



Supplementary Materials for

Selection of experience for memory by hippocampal sharp wave ripples

Wannan Yang *et al.*

Corresponding author: György Buzsáki, gyorgy.buzsaki@nyulangone.org

Science **383**, 1478 (2024)
DOI: 10.1126/science.adk8261

The PDF file includes:

Materials and Methods
Figs. S1 to S20
References

Other Supplementary Material for this manuscript includes the following:

MDAR Reproducibility Checklist

Materials and Methods

Implantation and recording

The data in this paper includes 3 different datasets. Dataset 1 was the main dataset used for both visualization and quantification. Only 1 session was taken from datasets 2 and 3, and shown in Fig. S2 only for visualization. The purpose of the visualization is to demonstrate that the trial block-specific population activity pattern can be observed in diverse tasks and different rodent species.

Dataset 1: As described in (11), chronic recordings were performed from freely moving adult C57BL/6J mice. The mice were anesthetized with 1.5–2% isoflurane and implanted with high-density silicon probes, mounted on a microdrive. The ASSY Int64-P32-1D or ASSY Int128-P64-1D silicon probes (Diagnostic Biochips) were dual-sided designs, equipped with recording sites on both sides of each shank. The spacing between shanks was 250- μ m. Craniotomy was drilled at -2-mm anteroposterior (AP) and 1.7-mm mediolateral and the probe was lowered to the deep neocortical layers, and the drive was cemented to the skull. A stainless-steel screw was placed over the cerebellum for grounding and reference. Craniotomies were sealed with a mix of dental wax and mineral oil, and a copper mesh cage was constructed to provide electrical and mechanical shielding. Postoperatively, animals received a single intramuscular injection of 0.06-mg/kg of buprenorphine (0.015-mg/ml) as needed for the next 1–3d. Animals were allowed a 7-d recovery period before the start of experiments. After a 7-d recovery period, neural signals were recorded in the homecage while probes were lowered into the CA1 pyramidal layer, which was identified physiologically via the sharp wave polarity reversal. Neural data were amplified and digitized at 30-kHz using Intan amplifier boards (catalog no. RHD2132/RHD2000, Evaluation System). The complete dataset is available at <https://dandiarchive.org/dandiset/000552/0.230630.2304>.

Dataset 2: As described in (17), male Long-Evans rats (250-350g) were bilaterally implanted in the dorsal hippocampus with two 8-shank (BUZ64) silicon probes. Each shank of the 8-shank silicon probes had 8 sites. All sites were vertically staggered along the shank with 20- μ m spacing between sites. Each site had an area of 160- μ m² and an impedance of 1–3-M Ω . In each rat, 50- μ m wires were placed gently abutting the left, and right mastoid masseter muscle as well as the back neck muscle for electromyographic (EMG) recordings used in sleep classification. All silicon probes were implanted parallel to the septo-temporal axis of the dorsal hippocampus. Finally, each rat was fitted with a small 3-dimensional accelerometer to record the animals' movement, or its absence, during sleep. Two stainless steel screws implanted above the cerebellum were used for referencing and grounding. Implantation was performed under isoflurane (1-1.5%) anesthesia. Each silicon probe was attached to a micromanipulator and lowered over several days until hippocampal layer CA1 was reached as determined by the appearance of hippocampal CA1 sharp wave ripples and pyramidal cell activity. Probe placement was histologically confirmed *post hoc*. The data is available at: https://buzsakilab.nyumc.org/datasets/GrosmarkAD/Achilles/Achilles_10252013/

Dataset 3: In this new dataset, procedures were identical to those for dataset 1 with the following exceptions: an 8-shank, single sided 128 channel probe (128-8, Diagnostic Biochips)

was implanted transversely into the right dorsal hippocampus (medial edge 2.2-mm posterior, 1.1-mm lateral of bregma, lateral edge 1.5-mm posterior and 1.8-mm lateral of bregma). Peri- and postoperative analgesia was performed with subcutaneous Ketoprofen 10 mg/kg daily for 3 days. The data is available at https://buzsakilab.nyumc.org/datasets/HainmuellerT/TH11_210605/.

Behavior

Dataset 1: Mice were trained on a spatial alternation task in a figure-eight maze and linear maze (11). Animals were water-restricted before the start of experiments and familiarized with the maze (Fig. 1b) raised 61 cm above the ground. Several days after the start of water deprivation, animals were shaped to visit alternate arms between trials to receive a water reward in the first corner reached after making a correct left/right turn. A 5-s delay in the start area was introduced between trials. Infrared (IR) sensors were used to detect the animal's progression through the task and 3D-printed doors mounted to servo motors were opened/closed to prevent the mice from backtracking. IR sensors and servo motors were controlled by a customized Arduino-based circuit. The position of head-mounted red LEDs (light-emitting diodes) was tracked with an overhead Basler camera (catalog no. acA1300-60 gmNIR, Graftek Imaging) at a frame rate of 30-Hz, and tracking data were aligned to the recording via TTL pulses from the camera, as well as a slow pulsing LED located outside the maze. In all sessions that included behavior, animals spent ~120-min in the homecage before running on the maze and another ~120-min in the homecage afterward. All behavioral sessions were performed in the mornings (the start of the dark cycle). Only sessions with more than 150 cells and >20 trials were selected for analysis. n = 26 sessions from 6 animals (21 out of 26 sessions were figure-8 maze sessions, and 5 of them were linear maze sessions).

Dataset 2: As described in (17), a rat (Achilles) was extensively handled both before and after surgery. Water restriction was initiated one week after the surgery; animals were restricted to 90% of their initial body weight and given one day a week of ad lib water access. The rat was well acclimatized and recorded in the ‘familiar’ room (where all sleep recordings were performed) for at least one week prior to novelty maze sessions. This time was used to gradually lower the silicon probes into position. All hippocampal, EMG and accelerometer signals were recorded continuously at 20-kHz using four identical 256-channel Amplipex Systems (16-bit resolution; analog multiplexing; one in the familiar room and one in each of the two novelty rooms). Only one session (Achilles_10252013) from this dataset was used to produce the visualization in Fig. S2.

Dataset 3: The mouse was handled and familiarized with the maze for multiple days prior to the commencement of experiments. It was then trained to forage for rewards on the radial 8-arm maze. The maze consisted of eight concentrically arranged arms of hard plastic (each arm 40-cm long) with 50-cm high side walls that were elevated 30-cm above the floor. In our version of the task, three out of the eight arms of the maze were chosen semi-randomly every day, with the constraints that a maximum of two adjacent arms were rewarded and that at least 2 arms had to be different from those of the previous day. These three arms were initially ‘baited’ with water rewards (they were released through a solenoid valve that was automatically triggered when the animal crossed an infrared barrier in the middle of the arm) but only re-plenished after the animal had visited all rewarded arms. There were no discrete trials and the animals were allowed to continuously forage the maze for 20-minutes. The animal was then placed in its home cage for 2-hours, during which neuronal activity was recorded. After this, the animal was given another 20-

minute session in the radial arm maze to assess for memory retention. Reward locations were kept constant between the two sessions of a given day but changed between subsequent days. Electrophysiological and behavior recording procedures were identical to those outlined for Dataset 1 above. Only one session (TH11_210605) from this dataset was used to visualize position coding by UMAP of a radial arm maze (Fig. S2).

Unit isolation and classification

Spikes were extracted and classified into putative single units using KiloSort1 and manual curation was performed in the Phy2 software with the aid of customized plugins (<https://github.com/petersenpeter/phy2-plugins>), as described in ref. (11).

Place field and spatial information

Place field analysis was done as in ref. (17). The code is available at <https://github.com/valegarman/HippoCookBook>. The firing rate distribution within place fields ('rate map') was generated by first binning spiking data into 5-cm wide bins, and spike counts were normalized by time occupancy (smoothing size: 2-bins). Trials for forward and backward directions were evaluated separately. The field boundaries were defined when the rate decreased below 20% of the peak firing rate. Place fields cleared all criteria when they were between 8.75-cm and 75-cm wide, had a minimum peak firing rate of 3-Hz, and had a spatial coherency > 0.7.

Spatial information was calculated in bits per spike as:

$$SPI = \sum_i^N pi \frac{\lambda_i}{\lambda} \log_2 \frac{\lambda_i}{\lambda} \quad (1)$$

where λ_i is the mean firing rate of a unit in the i^{th} bin, λ is the overall mean firing rate and pi is the probability of the animal being in the i^{th} bin (occupancy in the i^{th} bin/ total recording time).

Poisson process for modeling single cell activity across trials

The simulated spike was modeled based on a Poisson process:

$$Pr(X = \kappa) = \frac{\lambda^\kappa e^{-\lambda}}{\kappa!} \quad (2)$$

The procedure to generate Poisson spikes followed ref. (61). For each simulated neuron, the spike train progresses through time in small windows of dt (1-ms), and spikes were generated via Poisson process using the cross-trial mean firing rate $fr(pos)$ of the real neuron at position pos .

Theta-cycle detection

Theta cycle detection was done following ref. (17). As theta-phase shifts along the radial axis, a channel with a positive sharp wave (that is, above the center of the pyramidal layer) was selected to ensure consistency of extracted phases across recordings. Broadband LFP was bandpass filtered between 6 and 12-Hz using a fourth-order Chebyshev filter. The Hilbert transform of the filtered signal was computed, and its absolute value and angle at each timepoint were taken to be the theta-band amplitude and phase, respectively. Intervals with theta-band amplitude 1 s.d. above the mean were considered for theta cycle detection. Within these intervals, timepoints, where the phase crossed 0° , were identified as peaks of theta, and timepoints of consecutive theta peaks were considered the onsets and offsets of individual theta cycles (peaks are at 0° and 360° and troughs at 180°). Only theta cycles occurring within identified awake periods (see section State scoring) were considered for analysis.

State scoring

State scoring was performed as described previously (<https://github.com/buzsakilab/buzcode/blob/dev/detectors/detectStates/SleepScoreMaster/SleepScoreMaster.m>). First, the LFP was extracted from wideband data by low-pass filtering (sinc filter with a 450-Hz cut-off band) and downsampling to 1,250-Hz. Three signals were used for state scoring: broadband LFP, narrowband theta frequency LFP and electromyogram (EMG). Spectrograms were computed from broadband LFP with fast-Fourier transform in 10-s sliding windows (at 1-s) and principal component analysis (PCA) was computed after a z transform. The first PC reflected power in the low (<20-Hz) frequency range, with oppositely weighted power at higher (>32-Hz) frequencies. Theta dominance was quantified as the ratio of powers in the 5 to 10-Hz and 2 to 16-Hz frequency bands. EMG was estimated as the zero-lag correlation between 300 and 600-Hz filtered signals across recording sites. Soft sticky thresholds on these metrics were used to identify states. Briefly, high LFP PC1 and low EMG were taken to be NREM, high theta and low EMG were considered to be REM and the remaining data were taken to reflect the awake state. All assignments were inspected visually and manually curated wherever appropriate (<https://github.com/buzsakilab/buzcode/blob/dev/GUITools/TheStateEditor/TheStateEditor.m>).

Ripple detection

Ripple events were conditioned on the coincidence of both population synchrony events, and LFP detected ripples as described in ref. (17). For each session the combined spiking of all recorded CA1 pyramidal cells was binned in 1-ms bins and convolved with a 15-ms Gaussian kernel. For each session, a trigger rate was defined as being 3 standard deviations above the mean of all 1-ms bins within NREM epochs of both PRE and POST epochs combined. Putative population synchrony events were detected when the smoothed firing rate vector crossed the trigger rate. The beginning and end of the putative synchrony events were defined as the time points at which the convolved firing rate vector returned to the mean of all within-NREM firing rate bins. Independently, ripple events were detected from the pyramidal layer LFP. Population synchrony events that did not contain at least one LFP-detected ripple were discarded. Population synchrony events that (1) contained at least one LFP-detected sharp wave-ripple, (2) lasted between 50 to 500-ms, and (3) occurred during non-theta or ‘off-line’ states (quite awake immobility or NREM) and 4) in which at least five distinct pyramidal cells each fired at least one spike were termed ‘Ripple events’ and considered for further analysis. This ‘five pyramidal cells’ inclusion criterion was used as it is the minimum number of cells needed to establish significance

for sequence relationship between cells at $p < 0.05$ ($5! = 120$ possible orderings, of which half are mirror ‘forward’ and ‘reverse’ sequences and hence equivalent to the sequence analysis, resulting in 60 unique orderings, $1/60 = 0.0167$).

SeqNMF

SeqNMF algorithm was implemented following ref. (12) and code from (<https://github.com/FeeLab/seqNMF>). Briefly, seqNMF started by taking a data matrix X which contains the activity of N neurons at T time points. If the neurons exhibit a single repeated pattern of synchronous activity, the entire data matrix can be reconstructed using a column vector w representing the neural pattern and a row vector h representing the times and amplitudes at which the pattern occurs. Data matrix X was mathematically reconstructed as the outer product of w and h . If multiple component patterns are present in the data, then each pattern can be reconstructed by a separate outer product, where the reconstructions are summed to approximate the entire data matrix as follows:

$$X_{nt} \approx \sum_{k=1}^K W_{nk} H_{kt} = (WH)_{nt} \quad (3)$$

Where X_{nt} is the $(nt)^{th}$ element of matrix X , which represents the activity of neuron n at time t . To store K different patterns, W is a N by K matrix containing the K exemplar patterns or sequence factors (Suppl Fig. 1A) and H is a K by T matrix. Here we choose $K = 8$, given the topology of the figure 8 maze.

