Stable task information from an unstable neural population

Adrianna R. Loback[1,†], Michael E. Rule[1,†], Dhruva V. Raman[1],
Laura N. Driscoll[2], Christopher D. Harvey[3], Timothy O'Leary[1*]

1 *Department of Engineering, University of Cambridge, Cambridge, CB2 1PZ, UK*
2 *Department of Electrical Engineering, Stanford University, Stanford, CA, CA 94305-9505, USA*
3 *Department of Neurobiology, Harvard Medical School, Boston, MA, 02115, USA*

[†]Equal Contribution, [*]Correspondence (tso24@cam.ac.uk), Dated: October 3, 2019

Abstract

Over days and weeks, neurons in mammalian sensorimotor cortex have been found to continually change their activity patterns during performance of a learned sensorimotor task, with no detectable change in behaviour. This challenges classical theories of neural circuit function and memory, which posit that stable engrams underlie stable learned behavior [1, 2]. Using existing experimental data we show that a simple linear readout can accurately recover behavioural variables, and that fixed linear weights can approximately decode behaviour over many days, despite significant changes in neural tuning. This implies that an appreciable fraction of ongoing activity reconfiguration occurs in an approximately linear subspace of population activity. We quantify the amount of additional plasticity that would be required to compensate for reconfiguration, and show that a biologically plausible local learning rule can achieve good decoding accuracy with physiologically achievable rates of synaptic plasticity.

Introduction and Results

Several recent experimental studies show that the mapping between neural activation and behavior reconfigures over days in brain areas that are critical for performing a task, even when the task is fully learned [3–5]. One of our labs recently observed [4] that single-neuron activation across hundreds of neurons in the posterior parietal cortex (PPC) change over the course of weeks during a fixed, learned T-maze task. This reconfiguration continued for the duration of the experiment (20-52 days), revealing that a brain area crucial for executing a sensorimotor task continually reconfigures the relationship between neural activity and external task variables over several weeks.

Given these observations, it is unclear how activity in PPC can be used by the brain to perform a familiar task reliably. To address this question we reanalyzed data from a previous set of experiments performed by one of our labs [4]. Briefly, this dataset consists of optical recordings of calcium activity in populations of hundreds of PPC neurons during repeated trials of a virtual reality T-maze task (1a). Mice were trained to associate a visual cue at the start of the maze with turning left/right at a T-junction. The recordings were obtained once the mice consistently performed at 90% accuracy or above. Further details are provided in Methods and full experimental details can be found in the original study [4].

Previous analyses [4] showed that PPC neurons activated at specific locations in the maze on each day. When peak activation is plotted as a function of (linearized) maze location, the recorded population tiles the maze, as shown in Figure 1b. However, maintaining the same ordering in the same population of neurons revealed a loss of sequential activity over days to weeks (top row of 1b). Nonetheless, a different subset of neurons could always be found to tile the maze at these later experimental sessions. In all cases, the same gradual loss of ordered activation was observed (second and third rows, 1b). Figure 1c shows that PPC neurons gain or lose selectivity and occasionally change tuning locations. Together, these data show that PPC neurons form a continually reconfiguring representation of a fixed, learned task.

We asked whether precise task information can be extracted from this population of neurons, despite the continual activity reconfiguration evident in these data. We began by fitting a linear decoder for each task variable of interest (animal location, heading, and velocity) for each day. This model has the form $x(t)=M^T z(t)$, where $x(t)$ is the time-binned estimate of position, velocity or heading (view angle) in the virtual maze, M is a vector of weights, and $z(t)$ is the normalized time-binned calcium fluorescence (Methods: *Decoding analyses*).

Example decoding results for two mice are shown in Fig. 2a, and summaries of decoding performance for four mice in Fig. 2b. Position, speed, and view angle can each be recovered with a separate linear model. The average mean absolute decoding error for all animals included in the analysis was 47.2 cm $\pm$8.8 cm (mean $\pm$1 standard deviation) for position, 9.6 cm/s $\pm$2.2 cm/s for speed, and 13.8°$\pm$4.0° for view angle (Methods: *Decoding analyses*).

