bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

Stable task information from an unstable neural population

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

1 *Department of Engineering, University of Cambridge, Cambridge, Cambridgeshire, 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*
[*]correspondence: (tso24@cam.ac.uk)

(Dated: May 12, 2020)

Abstract

Over days and weeks, neural activity representing an animal's position and movement in sensorimotor cortex has been found to continually reconfigure or 'drift' during repeated trials of learned tasks, with no obvious change in behavior. This challenges classical theories which assume stable engrams underlie stable behavior. However, it is not known whether this drift occurs systematically, allowing downstream circuits to extract consistent information. We show that drift is systematically constrained far above chance, facilitating a linear weighted readout of behavioural variables. However, a significant component of drift continually degrades a fixed readout, implying that drift is not confined to a null coding space. We calculate the amount of plasticity required to compensate drift independently of any learning rule, and find that this is within physiologically achievable bounds. We demonstrate that a simple, biologically plausible local learning rule can achieve these bounds, accurately decoding behavior over many days.

Keywords: neural plasticity, parietal cortex, population coding, spatial navigation, computational model, learning and memory

Introduction and Results

A core principle in neuroscience is that behavioral variables are represented in neural activity. Such representations must be maintained to retain learned skills and memories. However, recent work has challenged the idea of long-lasting neural codes [1]. In our recent work [2], we found that neural activity-behavior relationships in individual posterior parietal cortex (PPC) neurons continually changed over many days during a repeated virtual navigation task. Similar 'representational drift' has been shown in other neocortical areas and hippocampus [3–5]. Importantly, these studies showed that representational drift is observed in brain areas essential for performing the task long after the task has been learned.

These experimental observations raise the major question of whether drifting representations are fundamentally at odds with the storage of stable memories of behavioral variables (e.g. 6, 7). Theoretical work has proposed that a consistent readout of a representation can be achieved if drift in neural activity patterns occurs in dimensions of population activity that are orthogonal to coding dimensions - in a 'null coding space' [8–11]. This can be facilitated by neural representations that consist of low-dimensional dynamics distributed over many neurons [12–16]. Redundancy could therefore permit substantial reconfiguration of tuning in single cells without disrupting neural codes [9, 17–20]. However, the extent to which drift is confined in such a null coding space remains an open question.

Purely random drift, as would occur if synaptic strengths and other circuit parameters follow independent random walks, would eventually disrupt a population code. Several studies have provided evidence that cortical synaptic weights and synaptic connections exhibit statistics that are consistent with a purely random process [21–23]. Indeed, our previous experimental findings reveal that drift includes cells that lose representations of task relevant variables, suggesting that some component of drift affects coding dimensions [2].

Together, these observations raise fundamental questions that have not been directly addressed with experimental data, and which we address here. First, to what extent can ongoing drift in task representations be confined to a null coding space over extended periods while maintaining an accurate readout of behavioural variables in a biologically plausible way? Second, how might we estimate how much additional ongoing plasticity (if any) would be required to maintain a stable readout of behavioural variables, irrespective of specific learning rules? Third, is such an estimate of ongoing plasticity biologically feasible for typical levels of connectivity, and typical rates of change observed in synaptic strengths? Fourth, can a local, biologically plausible plasticity mechanism tune readout weights to identify a maximally stable coding subspace and compensate any residual drift away from this subspace?

We addressed these questions by modelling and analysing data from **(author?)** [2]. This dataset consists of optical recordings of calcium activity in populations of hundreds of neurons in Posterior Parietal Cortex (PPC) 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 or right at a T-junction. Behavioral performance and

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

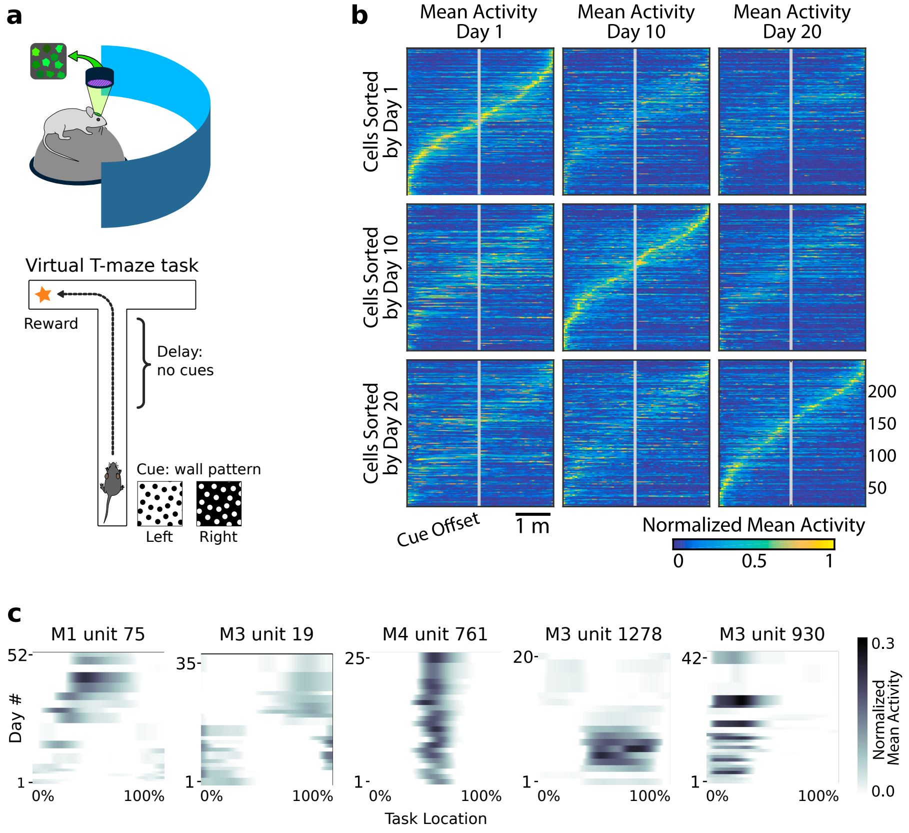

FIG. 1. **Neural population coding of spatial navigation reconfigures over time in a virtual-reality maze task (a)** Mice were trained to use visual cues to navigate to a reward in a virtual-reality maze; neural population activity was recorded using Ca^{2+} imaging [2]. **(b)** (Reprinted from [2]) Neurons in PPC (vertical axes) fire at various regions in the maze (horizontal axes). Over days to weeks, individual neurons change their tuning, reconfiguring 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, and intermittent responsiveness.

kinematic variables were stable over time with some per-session variability (mouse 4 exhibited a slight decrease in forward speed; Fig. 2-S1). Full experimental details can be found in the original study.

Previous studies identified planning and choice-based roles for PPC in the T-maze task [24], and stable decoding of such binary variables was explored in [2]. However, in primates PPC has traditionally been viewed as containing continuous motor-related representations [25–27], and recent work [28, 29] has confirmed that PPC has an equally motor-like role in spatial navigation in rodents [30]. It is therefore important revisit these data in the context of continuous kinematics encoding.

Previous analyses 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 in 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.

PPC representations facilitate a linear readout

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 $x(t){=}M^{\top}z(t)$, 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).

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 ±8.8 cm (mean ±1 standard deviation) for position, 9.6 cm/s ±2.2 cm/s for speed, and 13.8°±4.0° for view angle (Methods).