Assessment of single unit stability across trials

The displacement of the estimated positions of single units relative to the probe sites was measured, following the description from ref. (25). Briefly, the centroid of each single units was measured by a spatial average across electrode positions weighted by the square of the mean waveform amplitude at each electrode. We estimated single unit centroid position (x, y, z) as

$$(x, y, z) = \left[\frac{\sum_{i=1}^N x_i a_i^2}{\sum_{i=1}^N a_i^2}, \frac{\sum_{i=1}^N y_i a_i^2}{\sum_{i=1}^N a_i^2}, \frac{\sum_{i=1}^N z_i a_i^2}{\sum_{i=1}^N a_i^2} \right] \quad (4)$$

Where N is the number of electrodes, x_i is the lateral position of the i^{th} electrode, y_i is the vertical position of the i^{th} electrode, z_i indicates the position of the i^{th} electrode on either the front or back side of the probe (since there were two sides for the dual-sided probe we used), and a_i is the peak-to-peak amplitude of the spike waveform recorded at the i^{th} electrode. Single-unit displacement was then computed using the Euclidean distance between single-unit centroids at different time points of the sessions. The results of this analysis are shown in Fig. S5C,D and Fig. S6D.

Low-dimensional manifold visualization with UMAP (unsupervised)

Neural data were first preprocessed before dimensionality reduction. Neural spiking data (spike count) during maze learning was binned into 100-ms bin. The data was then smoothed using a 500-ms wide Gaussian kernel and z-scored. The UMAP dimensionality reduction algorithm in

ref. (13) was then applied to this matrix. Each point in the low-dimensional manifold corresponds to the population activity at a single time bin in the session. The code is available at https://github.com/lmcinnes/umap/blob/master/doc/how_umap_works.rst.

UMAP hyperparameters used were: n_neighbors = 20, metric = 'cosine', output_metric = 'euclidean', learning_rate = 1.0, init = 'spectral', min_dist = 0.1, spread = 1.0, repulsion_strength = 1.0, negative_sample_rate = 5, target_metric = 'categorical', dens_lambda = 2.0, dens_frac = 0.3, dens_var_shift=0.1.

Low-dimensional manifold visualization with UMAP (supervised)

With supervised methods, data from the same trial were given the same label. Data points that share the same labels were leveraged so that similar points were embedded closer together. For example, visualizations in Fig. 1G, H were produced by supervised UAMP dimensionality reduction, where we exploited this option by using trial block labels in the behavioral data as supervised information. This was implemented by passing the trial block number as target when calling the `fit_transform` function from the UMAP python package https://github.com/lmcinnes/umap/blob/master/doc/how_umap_works.rst.

Decoding maze data with UMAP

Although the mathematical framework for UMAP was based on a theoretical framework in algebraic topology and topological data analysis, at the level of implementation, it boiled down to simply building a neighborhood graph on the data and then optimizing the low-dimensional representation to preserve the structure and relationship of the graph as much as possible. The UMAP dimensionality reduction process was comprised of the following two main steps:

- Step 1, computing a graph representing the data. In default UMAP, a weighted k nearest neighbor (kNN) graph, which connects each datapoint to its k nearest neighbors, was constructed, and used to generate the initial topological representation of the data from the high-dimensional space (Fig. S4B, top panel).
- Step 2, learning a low-dimensional embedding for that graph. UMAP optimizes the low-dimensional representation to have as close a topological representation as possible to that in the original high-dimensional space (measured by cross entropy, Fig. S4B, middle panel).

There were 4 steps for UMAP decoding: (1) Data preprocessing; (2) Manifold embedding with training data; (3) Embedding test data based on the learned metric generated from training data and (4) k-nearest neighbor (kNN) decoding. Neural data were first preprocessed before dimensionality reduction as described above.

To decode the trial block identity from population activity, metric learning was implemented (13). Briefly, metric learning is the process where a labeled set of points is used to learn a metric on data, and the learned metric is used as a measure of distance between new unlabeled points. In our case, a subset of data was selected as training data, based on which the manifold embedding was generated. The training data was supervised with trial block labels by concatenating 5 trials as one block. After generating the manifold with training data, previously unseen (and unlabeled)

test points were then embedded into the learned space using the *transform* function in the Python package (<https://umap-learn.readthedocs.io/en/latest/supervised.html>).

The *scikitlearn* implementation of k-Nearest Neighbors (kNN) algorithm (<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>) was used to decode the trial block identity of the test data embedded in the low-dimensional manifold. The mode of the three nearest neighbors' trial block membership was taken as the decoded trial block. Position bin membership was decoded following an analogous procedure, using the mode of the three nearest neighbors' position bin membership (5-cm bin).

Decoding error and cross-validation method

Decoding error during behavior was cross-validated by tenfold cross-validation, which was implemented by splitting the data into 10 subsamples, each randomly drawn from across the entire session. Decoders were then trained on 9 of the subsamples and tested on the remaining subsample. All 10 subsamples were used as test data during successive iterations. Decoding error was taken as the average error across all 10 trained decoders. For PCA decoding, cross-validation was repeated 1,000 times with overall decoding accuracy taken as the mean across the 1,000 repetitions. Because of computational constraints, cross-validation was repeated 10 times when using UMAP.

To test whether the state space of different trials evolved systematically along a single axis (whereby neighboring trial blocks were embedded closer together in the neural manifold), a targeted validation method was also implemented. In contrast to tenfold cross-validation, *an entire trial block* was held out for each subset of training data. Since the training data did not contain any data that shared the same trial block label with the test data, test data was expected to be decoded to the neighboring trial block only if the population activity of neighboring trial blocks was more similar than that of distant trials (Fig. S7A). Statistical significance for decoding error was determined by comparing the mean decoding error from the original data against the mean decoding error from trial-shuffled data (Fig. S8). Trial-shuffled comparisons were generated by shuffling the trial block identity of the data. This process was repeated 1,000 times with different random seeds.

For comparing decoding accuracies across different subpopulations of place cells and non-place cells, size-matched populations were used as control. Size-matched populations were generated by subsampling the total population of cells for a given session without replacement using different random seeds.

For the downsampling analysis, different subsets of cells were randomly drawn from the total population of cells for a given session without replacement. For each sample size, 1000 different random seeds were used to randomly subsample different cells. Decoding accuracy was taken as the average accuracy across all 1000 different random samples.

Decoding from the original high-dimensional (related to Fig. S4)

For decoding from the high-dimensional space, the mode of the k nearest neighbors' trial block membership was computed (Fig. S4A). Two types of distance metrics, Euclidean and Cosine

similarity were used to build the k-neighbor graph. The main summary statistics in the main figures were performed in the original dimensional space.

Decoding SPW-R content with UMAP

The flow diagram illustrating the procedures for decoding SPW-R events was shown in Fig. S12.

- Step 1: Preprocessing data. Spike trains during maze running were binned in 100-ms time bins, smoothed, and z-scored as described. Spike trains of candidate SPW-R replays were binned in 20-ms time bins, smoothed, and z-scored.
- Step 2: Generating manifold embedding using unsupervised manifold learning (UMAP). The preprocessed data matrix was then passed through the nonlinear dimensionality reduction step by UMAP, which generated a six-dimensional embedding. For further details related to steps 1 and 2, refer to Fig. S12,13.
- Step 3: Embedding candidate ripple events with the manifold generated at step 2.
- Step 4: Measuring distance to manifold and trajectory length for each event.
- Step 5: Testing whether the candidate events were significant according to two criteria: (1) the mean *distance* to the manifold of a candidate should be significantly smaller than shuffled events (circular shuffle). (2) the mean *trajectory length* needs to be significantly smaller (i.e. no large stochastic jumps on the manifold) than shuffled events (shuffle time).
- Step 6: If a candidate event was a significant replay event, proceed to decode both the position bin and trial block membership of the replay event. To decode the position relayed by the candidate event, the population activity during ripple was embedded with the position manifold, where each point on the manifold was associated with a position bin label (positions were binned into 5-cm bins). The low-dimensional embedding along with the position label was used to train a kNN decoder. Similarly, the ripple activity vectors were embedded with the trial manifold for trial block membership decoding (trials were binned into 5-trial bins).

UMAP decoding and negative sampling (refer to Fig. S13)

At the implementation level, UMAP used a sampling-based approach called negative sampling (13) to optimize the low-dimensional embedding. It is a contrastive method to optimize the low-dimensional embedding. Briefly, the intuition behind negative sampling is that data far away from the original high-dimensional space is sampled as negative samples from which data points are repulsed away, whereas similar data in the high-dimensional space are attracted to each other. Based on the principle of negative sampling, we generated negative samples from shuffled data (shuffling the rows and columns of the maze firing rate matrix). With the presence of the negative samples, population vectors similar to the maze manifold were attracted to the maze manifold in the low-dimensional space and those dissimilar to the maze manifold were pulled closer to the negative sample cloud (Fig. S13E-H). Shuffled ripple events were almost exclusively attracted to the negative sample cloud (Fig. S13G,H).

Criteria for significant replay

Candidate events were binned into 20-ms bins. Two kinds of shuffles were used to generate a null distribution for distance-to-manifold and trajectory length, respectively. (1) *Circular shuffle*. The purpose of the cell identity shuffle was to degrade the coordinated population pattern without affecting trajectory length and other properties, such as firing rate and firing rate variance of each neuron. (2) *Temporal shuffle*. It specifically disrupted the sequence structure of the event while preserving the distance-to-manifold. A replay event was classified as a significant event only when its distance to the manifold was significantly shorter than the circular shuffle and its trajectory length was significantly shorter than the temporal shuffle.

Decoding with a Bayesian decoder

For Bayesian decoding, neural spiking data was preprocessed similarly for UMAP decoding. The data was then smoothed using a 1000-ms wide Gaussian kernel. Only data with speed larger than 2-5-cm/s was used to build the template. Bayesian decoding was performed independently for the left and right arm trials. For each ripple event, the position and trial block identity of the event were decoded separately.

Bayesian position decoding

Bayesian decoding of position was performed as described in ref. (17). Briefly, for each of the binned firing rate vectors, Bayesian classification of virtual position was performed utilizing a ‘template’ comprising of all place cell’s firing rate-by-position vector as:

$$Pr_{(pos|spikes)} = \left(\prod_{i=1}^N f_i(pos)^{sp_i} \right) e^{-\tau \sum_{i=1}^N f_i(pos)} \quad (5)$$

Where $f_i(pos)^{sp_i}$ is the value of the firing rate-by-position vector of the i^{th} cell at position pos , sp_i is the number of spikes fired by the i^{th} cell in the time bin being decoded, τ is the duration of the time bin (100-ms) and N is the total number cells. Posterior probabilities were subsequently normalized to one:

$$Pr(pos|spikes) \propto \frac{Pr(pos|spikes)}{\sum_{i=1}^{p_n} Pr(pos_i|spikes)} \quad (6)$$

Where p_n is the total number of position bins (5-cm bins).

Bayesian trial decoding

Bayesian classification of trials utilized a template comprising of all cell’s smoothed firing rate-by-position-trial matrix from each arm as:

$$Pr_{(trial, pos|spikes)} = \left(\prod_{i=1}^N f_i(trial, pos)^{sp_i} \right) e^{-\tau \sum_{i=1}^N f_i(trial, pos)} \quad (7)$$

Where $f_i(trial, pos)^{sp_i}$ is the value of the firing rate-by-trial vector of the i^{th} cell at position pos and trial $trial$, sp_i is the number of spikes fired by the i^{th} cell in the time bin being decoded, τ is the duration of the time bin (100-ms) and N is the total number of place cells. Posterior probabilities were subsequently normalized to one:

$$Pr(pos|spikes) \propto \frac{Pr(pos|spikes)}{\sum_{i=1}^n Pr(pos|spikes)} \quad (8)$$

Mixed-effect linear regression analysis

To quantify the relative magnitude of the effects of different variables to explain the trial block distribution pattern of pos-sleep replay, we applied mixed-effect linear regression analysis (MATLAB function *fitlme*). The model was fitted with the following formula:

$$\begin{aligned} post\ sleep\ trial\ replay\ pattern &\sim maze\ replay\ trial\ pattern + \\ &theta\ cycle\ trial\ pattern + theta\ power\ trial\ pattern + pre\ sleep\ trial\ pattern + \\ &shuffle\ trial\ pattern + (session_{ID}) \end{aligned} \quad (9)$$

Statistical analyses

All statistical analyses were performed with custom-written scripts in MATLAB and Python. Power analysis was not used to determine sample sizes.

