Methods

M.1 Subjects

Experimental procedures were approved by the Harvard Medical School Institutional Animal Care and Use Committee. Data were acquired from five male C57BL/6J mice (Jackson Labs), aged 8-10 weeks at the start of behavioral training and 14-22 weeks during imaging. Mice were housed as pairs in cages in a room with a reverse light/dark cycle. Mice had no previous history of any other experiments. A titanium headplate was affixed to the mouse's skull using dental cement (Metabond, Parkell). Mice were placed on a water schedule, in which they received 800 μl of water each day. Each mouse's weight was measured daily to ensure that it was $\geq$ 80% of the mouse's pre-water-restriction weight.

M.2 Behavior

M.2.1 Virtual reality system

We used a modified version of the virtual reality (VR) system described previously[51]. Images were back-projected onto a half-cylindrical screen (24-inch diameter) using a PicoP microprojector (MicroVision). Spherical treadmill movement was recorded using an optical sensor positioned beneath the ball. Forward/backward translation in VR was controlled by changes in pitch, and rotation in VR was controlled by changes in roll (both relative to the mouse's body). The recorded behavioral parameters were the mouse's position and view angle in the virtual environment along with the rotational velocity of the spherical treadmill. Virtual environments were made using ViRMEn[52].

M.2.2 Task description (Fig. 1, Supplementary Fig. 1)

While running down the stem of the T with predominantly gray walls, mice encountered six visual cues (white wall segments with black dots) at fixed locations. Each cue could appear on either the left or right wall, and only one cue was visible at a time. Cue visibility was determined by the mouse's position such that the duration of each cue was determined by the mouse's running speed. On average, each cue was visible for ~0.8 seconds. To receive a reward, mice had to determine if more cues were presented on the left or the right and, after a short stretch of maze without additional cues (90 cm, ~1 second), turn at the T-intersection toward the direction that had more cues. Task difficulty was modulated by varying the difference between the number of left and right cues (net evidence; 6 total cues per trial, ranging from 0 to 6 presented on the left and the remainder on the right). The sequence of cues was determined randomly for each trial of a given net evidence. On 3-3 trials, the rewarded location was selected randomly. Following the completion of the trial, the screen changed to black for the duration of the inter-trial interval (2 s for correct choice and 4 s for incorrect choice).

M.2.3 Behavioral training procedure (Supplementary Fig. 1)

Behavioral sessions lasted 45-60 minutes. Mice received liquid rewards through a lick spout (4 μl/reward, 10% sweetened condensed milk). Mice were trained to perform the evidence accumulation task over a series of eight mazes (Supplementary Fig. 1). We implemented bias correction throughout training. Bias correction was not used during imaging sessions. On each trial, we determined a probability that a trial would be a right choice trial as the fraction of left choices over the previous 20 trials. To maintain a high level of performance throughout the session, we introduced a small fraction of easy trials ('crutch trials') interleaved with the evidence accumulation trials. Crutch trials were identical to trials from maze 5 (Supplementary Fig. 1) in which no evidence accumulation or delay were present. The probability of a crutch trial on a given trial was equal to the fraction of error trials over the previous 20 trials. Crutch trials were used during imaging sessions and were excluded from all analyses.

M.2.4 Behavioral analyses

Behavioral performance was calculated as the fraction of trials in which the mouse performed the correct choice, excluding crutch trials.

M.2.4.1 Behavioral analysis of evidence accumulation (Supplementary Fig. 2)

To test if mice used more than one piece of evidence per trial, we fit the behavioral performance as a function of number of left cues with a logistic function (assuming more than one piece of evidence used per trial) and a linear function (assuming a single piece of evidence used per trial). To compare model fits, for each mouse, each behavioral day was fit separately by each model, and the distribution of root mean squared errors (RMSE) was compared with a two-sample Student's t-test. Across mice, the logistic function fit the data significantly better than the linear function (Supplementary Fig. 2b-c), indicating that mice used more than one piece of evidence per trial.