We chose a linear decoder specifically because it can be interpreted biologically 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 a 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 to-

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

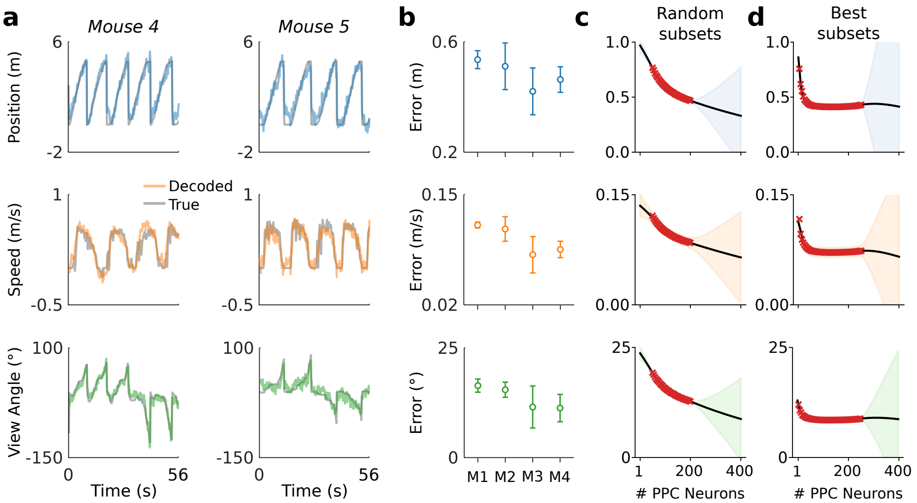

FIG. 2. **A linear decoder can extract kinematic information from PPC population activity on a single day. (a)** Example decoding performance for a single session for mice 4 and 5. Grey denotes held-out test data; colors denote the prediction for the corresponding kinematic variable. **(b)** Summary of the decoding performance on single days; each point denotes one mouse. Error bars denote one standard deviation over all sessions that had at least N=200 high-confidence PPC neurons for each mouse. (Mouse 2 is excluded due to an insufficient number of isolated neurons). Chance level is ~1.5 m for forward position, and varies across subjects for forward velocity (~0.2-0.25 m/s) and head direction (~20-30 deg). **(c)** Extrapolation of the performance of the static linear decoder for decoding position as a function of the number of PPC neurons, done via Gaussian process regression (Methods). Red "×" marks denote data; solid black line denotes the inferred mean of the GP. Shaded regions reflect $\pm 1.96\sigma$ Gaussian estimates of the 95th and 5th percentiles. **(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).

tal PPC population. To assess whether 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 improved rapidly up to ~30 neurons and saturated at ~30% (50-100 neurons) of the neurons recorded (Fig. 2d). On a given day task variables could be decoded well with relatively few (~10) neurons. However, the identity of the neurons in this optimal subset changed over days. For all subjects, no more than 1% of cells were consistently ranked in the top 10%, an no more than 13% in the top 50%. We confirmed that this instability was not due to under-regularization in training (Methods).

Of the neurons with strong location tuning, the previous study [2] found that 60% changed their location tuning over two weeks and a total of 80% changed over the 30 day period examined. We find that even the small remaining 'stable' subset of neurons exhibited daily variations in their Signal-to-Noise Ratio (SNR) with respect to task decoding, consistent with other studies [31]. For example, no more than 8% of neurons that were in the top 25% in terms of tuning-peak stability were also consistently in the top 25% in terms of SNR for all days. If a neuron becomes relatively less reliable, then the weight assigned may become inappropriate for decoding. This affects our analyses, and would also physiologically affect a downstream neuron with fixed synaptic weights.

Representational drift is systematic and significantly degrades a fixed readout

Naively fitting a linear model to data from any given day shows that behavioural variables are encoded in a way that permits a simple readout, but there is no guarantee that this readout will survive long-term drift in the neural code. To illustrate this 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 unsurprising given the extent of reconfiguration reported in the original study [2] and depicted in Fig 1. Furthermore, because task-related PPC activity is distributed over many neurons, many different linear decoders can achieve similar error rates due to the degeneracy in the representation [8, 12, 18]. Since the directions in population activity used for inter-area communication might differ from the directions that maximally encode stimulus information in the local population [19, 32], single-day decoders might overlook a long-term stable subspace used for encoding and communication. This motivates the question of whether a drift-invariant linear decoder exists and whether its existence is biologically plausible.

To address this, we tested the performance of a single lin-

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

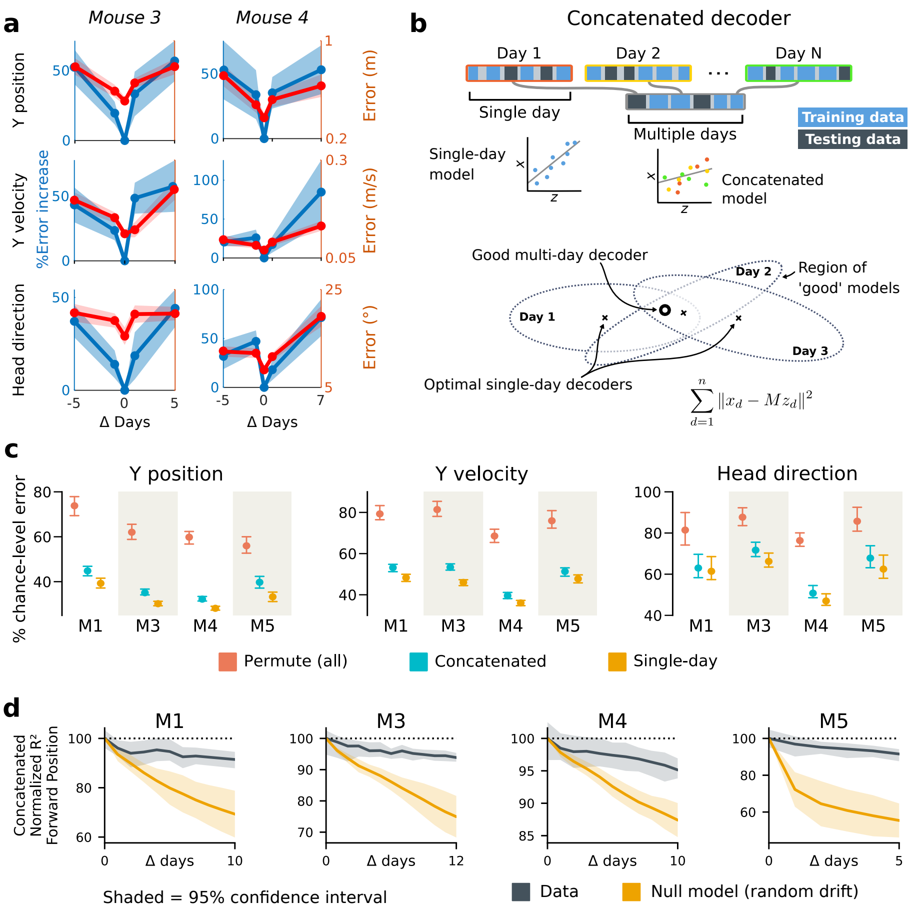

FIG. 3. **Single-day decoders generalize poorly to previous and subsequent days, but multi-day decoders exist with good performance. (a)** Blue: % increase in error over the optimal decoder for the testing day (mouse 3, 136 neurons; mouse 4, 166 neurons). Red: Mean absolute error for decoders trained on a single day ('0') and tested on past/future days. **(b)** Fixed decoders M for multiple days $d\in 1\ldots D$ ('concatenated decoders') are fit to concatenated excerpts from several sessions. The inset equation reflects the objective function to be minimized (Methods). Due to redundancy in the neural code, many decoders can perform well on a single day. Although the single-day optimal decoders vary, a stable subspace with good performance can exist. **(c)** Concatenated decoders (cyan) perform slightly but significantly worse than single-day decoders (ochre; Mann-Whitney U test, $p<0.01$). They also perform better than expected if neural codes were unrelated across days (permutation tests; red). Plots show the mean absolute decoding error as a percent of the chance-level error (points: median, whiskers: 5^{th}-95^{th}%). Chance-level error was estimated by shuffling kinematics traces relative to neural time-series (mean of 100 samples). For the permutation tests, 100 random samples were drawn with the neuronal identities randomly permuted. **(d)** Plots show the rate at which concatenated-decoder accuracy (normalized R^2) degrades as the number of days increase. Concatenated decoders (black) degrade more slowly than expected for random drift (ochre). Shaded regions reflect the inner 95% of the data (generated by resampling for the null model). The null model statistics are matched to the within- and between-day variance and sparsity of the experimental data for each animal (Methods).

ear decoder optimized across data from multiple days. We concatenated data from different days using the same subset of PPC neurons (Fig. 3b). In all four subjects, we found that such fixed multiple-day linear 'concatenated' decoders could recover accurate task variable information despite ongoing changes in PPC neuron tuning. However, the average performance of the multiple-day decoders was significantly 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 recover behavioral variables while allowing weights to vary in some region of weight space. This situation is illustrated in Figure 3b, which depicts regions of good performance of single-day linear decoders as ellipsoids. The existence of an approximate concatenated decoder implies that these ellipsoids intersect over several days for some allowable level of error in the decoder. For a sufficiently redundant neural code, one might expect to find an invariant decoder for some specified level of accuracy even if the underlying code drifts. However, there are many qualitative ways in which drift can occur in a neural code: it could resemble a random walk, as some studies suggest [21–23], or there could be a systematic component. Is the accuracy we observe in the concatenated decoder expected for a random walk? In all subjects, we found that a concatenated decoder performed substantially better on experimental data than on randomly drifting synthetic data with matched sparseness and matched within/between-session variability (Fig. 3d). This suggests that the drift in the neural data is not purely random.