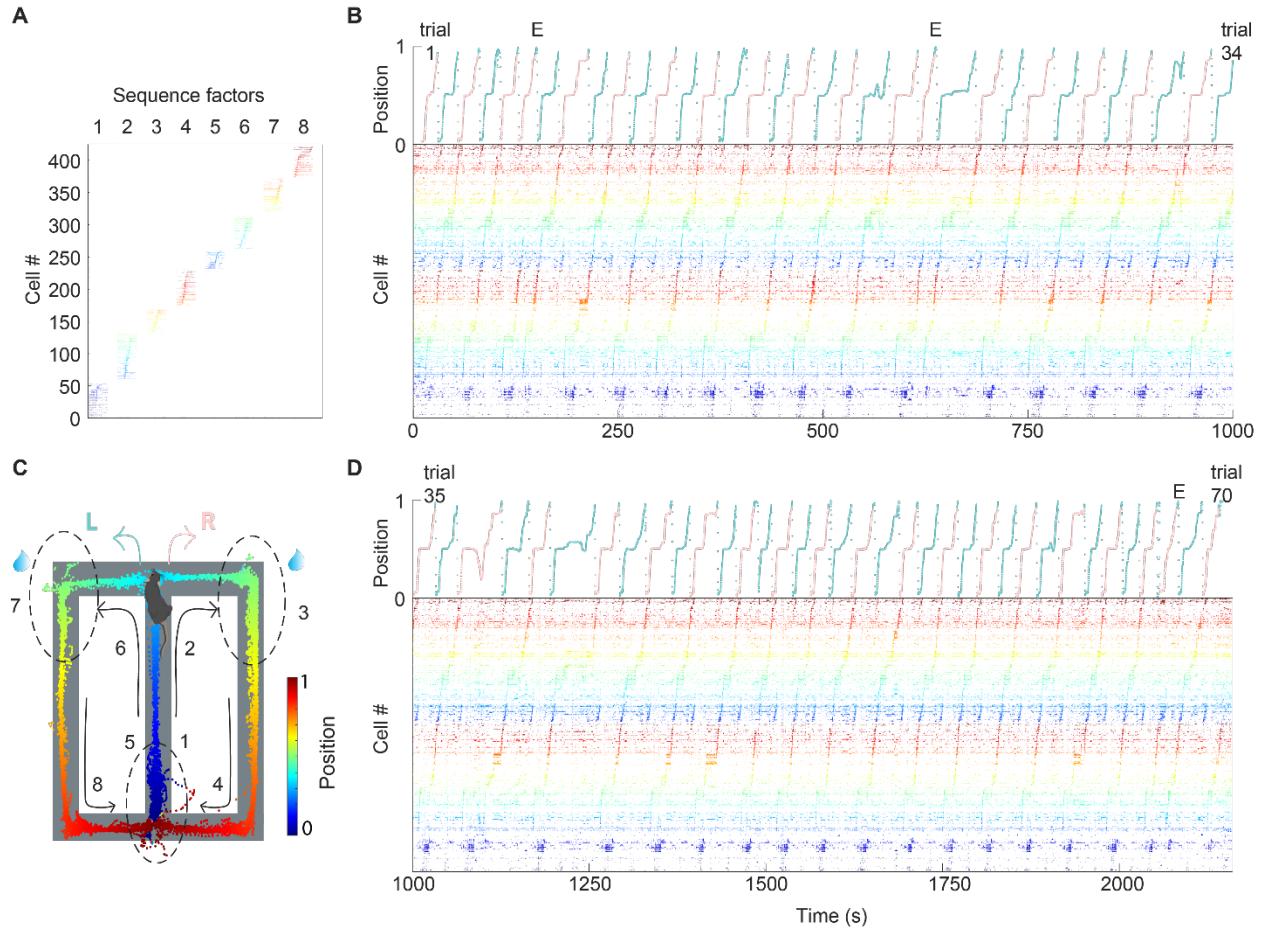


Fig. S1. Neural spiking data of the entire example session sorted by seqNMF.

(A) Eight sequence factors identified by seqNMF.

(B) Trials 1-34 (out of 70 trials in total) during the figure-8 maze task (shown in Fig.1D). Top, linearized position of the animal. Red, right traversal; green, left traversal. Bottom, raster plot of 422 pyramidal cells simultaneously recorded from the right dorsal CA1 region, sorted by seqNMF, and color-coded according to the linearized position (the same color scheme as in C).

(C) Figure-8 maze task, where the mouse alternates between left and right arms to gain water reward (blue droplets). Animal's trajectory in the maze was colored by its linearized position. Numbers 1 to 8 indicate the mapping of the 8 seqNMF sequence factors (in panel A) onto the corresponding segments of the maze.

(D) Trials 35-70 of the same session as in (B).

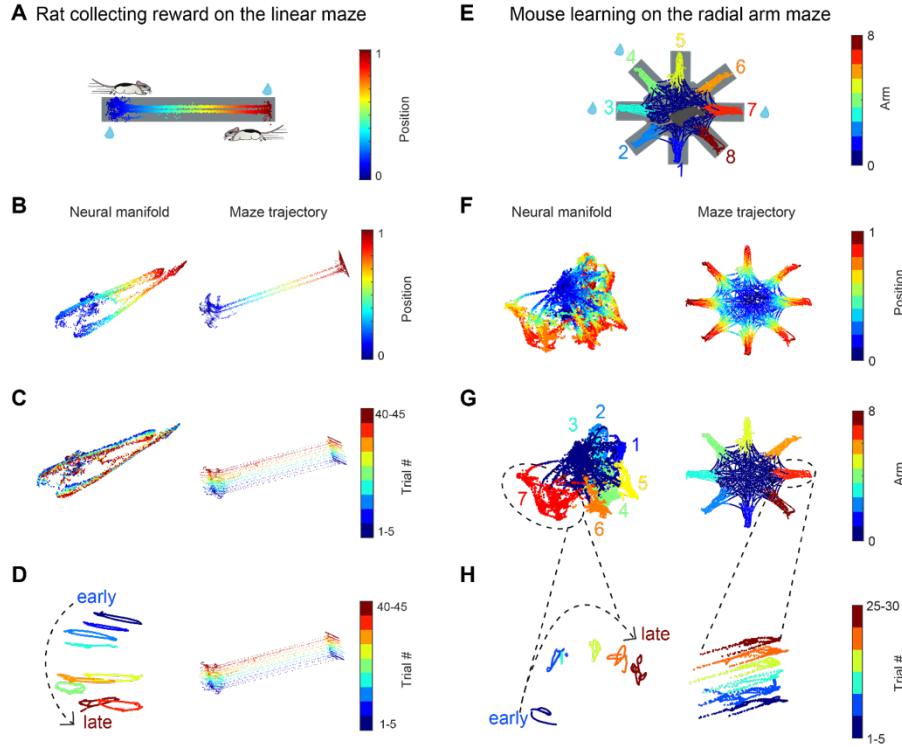


Fig. S2. Trial-specific evolution of neuronal population activity can be observed in different tasks and rodent species.

Maze topology and trial-specific population representation in the manifold of figure-8 maze (Fig.1) were also observed in simpler tasks like a linear maze, as well as more complex tasks like radial arm maze.

- (A) Rat running back and forth between the two reward ports located at either end of the linear track to collect water reward (blue droplets).
- (B) Animal's trajectory along the track was color-coded by animals' linearized position along the track. Only data when the animal's speed was larger than 1 cm/s was included. Left, UMAP embedding of population activity, colored by the rat's linearized position. Right, running trajectory of a rat on the linear maze.
- (C) Same session as (B) but the manifold (left) and maze trajectory (right) were colored by trial block number.
- (D) Same session as (B) and (C), but using semi-supervised UMAP trained on blocks of 5 trials. The neural manifold (left) and maze trajectory (right) were colored by trial block number.
- (E) In the radial-arm maze experiment, mouse collect water reward (blue droplets) from 3 out of the 8 potential reward sites at the end of each arm of the radial arm maze (arms 3,4 and 7 were rewarded in this example session). The animal's trajectory along the track was color-coded by the arm's identity.
- (F) Left, UMAP embedding of population activity, colored by animal's linearized position on each of the 8 arms. Right, running trajectory of a mouse on the radial arm maze.
- (G) Same session as (F) but the neural manifold and maze trajectory were colored by arm identity. 0 represents the center area.
- (H) Since the manifold of figure-8 maze was complex, we focused only on one of the 8 arms to have a clear visualization of the state space evolution with trial number. We focused on arm 7 because the mouse visited this arm for 30 trials on this recording day. The neural manifold (left) was trained using semi-supervised UMAP on blocks of 5 trials. The manifold (left) and maze trajectory (right) were colored by trial block number.

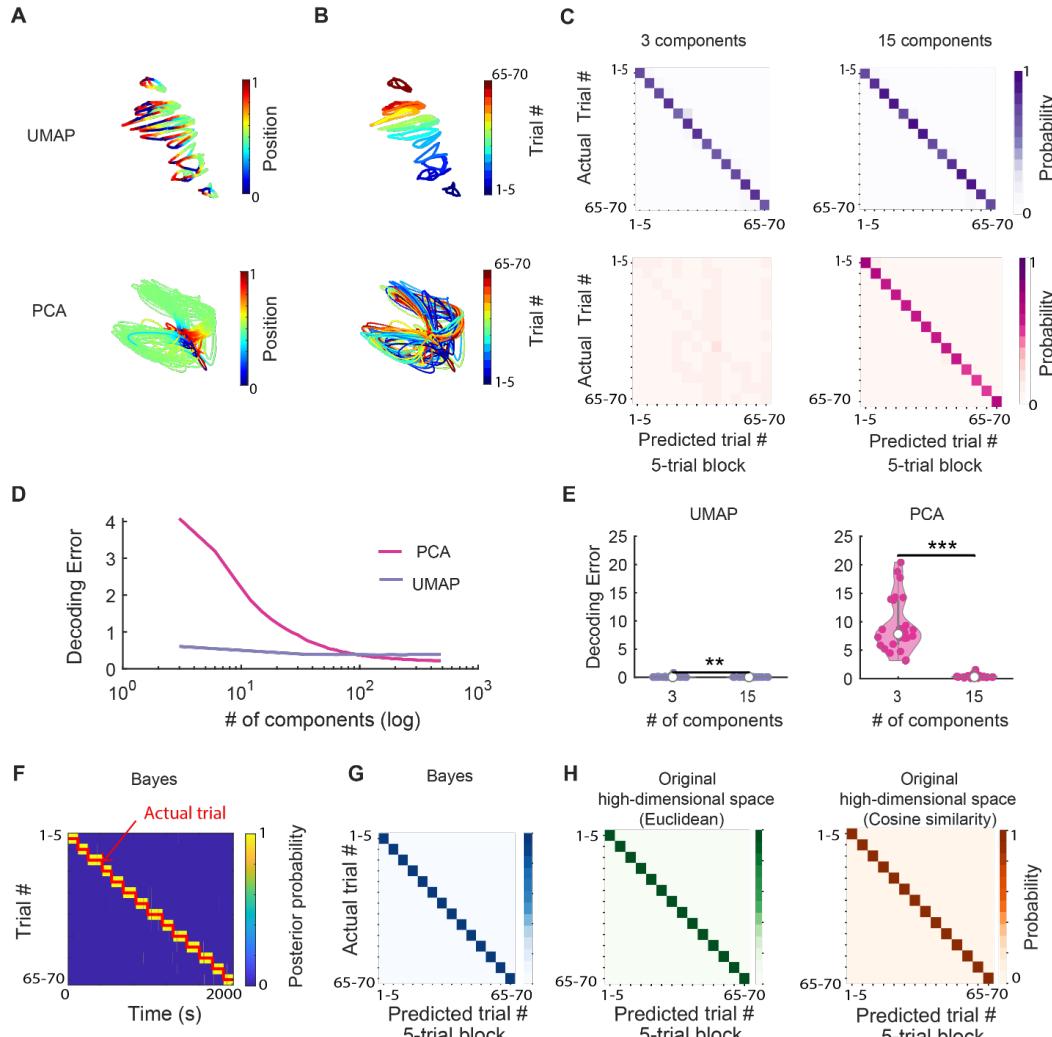
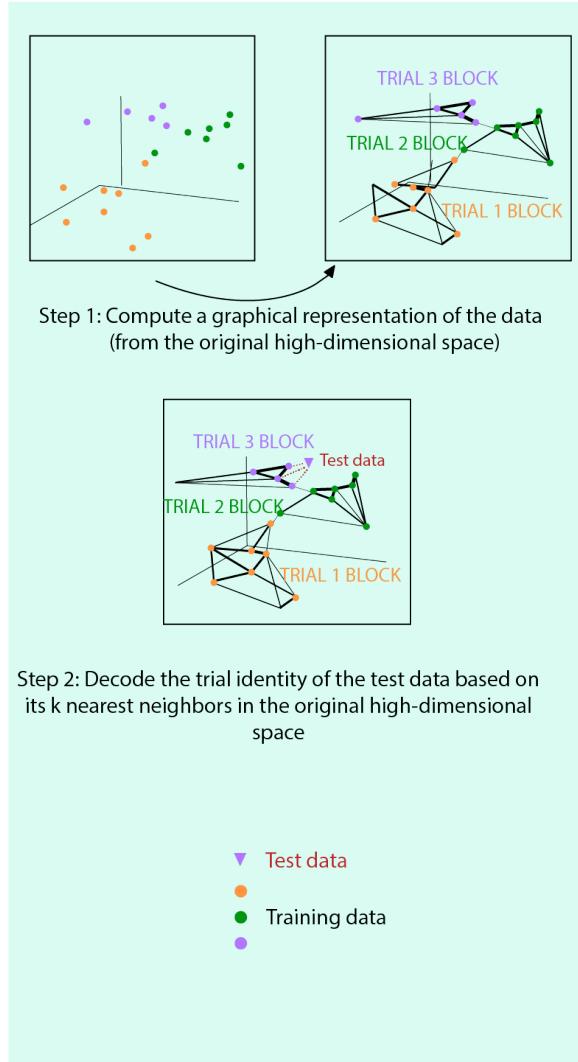