To test which cues mice used across trials, we used multivariate linear regression, with the behavioral choice as the response variable and the cue identities as the explanatory variables. To include large numbers of trials, multiple consecutive sessions (mean: 9, range: 7-12) were combined. All cues had significant regression coefficients with a preference toward earlier segments, suggesting that mice accumulated evidence with a primacy bias (Supplementary Fig. 2d-f). This unequal weighting of cues was not present in all mice.

M.2.4.2 Analysis of across-trial behavioral effects

To test whether there was a relationship between the mouse's choice on a trial and the outcome of the previous trial, we used a multivariate logistic regression with interactions with the previous trial's choice and reward as binary explanatory variables and the mouse's choice on the test trial

as the response variable. We combined multiple consecutive sessions (mean: 9, range: 7-12) to include large numbers of trials. This model was unable to predict the mouse's choice, suggesting that there was no easily detectable behavioral relationship between the mouse's choice and the outcome of the previous trial (R^2: 0.02 ± 0.01, mean ± s.e.m across datasets, $p > 0.05$).

M.2.5 Contribution of behavioral variability to neuronal activity results

We performed multiple analyses to test the possibility that our results were due to contributions from behavioral parameters such as changes in the visual scene or running patterns, rather than features such as evidence accumulation or variability in internally-driven neuronal population activity. In all cases, we found that behavioral variability could not entirely explain the neuronal activity patterns we observed.

One possibility is that the mouse began to turn left or right as it saw evidence cues such that accumulation of evidence was performed through the mouse's viewing angle in the maze (e.g. left of center viewing for more accumulated left cues) rather than through an internal representation of net evidence. In such a case, net evidence could be correlated with different heading directions (view angle in the maze), motor signals (turning on the treadmill), and direct visual input (combination of view angle and position in the maze). However, when we limited our analysis to only trials with similar view angles (± 2.5 degrees), the SVR analysis based on population activity predicted above chance levels the actual net evidence (Supplementary Fig. 5e-f). Additionally, when we trained SVR models on behavioral parameters alone (view angle, maze position, two axes of treadmill rotational velocity in a single model) or on behavioral parameters in addition to neuronal population activity, models trained on both behavioral parameters and neuronal population activity consistently predicted the net evidence better than those trained on behavioral parameters alone, despite modest predictability from behavioral parameters alone (comparison of models with different numbers of parameters was made possible by the use of non-overlapping training and testing sets; Supplementary Fig. 5g-h). These results suggest that a representation of net evidence was present independent of heading direction, running patterns, and direct visual input.

Another possibility is that trial-trial differences in behavioral parameters could have generated the structured trial-trial variability in neuronal activity and the presence of history signals across long timescales. We ruled out these possibilities using a series of tests to see if neuronal activity explained additional variability beyond what could be explained by the behavioral variability, by building both neuronal activity and behavioral features into a single logistic regression model. We found that the current behavioral parameters alone explained above chance, but poorly, past and future activity pattern clusters for only ~1 epoch into the past or future (Supplementary Fig. 8a-c). In contrast, models using the current behavioral parameters and the current activity

pattern cluster (or the current activity pattern alone) predicted the past and future epochs significantly better, including across a longer timescale of 5-6 epochs (determined by adjusted R^2 to compare models with different numbers of parameters; Supplementary Fig. 8a-c). Consistently, using behavioral parameters for visual scene and running patterns, we were unable to classify above chance levels history signals from the previous trial in the subsequent trial (Supplementary Fig. 8d-g). Together these results indicate that the trial-trial variability and history signals we observed included neuronal signals that could not be explained by major behavioral variability.