We further investigated the dynamics of drift by quantifying the direction of changes in neural variability over time (Fig. 4c,d, Methods). We found that drift is indeed aligned above chance to within-session neural population variability. This suggests that the biological mechanisms underlying drift are in part systematic and constrained by a requirement to keep a consistent population code over time. In comparison, the projection of drift onto behavior-coding directions was small, but still above chance. This is consistent with the hypothesis that ongoing compensation might be needed for a long-term stable readout.

To quantify the systematic nature of drift further, we modified the null model to make drift partially systematic by constraining the null-model drift within a low rank subspace (Fig. 4-S1). This reflects a scenario in which only a few components of the population code change over time. We found that the performance of a concatenated decoder for low-rank drift better approximated experimental data. For

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

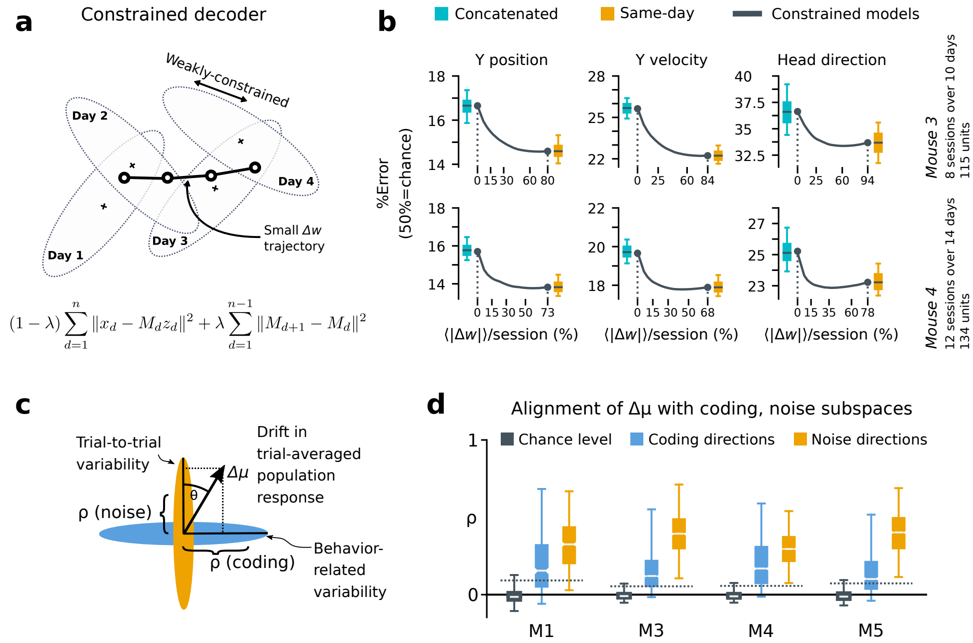

FIG. 4. **A slowly-varying component of drift disrupts the behavior-coding subspace. (a)** The small error increase when training concatenated decoders (Fig. 3) suggests that plasticity is needed to maintain good decoding in the long term. We assess the minimum rate for this plasticity by training a separate decoder M_d for each day, while minimizing the change in weights across days. The parameter λ controls how strongly we constrain weight changes across days (the inset equation reflects the objective function to be minimized; Methods). **(b)** Decoders trained on all days (cyan) perform better than chance (red), but worse than single-day decoders (ochre). Black traces illustrate the plasticity-accuracy trade-off for adaptive decoding. Modest weight changes per day are sufficient to match the performance of single-day decoders (Boxes: inner 50% of data, horizontal lines: median, whiskers: 5–95$^{\text{th}}$%). **(c)** Across days, the mean neural activity associated with a particular phase of the task changes ($\Delta\mu$). We define an alignment measure ρ (Methods) to assess the extent to which these changes align with behavior-coding directions in the population code (blue) verses directions of noise correlations (ochre). **(d)** Drift is more aligned with noise (ochre) than it is with behavior-coding directions (blue). Nevertheless, drift overlaps this behavior-coding subspace much more than chance (grey; dashed line: 95% Monte-Carlo sample). Each box reflects the distribution over all maze locations, with all consecutive pairs of sessions combined.

three of the four mice we could match concatenated decoder performance when the dimension of the drift process was constrained within a range of 14-26, a relatively small fraction (around 20%) of the components of the full population.

Biologically achievable rates of plasticity can compensate drift

Together, these analyses show that the observed dynamics of drift favor a fixed linear readout above what would be expected for random drift. However, our results also show that a substantial component of drift cannot be confined to the null space of a fixed downstream linear readout. We asked how much ongoing weight change would be needed to achieve the performance of single-day decoders while minimizing day-to-day changes in decoding weights. We first approached this without assuming a specific plasticity rule, by simultaneously optimizing linear decoders for all recorded days while penalizing the magnitude of weight change between sessions (Fig. 4a, Methods). By varying the magnitude 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 4b. Performance improves rapidly once small weight changes are permitted (~12-25% per session). Thus, relatively modest amounts of synaptic plasticity might suffice to keep encoding consistent with changes in representation, provided a mechanism exists to implement appropriate weight changes.

A biologically plausible local learning rule can compensate drift

The results in Figure 4b suggest that modest amounts of synaptic plasticity could compensate for drift, but do not suggest a biologically plausible mechanism for this compensation. Could neurons track slow reconfiguration using locally available signals in practice? To test this, we used an adaptive linear neuron model based on the least mean square learning (LMS) rule [33, 34] (Methods). This algorithm is biologically plausible because it only requires each synapse to access its current weight and recent prediction error (Fig. 5a, Methods).

Fig. 5b shows that this online learning rule achieved decoding performance comparable to the offline constrained decoders. Over the timespan of the data, LMS allows a linear decoder to track representational drift observed (Fig. 5c), exhibiting weight changes of ~10%/day across all animals (learning rate 4×10^{-4}/sample, Fig. 5-S1). These results sug-

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

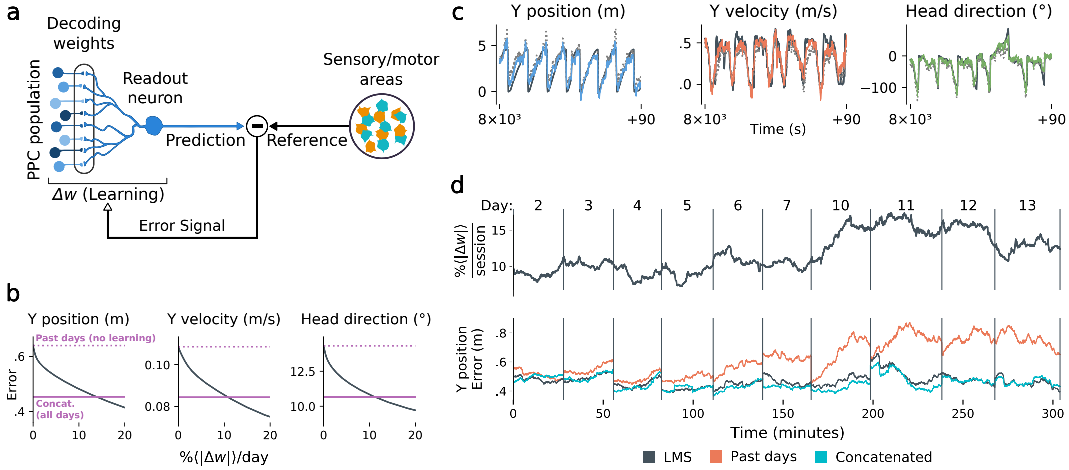

FIG. 5. **Local, adaptive decoders can track representational drift over multiple days. (a)** The Least Mean-Squares (LMS) algorithm learns to linearly decode a target kinematic variable based on error feedback. Continued online learning can track gradual reconfiguration in population representations. **(b)** As the average weight change per day (horizontal axis) increases, the average decoding error (vertical axis) of the LMS algorithm improves, shown here for three kinematic variables (Mouse 4, 144 units, 10 sessions over 12 days; Methods). (Dashed line: error for a decoder trained on only the previous session without online learning; Solid line: performance of a decoder trained over all testing days). As the rate of synaptic plasticity is increased, LMS achieves error rates comparable to the concatenated decoder. **(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 day six. Dashed traces indicate the performance of the decoder without ongoing re-training. **(d)** (top) Average percent weight-change per session for online decoding of forward position (learning rate: 4×10^{-4}/sample). The horizontal axis reflects time, with vertical bars separating days. The average weight change is 10.2% per session. To visualize $\%\Delta w$ continuously in this plot, we use a sliding difference with a window reflecting the average number of samples per session. (bottom) LMS (black) performs comparably to the concatenated decoder (cyan) (LMS mean absolute error of 0.47 m is within $\leq$ 3% of concatenated decoder error). Without ongoing learning, the performance of the initial decoder degrades (orange). Error traces have been averaged over ten minute intervals within each session. Discontinuities between days reflect day-to-day variability and suggest a small transient increase in error for LMS decoding at the start of each day.

gest that small weight changes could track representational drift in practice. In contrast, we found that LMS struggled to match the unconstrained drift of the null model explored in Figure 3d. Calibrating the LMS learning rate on the null model to match the mean performance seen on the true data required an average weight change of 93% per day. In comparison, matching the average percent weight change per day of 10%, the null model produced a normalized mean-squared-error of $1.3\sigma^2$ (averaged over all mice), worse than chance. This further indicates that drift is highly structured, facilitating online compensation with a local learning rule.

