernel width and each learning rule. To gain insight into the relevance of a temporal feature for classification, we computed the average weight for each kernel width. We found that NC is reported with a net positive contribution only for the most temporally precise kernels. NC is also distinguished by relatively low weights for the firing rate kernel towards the end of learning steps, as seen by the presence of more negative weights on the right side of Fig. 4B. BPTT1 and BPTT2 rely on the positive contribution of the most temporally precise kernels of specific timings. Consistent with the larger proportion of bursts in BDP, this learning rule is distinguished with an intermediate kernel width and a negative contribution from a temporally precise kernel. The STDP relies on a negative contribution from the temporally precise kernel at the beginning and positive contributions from the rate-computing kernels in the end. These observations help provide an explanation of how the classifier is able to discriminate plasticity rules.

3.5 Perturbation for two conditions verify the feasibility of practical use

Out-distribution test is a necessary step toward the application of an artificial neural network. Here, we test our classifiers after implementing a perturbation. We considered two types of perturbations and tested the accuracy of our classifiers: 1) Burst fraction perturbation, where we trained with sparse burst fraction baseline (0.2) and tested with a higher one (0.4; Fig 5A), this perturbation was only

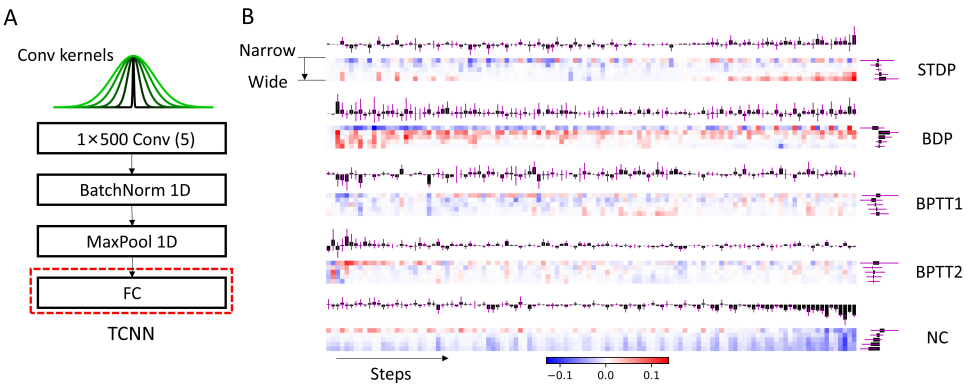

Figure 4: **TCNN relies on different temporal features to predict learning rule identities. (A)** Five Gaussian temporal convolution kernels are used in the TCNN. The architecture of the TCNN is shown below the kernels. **(B)** Heat map of the fully connected layer's weights (average value across 50 initializations) from each channel (lines) and each time unit (columns) for each learning rule. The column-wise (top) and line-wise (right) weight average and standard deviation are shown.

applied to the fixed end-representation protocol. 2) Background perturbation, where we set the noise level to 0.05 in training and to 0.1 in the test set (Fig 5B). This perturbation was only applied to the task-oriented protocol.

As shown in Figure 5, the total prediction accuracy under the fixed end-representation protocol decreases under burst fraction perturbation. For both classifiers, the overall mean accuracy is affected, although weakly. This decrease in accuracy arises in large part from the classification of the BPTT1 learning rule. However, background perturbation in the task-oriented protocol (Fig. 5) causes a significant overall accuracy decrease for both classifiers. This serious decrease in accuracy arises in large part from the classification of the BPTT1 and BPTT2 learning rules. The mean individual accuracy for BPTT1 and BPTT2 all fall close to the chance level (0.25), which causes overall prediction accuracy to decrease to 0.534 (TCNN) and 0.531 (TTE). This result indicates that the separability of learning trajectories of BPTT1 and BPTT2 is sensitive to the background noise. Together, out-of-distribution testing retains above-chance performance in overall accuracy but highlights the importance of training over a wide range of states to preserve testing accuracy.

4 Conclusions and discussion

To verify the feasibility of inferring biological plasticity rules by observing single-neuron spike trains, we trained multiple SNN models to classify images using distinct plasticity rules. The prediction accuracy for both learning protocols is largely above chance, despite the presence of representation drift and background noise. Different temporal components of the response are exploited by the machine learning algorithms to distinguish plasticity rules in a manner consistent with expectations. We found, however, that the methodology tended to be sensitive to the particular training distribution. While well-trained classifiers were robust to testing in higher levels of bursting, they suffered greatly upon testing with different levels of background noise. Our results illustrate the importance of training model using data covering a wide range of biological states.