M.3 Imaging

M.3.1 Surgical procedure

When mice performed well on maze 7 (Supplementary Fig. 1a), they underwent a surgery to implant a cranial window. For three days prior to surgery, mice were given 5 mL of water per day. A circular craniotomy (3.1 mm diameter) was made over left PPC (2 mm posterior, 1.75 mm lateral of bregma). A virus mixture containing a 4:1 volumetric ratio of tdTomato (AAV2/1-*CAG*-tdTomato) to GCaMP6 (AAV2/1-*synapsin-1*-GCaMP6f or AAV2/1-*synapsin-1*-GCaMP6m) was delivered by three injections of ~20 nL (~5 min/injection, ~150 µm spacing between injections). Viruses were obtained from the University of Pennsylvania Vector Core Facility. Injections were made near the center of the craniotomy, ~275 µm below the dura, using a beveled glass pipette (~15 µm tip diameter) and a custom air pressure injection system. The pipette was advanced using a micromanipulator (Sutter MP285) at a 30-degree angle to minimize compression of the brain. A window with glass plug (5 mm diameter coverslip plus two 3 mm diameter coverslips; #1 thickness; CS-3R and CS-5R, Warner Instruments) was made using UV-curable optically transparent adhesive (Norland Optics). The window was affixed to the brain using a drop of Kwik-Sil (World Precision Instruments) and affixed to the skull using Metabond mixed with ~ 5% vol/vol India ink, to prevent light leakage. A headplate was affixed to the skull using Metabond mixed with India ink. A titanium ring was mounted on top of the headplate to interface with a cylinder of black rubber to surround the microscope's objective lens, thus preventing light leak from the VR display into the microscope[53]. Imaging began at least 4 weeks post-injection and was continued for up to 12 weeks. Fields-of-view containing cells with GCaMP6 in the nucleus were excluded. In a given session, we imaged ~350 neurons simultaneously during ~300 trials (range, 188-648 neurons; range, 231-414 trials; n = 5 mice; Supplementary Table 1).

M.3.2 Two-photon microscope design

Imaging was performed using a custom-built two-photon microscope. The microscope scan head included a resonant scanning mirror and a galvanometric mirror separated by a scan lens-based relay telescope. Fluorescence light collection optics were based on a custom design to collect wide dispersion angles from large (~20 mm) back aperture objectives. The microscope was

stationary, and the mouse was mounted on an XYZ translation stage (Dover Motion). Green and red emission light were separated by a dichroic mirror (580 nm long-pass, Semrock) and bandpass filters (525/50 and 641/75 nm, Semrock) and collected by GaAsP photomultiplier tubes (Hamamastu). Excitation light was delivered from a Ti:sapphire laser (Coherent) operated at 920 nm. The microscope was controlled by ScanImage (version 5; Vidrio Technologies)[54].

M.3.3 Imaging data acquisition

Imaging data were acquired at ~30 Hz at a resolution of 512 x 512 pixels (~700 µm x ~700 µm field-of-view) using a Nikon 16x 0.8 NA objective lens. Imaging and behavioral data were synchronized using custom-written MATLAB software by simultaneously recording the frame clock from Scanimage and an iteration counter from ViRMEn (see Methods M.2.1). Up to 100,000 frames were acquired from each imaging session over the course of ~1 hour. Imaging data were acquired at depths between 100 and 200 µm below the dura. Data were analyzed from 11 fields-of-view from 5 mice.

M.3.4 Pre-processing of imaging data

Motion correction, the definition of putative cell bodies, and extraction of fluorescence traces ($\Delta F/F$) were performed in a semi-automatic fashion using custom-written software. In brief, following motion correction[55], the correlation of fluorescence timeseries was calculated for each pair of pixels within ~60 µm of one another. Fluorescence sources (putative cells) were then identified by applying a continuous-valued, eigenvector-based approximation of the normalized cuts objective[56] to the correlation matrix, followed by discrete segmentation by k-means clustering, yielding binary masks for all identifiable fluorescence sources. For each putative cell, the local neuropil fluorescence was estimated by averaging across nearby pixels devoid of fluorescence sources. The scale of neuropil contamination of the cell fluorescence was estimated by regressing the background timeseries against low-activity regions of the cell timeseries, and the scaled background timeseries was then subtracted from the cell timeseries. Cell selection and neuropil subtraction were performed using a tool that allowed manual examination of clustering results and parameters, in combination with anatomical information and fluorescence traces corresponding to each cluster. All neuropil contamination fits were also examined by eye and adjusted when necessary. All fluorescence traces were deconvolved to estimate the probability of a spike in each frame (estimated spike count)[16]. Similar results were obtained from the non-deconvolved $\Delta F/F$ traces (Supplementary Fig. 10).