We stress that modelling assumptions mean that these results are necessarily a proxy for the rates of synaptic plasticity that are observed *in vivo*. Nonetheless, we believe these calculations are conservative. We were restricted to a sample of ∼100-200 neurons, at least an order of magnitude less than the typical number of inputs to a pyramidal cell in cortex. The per-synapse magnitude of plasticity necessarily increases when smaller subsets are used for a readout (Fig. 5-S2). One would therefore expect lower rates of plasticity for larger populations. Indeed, when we combined neurons across mice into a large synthetic population (1238 cells), we found that the plasticity required to achieve target error asymptotes at less than 4% per day (Fig. 5-S3). Together, these results show a conservatively achievable bound on the rate of plasticity required to compensate drift in a biologically plausible model.

Discussion

Several theories have been proposed for how stable behavior could be maintained despite ongoing changes in connectivity and neural activity. Here, we found that representational drift occurred in both coding and non-coding subspaces. On a timescale of a few days, redundancy in the neural population could accommodate a significant component of drift, assuming a biological mechanism exists for establishing appropriate readout weights. Simulations suggested that the existence of this approximately stable subspace were not simply a result of population redundancy, since random diffusive drift quickly degraded a downstream readout. Drift being confined to a low-dimensional subspace is one scenario that could give rise to this, although we do not exclude other possibilities. Nevertheless, a non-negligible component of drift resides outside the null space of a linear encoding subspace, implying that drift will eventually destroy any fixed-weight readout.

However, we showed that this destructive component of drift could be compensated with small and biologically realistic changes in synaptic weights, independently of any specific learning rule. Furthermore, we provided an example of a simple and biologically plausible learning rule that can achieve such compensation over long timescales with modest rates of plasticity. If our modeling results are taken literally, this would suggest that a single unit with connections to ∼100 PPC neurons can accurately decode task information with

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

modest changes in synaptic weights over many days. This provides a concrete and quantitative analysis of the implications of drift on synaptic plasticity and connectivity. Together, our findings provide some of the first evidence from experimental data that representational drift could be compatible with long-term memories of learned behavioral associations.

A natural question is whether a long-term stable subspace is supported by an unobserved subset of neurons that have stable tuning [35]. We do not exclude this possibility because we measured a subset of the neural population. However, over multiple samples from different animals our analyses consistently suggest that drift will reconfigure the code entirely over months. Specifically, we found that past reliability in single cells is no guarantee of future stability. This, combined with an abundance of highly-informative cells on a single day, contributes to poor (fixed) decoder generalization, because previously reliable cells eventually drop out or change their tuning. Consistent with this, 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, 36–38].

We stress that the kind of reconfiguration observed in PPC is not seen in all parts of the brain; primary sensory and motor cortices can show remarkable stability in neural representations over time [**?**]gallego2020long). However, even if stable representations exist elsewhere in the brain, PPC still must communicate with these areas. We suggest that a component of ongoing plasticity maintains congruent representations across different neural circuits. Such maintenance would be important in a distributed, adaptive system like the brain, in which multiple areas learn in parallel. How this is achieved is the subject of intense debate [39]. 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. Our study therefore motivates new experiments to search for neural correlates of error feedback between areas, and suggests further theoretical work to explore the consequences of such feedback.

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

Designed the study: T.O., D.V.R.; Methodology (statistical): M.E.R., D.V.R., A.R.L., T.O.; Modelling and data analysis: M.E.R., D.V.R., A.R.L.; Analysed, contributed and consulted on experimental data: L.D., C.D.H.; Validation: M.E.R.; Visualization: M.E.R.; Interpreted results: M.E.R, D.V.R, A.R.L, L.D., C.D.H., T.O.; Wrote the manuscript: M.E.R., C.D.H., T.O.; Drafted early versions of the manuscript: A.R.L., M.E.R., C.D.H, T.O. Writing (revisions and editing): M.E.R., D.V.R., L.D., 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 [2].

Virtual reality task Details of the virtual reality environment, training protocol, and fixed association navigation task are described in [2]. In brief, virtual reality environments were constructed and operated using the MATLAB-based ViRMEn software (Virtual Reality Mouse Engine) [24]. Data were obtained from mice that had completed the 4-8 week training program 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 cues were patterns on the walls (black with white dots or white with black dots), and were followed by a gray striped 'cue recall' 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 [2]. Processed data consisted of normalized Ca^{2+} fluorescence transients ("$\Delta F/F$") and behavioral variables (mouse position, view angle, and velocity). Inter-trial 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.

When considering sequences of days, we restricted analysis to units that were continuously tracked over all days. For figures 3 and 4, we use the following data: M1: 7 sessions, 15 days, 101 neurons; M3: 10 sessions, 13 days, 114 neurons; M4: 10 sessions, 11 days, 146 neurons; M5: 7 sessions, 7 days, 112 neurons. We allowed up to two-day recording gaps between consecutive sessions from the same mouse.

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

QUANTIFICATION AND STATISTICAL ANALYSIS

Decoding analyses

We decoded kinematics time-series $\mathbf{x}=\{x_1,...,x_T\}$ with T time-points 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, i.e. $\hat{\mathbf{x}} = M^\top \mathbf{z}$. We used the ordinary least-squares (OLS) solution for M, which minimizes the squared (L2) prediction error $\varepsilon=\|\mathbf{x}-M^\top\mathbf{z}\|^2$ 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 $\langle|\ \mathbf{x}-\hat{\mathbf{x}}\ |\rangle$. Here, $|\ .\ |$ denotes the element-wise absolute value, and $\langle . \rangle$ denotes expectation.

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. We added L2 regularization to this analysis by adding a constant λI to the covariance matrix of the neural data. The optimal regularization strength ($\lambda = 10^{-4}$ to 10^{-3}) slightly reduced decoding error, but did not alter the ranking of cells.

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 mean over 20 randomly-drawn neuronal subpopulations, 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, [40]).

Concatenated and constrained analyses For both the concatenated (Fig. 3b,e) and constrained analyses (Fig. 4a,b), we used the set of identified neurons included in all sessions considered. In the concatenated analyses, we solved for a single decoder M_c for all days:

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

where ε denotes the quadratic objective function to be minimized. In the constrained analysis, 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:

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

For the purposes of the constrained analysis, missing days were ignored and the remaining days treated as if they were contiguous. Two sessions were missing from the 10 and 14-day spans for mice 3 and 4, respectively (Fig. 4b). Figure 3c also shows the expected performance of a concatenated decoder for completely unrelated neural codes. To assess this, we permuted neuronal identities within individual sessions, so that each day uses a different "code".

Null model

We developed a null model to assess whether the performance of the concatenated decoder was consistent with random drift. For this, we matched the amount of day-to-day drift based on the rate at which single-day decoders degrade. We also sampled neural states from the true data in order to preserve sparsity and correlation statistics. The null model related neural activity to a 'fake' observable readout (e.g. mouse position) via an arbitrary linear mapping. The null model changed from day to day, reflecting drift in the neural code. The fidelity of single day and across day decoders in inferring a readout from the null model was matched to the true data.