Our choice of linear decoder can be interpreted bio-

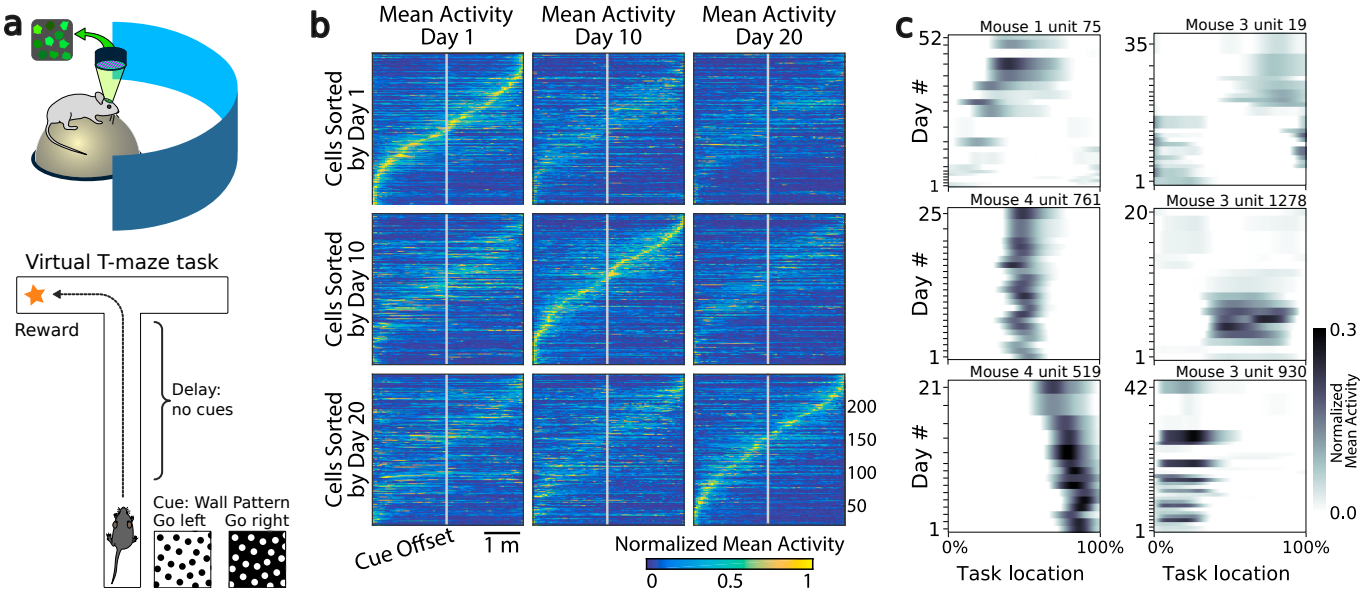

FIG. 1. **Neural population coding of spatial navigation reconfigures over time in a virtual-reality maze task** **(a)** Mice were trained to use visual cues to navigate to a reward in a virtual-reality maze; neural population activity was recorded using Ca^{2+} imaging [4]. **(b)** (Reprinted from [4]) Neurons in PPC (vertical-axes) fire at various regions in the maze (horizontal-axes). Over days to weeks, individual neurons change their tuning, re-configuring the population code. This occurs even at steady-state behavioral performance (after learning). **(c)** Each plot shows how location-averaged normalized activity changes for single-cells over weeks. Missing days are interpolated to the nearest available sessions, and both left and right turns are combined. Neurons show diverse changes in tuning over days, including instability, relocation, long-term stability, gain/loss of selectivity, advancement, and intermittent responsiveness.

logically as a single 'readout' neuron that receives input from a few hundred PPC neurons, and whose activity approximates a linear weighted sum. The fact that the linear decoder recovers behavioral variables to reasonable accuracy suggests that brain areas with sufficiently dense connectivity to PPC can extract this information via simple weighted sums.

The number of PPC neurons recorded is a subset of the total PPC population. To assess the extent to which additional neurons might improve decoding accuracy, we evaluated decoding performance of randomly drawn subsets of recorded neurons (Fig. 2c). Extrapolation of the decoding performance suggested that better performance might be possible with a larger population of randomly sampled PPC neurons than we recorded.

It is possible that a random sample of neurons misses the 'best' subset of cells for decoding task variables. When we restricted to optimal subsets of neurons we found that performance saturated at ≈30% (50-100 neurons) of the neurons recorded (Fig. 2d). However, the identity of the neurons in the optimal subset changed over days. For both mouse 3 (10 days, 104 neurons) and 4 (13 days, 126 neurons), 6% of neurons are consistently in the top 50% for all sessions. No neurons were consistently in the top 10% for all days. Biologically, this suggests that even though a subset of neurons might be sufficient for encoding task information, this subset cannot be identified from short-term recordings.

An approximately stable coding subspace exists despite reconfiguration

We next asked how a linear decoder optimized on a given day would perform on prior and subsequent days. We compared the decoding performance of models fitted on a given day with decoders optimized on data from earlier or later days. We restricted this analysis to those neurons that were identified with high confidence on all days considered. We found that decoding performance decreased as the separation between days grew (Fig 3a). This is not surprising given the extent of reconfiguration reported in the original study [4] and depicted in Fig 1.

Because task-related PPC activity is distributed in a large number of neurons, many different linear decoders can in principle achieve similar error rates due to degeneracy in the representation [6, 7]. Thus, the observed decay in performance of a decoder fit on a single day might reflect misidentification of the 'true' set of weightings that perform better over time. This would imply that the observed ongoing reconfiguration of neural activity occurs in the null space of the 'true' decoder being used internally by the brain, consistent with some proposed models [7–9].