In conclusion, our work shows it is feasible to infer the plasticity categories only using single-neuron post-synaptic spike trains by deep learning methods. This progress paves the way for a more comprehensive understanding of the learning rules in biological brains, which potentially inspires the solutions to the credit assignment problem in advanced artificial neural networks.

References

[1] Blake A Richards, Timothy Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Archy De Berker, Surya Ganguli, Colleen Gillon, Adam Hafner, Danijar Kepecs, Nikolaus Kriegeskorte, Peter Latham, Grace Lindsay, Richard Miller, Ken-

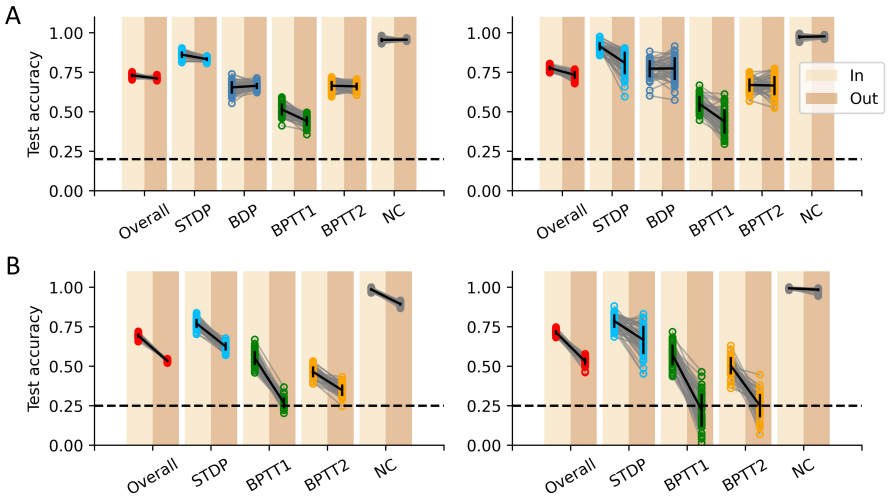

Figure 5: **Plasticity prediction in out-of-distribution learning trajectories. (A)** For the fixed end-representation protocol, the TCNN (left panel) and TTE (right panel) are tested with neurons either having the same burst fraction baseline (left points in a pair) or increasing the burst fraction baseline to 0.4 (right points in a pair). **(B)** Same as A but for background noise perturbation in the task-oriented protocol.

neth Naud, Christopher Pack, Panayiota Poirazi, Rui Ponte Costa, Pieter Roelfsema, João Sacramento, Andrew Saxe, Anna Schapiro, Walter Senn, Greg Wayne, Daniel Yamins, Friedemann Zenke, Denis Zylberberg, Joel Therien, and Konrad Kording. A deep learning framework for systems neuroscience. *Nat Neurosci*, 2019.

[2] E.L. Bienenstock, L.N. Cooper, and P.W. Munroe. Theory of the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J Neurosci*, 2:32–48, 1982. reprinted in Anderson and Rosenfeld, 1990.

[3] Kenneth D Miller. Synaptic economics: competition and cooperation in synaptic plasticity. *Neuron*, 17(3):371–374, 1996.

[4] Kenneth D Miller. Equivalence of a sprouting-and-retraction model and correlation-based plasticity models of neural development. *Neural Computation*, 10(3):529–547, 1998.

[5] H. Markram and M. Tsodyks. Redistribution of synaptic efficacy between neocortical pyramidal neurons. *Nature*, 382:807–810, 1996.

[6] Yanis Inglebert, Johnatan Aljadeff, Nicolas Brunel, and Dominique Debanne. Synaptic plasticity rules with physiological calcium levels. *Proceedings of the National Academy of Sciences*, 117(52):33639–33648, 2020.

[7] Per Jesper Sjöström, Gina G Turrigiano, and Sacha B Nelson. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, 32(6):1149–1164, 2001.

[8] Robert C Froemke, Ishan A Tsay, Mohamad Raad, John D Long, and Yang Dan. Contribution of individual spikes in burst-induced long-term synaptic modification. *J. Neurophys.*, 95(3):1620–1629, 2006.