M.4 Data analysis

M.4.1 General analysis procedures

Data were grouped into spatial bins (3.75 cm/bin). The T-maze was linearized prior to binning by folding the arms such that they were a continuation of the stem. Neuronal activity and behavioral

parameters were averaged in each bin (2-3 imaging frames per bin per trial). Unless otherwise noted, all analyses were performed on both correct and error trials together. All correlation coefficients were from Pearson's correlations. Portions of this research were conducted on the Orchestra High Performance Compute Cluster at Harvard Medical School (supported by grant NCRR 1S10RR028832-01).

M.4.4 Classifiers (without clustering)

M.4.4.1 General procedures

Unless otherwise noted, all population classifiers were support vector machines (SVMs) with a radial basis function (Gaussian) kernel[57,58] implemented using the libsvm library (version 3.20)[59]. All population classification was performed on the concatenated activity of all individual neurons. Data were divided into non-overlapping training/validation and test sets (50% of trials each). To prevent overfitting, models were trained exclusively on the training/validation set, with the test set left untouched until final testing. Hyperparameter (C and γ; regularization weight and radial basis function width, respectively) selection was performed using a random search method with 10-fold cross-validation on the training/validation set of only a single dataset, and the same hyperparameters were used for all datasets.

M.4.4.2 Two-class classifications (Fig. 2c, e, g, Supplementary Figs. 5c, j, 8d-g, 10a, d-e)

For two-class classification problems, such as the classification of the choice (Fig. 2c, e, g, Supplementary Fig. 5c), previous trial's choice (Fig. 5b), and previous trial's reward outcome (Fig. 5c), a *C*-Support Vector Classification (*C*-SVC) approach was used. For the single neuron choice classifier, neuronal activity was averaged across each left 6-0 and right 0-6 trial (Fig. 2c). For population classifiers, independent SVMs were trained on each spatial bin (Fig. 2e, g, Supplementary Figs. 5c, 9d-g, 10a, d-e). In cases where classification accuracy based on neuronal data was compared with accuracy based on behavioral data (Supplementary Fig. 8d-g), hyperparameters were optimized separately for each. In cases where the weights placed on individual neurons were analyzed (Supplementary Fig. 5j), a linear kernel was used to allow for interpretability of the weights.

M.4.4.3 Net evidence support vector regression (Fig. 2d, f, h, Supplementary Figs. 5e-h, 10b)

For the prediction of net evidence based on neuronal population activity (Fig. 2d, f, h; Supplementary Fig. 5e-h), an ϵ-Support Vector Regression (ϵ-SVR) approach was used[57,58,60]. For the net evidence models, the average activity during the third quarter of each cue's presentation (~200 ms) was calculated for each neuron. For training and testing, each cue was treated as a separate trial with class labels corresponding to the net evidence including that cue. Training and testing sets were divided based on whole trials to prevent similar activity on different cues within the same trial from corrupting the results. To determine prediction accuracy, the predicted net

evidence was compared to the actual net evidence via a correlation coefficient. To calculate statistical significance, the results were compared to the distribution resulting from 1000 shuffles of class labels. To rule out categorization, which would result in identical guesses within left and right net evidence conditions, but a positive slope across all net evidence conditions, we calculated significance separately within left and right net evidence conditions; both were statistically significant across mice for the population classifiers ($p < 0.001$; Fig. 2f).