To address this, we tested whether a single linear de-

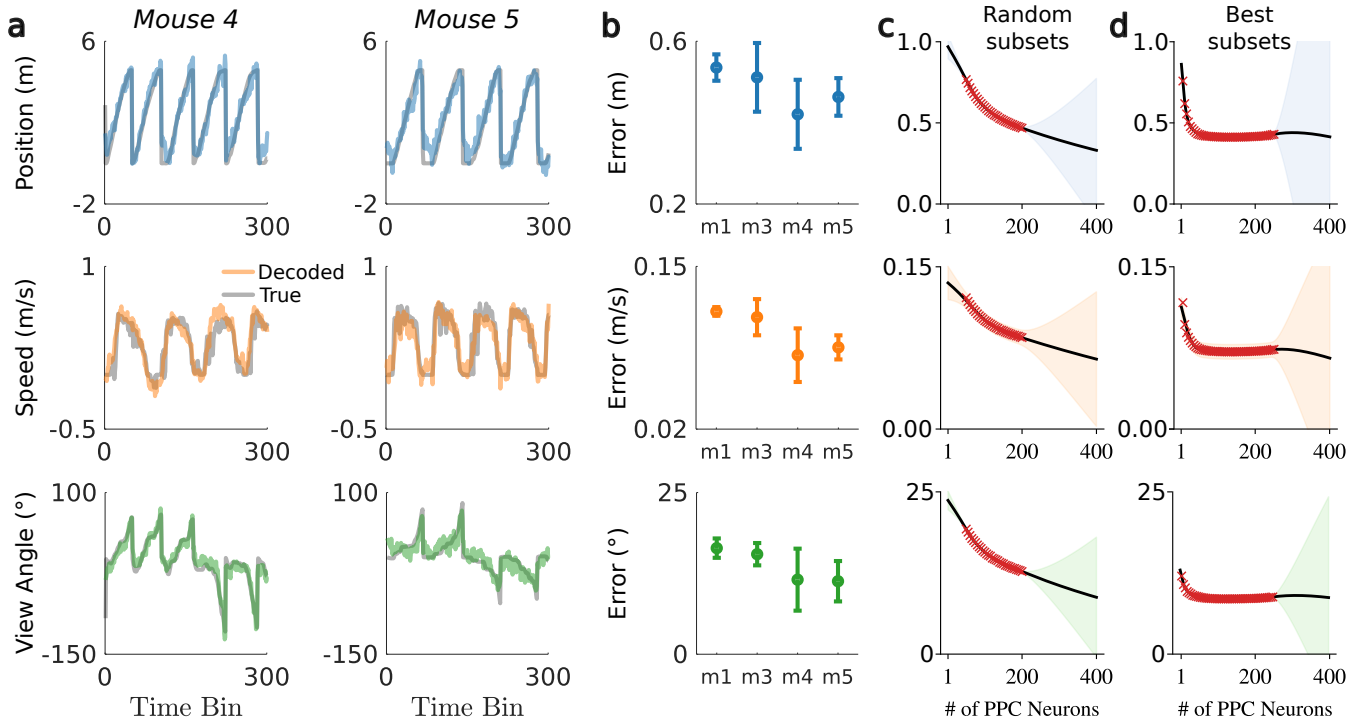

FIG. 2. **A linear decoder can extract kinematic information from PPC population activity on a single day.** **(a)** Example decoding performance for a single session for mouse 4 and 5. Grey denotes held-out test data; colors denote the prediction for the corresponding kinematic variable. **(b)** Summary of the decoding performance of the static linear model; each point denotes one mouse. Error bars denote one standard deviation over all sessions for the corresponding mouse that had at least N=200 high-confidence PPC neurons. (Results for mouse 2 are not shown, since 4no session had $\geq$ 200 identified neurons). **(c)** Gaussian Process (GP) extrapolation of the MAE performance of the static linear decoder for decoding position as a function of the number of PPC neurons, done via Gaussian process regression (see Methods). Red x's denote data; solid black line denotes the mean function of the GP. **(d)** Same as panel (c), but where the neurons have been ranked such that the "best" subset of size $1 \leq K \leq N$ is chosen, selected by greedy search based on explained variance (Methods: Best K-Subset Ranking).

coder could be optimized across multiple days. We fitted linear decoders using concatenated data from different days using the same subset of PPC neurons (Fig. 3b). The requirement to have large overlapping subsets of cells over many days reduced the number of animals that could be used to two. In both animals we found that fixed multiple-day linear decoders could recover somewhat accurate task variable information in spite of ongoing changes in PPC tuning. However, the average performance of the multiple-day decoders was detectably worse than single-day linear decoders for each day (Fig. 3c).