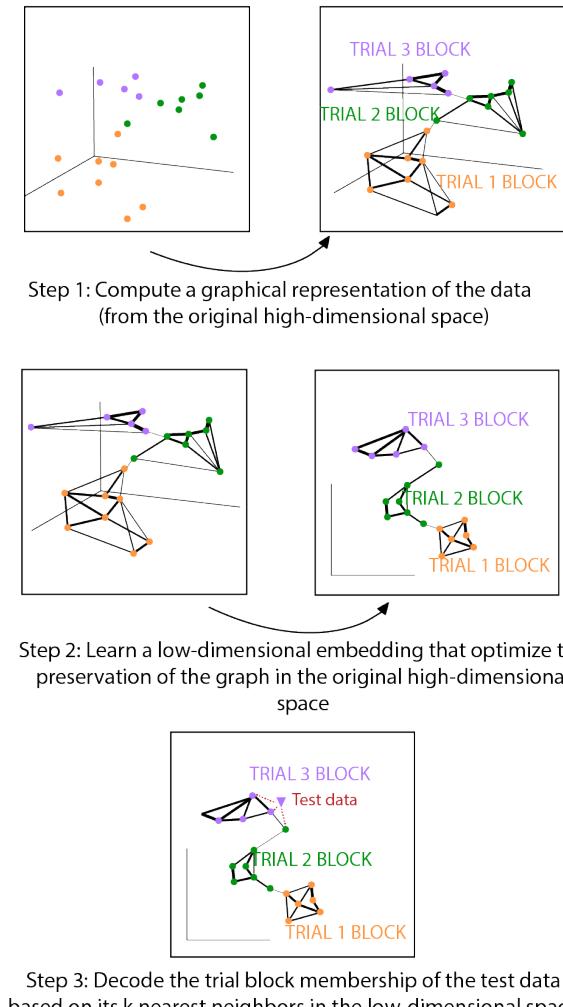
Fig. S3. Trial block identity can be decoded using five different methods.

- (A) Low-dimensional embedding of CA1 population activity, colored by animal's linearized position on the figure-8 maze. Top, UMAP manifold. Bottom, PCA manifold.
- (B) Same as A but colored by trial block number.
- (C) Confusion matrix of trial block decoding result using 3 (left) and 15 (right) components of the low-dimensional data from an example session. Top, UMAP decoding results. Bottom, PCA decoding results.
- (D) Decoding Error as a function of the number of components in the low-dimensional space for both UMAP (purple) and PCA (pink) decoding.
- (E) Summary statistics across the whole dataset, comparing decoding accuracy using UMAP and PCA using the first 3 and 15 components of the low-dimensional space. PCA decoding achieved a similar level of decoding accuracy as UMAP when decoded based on the first 15 principal components. (** P < 0.01 and *** P < 10⁻⁹ for UMAP and PCA respectively; paired t-test; n = 26 sessions from 6 animals).
- (F) Posterior probability of Bayesian trial block decoding across time in one session. Redline, actual trial block number.
- (G) Confusion matrix of trial block decoding results from Bayesian decoding for the same example session shown in (C).
- (H) Confusion matrix of trial block decoding results from the original high-dimensional space for the same example session shown in (C) and (G). Left, Euclidean distance was used as the distance metric. Right, cosine similarity was used as the distance metric.

A Decoding from the original high-dimensional space



B Decoding from the low-dimensional space (UMAP)



C

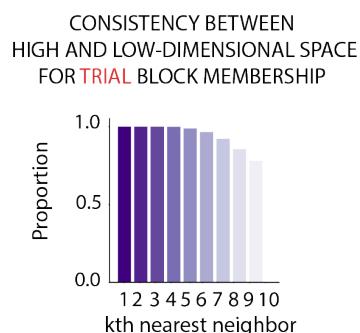


Fig. S4. Consistent trial block membership between the low-dimensional and the original high-dimensional space.

(A) Diagram explaining the main steps of trial block decoding from the original high-dimensional space. Step 1: by default, a weighted k nearest neighbor (kNN) graph, which connects each

datapoint to its k nearest neighbors, was constructed and used to generate the initial topological representation of the training dataset (dots in three different colors). Step 2: the trial block membership of the test data (purple triangle) was decoded by taking the mode of its k nearest neighbors. For example, the example test data point (purple triangle) was decoded to be trial block 3 because all of its three nearest neighbors in the original high-dimensional space belong to trial block 3 (purple).

- (B) Diagram explaining the main steps of trial block decoding from the low-dimensional space generated by UMAP. Step 1 was the same as in panel (A). Step 2: optimize the low-dimensional representation to preserve the topological representation as much as possible to that in the original high-dimensional space. Step 3: The trial block membership of the test data (purple triangle) was decoded by taking the mode of its k nearest neighbors in the low-dimensional embedding. For example, the example test data point (purple triangle) was decoded to be trial block 3 because all its three nearest neighbors in the low-dimensional space belong to trial 3 block (purple).
- (C) To check whether the UMAP dimensionality reduction process introduced distortions that affected the decoded trial block membership (5-trial block), the consistency between the original high-dimensional space and reduced low-dimensional space was plotted. For each data point, the decoded trial block membership of its ten nearest neighbors in the high-dimensional space (*trial_member_high*) and the reduced low-dimensional space (*trial_member_low*) were compared. A data point was considered as having consistent trial block membership if $trial_member_high - trial_member_low = 0$ (i.e. being decoded to the same trial block). For the first four nearest neighbors, the decoded trial block membership between the high and low-dimensional space was 100% consistent. Thus, UMAP dimensionality reduction did not introduce any distortion at the level of trial block decoding.

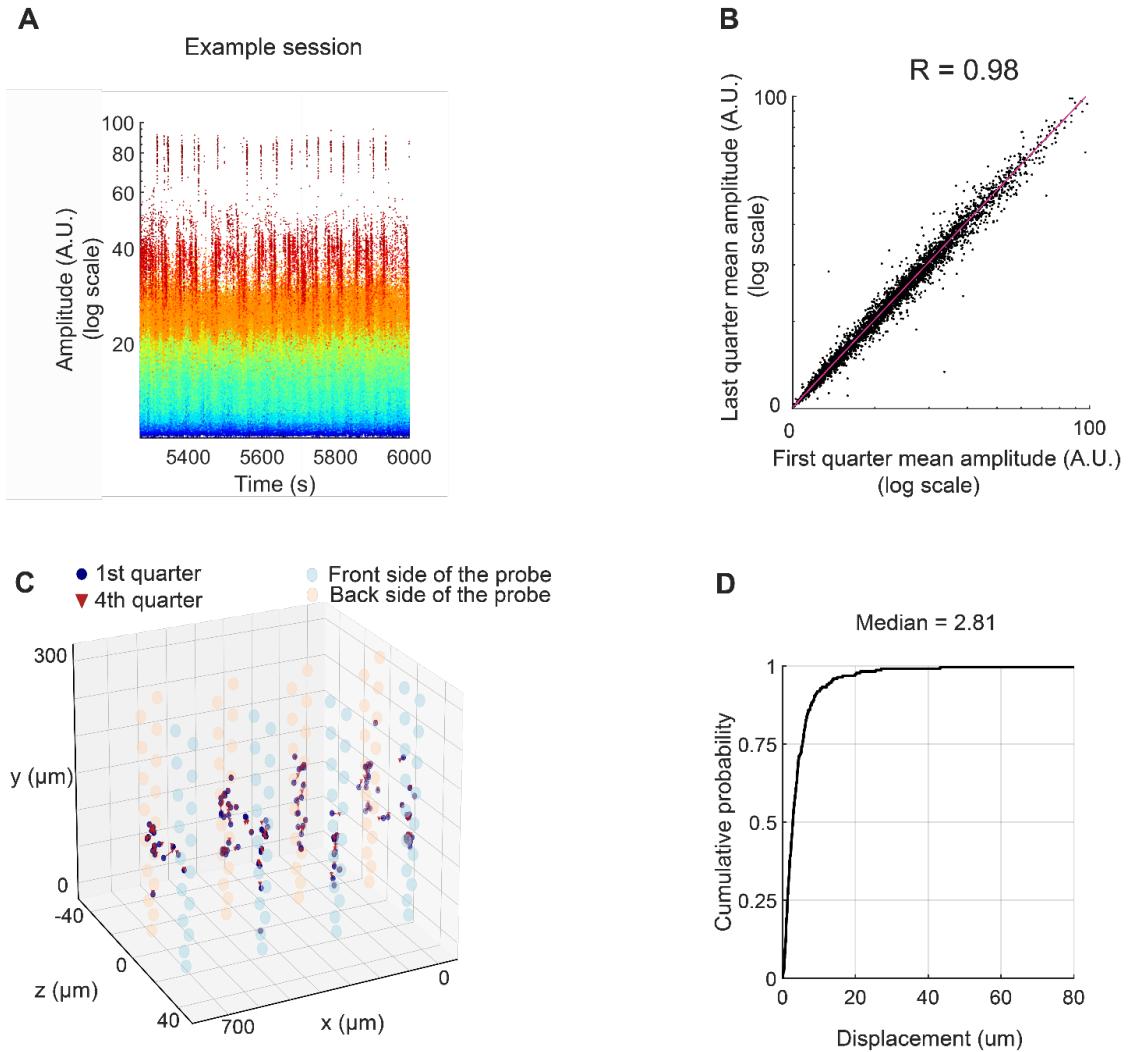


Fig. S5. Assessment of single-unit stability over time.

- (A) Waveform amplitude over time (on-maze duration) for all the cells in an example session. Spikes of individual neurons are color-coded by amplitude. The spike amplitude value was extracted from *Amplitudes.npy*, which was generated by Kilosort/Phy. It is a measure of each spike's amplitude in the template space (A.U.).
- (B) Scatter plot of waveform amplitude in the first quarter of on-maze recording compared to the last quarter, for all the isolated units from the entire figure-8 maze dataset (Pearson correlation coefficient, $R = 0.98$, $P < 10^{-32}$; $N = 4469$ cells from 6 animals).
- (C) Single-unit waveform centroid displacement across time, from a representative session, plotted in 3-D space. Centroid was computed by taking the spatial average across electrode positions weighted by the squared mean waveform amplitude at each electrode (see Methods for details). Waveform centroid for each single unit during the first quarter (blue circles) and last quarter of recording (red triangles) were connected by red lines (note: most of them were very short and not visible). The large semi-transparent orange and blue circles indicate the positions of the dual-sided probe's 128 electrode sites (blue, and orange, front and back sides of the dual-sided probe, respectively).
- (D) Cumulative distribution of unit centroid displacement between the first and the last quarter of maze recording for all the cells in the dataset (median = 2.81- μm , Q1 = 1.37- μm , Q3 = 5.46- μm ; $N = 4469$ cells from 6 animals).

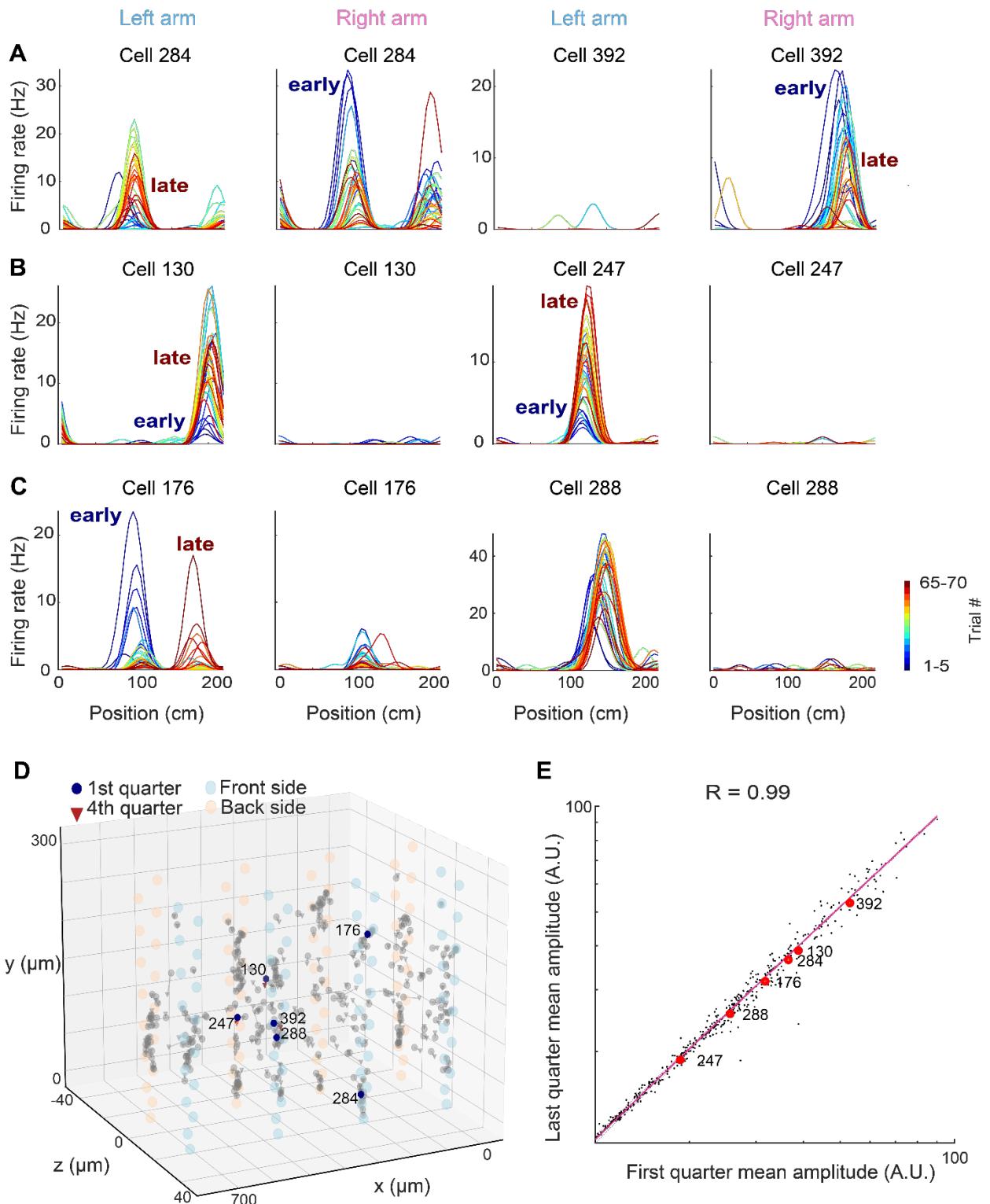


Fig. S6. Spatial tuning curve across trials of example neurons.