M.4.4.4 Classifiers built by adding-in subsets of neurons (Fig. 2g-h, Supplementary Fig. 5i)
Classifiers were built similar to the population classifiers described in M.4.4.1, except with the input being a subset of neurons. The single neuron choice SVM and SVR net evidence correlation were used to determine single neuron selectivity for choice and net evidence, respectively. Neurons were sorted in ascending order based on their selectivity. A separate classifier was trained for increasingly larger populations of neurons with neurons added in from least to most selective.

The classifier's accuracy increased as neurons with low individual classification accuracy were incorporated (Fig. 2g-h). Using the least selective 40% of neurons, the classifier for choice reached an overall accuracy of ~75%, despite the fact that the most selective neuron included only reached an individual accuracy of 52.5%. While this result may appear counterintuitive at first, there are several ways in which it can occur. For example, consider a neuron which is active on only 10 out of 200 trials. However, all ten of the trials during which the neuron is active result in a left choice. The neuron's individual classification accuracy will necessarily be low: the best it can achieve is 52.5% (chance performance on the 190 trials during which it is silent and perfect performance on the 10 trials during which it is active). However, a population classifier combining many such neurons, each active on a different subset of 10 trials, may still achieve a high accuracy overall.

As another example, consider a neuron which is active on all trials with 10% higher activity on left trials than on right trials. In a noiseless environment, such a neuron would result in perfect classification accuracy. However, as noise increases, the neuron's classification accuracy will quickly fall, and if the noise is much larger than the difference in average activity between left and right trials, the neuron will display poor individual accuracy. Nevertheless, if a population classifier averages the responses of many such neurons (assuming independent noise), its accuracy will increase steadily as more neurons are included (and as more noise is averaged out), eventually reaching perfect classification accuracy.

We tested if our classifier's performance was due to weak information in single cells or correlations between neurons by shuffling the trial identities within a trial-type category

separately for each neuron, thus preserving each neuron's activity but breaking inter-neuronal correlations (i.e., simulating a pseudo-population). In the shuffled case, the classifier performed with high accuracy, indicating that many cells beyond the highly selective ones contained small amounts of choice- and net evidence-related information (Supplementary Fig. 5i). Importantly, these analyses indicate that our classifier was not dependent on correlations but they do not rule out that population-level correlations may play a key role in information encoding.

M.4.4.5 Classification of cue sequences (Fig. 6a-b, Supplementary Figs. 5d, 10g-h)

To determine if the identity of previous cues could be read out at a current epoch, given identical current cue and net evidence (Fig. 6a-b, Supplementary Fig. 5d), we examined pairwise trial-trial population activity correlations. At each of cues 3-6, we separated trials into those which contained the patterns *LRL* (left-right-left) and *RLL* (right-left-left) or *LRR* and *RLR* with the last cue in the pattern matching the currently analyzed cue. For example, at the third cue, the pattern *LRLRRR* would be a match, while at fifth cue, the pattern *RRLRLR* would be a match. To rule out predictability due to differences in net evidence, a subset of trials in which the distribution of net evidence was equivalent across the two groups was used. This procedure resulted in eight groups (2 choices x 4 patterns). At each cue epoch, pairwise trial-trial correlations of the mean neuronal population activity vector for *n* simultaneously imaged neurons were calculated for trials with the same previous cue or different previous cues. To predict the previous cue based on these correlations, in a leave-one-out fashion, the mean population activity correlations between the test trial and all other trials with a left or right previous cue were calculated. Accuracy was compared to the accuracies from 1000 shuffles of the labels assigning previous cues to trials.

The activity difference due to different previous cues could reflect different internal accumulated evidence values owing to unequal weighting of early and late cues. However, similar results were obtained when we restricted our analysis to the fifth and sixth cues, which were weighted similarly behaviorally, and when we considered data from a mouse that weighted all cues equally (Supplementary Fig. 2d, see mouse marked in red; for both cases: $p < 0.05$, comparison of pairwise activity correlations for trials with the same or different previous cue, two-sample KS test; Supplementary Fig. 5d).