The existence of a fixed, approximate decoder implies a degenerate representation of task variables in the population activity of PPC neurons. In other words, there is a family of linear decoders that can achieve reasonably accurate performance while allowing weights to vary in some region of weight space. This situation is illustrated in Figure 3c, which depicts regions of good performance of single-day linear decoders as ellipsoids. The existence of a reasonably good concatenated decoder implies that these ellipsoids intersect over several days.

We asked how much synaptic plasticity would be needed to more accurately compensate for representational drift and achieve the performance of single-day decoders. We approached this without invoking a specific plasticity rule, by simultaneously training separate linear decoders for each day over several sessions (mouse 3: 7 days, mouse 4: 10 days), and penalizing the magnitude of weight change between days (Fig. 3d, Methods: *Concatenated and constrained analyses*). By varying the strength of the weight change penalty we interpolated between the concatenated decoder (no weight changes) and the single-day decoders (optimal weights for each day). The result of this is shown in Figure 3e. Performance improves rapidly once small amounts of synaptic plasticity are permitted ($\approx$25% per session). Thus, relatively modest amounts of synaptic plasticity would be needed to keep encoding consistent with changes in representation.

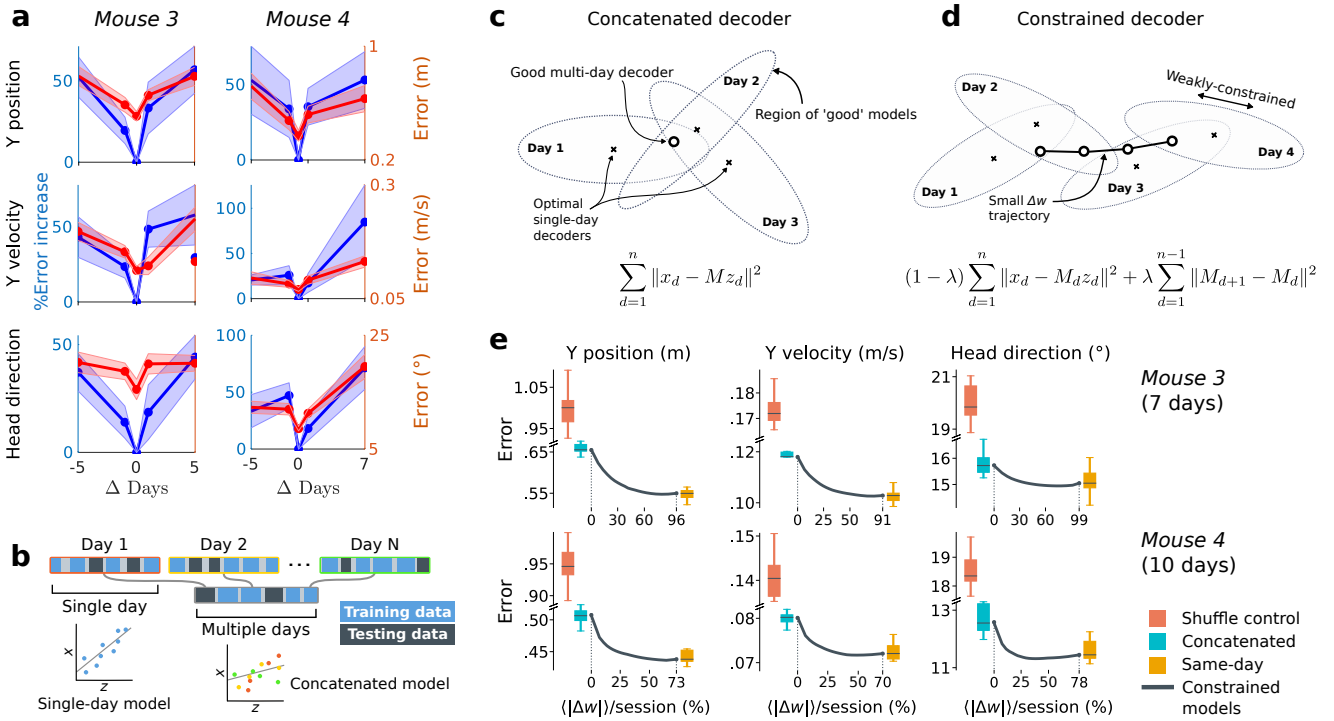

FIG. 3. **Single-day decoders generalize poorly to previous and subsequent days, but multi-day decoders exist with good performance. (a)** Blue: % increase in error over the optimal decoder for the testing day (mouse 3, 136 neurons; mouse 4, 166 neurons). Red: Mean absolute error for decoders trained on a single day ('0') and tested on past/future days. **(b)** Schematic of procedure for fitting multiple day models: trials over multiple days are concatenated (Methods). **(c)** Although the single-day optimal decoders may change over time, a stable subspace with good performance may exist. We assess this by training a single decoder M applied to multiple days $d \in 1..D$. **(d)** Over longer periods of time a stable encoding subspace may change. We assess this by training a separate decoder M_d for each day, while also minimizing the change in weights across days. The parameter λ controls how strongly we constrain weight changes across days (Methods). **(e)** Decoding error on concatenated data (cyan) is slightly but significantly worse than the average error of the best single day decoder (ochre) (Mann-Whitney U test, $p<0.01$) (Boxes: inner 50% of data, horizontal lines: median, whiskers: 5th, 95th percentiles). Both single day and multiple day decoders perform better than a shuffled control (red). Black traces show performance of aonstrained decoder for varying strength of weight change penalty.