The firing rate in space (linearized position on the figure-8 maze) of 6 example neurons across different trials in one representative session. The tuning curves were color-coded by trial block

number (cold color for early trials and warm color for late trials). For each example cell, its firing rates as a function of the linearized position (from 0 to 200-cm) for each of the two arms of the figure-8 maze were plotted separately. Most cells exhibited rate remapping or global remapping across the two arms of the radial arm maze.

- (A) Two example cells that fired with higher firing rates during early trials. Left, left arm trials. Right, right arm trials.
- (B) Two example cells that fired with higher firing rate during late trials.
- (C) Two example cells with place field remapping across trials.
- (D) Single unit waveform centroid (grey) displacement across time, plotted in 3D space, highlighting the 6 example cells (out of 482) shown in (A)-(C). Notice that the firing rates or place fields of all 6 example neurons varied across different trials but their location with respect to the electrode sites (as measured by centroid displacement) remained relatively stable.
- (E) Scatter plot for waveform amplitude during the first versus the last quarter of recording (as in Fig. S5B), highlighting the 6 example cells (red dots) shown in (A)-(C) (Pearson correlation coefficient, $R = 0.99$, $P < 10^{32}$). The spike amplitude value was extracted from Amplitudes.npy, which was generated by Kilosort/Phy. It is a measure of each spike's amplitude in the template space (A.U.).

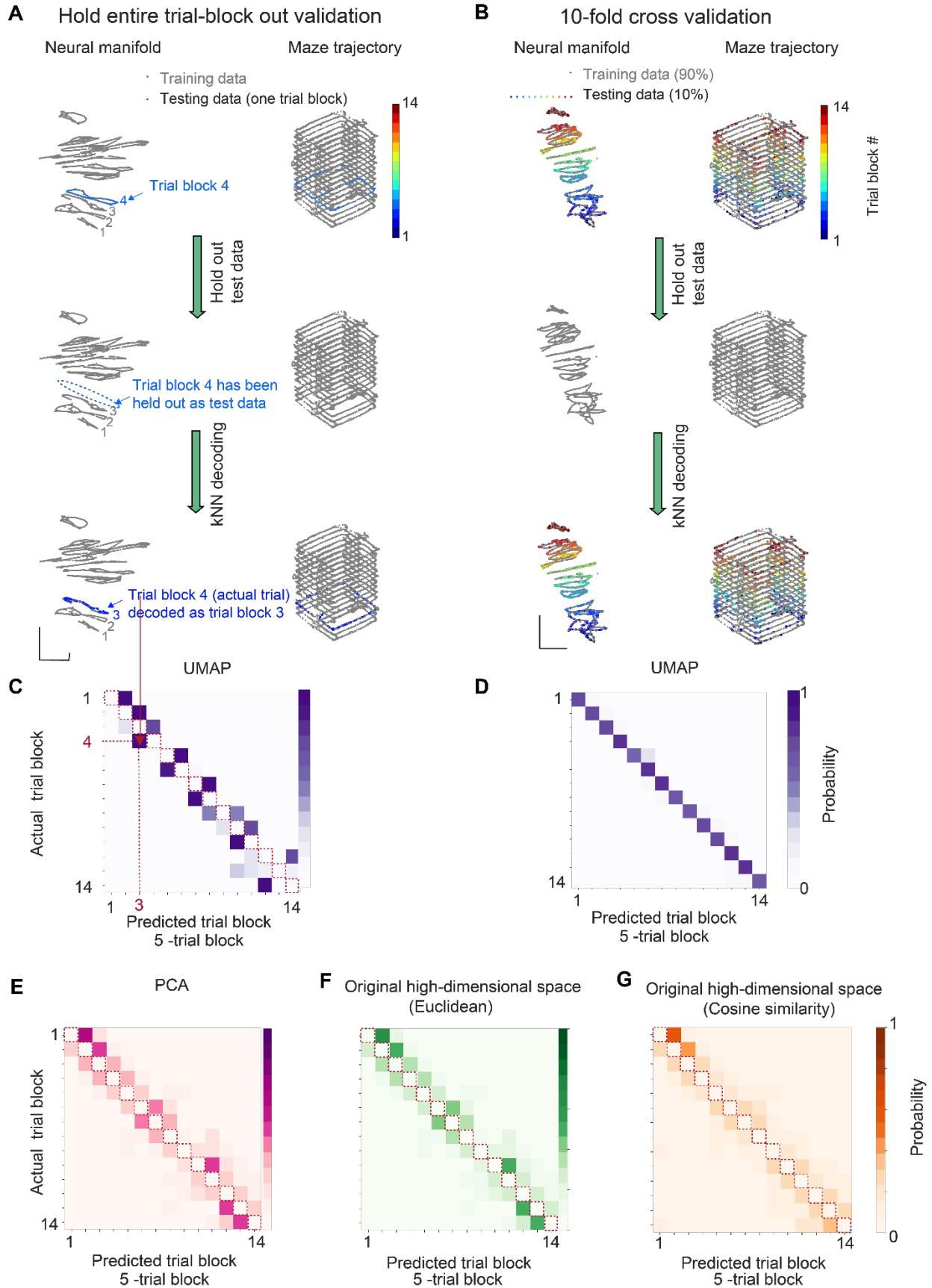


Fig. S7. Hold entire-trial-block-out validation results from four different decoding methods.

- (A) Illustration of the hold entire-trial-block-out validation method. Top, neural manifold from all the data in one example session. To give an example of the cross-validation procedure, data from one trial block (trial block 4; blue) was highlighted. Middle, the highlighted trial block in A was held out as test data. After removing the test data, a kNN decoder was trained only using training data (grey). Bottom, the test data was then embedded with the learned manifold (generated from training data), and the kNN decoder was used for trial block membership decoding. Test data was decoded to the trial block label of its nearest neighbor (trial block 3) on the training data manifold. Training data was gray and test data was colored by the identity of the decoded trial block.
- (B) Illustration of the 10-fold cross-validation method. Top, neural manifold generated from all the data in one example session. 10% of the data were highlighted (colored by their true trial block identity). The highlighted data was held out as test data. Middle, after removing the test data, a kNN decoder was trained only using training data (grey). Bottom, embedding the test data with the training manifold and using the kNN decoder for trial block identity decoding. Training data was gray and test data was colored by the identity of the decoded trial block (trial block 3).
- (C) Confusion matrix obtained using hold entire-trial-block-out validation. Note that the diagonal (red dashed line) of the confusion matrix had 0 decoding probability because the training and test data did not share any data from the same trial block. Since the neural embedding was structured according to the progression of trial events, the test data were correctly decoded to their nearest neighbors.
- (D) Confusion matrix from UMAP trial block decoding using hold-entire-trial-block-out validation.
- (E) Confusion matrix from UMAP trial block decoding using 10-fold cross-validation.
- (F) Confusion matrix for trial block decoding from the original high-dimensional space, with Euclidean distance as the distance metric. Hold-entire-trial-block-out validation was used.
- (G) Similar to (F), but using the cosine similarity as the distance metric. Hold-entire-trial-block-out validation was used.

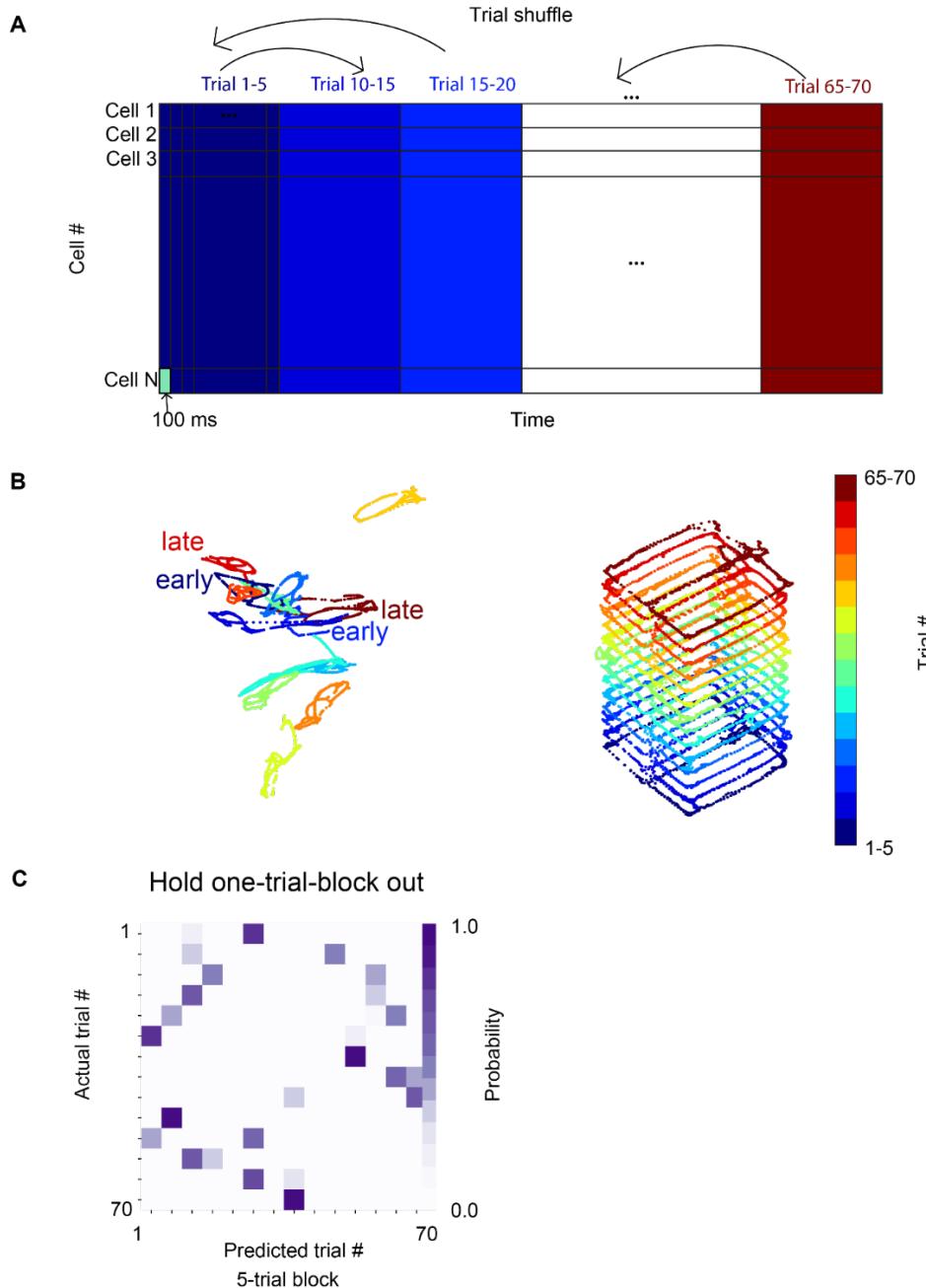


Fig S8. Illustration of trial shuffle procedure and its decoding results.

- (A) Illustration of the trial shuffle procedure. Neural spiking data was binned by 100-ms bins and concatenated into a matrix (each row is a cell, and each column is a 100-ms bin). Trial shuffle was performed by shuffling the trial block identity of the entire data matrix (same permutation for all the cells).
- (B) Left, neural manifold for the trial-shuffled data. Both the neural manifold (left) and the maze trajectory (right) were colored by the trial block number. Notice that the shuffle procedure disrupted the structure of the data such that early and late trials may be embedded very close to each other.
- (C) Confusion matrix of the trial shuffled data for hold-entire-trial-out validation. Note that trial shuffled data lost the trial structure in real data. Trial blocks close in event order were no longer embedded close in the low-dimensional space. Thus, the hold-one-trial out validation method yielded low decoding accuracy for trial shuffled data.