[9] Katie C Bittner, Aaron D Milstein, Christine Grienberger, Sandro Romani, and Jeffrey C Magee. Behavioral time scale synaptic plasticity underlies ca1 place fields. *Science*, 357(6355):1033–1036, 2017.

[10] Lea Caya-Bissonnette, Richard Naud, and Jean-Claude Beique. Cellular substrate of eligibility traces during behavioral timescale synaptic plasticity. *BioRxiv*, 2023.

[11] Kaiwen He, Marco Huertas, Su Z Hong, XiaoXiu Tie, Johannes W Hell, Harel Shouval, and Alfredo Kirkwood. Distinct eligibility traces for ltp and ltd in cortical synapses. *Neuron*, 88(3):528–538, 2015.

[12] Verena Pawlak and Jason N D Kerr. Dopamine receptor activation is required for corticostriatal spike-timing-dependent plasticity. *J Neurosci*, 28(10):2435–2446, Mar 2008.

[13] Nicolas Frémaux, Henning Sprekeler, and Wulfram Gerstner. Functional requirements for reward-modulated spike-timing-dependent plasticity. *J Neurosci*, 30(40):13326–13337, 2010.

[14] Patricio T Huerta and John E Lisman. Bidirectional synaptic plasticity induced by a single burst during cholinergic theta oscillation in ca1 in vitro. *Neuron*, 15(5):1053–1063, 1995.

[15] Peter U Diehl and Matthew Cook. Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Frontiers in computational neuroscience*, 9:99, 2015.

[16] Jordan Guerguiev, Timothy P Lillicrap, and Blake A Richards. Towards deep learning with segregated dendrites. *Elife*, 6:e22901, 2017.

[17] Alexandre Payeur, Jordan Guerguiev, Friedemann Zenke, Blake A Richards, and Richard Naud. Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits. *Nature neuroscience*, 24(7):1010–1019, 2021.

[18] João Sacramento, Rui Ponte Costa, Yoshua Bengio, and Walter Senn. Dendritic cortical microcircuits approximate the backpropagation algorithm. In *Advances in Neural Information Processing Systems*, pages 8721–8732, 2018.

[19] Will Greedy, Heng Wei Zhu, Joseph Pemberton, Jack Mellor, and Rui Ponte Costa. Single-phase deep learning in cortico-cortical networks. *arXiv preprint arXiv:2206.11769*, 2022.

[20] Sukbin Lim, Jillian L McKee, Luke Woloszyn, Yali Amit, David J Freedman, David L Sheinberg, and Nicolas Brunel. Inferring learning rules from distributions of firing rates in cortical neurons. *Nature neuroscience*, 18(12):1804–1810, 2015.

[21] J.M. Young, W.J. Waleszczyk, C. Wang, M.B. Calford, B. Dreher, and K. Obermayer. Cortical reorganization consistent with spike timing–but not correlation-dependent plasticity. *Nature Neuroscience*, 10:887–895, 2007.

[22] Katie C Bittner, Christine Grienberger, Sachin P Vaidya, Aaron D Milstein, John J Macklin, Junghyup Suh, Susumu Tonegawa, and Jeffrey C Magee. Conjunctive input processing drives feature selectivity in hippocampal ca1 neurons. *Nat. Neurosci.*, 18(8):1133–1142, 2015.

[23] C Daniel Meliza and Yang Dan. Receptive-field modification in rat visual cortex induced by paired visual stimulation and single-cell spiking. *Neuron*, 49(2):183–189, 2006.

[24] Verena Pawlak, David S Greenberg, Henning Sprekeler, Wulfram Gerstner, and Jason ND Kerr. Changing the responses of cortical neurons from sub-to suprathreshold using single spikes in vivo. *Elife*, 2:e00012, 2013.

[25] Sadegh Nabavi, Rocky Fox, Christophe D Proulx, John Y Lin, Roger Y Tsien, and Roberto Malinow. Engineering a memory with ltd and ltp. *Nature*, 511(7509):348–352, 2014.

[26] Sami El-Boustani, Jacque PK Ip, Vincent Breton-Provencher, Graham W Knott, Hiroyuki Okuno, Haruhiko Bito, and Mriganka Sur. Locally coordinated synaptic plasticity of visual cortex neurons in vivo. *Science*, 360(6395):1349–1354, 2018.