Biologically plausible weight adjustment can compensate for ongoing reconfiguration of PPC activity

The results in Figure 3e suggest that small amounts of synaptic plasticity could compensate drift, but do not suggests a biologically plausible mechanism for compensation. Could neurons, in practice, track slow reconfiguration using locally available signals? To test this, we used an adaptive linear neuron model first proposed by Widrow & Hoff in 1962 [10–12] on mouse 4 (Methods). This Least Mean Squares (LMS) model uses prediction error of a linear readout to update weights continuously (Fig. 4a). This is biologically plausible because it only requires each synapse to access its current weight and recent prediction error. To test how well this model could track drift with small weight changes, we limited the amount of weight change per time-step (Methods: *Online LMS algorithm*).

Fig. 4b illustrates that, by varying the limits on weight change, the LMS model can achieve online decoding results analogous to the offline constrained decoders discussed previously. Decoding error improves for larger rates of weight change, approaching the performance of a concatenated, fixed, decoder trained and tested over ten days. Over the ten-day span evaluated, LMS can keep a linear decoder in register with the slow representational drift observed (Fig. 4c).

In our analysis, learning rates of a few percent per minute were needed to track long-term drift. This included fast fluctuations in synaptic weights superimposed over a slowly-varying mean trend. For example, for position decoding from mouse 4 (Fig. 4d), an average plasticity rate of 2% per minute kept decoding error comparable to that of the concatenated decoder. This amounted to an accumulated weight change averaging 7.67% per session.