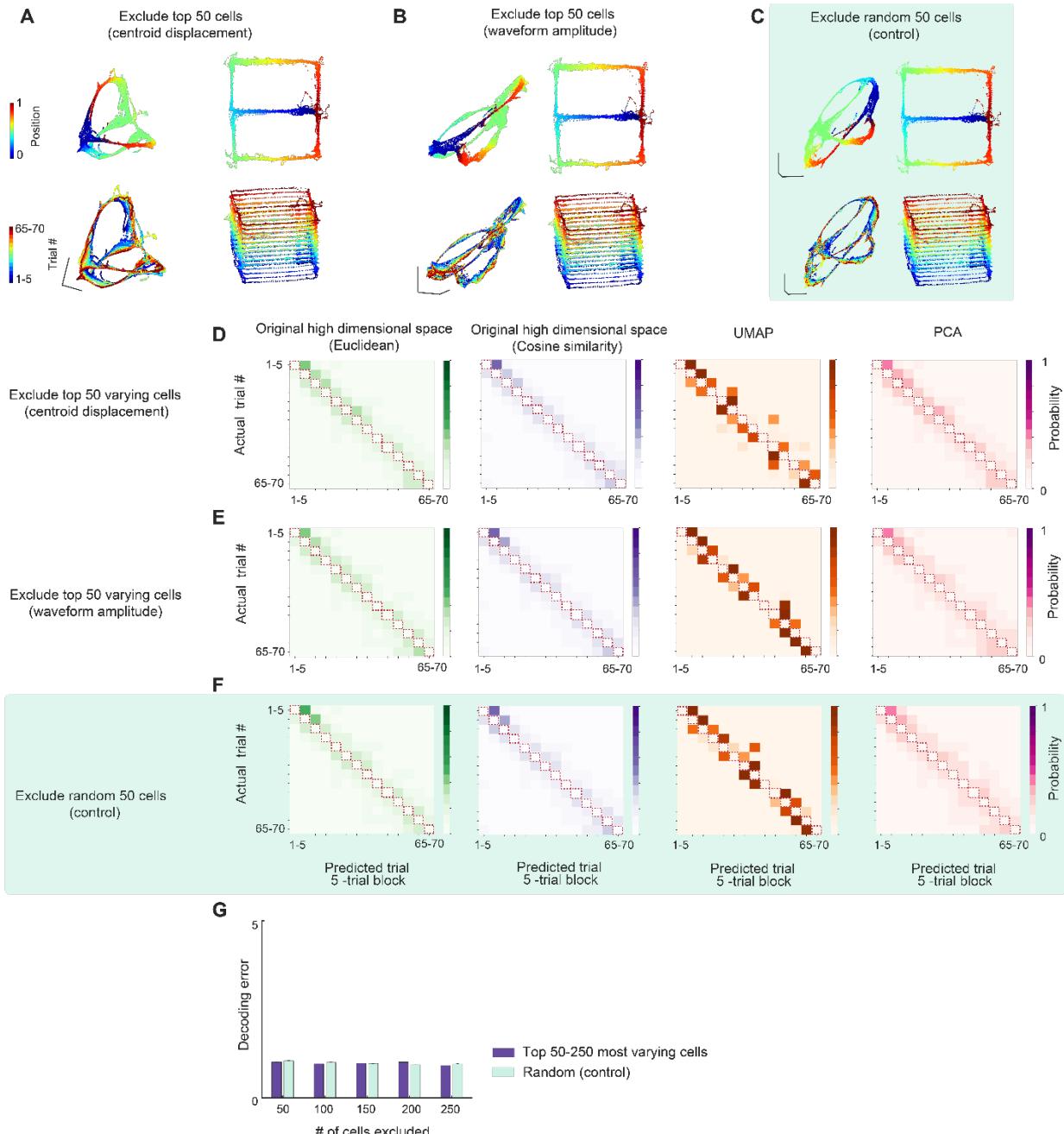


Fig S9. Trial-by-trial representation drift cannot be explained by variations in waveform amplitude or waveform centroid displacement.

The purpose of this figure was to test whether the trial-by-trial variation at the population level can be explained by electrode drift. Two metrics were measured: (1) waveform amplitude change and (2) waveform centroid displacement. Units were then sorted by those two metrics. Next, the top k cells with greatest variance for each of the metric were excluded from decoding analysis. If the trial-by-trial representation could be fully explained by non-stationarity of recording, then excluding the most unstable units would introduce higher trial block decoding error (comparing to randomly excluding k units to control for the reduction in cell number). In this figure, the effect

of excluding the most unstable units was demonstrated via both visualization (panels A-C) and quantification (panels D-G).

- (A) The low-dimensional embedding after excluding 50 cells that showed the largest centroid displacement was plotted. The manifold was colored by animal's linearized position (top) and trial number (bottom).
- (B) The low-dimensional embedding after excluding 50 cells with the greatest amplitude variance over time was plotted.
- (C) The low-dimensional embedding after randomly excluding 50 cells (control) was plotted.
- (D) Confusion matrices of trial block decoding results after excluding 50 cells with the largest centroid displacement were plotted. From left to right, decoding results from the original high-dimensional space using Euclidean distance (green), decoding from original high-dimensional space using cosine similarity (purple), UMAP decoding (orange), and PCA decoding (pink).
- (E) Confusion matrices of trial block decoding results after excluding 50 cells that showed the largest amplitude variance are plotted. The sub-panel arrangement is the same as in (D).
- (F) Confusion matrices of trial block decoding results after randomly removing 50 cells (control). The sub-panel arrangement is the same as in (D) and (E).
- (G) Decoding error after excluding the top k most variant cells versus k random cells, where k ranged from 50 – 250 cells. The decoding error was not significantly different from control across different range of k .

The visualization as well as the quantification results suggest that the trial-by-trial variation at the population level could not be explained by electrode drift.

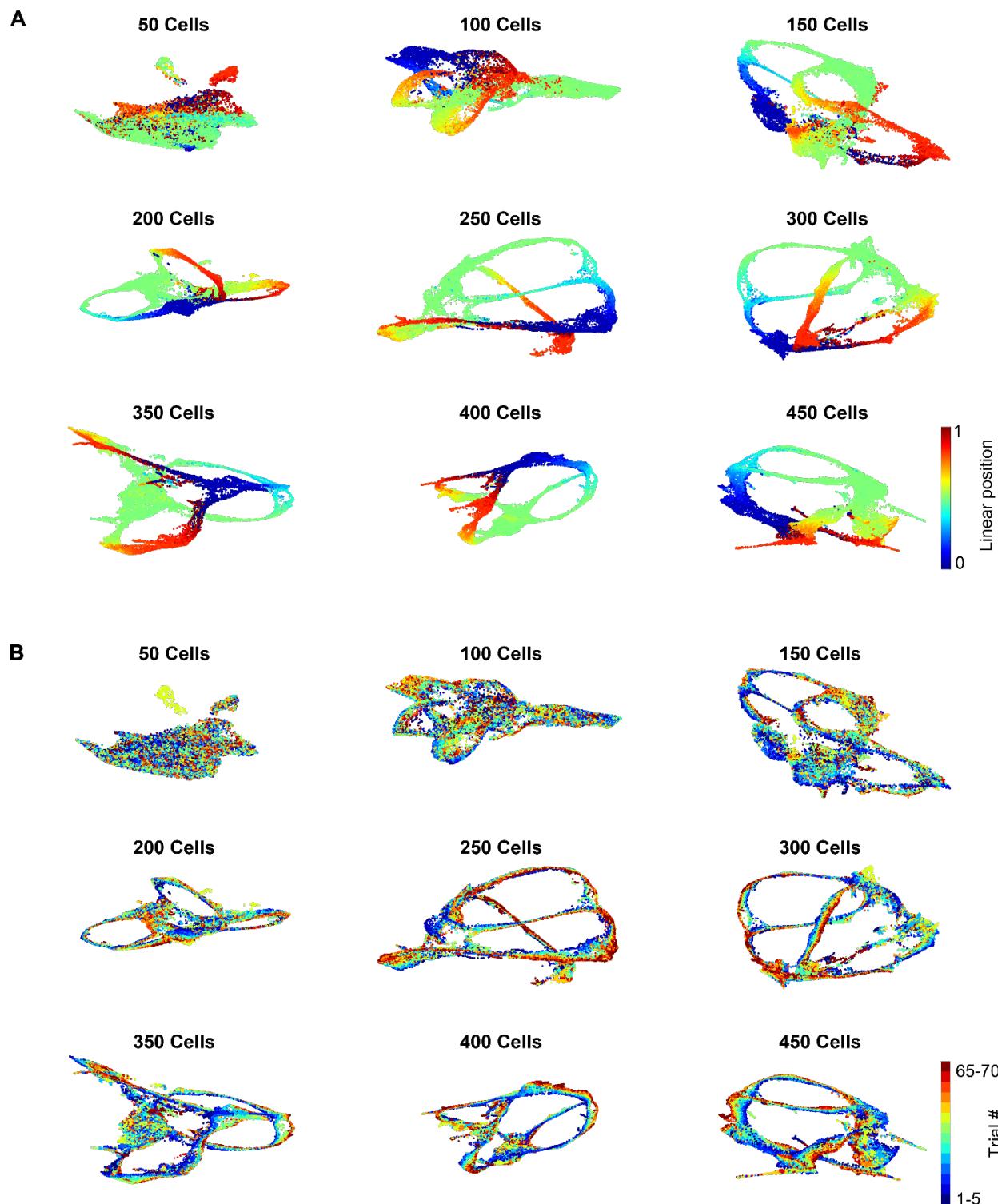


Fig S10. Visualization of the downsampling analysis using UMAP (unsupervised)

(A) Manifolds generated from downsampling the number of cells included for UMAP embedding. Manifolds were colored by the animal's linearized position.

(B) Same as in (A) but manifolds were colored by trial block number.

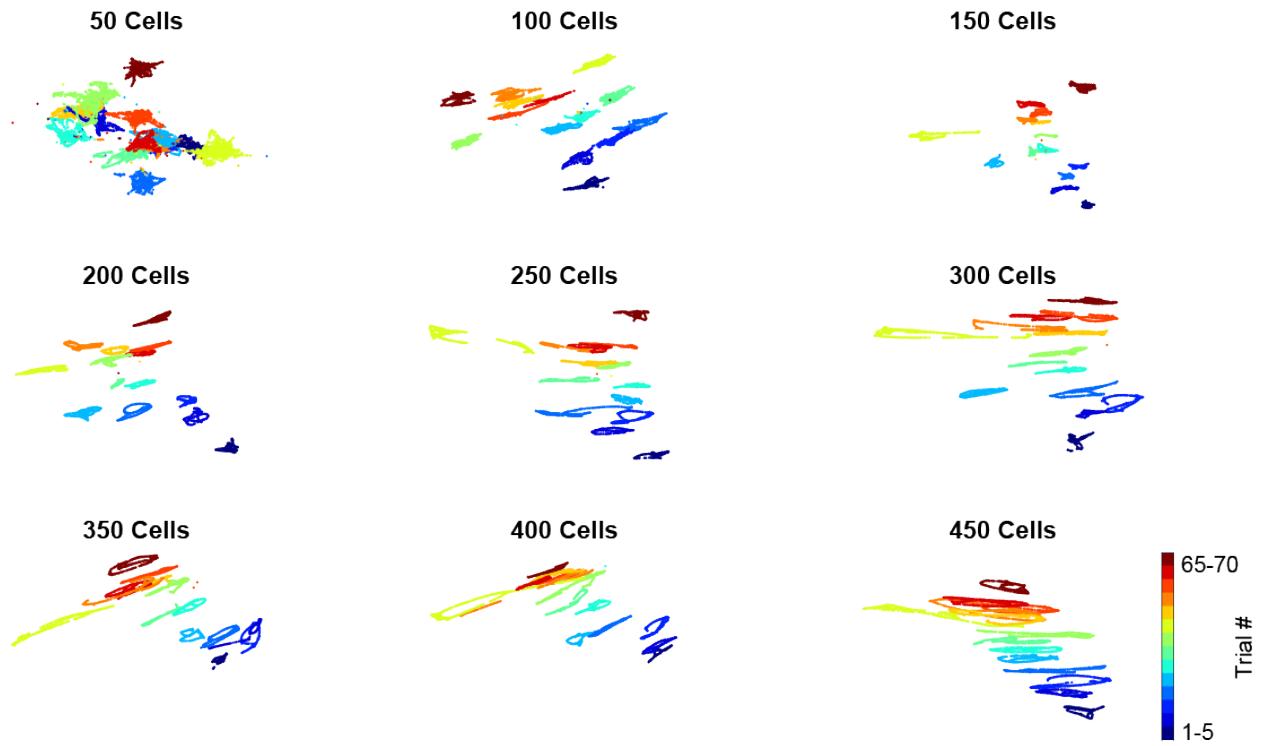


Fig. S11. Visualization of downsampling analysis UMAP (semi-supervised)

UMAP manifolds generated from downsampling the number of cells. Here, the manifold embeddings were trained by the semi-supervised method. Manifolds were colored by trial block number.