M.4.5 Visualization of activity in high-dimensional state space via factor analysis (Fig. 5a, Supplementary Fig. 9)

For visualizations using factor analysis (Fig. 5a, Supplementary Fig. 8), dimensionality was reduced to 5-factors and two were selected for visualization.

M.4.6 Clustering methods (Figs. 3, 4, 6c, Supplementary Figs. 6, 7, 9a-c, 10c, f)

We used clustering to reduce the dimensionality of the population activity without inclusion of information about behavioral parameters and without encouraging projection of the data onto dimensions which maximize variance due to specific task features, such as choice. We used clustering because it did not assume linearity in the data structure and facilitated analyses by discretizing activity patterns, allowing for the calculation of transition probabilities between discrete activity states. However, clusters were not considered an indication of discreteness of the underlying activity patterns. Clustering provided the advantage of allowing analysis of the rules that govern the transitions between activity patterns from moment-to-moment within a trial. Although other analysis approaches may have been possible, standard methods have not been developed previously to study the rules governing transitions between transient and largely different patterns of population activity, as we observed here.

M.4.6.1 Pre-processing for clustering

Prior to clustering, each spatially binned trial was divided into ten non-overlapping epochs, corresponding to the start of the trial, cues 1-6, the early and late delay, and the turn. For the trial-start epoch, neuronal activity was averaged over the 4 spatial bins (15 cm) immediately preceding onset of the first cue. For cues 1-6, activity was averaged over the third quarter of each cue's presentation (4 spatial bins). For the early and late delay, respectively, activity was averaged across the 4 spatial bins beginning 15 and 37.5 cm after offset of the final cue. For the turn, activity was averaged across the final 4 spatial bins in the maze. Each epoch corresponded to approximately 200 ms.

M.4.6.2 Clustering via affinity propagation

Within each epoch, trials were clustered into groups based on their neuronal activity using affinity propagation (code from Brendan Frey)[23]. Affinity propagation has two inputs: a distance matrix and a 'preference' for each data point. We calculated the distance as the negative sum of pairwise Euclidean distance and one minus the pairwise cosine similarity between every trial in the *n*-dimensional activity space (each dimension is the activity of a single neuron). In contrast to other clustering methodologies, such as k-means clustering, the preference parameter does not specify the number of clusters, but rather a general range. For example, in our experience, clustering on different datasets using the same preference parameter can result in anywhere from 1 to 30 clusters. To determine the preference parameter, we calculated the number of clusters generated across a range of preference parameter values. The tenth percentile of the difference matrix was within a stable range, such that small modifications in the preference parameter did not greatly influence the number of clusters identified. The choice of this parameter and the resulting number of clusters had little impact on our results (Supplementary Fig. 6j).

M.4.6.3 Transition probabilities between clusters (Figs. 3, 4)

To calculate transition probabilities between clusters (cluster *a* at epoch 1, cluster *b* at epoch 2), we calculated the fraction of trials in cluster *a* which were also in cluster *b*. To superimpose behavioral variables on clusters (Figs. 3a-d, 4a-c), for each cluster, we calculated the fraction of trials that had a given feature. To validate that the clustering found meaningful groups, we analyzed whether the distribution of behavioral variables across clusters at a given epoch was significantly different than chance. To calculate chance, we shuffled the cluster labels such that the number of trials in each cluster was maintained, but with the trials assigned to a given cluster determined randomly (Supplementary Fig. 6a-d). Significance was established by summing the absolute difference in behavioral variables from the expected uniform distribution for the real data and comparing this total difference to the distribution of the same metric obtained from 1000 shuffles (Supplementary Fig. 6b, d).