These results suggest that small weight changes could

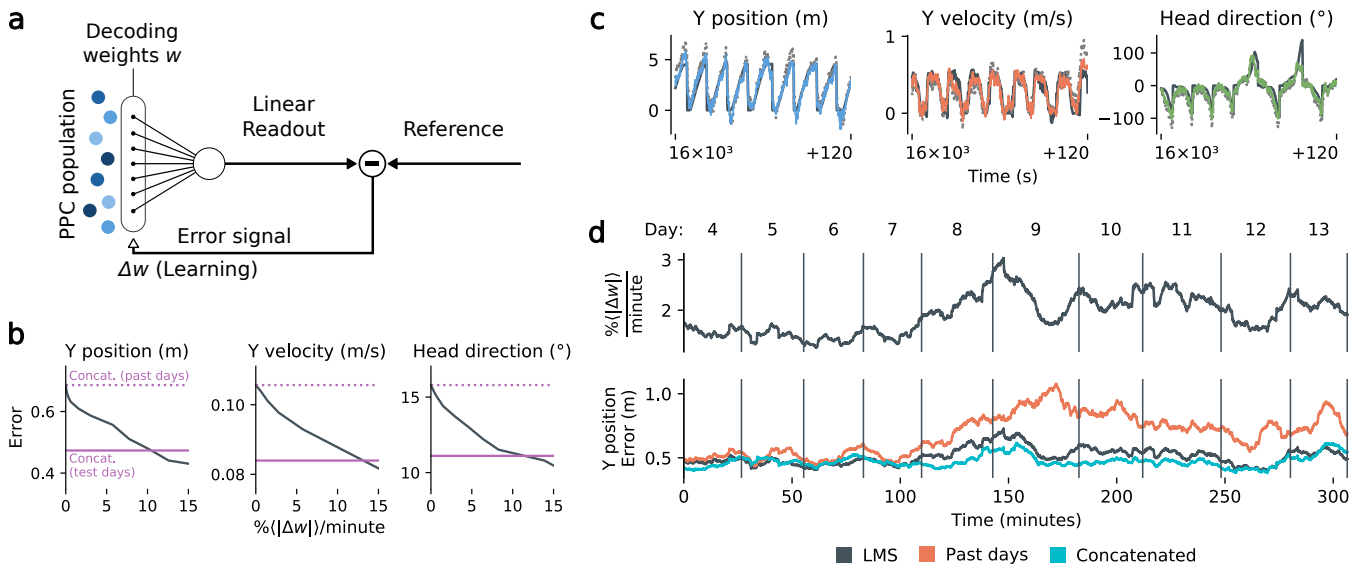

FIG. 4. **Local, adaptive decoders can track representational drift over multiple days. (a)** The Least Mean-Squares (LMS) algorithm learns to linearly decode a target kinematic variable based on error feedback. Continued online learning can track gradual reconfiguration in population representations. **(b)** Plots show the trade-off between the average rate of weight change (horizontal axis) and the mean absolute decoding error (vertical axis) of the LMS algorithm, for three kinematic variables (Mouse 4, 144 units; Methods: Online LMS algorithm). The dashed line reflects the decoding generalization error for a decoder trained on only the previous two sessions. The solid line reflects the cross-validated performance of a concatenated decoder over ten testing days. As the rate of synaptic plasticity is increased, LMS achieves error rates comparable to the concatenated decoder (solid horizontal line). **(c)** Example LMS decoding results for three kinematic variables. Ground truth is plotted in black, and LMS estimate in color. Sample traces are taken from the ninth day. Dashed traces indicate the performance of the decoder without ongoing re-training. **(d)** *(top)* Average percent weight-change while decoding position using the LMS algorithm, averaged over ten-second intervals. The horizontal axis reflects time, and vertical black bars mark the boundaries between days. The average weight change is 1.92% per minute, with an accumulated weight change averaging 7.67% per session. *(bottom)* LMS (black) is able to track population reconfiguration with decoding errors comparable to that of the concatenated decoder (cyan). Without ongoing learning, the performance of the initial decoder trained on the previous two days degrades (red). Error traces have been averaged over ten-second intervals.

track representational drift in practice, but are only a weak proxy for the rates of spine plasticity that one might expect *in vivo*. These analyses were done for ≈100-200 neurons, at least an order of magnitude less than the typical number of inputs to a pyramidal cell in cortex. We thus interpret these results as an approximate upper-bound on the amount of plasticity required for a linear decoder. We expect better performance with lower rates of plasticity for larger neuronal populations. Additionally, more sophisticated plasticity rules might alter synaptic weights on both fast and slow timescales to compensate rapid fluctuations in activity while tracking the slow component of drift.

Discussion

Several theories have been presented for how stable behavior could be maintained despite ongoing changes in connectivity and neural activity [6, 8, 13, 14]. Here we show—in principle—that a biologically plausible readout of neural activity in PPC neurons can decode task information while compensating for ongoing changes in how this information is represented. If our modeling assumptions are taken literally, this would suggest that a single unit with connections to ∼100 PPC neurons can accurately decode task information with modest changes in synaptic weights over several days.

It is also possible that long-term representations are supported by a core subset of neurons with stable tuning and connectivity. Our results show that it is difficult to identify this subset based on single-day recordings. Due to redundancy, the number of 'good' models (and potentially good subsets of neurons) is large, and informativeness on a given day does not imply a stable tuning on subsequent days.

We focused on association areas in parietal cortex, but other studies have found potentially stable codes in motor areas [15]. If this is the case, stable behavior might be supported by long-term representations outside of parietal cortex. However, the question of why representations of habitual tasks should reconfigure in parietal cortex remains.

Consistent with these observations, studies have shown

that connectivity in mammalian cortex is surprisingly dynamic. Connections between neurons change on a timescale of hours to days with a small number of stable connections [3, 16–18]. Some of this turnover likely reflects ongoing learning and unavoidable fluctuations in biological systems. We suggest that another component of this ongoing plasticity maintains congruent representations of information across different neural circuits. Such ongoing maintenance might be necessary in a distributed, adaptive system such as the brain, with multiple areas continually learning new representations while maintaining old information. How this is achieved is the subject of intense debate [19]. Based on our study, we hypothesize that neural circuits have continual access to two kinds of error signals. One kind should reflect mismatch between internal representations and external task variables, and another should reflect prediction mismatch between one neural circuit and another.

Acknowledgements

We thank Fulvio Forni, Yaniv Ziv and Alon Rubin for in depth discussions. This work was supported by the Human Frontier Science Program (RGY0069), ERC Starting Grant (StG FLEXNEURO 716643) and grants from the NIH (NS089521, MH107620, NS108410)

Author Contributions

Performed research A.R.L., M.E.R.; designed the study: T.O., D.V.R. and C.D.H; analysed, contributed and consulted on experimental data: L.D., C.D.H.; interpreted results: A.R.L., M.E.R, D.V.R, L.D., C.D.H., T.O.; wrote the manuscript: M.E.R., C.D.H., T.O.; drafted and revised the manuscript: A.R.L., M.E.R., L.D., C.D.H, T.O.

Declaration of Interests

The authors declare no conflict of interest.

Methods

Data acquisition

The behavioral and two-photon calcium imaging data analyzed here was provided by the Harvey lab. Details regarding the experimental subjects and methods are provided in [4].

Virtual reality task

Details of the virtual reality environment, training protocol, and fixed association navigation task are described in [4]. In brief, Virtual reality environments were constructed and operated using MATLAB-based ViRMEn software (Virtual Reality Mouse Engine) [20]. Data were obtained from mice that had completed the 4-8 week training program (reaching a behavioral performance of >80% correct for 2-3 consecutive days) for the two-alternative forced choice T-maze task. The length of the virtual reality maze was fixed to have a total length of 4.5 m. The colored walls were either black with white dots or white with black dots, followed by a gray striped 'delay' segment (2.25 m long) that was identical across trial types.