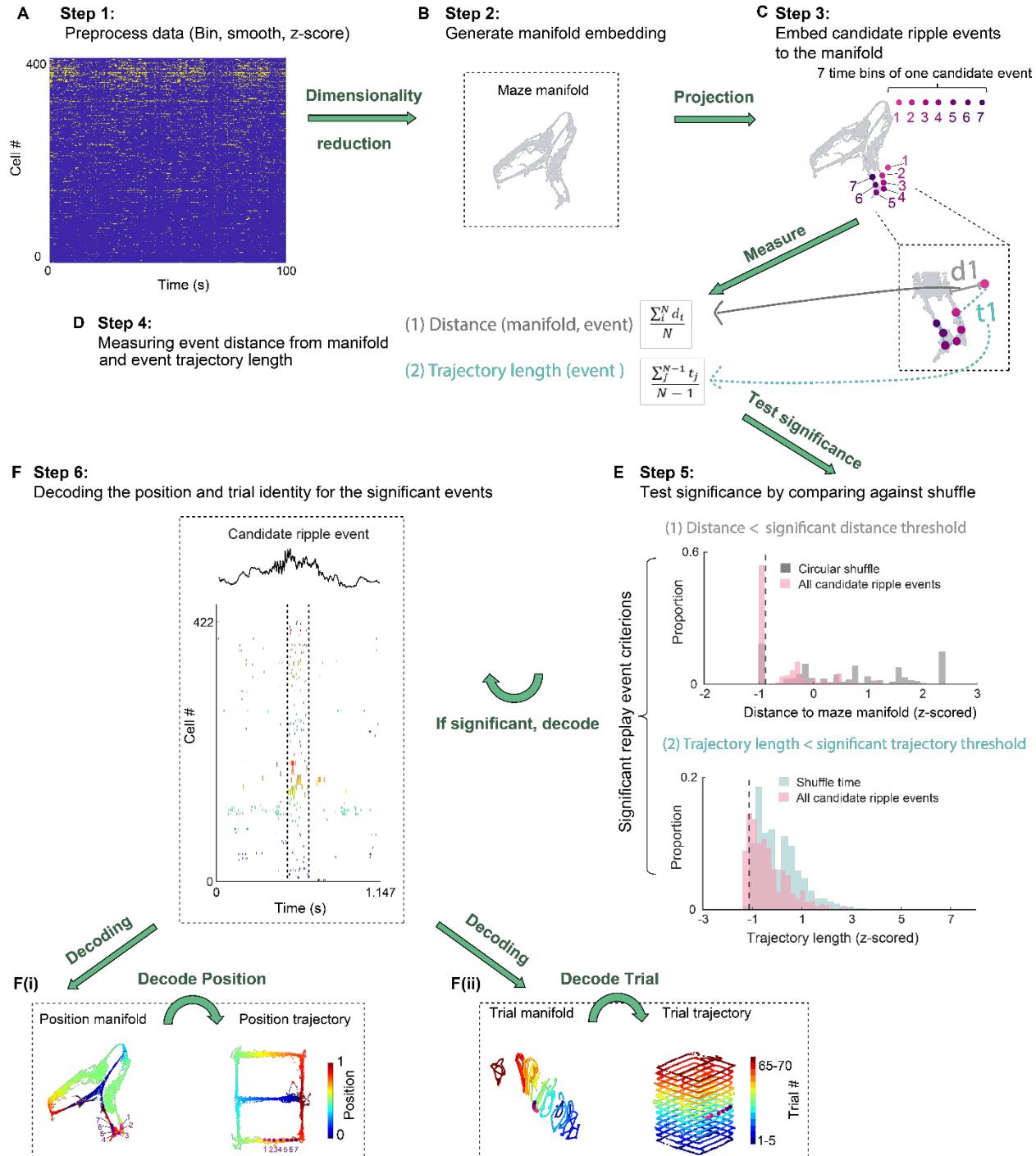


Fig. S12. Flow diagram demonstrating procedures for detecting and decoding significant SPW-R replays.

(A) Step 1: Preprocessing. The animal performed an alternation task in the figure-8 maze while spikes from 422 pyramidal cells (same neurons were shown in Fig. 1, ordered by firing rate). Spike trains during maze learning were binned in 100-ms bins, z-scored, and then smoothed, yielding a matrix of 422-dimensional population activity vectors. Spike trains of candidate SPW-R events were binned into 20-ms bins, smoothed, and then z-scored.

(B) Step 2: Generating maze manifold embedding. The preprocessed data matrix was passed through dimensionality reduction step by either UMAP or PCA, which generated a low-dimensional embedding, allowing the topological structure to be visualized. Notice that this step was un-supervised.

(C) Step 3: Embedding candidate ripple events to the embedding metric defined by the maze manifold. The 7 pink-purple dots on the manifold represent low-dimensional embeddings of an example candidate SPW-R event. The 7 dots were the low-dimensional embeddings corresponding to the 7 successive 20-ms bins of the example SPW-R event. For further details related to step 2 and 3, see Fig. S13.

(D) Step 4: Measuring distance to the manifold and trajectory length. The event distance to the manifold is defined as the mean Euclidian distance across all time bins within the event. For example, grey bar in the zoomed-in inset represents the distance (d_1) between the first time bin and the maze manifold. The event trajectory length was defined as the mean trajectory length between all successive time bins. For example, cyan dashed line denotes the trajectory length (t_1) between the first and second time bins of the event.

(E) Step 5: Testing whether the candidate events were significant against shuffled distributions. A candidate event was classified as a significant replay event only if it satisfied two criteria: (1) The mean distance to manifold (as defined in D) of a candidate event is significantly shorter than the shuffled data (circular). (2) The mean trajectory length is significantly shorter than shuffled data (shuffle time).

(F) Step 6: Only significant replays were included for further analysis from this step onward. F(i). To decode the virtual position relayed by the candidate event, the population activity during SPW-R was embedded with the position manifold, where each point on the manifold was associated with a position bin label (positions were binned into 5-cm bins). The low-dimensional embeddings along with the position labels were used to train a kNN decoder. The replay content of each time bin was decoded by taking the mean of the position label of its k nearest neighbors on the ‘position manifold’. F(ii). Similarly, the SPW-R activity vectors were embedded with the ‘trial manifold’ for trial block identity decoding. To decode the trial block membership, an analogous procedure was applied.

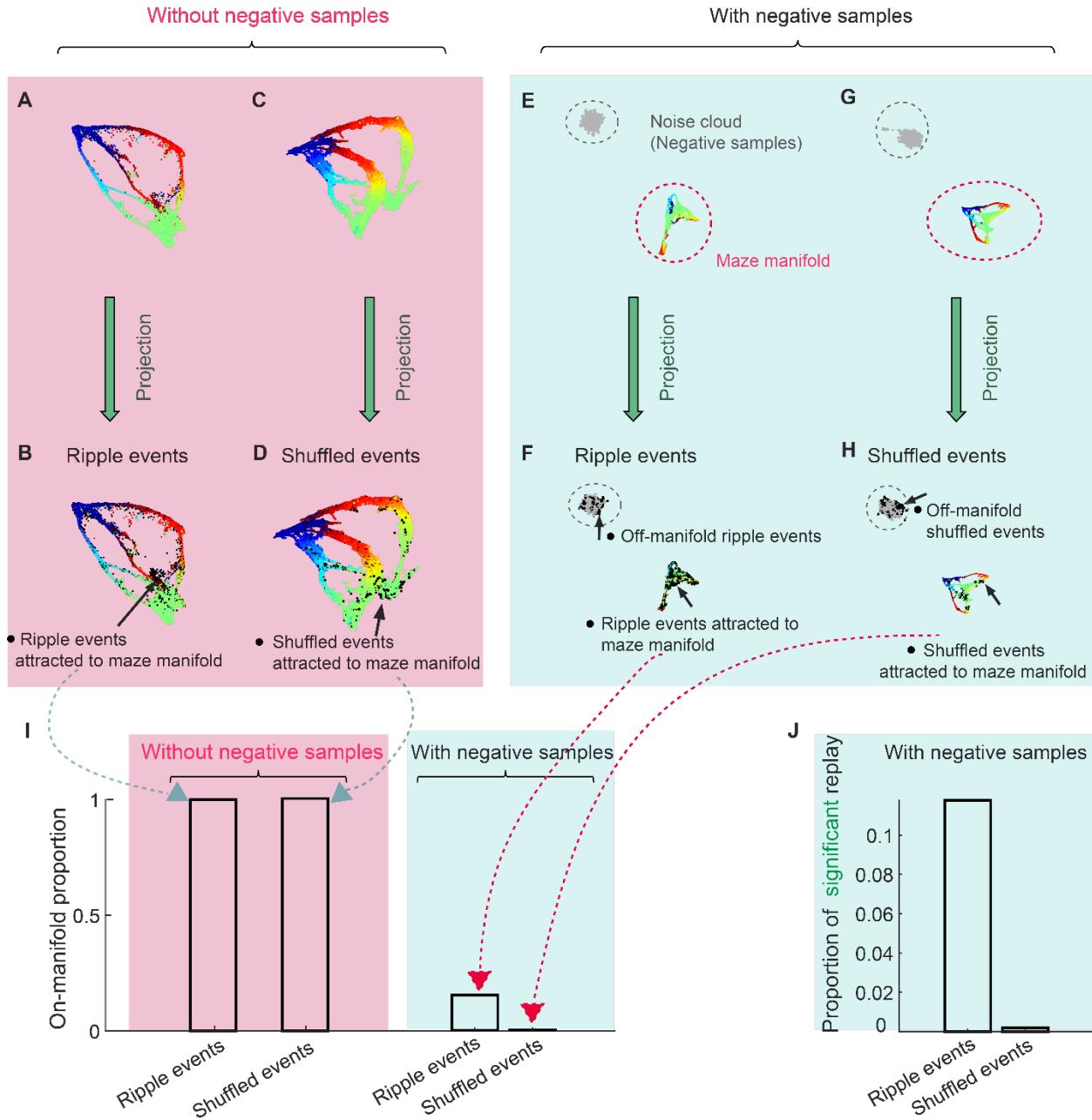


Fig. S13. Negative sampling algorithm and the role of negative samples.

This figure is related to Steps 2 and 3 of the decoding procedure described in Fig. S12. The purpose of the data is to illustrate the importance of generating negative samples when using UMAP for decoding replay events. By default, UMAP used a sampling-based approach called negative sampling to optimize the low-dimensional embedding. It is fundamentally a contrastive method to optimize the low-dimensional embedding. Briefly, the intuition behind negative sampling is that data pairs that are dissimilar in the original high-dimensional space are repulsed away from each other, whereas similar data in the high-dimensional space are pulled closer to each other. The shuffle data (shuffle the rows and columns of the maze firing rate matrix) served as negative samples to generate the correct embedding for test data, such that population vectors similar to the maze manifold were pulled closer to the maze manifold in the low-dimensional space and those

dissimilar to the maze manifold were pulled closer to the noise cloud (negative samples). In the absence of the negative samples, all SPW-R data would be spuriously embedded close to the maze manifold. To illustrate the importance of the negative samples, the consequence of embedding without negative samples was examined in panels (A)-(D). In contrast, the results of adding negative samples were shown in panels (E)-(H).

- (A) Generating UMAP embedding from maze data.
- (B) Population vectors during SPW-R events were embedded with the maze manifold. Note that without negative samples, almost all SPW-R events were forced to be attracted to the maze manifold.
- (C)(C) and (D) were similar to (A) and (B), but embedded with shuffled SPW-R events.
- (D) Without negative samples, even shuffled events were embedded close to the maze manifold. This happened because the UMAP algorithm ultimately harnesses the negative sampling algorithm, which attracts the positive sample pairs and repulses the negative sample pairs. Without negative samples, mode collapse would occur. In the absence of negative samples, there was no repulsive force away from the maze manifold, and so every data point was forced to be attracted to the maze manifold. As a result, even shuffled data (those that were purposefully permuted to be dissimilar to maze data) were forced to be embedded close to the maze manifold.
- (E) To avoid mode collapse (shown in B and D), negative samples (grey dots) were generated and embedded along with the maze manifold so that data that were not similar to the maze data were attracted to the negative sample cloud. Negative samples were generated by shuffling the maze data.
- (F) With the presence of negative samples, only SPW-R events (black) that were sufficiently similar to the maze manifold replay were attracted to the maze manifold (15% of all SPW-R events are on-manifold). In contrast, SPW-R events that were sufficiently different from the maze manifold were attracted to the negative samples (circled cloud).
- (G)-(H) are similar to (E) and (F), but for the shuffled SPW-R events.
- (I) Proportion of on-manifold events under different conditions. Without negative samples, all SPW-R events and shuffled data were equally close to the manifold. In contrast, with negative samples, only 15% of real SPW-R events and 0.2% of shuffled events were on-manifold.
- (J) For an event to be classified as a significant replay event, it needs to be: (1) closer to the maze manifold compared to shuffle, (2) have a shorter trajectory length compared to shuffle (see Step 4 in Fig. S12). Approximately 11% of the SPW-R events were significant replay events, whereas only 0.1% of the shuffle events satisfy both significance criteria.