For each animal we take the matrix $z \in \mathbb{R}^{n\times d}$ of mean-centered neural activity on day one, where n represents the number of recorded neurons and d represents the number of datapoints. We relate this matrix to pseudo-observations of mouse position z via a null model of the form $z_r = M_r^\top z + \epsilon_r$, where $M_r^\top, \epsilon_r \in \mathbb{R}^{1\times n}$. Note that r indexes days. The vector ϵ_r is generated as scaled i.i.d. Gaussian noise. We scale ϵ_r such that the accuracy of a linear decoder trained on the data (z, x_r) matches the average (over days) accuracy of a single-day decoder trained on the true data.

Next, we consider the choice of the randomly-drifting readout, M_r. On day one, M_1 is generated as a vector of uniform random variables on $[0, 1]$. Given M_r, we desire an M_{r+1} that satisfies

- $\|M_{r+1}\|_2 = \|M_r\|_2$.
- The expected coefficient of multiple correlation of $x_{r+1} = M_{r+1}^\top z$ against the predictive model $M_r^\top z$ (between day R^2) matches the average (over days) of the equivalent statistic generated from the true data.

To do this, we first generate a candidate $\Delta M_r' \in \mathbb{R}^{n\times 1}$ as a vector of i.i.d. white noise. The components of $\Delta M_r'$ orthogonal and parallel to M_r are then scaled so that $M_{r+1} = M_r + \Delta M_r$ satisfies the constraints above.

In Figure 4-S1, a modification of the null model that confined inter-day model drift to a predefined subspace was

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

used. Before simulating the null model over days, we randomly chose k orthogonal basis vectors, representing a k-dimensional subspace. We then searched for a candidate $\Delta M_r'$, on each inter-day interval, that was representable as a weighted sum of these basis vectors. This requirement was in addition to those previously posed. Finding such a $\Delta M_r'$ corresponds to solving a quadratically-constrained quadratic program. This is non-convex, and thus a solution will not necessarily be found. However, solutions were always found in practice. We used unit Gaussian random variables as our initial guesses for each component of $\Delta M_r'$, before solving the quadratic program using the IPOPT toolbox [41].

Drift alignment

We examine how much drift aligns with noise correlations verses directions of neural activity that vary with the task ("behavior-coding directions"). We define an alignment statistic ρ that reflects how much drift projects onto a given subspace (i.e. noise vs. behavior). We normalize ρ so that 0 reflects chance-level alignment and 1 reflects perfect alignment of the drift with the largest eigenvector of a given subspace (e.g. the principal eigenvector of the noise covariance).

Let $z(x)$ denote the neural population activity, where x reflects a normalized measure of maze location, akin to trial pseudotime. Define drift $\Delta\mu_z(x)$ as the change in the mean neural activity $\mu_z(x)$ across days. We examine how much drift aligns with noise correlations verses directions of neural activity that vary with task pseudotime $(dz(x)/dx)$.

To measure the alignment of a drift vector $\Delta\mu$ with the distribution of inter-trial variability (i.e. noise), we consider the trial-averaged mean μ and covariance Σ of the neural activity (log calcium-fluorescence signals filtered between 0.03 and .3 Hz and z-scored), conditioned on trial location and the current/previous cue direction. We use the mean squared magnitude of the dot product between the change in trial-conditioned means between days ($\Delta\mu$), with the directions of inter-trial variability ($\Delta z{=}z{-}\langle z\rangle$) on the first day, which is summarized by the product $\Delta\mu^\top\Sigma\Delta\mu$:

$$\begin{aligned}\left\langle|\Delta\mu^\top\Delta z|^2\right\rangle &= \left\langle\Delta\mu^\top\Delta z\Delta z^\top\Delta\mu\right\rangle\\ &= \Delta\mu^\top\left\langle\Delta z\Delta z^\top\right\rangle\Delta\mu\\ &= \Delta\mu^\top\Sigma\Delta\mu.\end{aligned}\tag{3}$$

To compare pairs of sessions with different amounts of drift and variability, we normalize the drift vector to unit length, and normalize the trial-conditioned covariance by its largest eigenvalue $\lambda_{\max}$:

$$\phi^2_{\text{trial}} = \frac{\Delta\mu^\top\Sigma\Delta\mu}{|\Delta\mu|^2\cdot\lambda_{\max}}\tag{4}$$

The statistic ϕ_{trial} equals 1 if the drift aligns perfectly with the direction of largest inter-trial variability, and can be interpreted as the fraction of drift explained by the directions of noise correlations.

Random drift can still align with some directions by chance, and the mean squared dot-product between two randomly-oriented D-dimensional unit vectors scales as $1/D$. Accounting for the contribution from each dimension of Σ, the expected chance alignment is therefore $\phi_0^2 = \operatorname{tr}(\Sigma)/(D\cdot\lambda_{\max})$. We normalize the alignment coefficient ρ_{noise} such that it is 0 for randomly oriented vectors, and 1 if the drift aligns perfectly with the direction of largest variability:

$$\rho_{\text{noise}} = \frac{\phi_{\text{trial}}-\phi_0}{1-\phi_0}\tag{5}$$

We define a similar alignment statistic ρ_{coding} to assess how drift aligns with directions of neural variability that encode location. We consider the root-mean-squared dot product between the drift $\Delta\mu$, and the directions of neural activity (z) that vary with location (x) on a given trial, i.e. $\nabla_x z(x)$:

$$\begin{aligned}\left\langle|\Delta\mu^\top\nabla_x z(x)|^2\right\rangle &= \left\langle\Delta\mu^\top[\nabla_x z(x)][\nabla_x z(x)]^\top\Delta\mu\right\rangle\\ &= \Delta\mu^\top\left\langle[\nabla_x z(x)][\nabla_x z(x)]^\top\right\rangle\Delta\mu\\ &= \Delta\mu^\top\left[\Sigma_\nabla+\mu_\nabla\mu_\nabla^\top\right]\Delta\mu\end{aligned}\tag{6}$$

In contrast to the trial-to-trial variability statistic, this statistic depends on the second moment $\Sigma_\nabla+\mu_\nabla\mu_\nabla^\top$, where $\nabla_x z(x){\sim}\mathcal{N}(\mu_\nabla,\Sigma_\nabla)$. We define a normalized ϕ^2_{coding} and ρ_{coding} similarly to ϕ^2_{trial} and ρ_{noise}. For the alignment of drift with behavior, we observed ρ_{coding}=0.11–0.24 (μ=0.15, σ=0.03), which was significantly above chance for all mice. In contrast, the 95th percentile for chance alignment (i.e. random drift) ranged from 0.06–0.10 (μ=0.07, σ=0.02). Drift aligned substantially more with noise correlations, with ρ=0.29–0.43 (μ=0.36, σ=0.04).

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 [33, 34, 42]. 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.

By varying the learning rate, we obtained a trade-off (Fig. 4b) between the rate of weight changes and the decoding error, with the most rapid learning rates exceeding the performance of offline (static) decoders. In Fig. 4d, we selected an example with a learning rate of η=4×10^{-4}. To provide a continuous visualization of the rate of weight change in Fig. 4d, we used a sliding difference with a duration matching the average session length. This was normalized by the average weight magnitude to report percent weight change per day. In all other statistics, per-day weight change is assessed as

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

the difference in weights at the end of each session, divided by the days between the sessions.

DATA AND CODE AVAILABILITY

Datasets recorded in Driscoll et al. [2] 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] S. Rumpel and J. Triesch, "The dynamic connectome," *e-Neuroforum*, vol. 22, no. 3, pp. 48–53, 2016.

[2] 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.

[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] Y. Ziv, L. D. Burns, E. D. Cocker, E. O. Hamel, K. K. Ghosh, L. J. Kitch, A. El Gamal, and M. J. Schnitzer, "Long-term dynamics of ca1 hippocampal place codes," *Nature neuroscience*, vol. 16, no. 3, p. 264, 2013.

[5] S. J. Levy, N. R. Kinsky, W. Mau, D. W. Sullivan, and M. E. Hasselmo, "Hippocampal spatial memory representations in mice are heterogeneously stable," *bioRxiv*, p. 843037, 2019.

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

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] A. Singh, A. Peyrache, and M. D. Humphries, "Medial prefrontal cortex population activity is plastic irrespective of learning," *Journal of Neuroscience*, vol. 39, no. 18, pp. 3470–3483, 2019.

[12] J. S. Montijn, G. T. Meijer, C. S. Lansink, and C. M. Pennartz, "Population-level neural codes are robust to single-neuron variability from a multidimensional coding perspective," *Cell reports*, vol. 16, no. 9, pp. 2486–2498, 2016.