[27] Aaron D Milstein, Yiding Li, Katie C Bittner, Christine Grienberger, Ivan Soltesz, Jeffrey C Magee, and Sandro Romani. Bidirectional synaptic plasticity rapidly modifies hippocampal representations. *Elife*, 10:e73046, 2021.

[28] Gayle M Wittenberg and Samuel S-H Wang. Malleability of spike-timing-dependent plasticity at the ca3–ca1 synapse. *Journal of Neuroscience*, 26(24):6610–6617, 2006.

[29] John Lisman and Nelson Spruston. Postsynaptic depolarization requirements for LTP and LTD: a critique of spike timing-dependent plasticity. *Nature Neuroscience*, 8(7):839–841, 2005.

[30] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[32] Michael E Rule, Timothy O'Leary, and Christopher D Harvey. Causes and consequences of representational drift. *Current opinion in neurobiology*, 58:141–147, 2019.

[33] Jérôme Epsztein, Michael Brecht, and Albert K Lee. Intracellular determinants of hippocampal ca1 place and silent cell activity in a novel environment. *Neuron*, 70(1):109–120, 2011.

[34] Laura N Driscoll, Noah L Pettit, Matthias Minderer, Selmaan N Chettih, and Christopher D Harvey. Dynamic reorganization of neuronal activity patterns in parietal cortex. *Cell*, 170(5):986–999, 2017.

[35] Joshua H Siegle, Xiaoxuan Jia, Séverine Durand, Sam Gale, Corbett Bennett, Nile Graddis, Greggory Heller, Tamina K Ramirez, Hannah Choi, Jennifer A Luviano, et al. Survey of spiking in the mouse visual system reveals functional hierarchy. *Nature*, 592(7852):86–92, 2021.

[36] Motoharu Yoshida and Hatsuo Hayashi. Emergence of sequence sensitivity in a hippocampal ca3–ca1 model. *Neural Networks*, 20(6):653–667, 2007.

[37] Tom M George, William de Cothi, Kimberly L Stachenfeld, and Caswell Barry. Rapid learning of predictive maps with stdp and theta phase precession. *Elife*, 12:e80663, 2023.

[38] Simon Nikolaus Weber and Henning Sprekeler. Learning place cells, grid cells and invariances with excitatory and inhibitory plasticity. *Elife*, 7:e34560, 2018.

[39] Aran Nayebi, Sanjana Srivastava, Surya Ganguli, and Daniel L Yamins. Identifying learning rules from neural network observables. *Advances in Neural Information Processing Systems*, 33:2639–2650, 2020.

[40] Jacob Portes, Christian Schmid, and James M Murray. Distinguishing learning rules with brain machine interfaces. *Advances in neural information processing systems*, 35:25937–25950, 2022.

[41] Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Advances in neural information processing systems*, 21, 2008.

[42] Chiaki Itami and Fumitaka Kimura. Developmental switch in spike timing-dependent plasticity at layers 4–2/3 in the rodent barrel cortex. *Journal of Neuroscience*, 32(43):15000–15011, 2012.

[43] L Abbott and S Nelson. Synaptic plasticity: taming the beast. *Nature Neuroscience*, Jan 2000.

[44] Yanis Inglebert, Johnatan Aljadeff, Nicolas Brunel, and Dominique Debanne. Altered spike timing-dependent plasticity rules in physiological calcium. *bioRxiv*, 2020.

[45] Wulfram Gerstner, Werner Kistler, Richard Naud, and Liam Paninski. *Neuronal Dynamics*. Cambridge University Press, Cambridge, 2014.

[46] Christine Grienberger and Jeffrey C Magee. Entorhinal cortex directs learning-related changes in ca1 representations. *Nature*, 611(7936):554–562, 2022.

[47] Alexandre Payeur, Jordan Guerguiev, Friedemann Zenke, Blake A Richards, and Richard Naud. Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits. *Nature Neuroscience*, pages 1–10, 2021.

[48] Friedemann Zenke and Surya Ganguli. Superspike: Supervised learning in multilayer spiking neural networks. *Neural Comput.*, 30(6):1514–1541, 2018.

[49] Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019.

[50] Lee Susman, Naama Brenner, and Omri Barak. Stable memory with unstable synapses. *Nature communications*, 10(1):4441, 2019.

[51] Yann LeCun, Corinna Cortes, Chris Burges, et al. Mnist handwritten digit database, 2010.

[52] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 357–366, 2021.

[53] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4804–4814, 2022.