Data preparation and pre-processing

Raw Ca^{2+} fluorescence videos (sample rate=5.3 Hz) were corrected for motion artefacts, and individual sources of Ca^{2+} fluorescence were identified and extracted [4]. Processed data consisted of normalized Ca^{2+} fluorescence transients ("$\Delta F/F$") and behavioral variables (mouse position, view angle, and velocity). Intertrial intervals (ITIs) were removed for all subsequent analyses. For offline decoding, we considered only correct trials, and all signals were centered to zero-mean on each trial as a pre-processing step.

QUANTIFICATION AND STATISTICAL ANALYSIS

Decoding analyses

We decoded kinematics time-series $\mathbf{x}=\{x_1, ..., x_T\}$ from the vector of instantaneous neural population activity $\mathbf{z}=\{z_1, .., z_T\}$, using a linear decoder with a fixed set of weights M:

$$\hat{\mathbf{x}} = M\mathbf{z}. \tag{1}$$

We used the ordinary least-squares (OLS) solution for M, which minimizes the squared (L2) prediction error $\varepsilon=\|\mathbf{x}-M\mathbf{z}\|^2$ over all time-points. For the 'same-day' analyses, we optimize a separate M_d for each day d (Fig. 2), restricting analysis to sessions with at least 200 identified units. We assessed decoding performance using 10-fold cross-validation, and report the mean absolute error, defined as $\langle | \mathbf{x}-\hat{\mathbf{x}} | \rangle$.

Best K-Subset Ranking

For Fig. 2d, we ranked cells in order of explained variance using a greedy algorithm. Starting with the most predictive cell, we iteratively added the next cell that minimized the MSE under ten-fold cross-validated linear decoding. To accelerate this procedure, we pre-computed the mean and covariance structure for training and testing datasets. MSE fits and decoding performance can be computed directly from these summary statistics, accelerating the several thousand evaluations required for greedy selection.

Concatenated and constrained analyses

For both the concatenated and constrained analyses (Fig. 3b-e), we used the set of identified neurons common to all included days. In the concatenated analyses (Fig. 3b,c,e), we solved for a single decoder M_c for all days:

$$\varepsilon = \sum_{d=1}^{n} \|\mathbf{x}_d - M_c \mathbf{z}_d\|^2, \tag{2}$$

where ε denotes the quadratic objective function to be minimized. In the constrained analysis (Fig. 3d,e), we optimized a series of different weights $\mathbf{M}=\{M_1,..,M_D\}$ for each day $d\in 1..D$, and added an adjustable L2 penalty λ on the change in weights across days. This problem reduces to the 'same-day' analysis for λ=0, and approaches the concatenated decoder as λ approaches 1:

$$\varepsilon = (1-\lambda)\sum_{d=1}^{n} \|\mathbf{x}_d - M_d \mathbf{z}_d\|^2 + \lambda \sum_{d=1}^{n-1} \|M_{d+1} - M_d\|^2. \tag{3}$$

Extrapolation via GP regression

To qualitatively assess whether decoding performance saturates with the available number of recorded neurons, we computed decoding performance on a sequence of random subsets of the population of various sizes (Fig. 2c,d). Results for all analyses are reported as the average over 20 permutations of the choice of neurons to include, and over all sessions that had at least N=150 units. Gaussian process (GP) regression was implemented in Python, using a combination of a Matérn kernel and an additive white noise kernel. Kernel parameters were optimized via maximum likelihood (Scikit-learn, [21]).

Online LMS algorithm

The Least Mean-Squares (LMS) algorithm is an online approach to training and updating a linear decoder, and corresponds to stochastic gradient-descent (Fig. 4a). The algorithm was originally introduced in [10–12]. Briefly, LMS computes a prediction error for an affine decoder (i.e. a linear decoder with a constant offset feature or bias parameter) at every time-point, which is then used to update the decoding weights. We analyzed twelve contiguous sessions from mouse 4 (144 units in common), and initialized the decoder by training on the first two sessions using OLS.

To demonstrate that LMS could perform well using physiologically plausible rates of synaptic plasticity, we limited the rate of weight change $-\delta \leq \Delta w \leq \delta$. We explored a variety of limits δ, and optimized the learning rate for each δ via grid-search. By varying δ, we obtained a trade-off (Fig. 4b) between the rate of weight changes and the decoding error, with the most rapid weight updates (∼10%/minute) approaching the performance of the offline decoders. In Fig. 4d, we selected an example with δ=5.4×10^{-4}/sample, and with a learning rate of η=2.2×10^{-4}. For position decoding, this corresponded to an average weight change of 1.92% per minute, and an accumulated weight change averaging 7.67% per session.