[13] J. A. Gallego, M. G. Perich, S. N. Naufel, C. Ethier, S. A. Solla, and L. E. Miller, "Cortical population activity within a preserved neural manifold underlies multiple motor behaviors," *Nature communications*, vol. 9, no. 1, p. 4233, 2018.

[14] J. A. Gallego, M. G. Perich, R. H. Chowdhury, S. A. Solla, and L. E. Miller, "Long-term stability of cortical population dynamics underlying consistent behavior," *Nature Neuroscience*, pp. 1–11, 2020.

[15] J. A. Hennig, M. D. Golub, P. J. Lund, P. T. Sadtler, E. R. Oby, K. M. Quick, S. I. Ryu, E. C. Tyler-Kabara, A. P. Batista, M. Y. Byron, *et al.*, "Constraints on neural redundancy," *Elife*, vol. 7, p. e36774, 2018.

[16] A. D. Degenhart, W. E. Bishop, E. R. Oby, E. C. Tyler-Kabara, S. M. Chase, A. P. Batista, and M. Y. Byron, "Stabilization of a brain–computer interface via the alignment of low-dimensional spaces of neural activity," *Nature Biomedical Engineering*, pp. 1–14, 2020.

[17] D. Huber, D. Gutnisky, S. Peron, D. O'connor, J. Wiegert, L. Tian, T. Oertner, L. Looger, and K. Svoboda, "Multiple dynamic representations in the motor cortex during sensorimotor learning," *Nature*, vol. 484, no. 7395, p. 473, 2012.

[18] 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.

[19] 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.

[20] 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.

[21] K. E. Moczulska, J. Tinter-Thiede, M. Peter, L. Ushakova, T. Wernle, B. Bathellier, and S. Rumpel, "Dynamics of dendritic spines in the mouse auditory cortex during memory formation and memory recall," *Proceedings of the National Academy of Sciences*, vol. 110, no. 45, pp. 18315–18320, 2013.

[22] Y. Loewenstein, A. Kuras, and S. Rumpel, "Multiplicative dynamics underlie the emergence of the log-normal distribution of spine sizes in the neocortex in vivo," *Journal of Neuroscience*, vol. 31, no. 26, pp. 9481–9488, 2011.

[23] Y. Loewenstein, U. Yanover, and S. Rumpel, "Predicting the dynamics of network connectivity in the neocortex," *Journal of Neuroscience*, vol. 35, no. 36, pp. 12535–12544, 2015.

[24] C. D. Harvey, P. Coen, and D. W. Tank, "Choice-specific sequences in parietal cortex during a virtual-navigation decision task," *Nature*, vol. 484, no. 7392, pp. 62–68, 2012.

[25] R. A. Andersen, L. H. Snyder, D. C. Bradley, and J. Xing, "Multimodal representation of space in the posterior parietal cortex and its use in planning movements," *Annual review of neuroscience*, vol. 20, no. 1, pp. 303–330, 1997.

[26] R. A. Andersen and C. A. Buneo, "Intentional maps in posterior parietal cortex," *Annual review of neuroscience*, vol. 25, no. 1, pp. 189–220, 2002.

[27] G. H. Mulliken, S. Musallam, and R. A. Andersen, "Forward estimation of movement state in posterior parietal cortex," *Proceedings of the National Academy of Sciences*, vol. 105, no. 24, pp. 8170–8177, 2008.

[28] M. Krumin, J. J. Lee, K. D. Harris, and M. Carandini, "Decision and navigation in mouse parietal cortex," *Elife*, vol. 7, p. e42583, 2018.

[29] M. Minderer, K. D. Brown, and C. D. Harvey, "The spatial structure of neural encoding in mouse posterior cortex during navigation," *Neuron*, 2019.

[30] J. L. Calton and J. S. Taube, "Where am i and how will i get there from here? a role for posterior parietal cortex in the integration of spatial information and route planning," *Neurobiology of learning and memory*, vol. 91, no. 2, pp. 186–196, 2009.

[31] J. M. Carmena, M. A. Lebedev, C. S. Henriquez, and M. A. Nicolelis, "Stable ensemble performance with single-neuron

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

variability during reaching movements in primates," *Journal of Neuroscience*, vol. 25, no. 46, pp. 10712–10716, 2005.

[32] J. D. Semedo, A. Zandvakili, C. K. Machens, M. Y. Byron, and A. Kohn, "Cortical areas interact through a communication subspace," *Neuron*, vol. 102, no. 1, pp. 249–259, 2019.

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

[34] 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.

[35] C. Clopath, T. Bonhoeffer, M. Hübener, and T. Rose, "Variance and invariance of neuronal long-term representations," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 372, no. 1715, p. 20160161, 2017.

[36] 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.

[37] 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.

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

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

[40] 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.

[41] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.

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

Supplemental Figures

Fig. 2-S1: Behavioral stability

It is possible that changes in population codes relate to systematic changes in behavior over time. As described in Driscoll et al. (2017), these experiments were performed only after mice achieved asymptotic performance in speed and accuracy on the task. Nevertheless, details of behavior are important.

Across all mice and behavioral variables, there was a statistically-significant difference in means between 91% pairs of sessions ($p<0.05$; Bonferroni multiple-comparison correction for a 0.05 false discovery rate (FDR)). However, the average effect size ($\Delta\mu/\sigma$, i.e. Cohen's d) was small, at 10–16% per animal. We could also partially predict the recording session based on 10-second kinematics trajectories (position, velocity, head-direction). Under cross-validation, kinematics could predict the recording session 9–17% above chance. We used a linear decoder to predict an indicator vector with a 1 in the session corresponding to the given kinematics trajectory, and 0 otherwise. The predicted session was assessed as the session with the largest predicted value under cross-validation, and chance level assessed by permuting session identities.

Most of this predictive power came from differences in the forward movement in the initial portion of the T-maze. Much of this appeared to be daily variability, rather than drift (Fig. 2-S1). We found a small but significant systematic decrease in forward velocity in mouse 4 (Pearson correlation between recording day and median forward velocity of -0.9, two-tailed $p<0.05$).

This suggests that each mouse exhibited small but detectable daily variability in their behavior. Most variability was unsystematic, and therefore unrelated to the slow changes in neural codes studied here. We expect changes in forward speed in mouse 4 to contribute to apparent drift in some cells. However, the results presented here generalize across mice 1, 3, and 5, which exhibited stable behavior.

Figure 4-S1, concatenated decoder performance depends on the rank of the drift

Figure 5-S1, online learning with LMS: additional subjects

We present an example of the LMS decoding algorithm on mouse 4 in the main text. We show similar results here for mice 1, 3, and 5. We observed inter-day weight changes of 7.6-10.4%, consistent with observed rates of change in the volume of dendritic spines in other studies. We used a learning rate of 4^{-4} 1/sample, which led to error rates within $5-15\%$ of the concatenated decoder (depending on the random selection of training and testing trials used for validation)

Figure 5-S2: The plasticity level required to track drift varies with population size

Figure 5-S3: Extrapolation to larger populations

We saw in Figure 5-S2 that the amount of synaptic plasticity required to track a given error decreases for larger populations. In this study, we examine recorded populations of $\sim$100 neurons. Typically, the number of inputs to a given neurons is much larger than this, on the order of thousands. The $\sim$10% weight change per day reported by LMS could therefore be an over-estimate of the amount of plasticity required. To address this, we extend the LMS analysis to larger populations by combining neurons from different mice. This procedure destroys population correlations, and requires aligning activity from different mice. Despite this, the pooled populations yields a useful study of how plasticity scales with population size.