M.4.6.4 Clustering based on all time points together (rather than epoch-by-epoch clustering) (Fig. 3m, Supplementary Fig. 6f)

To determine if clustering separately at each epoch resulted in multiple clusters with similar population activity patterns in different epochs, we also performed clustering on all epochs together. While the total number of clusters for epoch-by-epoch-based clustering was larger than those for all-epoch-based clustering (ratio: 1.6 ± 0.06), the ratio was relatively low, suggesting that activity patterns were distinguishable across epochs. Self-transitions were identified as transitions across epochs in the all-epoch clustering in which a trial was in the same cluster at two consecutive epochs (Supplementary Fig. 6f). To determine whether the variability of trials in cluster space changed over the course of a trial, we also calculated the fraction of the total clusters explored at each maze epoch (Fig. 3m). To remove outliers, only clusters containing three or more trials at a given epoch were counted as visited.

M.4.7 Classifiers based on activity in cluster-space

M.4.7.1 Classification of the cluster identity at past and future epochs based on the cluster identity at the current epoch (Figs. 4d-e, 6c, Supplementary Figs. 6j, 10c)

To predict the past or future cluster identity during a trial at a certain epoch (epoch *j*), given the cluster identity at another epoch (epoch *i*), we implemented a classifier built in cluster-space (Fig. 4d-e). In a leave-one-out fashion, we limited our analysis to only the trials which were in the same cluster as the test trial at epoch *i*. Of those trials, we then asked which cluster was most common at epoch *j*. The classifier then predicted that the test trial would also be in that cluster at epoch *j*. Prediction accuracy was calculated as the fraction of predicted clusters that matched the actual cluster. In Fig. 4d-e, clustering was performed separately on correct left 6-0 and right 0-6 trials to rule out structured variability due to different cues or behavioral choices. Accuracy was compared to 1000 shuffles of the assignment of trials to clusters. Cluster assignments were

shuffled independently at each epoch. This shuffle maintains the distribution of trials across clusters. This classifier was identical to the "cluster only" classifier used in Fig. 6c, except in Fig. 6c, it was applied to all trials together, independent of evidence or choice. The other classifiers used in Fig. 6c followed a similar logic. For the "chance" classifier, the same procedure was followed, except all trials except for the test trial were used to determine the most likely cluster at epoch *j*, not just those in the same cluster at epoch *i*. For the "cue only" classifier, only those trials with the same current cue (left or right) were used to determine the most likely cluster at epoch *j*. For the "cue + cluster" classifier, only those trials with both the same current cue as the test trial and which were in the same cluster at epoch *i* were used to determine the most likely cluster at epoch *j*.

M.4.7.2 Classification of past and future cluster identities with simulated pseudo-populations (Fig. 4e)

The temporal structure we observed could be caused by the prolonged activity of individual neurons, either due to persistent activity in the underlying neuronal activation or to the prolonged decay kinetics of the calcium indicator. To test if prolonged activity patterns could account for temporally structured and predictable trial-trial variability, we shuffled the trial labels independently for each neuron across trials with the same choice and sequence of evidence cues (e.g. correct left 6-0 trials; Fig. 4e). This shuffle therefore breaks neuron-neuron correlation structure, but maintains the temporal structure of each neuron's individual activity, simulating a pseudo-population. Following this shuffle, clustering was performed (separately for correct 6-0 left and 0-6 right trials), and past and future activity patterns were classified as described above using the "population activity only" classifier. In contrast to the unshuffled case, we were unable to predict past and future activity patterns over more than one epoch (Fig. 4e), suggesting that the predictability we observed in the real data could not be explained by prolonged activity or slow indicator kinetics.