DATA AND CODE AVAILABILITY

Datasets recorded in Driscoll et al. [4] are available upon request from CDH. The analysis code generated during this study are available on Github at github.com/michaelerule/Loback_et_al.

[1] K. Ganguly and J. M. Carmena, "Emergence of a stable cortical map for neuroprosthetic control," *PLoS biology*, vol. 7, no. 7, p. e1000153, 2009.

[2] S. Tonegawa, M. Pignatelli, D. S. Roy, and T. J. Ryan, "Memory engram storage and retrieval," *Current opinion in neurobiology*, vol. 35, pp. 101–109, 2015.

[3] A. Attardo, J. E. Fitzgerald, and M. J. Schnitzer, "Impermanence of dendritic spines in live adult ca1 hippocampus," *Nature*, vol. 523, no. 7562, p. 592, 2015.

[4] L. N. Driscoll, N. L. Pettit, M. Minderer, S. N. Chettih, and C. D. Harvey, "Dynamic reorganization of neuronal activity patterns in parietal cortex," *Cell*, vol. 170, no. 5, pp. 986–999, 2017.

[5] Y. Ziv, L. Burns, E. Cocker, E. Hamel, K. Ghosh, L. Kitch, A. El Gamal, and M. Schnitzer, "Long-term dynamics of ca1 hippocampal place codes," *Nature Neuroscience*, vol. 16, pp. 264–266, 2013.

[6] U. Rokni, A. G. Richardson, E. Bizzi, and H. S. Seung, "Motor learning with unstable neural representations," *Neuron*, vol. 54, no. 4, pp. 653–666, 2007.

[7] M. T. Kaufman, M. M. Churchland, S. I. Ryu, and K. V. Shenoy, "Cortical activity in the null space: permitting preparation without movement," *Nature neuroscience*, vol. 17, no. 3, p. 440, 2014.

[8] S. Druckmann and D. B. Chklovskii, "Neuronal circuits underlying persistent representations despite time varying activity," *Current Biology*, vol. 22, no. 22, pp. 2095–2103, 2012.

[9] A. Ni, D. Ruff, J. Alberts, J. Symmonds, and M. Cohen, "Learning and attention reveal a general relationship between population activity and behavior," *Science*, vol. 359, no. 6374, pp. 463–465, 2018.

[10] B. Widrow and M. E. Hoff, "Adaptive switching circuits," tech. rep., Stanford Univ Ca Stanford Electronics Labs, 1960.

[11] B. Widrow and M. E. Hoff, "Associative storage and retrieval of digital information in networks of adaptive 'neurons'," in *Biological Prototypes and Synthetic Systems*, pp. 160–160, Springer, 1962.

[12] B. Widrow and S. Stearns, *Adaptive Signal Processing*. Prentice-Hall, Inc., 1985.

[13] R. Ajemian, A. D'Ausilio, H. Moorman, and E. Bizzi, "A theory for how sensorimotor skills are learned and retained in noisy and nonstationary neural circuits," *Proceedings of the National Academy of Sciences*, vol. 110, no. 52, pp. E5078–E5087, 2013.

[14] D. Kappel, R. Legenstein, S. Habenschuss, M. Hsieh, and W. Maass, "A dynamic connectome supports the emergence of stable computational function of neural circuits through reward-based learning," *eNeuro*, vol. 5, no. 2, pp. ENEURO–0301, 2018.

[15] A. K. Dhawale, R. Poddar, S. B. Wolff, V. A. Normand, E. Kopelowitz, and B. P. Ölveczky, "Automated long-term recording and analysis of neural activity in behaving animals," *Elife*, vol. 6, p. e27702, 2017.

[16] A. J. Holtmaat, J. T. Trachtenberg, L. Wilbrecht, G. M. Shepherd, X. Zhang, G. W. Knott, and K. Svoboda, "Transient and persistent dendritic spines in the neocortex in vivo," *Neuron*, vol. 45, no. 2, pp. 279–291, 2005.

[17] A. Minerbi, R. Kahana, L. Goldfeld, M. Kaufman, S. Marom, and N. E. Ziv, "Long-term relationships between synaptic tenacity, synaptic remodeling, and network activity," *PLoS biology*, vol. 7, no. 6, p. e1000136, 2009.

[18] A. Holtmaat and K. Svoboda, "Experience-dependent structural synaptic plasticity in the mammalian brain," *Nature Reviews Neuroscience*, vol. 10, no. 9, p. 647, 2009.

[19] M. E. Rule, T. O'Leary, and C. D. Harvey, "Causes and consequences of representational drift," *Current opinion in neurobiology (in press)*, 2019.

[20] C. Harvey, P. Coen, and D. Tank, "Choice-specific sequences in parietal cortex during a virtual-navigation decision task," *Nature*, vol. 484, pp. 62–68, 2012.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.