To align data from different mice, we matched trials based on the current and previous trial cue, and converted the neural time-series into location-based pseudotime, representing

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

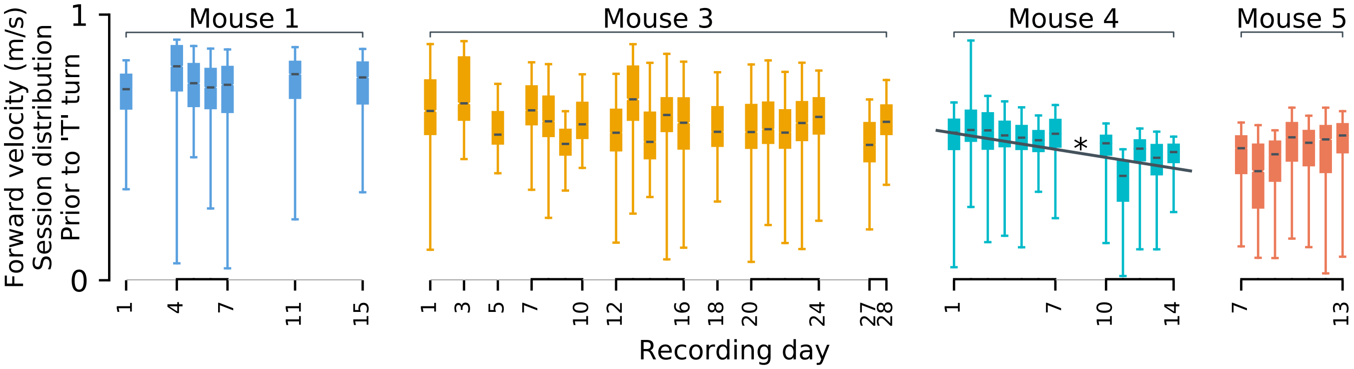

FIG. 6. *

Figure 2—figure supplement 1: Statistics of forward motion show small daily variations. Each mouse's velocity in the initial (forward) segment of the 'T' maze varies slightly between days. Across days, behavior is broadly similar. Differences in means (black lines), although minuscule, are often statistically significant. Systematic drift-like trends appear absent from mice 1 and 3. A statistically significant trend is present for mouse 4 ($p<0.05$). We show only forward velocity here, as other kinematics variables exhibited less variability.

the fraction of the maze completed between 0 and 100%. This allowed us to register neuronal signals from different trials on different mice. We constructed a synthetic population of 1238 cells, covering a six-session-long recording period. We allowed up to two-day recording gaps between consecutive sessions from the same mouse. We found that larger populations could achieve the same performance as ~100 cells with a ~4% weight change per day.

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

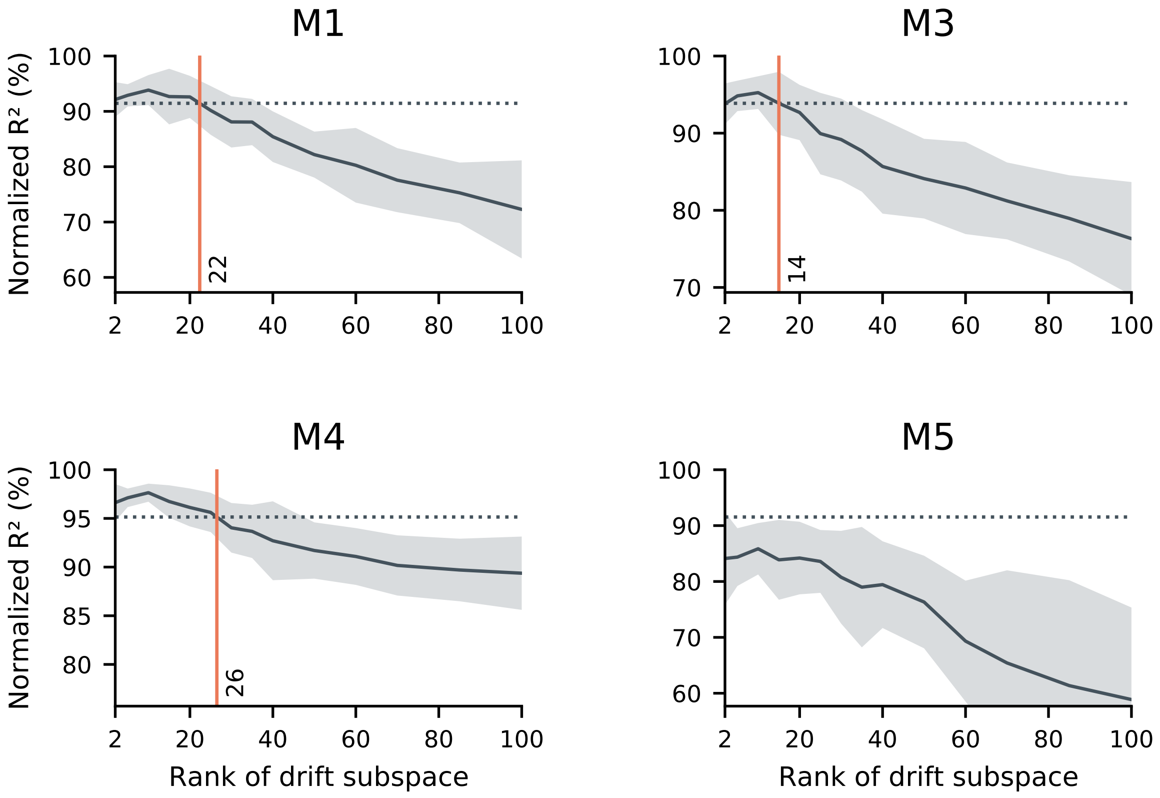

FIG. 7. *

Figure 4—figure supplement 1: sufficiently low-rank drift resembles the data in terms of the performance of a concatenated decoder. Here, we further explore the null model introduced in Figure 3d. As in Figure 3d, we simulated random drift in the neural readout. We matched the null model to the statistics of neural activity, the within-day decoding accuracy, and the performance degradation when generalizing between days. In these simulations, we explore the scenario that the drift may be confined to a (randomly-selected) low-dimensional subspace. We evaluated a range of dimensionalities for the drift subspace (horizontal axes), and evaluated the performance of a concatenated decoder on simulated data. While unconstrained drift prevents the identification of a concatenated decoder with good performance (Fig. 3d), sufficiently constrained drift does not. In these simulations, we found that constraining drift to a subspace of rank 14-26 (red vertical lines) led to similar performance as the data (dashed horizontal lines) in all subjects except for mouse 5. We speculate that this is because Mouse 5 had limited data and poor generalization of single-day decoders over time, but other scenarios are possible. Black traces reflect the mean over 20 random simulations, and shaded regions reflect one standard deviation.