M.4.7.3 Classification of past and future cluster identities based on behavioral data (Supplementary Fig. 8a-c)

To determine the fraction of past and future predictability accounted for by variability in behavioral parameters across clusters, we used logistic regression to compare predictability based only on behavioral parameters to that based on both behavioral parameters and neuron activity-defined clusters (Supplementary Fig. 8a-c). To allow for binary classification, only trials whose cluster identity contained either the most or second most trials during the prediction epoch were included. We used all recorded behavioral parameters (x/y position, spherical treadmill rotational velocities, view angle) for this analysis, which together account for the mouse's general running pattern and visual scene. At each epoch, we trained a logistic regression model to predict the activity pattern at the current epoch or at another epoch in the past or the

future based on either the behavioral variables alone (behavior only) or the behavioral variables in addition to the current cluster identity (behavior + neuronal activity clusters). Note that for the same epoch, we did not include neuronal clusters as an explanatory variable as they were identical to the response variable in that case. To compare model performance across the two cases, which have different numbers of predictors, we used adjusted R^2. Independent models were created for each combination of epochs and combined based on the number of epochs separating the predictor and response (e.g. cue 1 and cue 2 are separated by 1 epoch, while trial start and turn epochs were separated by 9 epochs). Separate models were calculated for left 6-0 and right 0-6 trials to rule out behavioral variability induced by the choice. Results were qualitatively similar when models included linear interaction terms and quadratic terms.

M.4.8 Analysis of the overlap of active neurons across clusters (Fig. 3k)

To calculate the overlap fraction for active neurons between clusters, each neuron's activity across all trials was z-scored. Within each cluster, each neuron's mean z-scored activity was calculated and compared to a z-score activity threshold of 1.5 (Fig. 3k), though similar results were obtained using different thresholds. Neurons whose mean z-scored activity was above this threshold were determined to be active. Using correct left 6-0 and right 0-6 trials separately to rule out differences due to evidence cues, the pairwise overlap fraction between all clusters at each epoch was calculated as $overlap\ index = \frac{number\ of\ neurons\ active\ in\ both\ clusters}{number\ of\ neurons\ active\ in\ either\ cluster}$. For the intra-cluster measure, trials within a cluster were randomly divided into two groups, the mean activity within each group was calculated for each neuron, compared to the z-score threshold, and the overlap fraction between the two groups was calculated. This process was repeated 100 times and the results were averaged. To reduce variability due to low trial numbers, only clusters (intra-cluster measure) or cluster pairs (inter-cluster measure) with greater than 20 combined trials were included. To determine the shuffled overlap fraction, the cell labels within each cluster were randomly assigned 1000 times, and the inter-cluster overlap fraction was recalculated.

Supplementary References

51. Harvey, C. D., Collman, F., Dombeck, D. A. & Tank, D. W. Intracellular dynamics of hippocampal place cells during virtual navigation. *Nature* **461,** 941–946 (2009).
52. Aronov, D. & Tank, D. W. Engagement of Neural Circuits Underlying 2D Spatial Navigation in a Rodent Virtual Reality System. *Neuron* **84,** 442–456 (2014).
53. Dombeck, D. A., Harvey, C. D., Tian, L., Looger, L. L. & Tank, D. W. Functional imaging of hippocampal place cells at cellular resolution during virtual navigation. *Nat. Neurosci.* **13,** 1433–1440 (2010).
54. Pologruto, T. A., Sabatini, B. L. & Svoboda, K. ScanImage: flexible software for operating laser scanning microscopes. *Biomed Eng Online* **2,** 13 (2003).
55. Greenberg, D. S. & Kerr, J. N. D. Automated correction of fast motion artifacts for two-photon imaging of awake animals. *Journal of Neuroscience Methods* **176,** 1–15 (2009).
56. Shi, J. & Malik, J. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **22,** 888–905 (2000).
57. Murphy, K. P. *Machine learning: a probabilistic perspective*. (2012).
58. Smola, A. & Vapnik, V. Support vector regression machines. *Advances in neural information processing …* (1997).
59. Chang, C. C. & Lin, C. J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and …* (2011).
60. Chang, C.-C. & Lin, C.-J. Training v-Support Vector Regression: Theory and Algorithms. *http://dx.doi.org.ezp-prod1.hul.harvard.edu/10.1162/089976602760128081* **14,** 1959–1977 (2006).