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

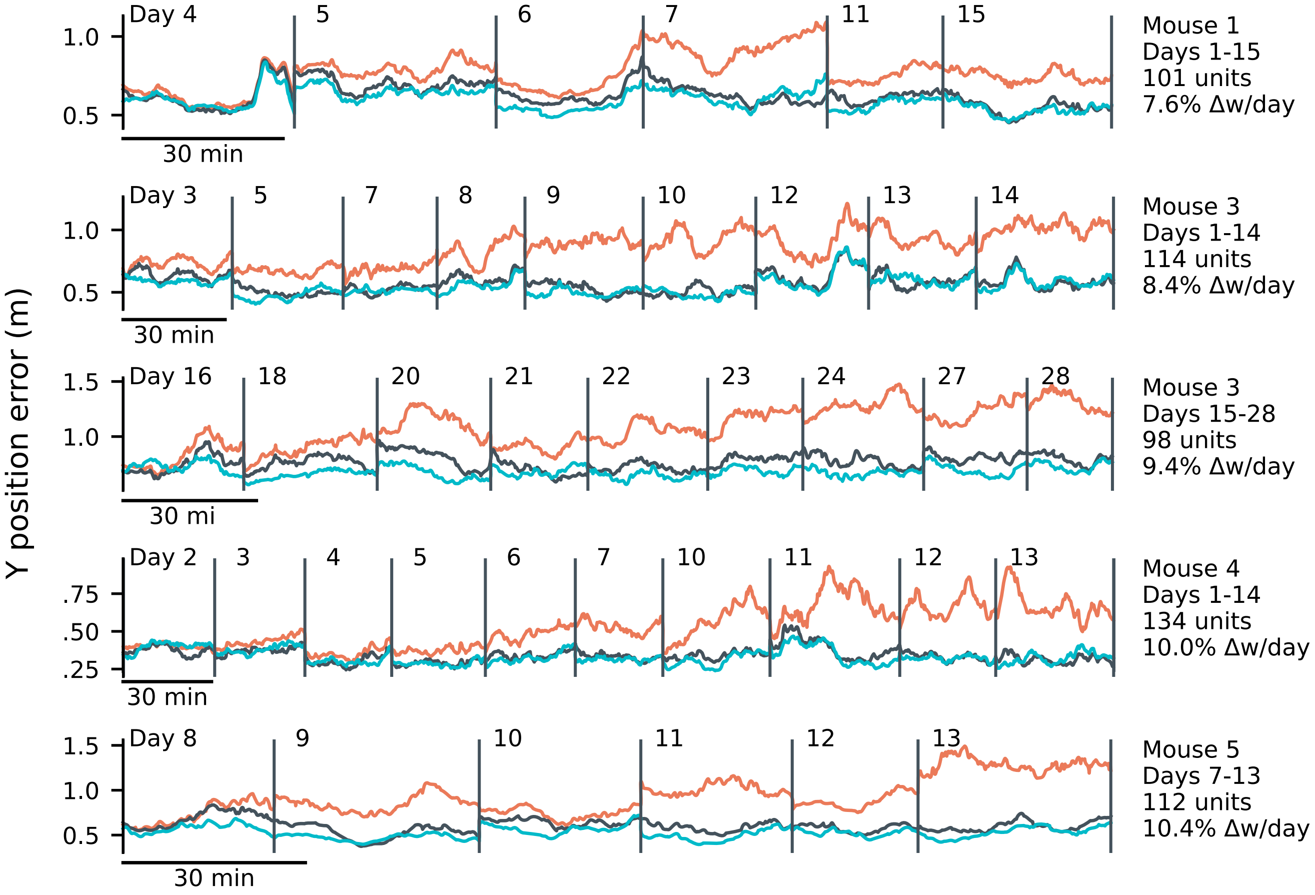

FIG. 8. *

Figure 5—figure supplement 1; LMS results for mice 1, 3, 4, and 5. Results of applying the online LMS algorithm with a learning rate of 4×10^{-4}/sample. Errors reflect the mean absolute error over ten minute intervals. LMS (black) achieves errors comparable to an offline decoder trained on all sessions ("concatenated", blue), and outperforms a fixed decoder trained on the initial day (red). Only times within a trial were used for training. We present two spans of time from Mouse 3, reflecting two largely non-overlapping populations of tracked neurons on non-overlapping spans of days.

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

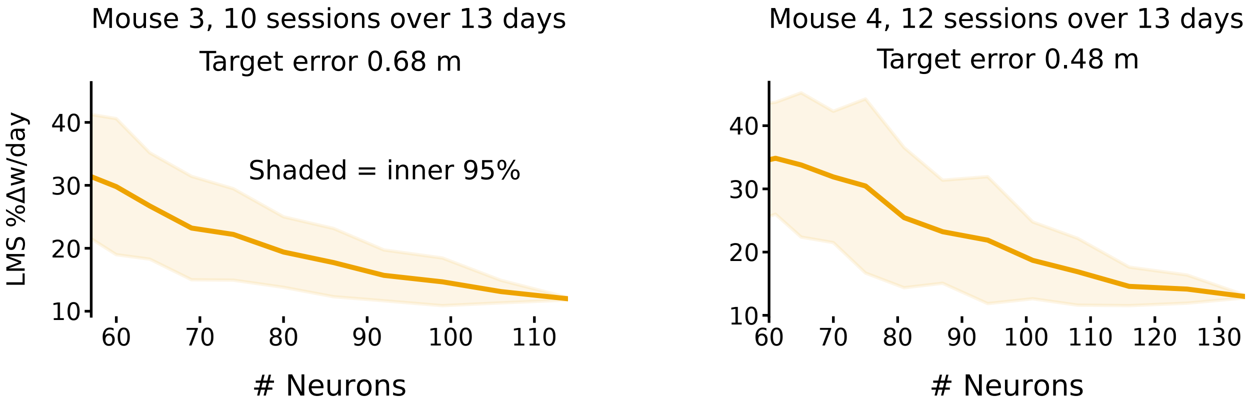

FIG. 9. *

Figure 5—figure supplement 2; Smaller populations require more plasticity to achieve target error levels. These plots show the daily weight changes required to track drift when decoding forward positions as a function of population size for mice 3 and 4. Smaller populations require more plasticity. The target error (M3: 0.68 m, M4: 0.48 m) was set based on the performance of LMS on the full population (M3: 114 neurons, M4: 134 neurons). For each sub-population size, 50 random sub-populations were drawn, and the learning rate was optimized to achieve the target error level. Shaded regions reflect the inner 95th percentile over all sampled sub-populations. Weight change was assessed as the weight change between the end of consecutive sessions and normalized by the overall average weight magnitude.

bioRxiv preprint doi: https://doi.org/10.1101/796334. this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. It is made available under a CC-BY-NC-ND 4.0 International license.

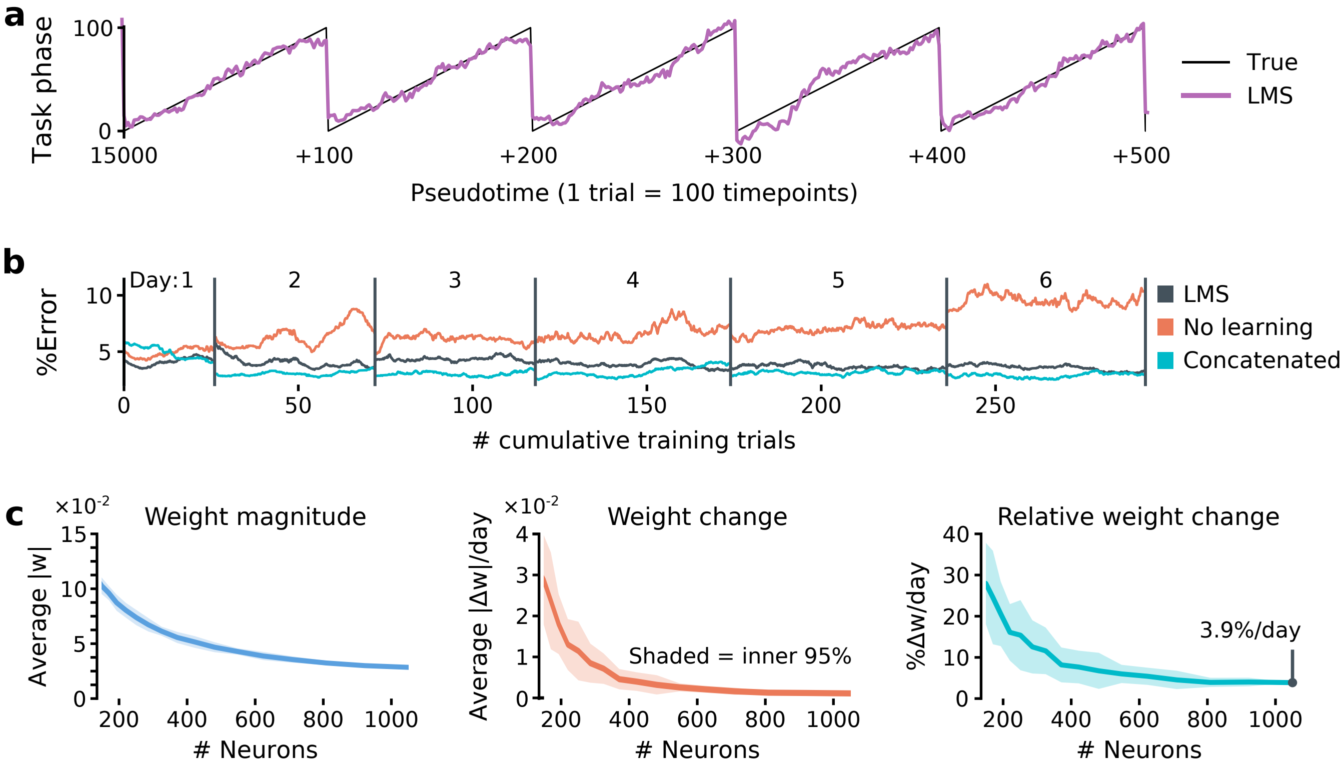

FIG. 10. *

Figure 5—figure supplement 3; The plasticity required to achieve a fixed error level decreases for larger populations. (a) Trial pseudotime (% of trial complete; black) can be decoded from a synthetic pooled population (1238 cells) using the LMS algorithm (violet: prediction). **(b)** Similarly to the single-subject results, LMS tracks changes in the population code over time. In this case, a learning rate of 8×10^{-4}/sample achieved comparable error to a concatenated decoder. The larger population permits better decoding error of ∼5%, compared to the ∼15–20% error in forward position decoded from ∼100 neurons. **(c)** As population size increases, both the weight magnitudes (left) and the rates of weight change (middle) decrease. Small populations could not achieve the error rates possible using the full population, even with very large learning rates. We therefore set the target error a bit higher, at 13% chance level. This is comparable to the error rates seen in individual mice using ∼100 cells. Overall, the required percentage weight change decreased for larger populations (right).