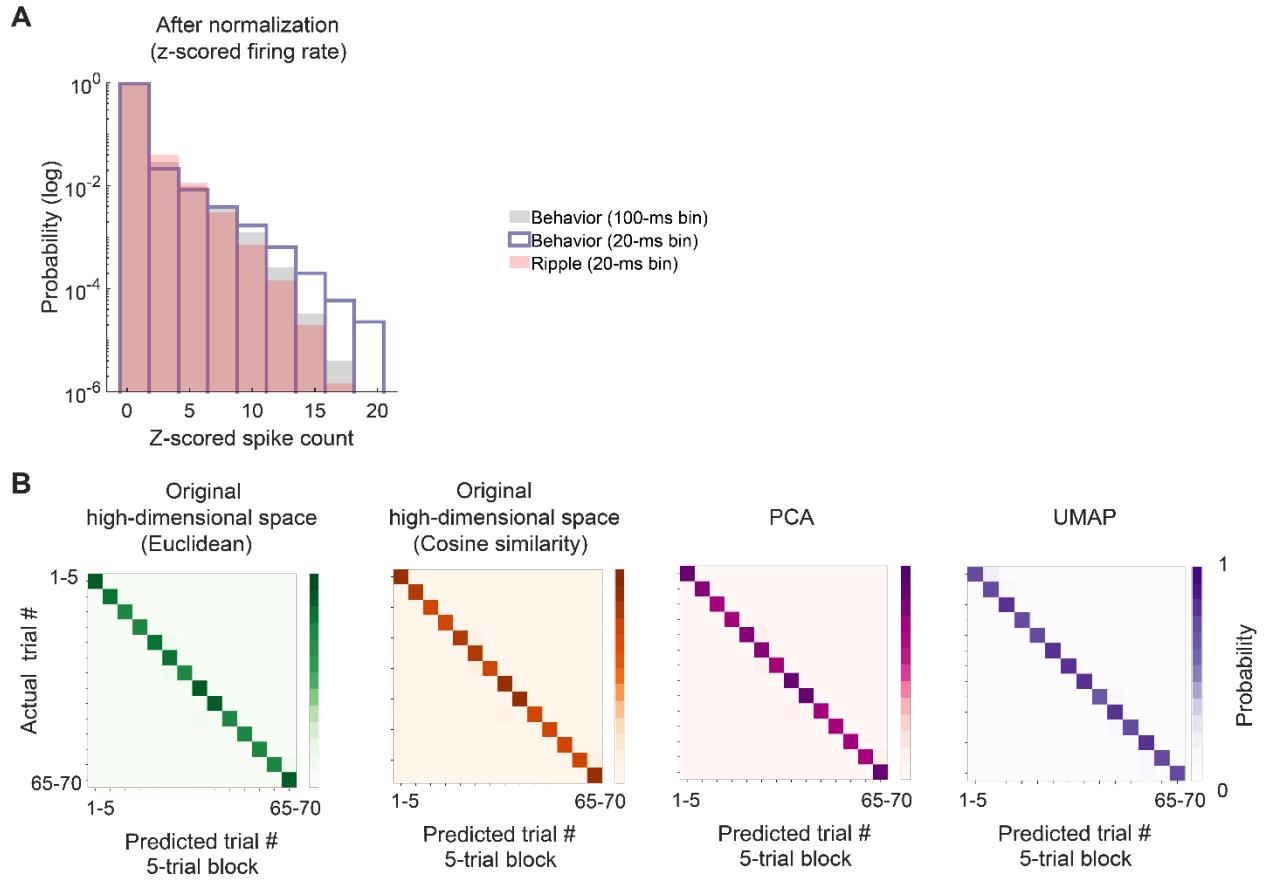


Fig. S14. Difference in time bin and its consequences for decoding.

The purpose of this figure was to demonstrate that the difference in time binning for SPW-R data and maze data did no hurt decoding accuracy. SPW-R data were binned into 20-ms bins and the maze data was binned into 100-ms bins. 100-ms bin was chosen for the maze data because it is at the theta timescale. 20-ms bin was used for SPW-Rs because it is at the fast gamma cycle range. This set of times bins are routinely used for position decoding in the literature.

(A) Distribution of spike counts with different time bins after normalization. As a standard preprocessing procedure, data were first z-scored before going through any dimensionality reduction analysis (including UMAP and PCA). After z-scoring, the distribution of spike counts was not different during maze behavior and SPW-Rs.

(B) To examine if difference in time bin affect the decoding result, test data with 20-ms bin were decoded from training data binned in 100-ms bins. The test exploited the fact that data during maze running had ground truth (i.e., position and trial block labels). Confusion matrix of decoding results obtained by embedding the data with 20-ms bins with the manifold generated by maze data with 100-ms bins. The decoding probability was concentrated at the diagonal of the confusion matrix, suggesting that trial block membership can still be accurately decoded, even with different time bins.

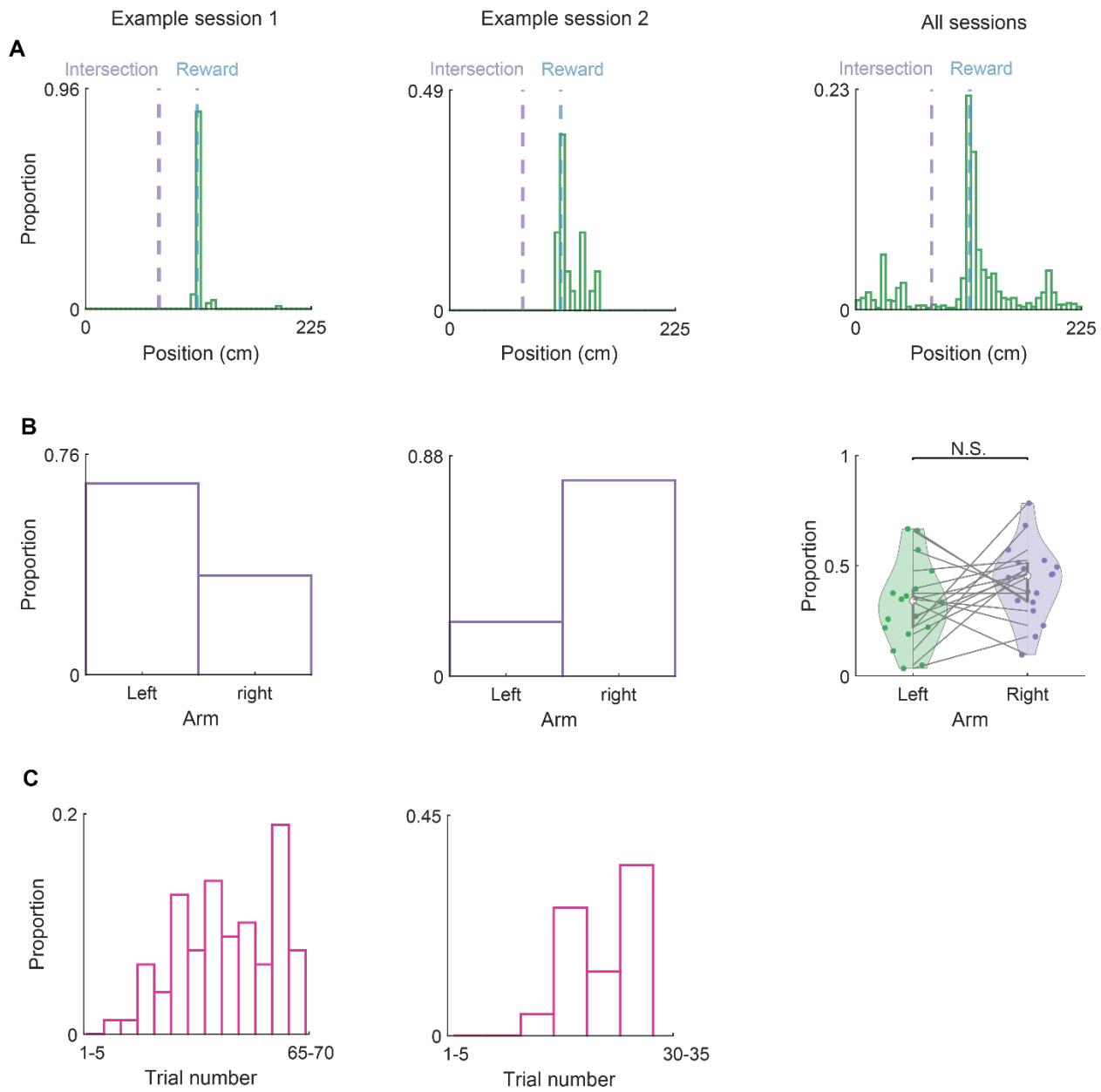


Fig. S15. Distribution of SPW-Rs across different locations, trials, and arms of the maze.

- (A) Distributions of SPW-Rs across different maze locations of the figure-8 maze from two example sessions (left and middle) and the summary data (right) across all figure-8 maze sessions ($n = 21$ sessions from 6 animals). Intersection (purple dashed line), T junction of the figure-8 maze. Reward (blue dashed line), location of the water reward in the left or right-side arm. Most of the awake replay events occur at the reward site.
- (B) Distributions of SPW-Rs across different sides of the arms of the figure-8 maze from two example sessions (left and middle) and the summary data (right) across all figure-8 maze sessions (N.S., not significant; unpaired t-test; $n = 21$ sessions from 6 animals).
- (C) Distributions of SPW-Rs across different trial blocks of the figure-8 maze from two example sessions (left and middle).

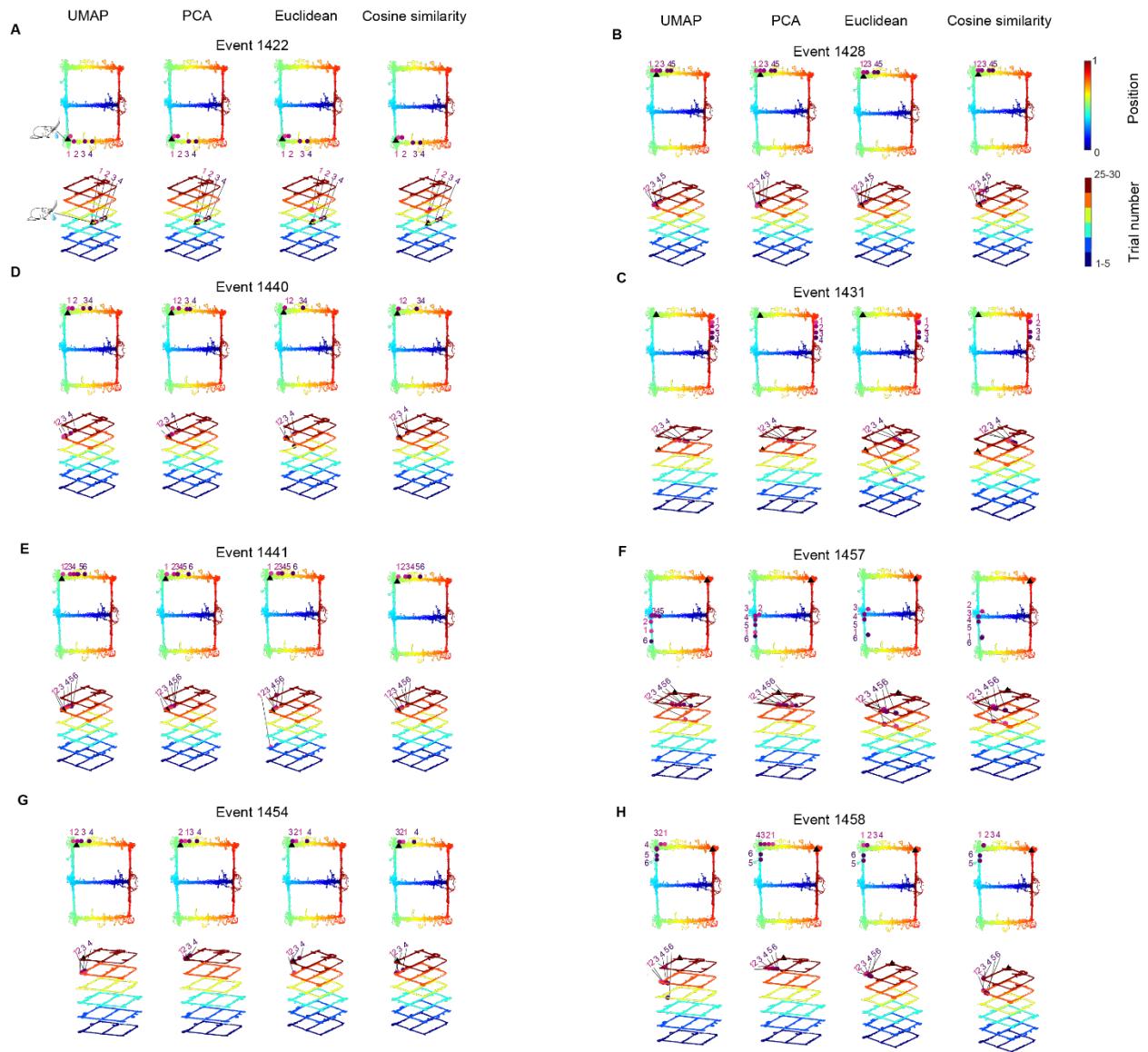


Fig. S16. Decoding results of example replay events using different decoding methods.

(A)-(H) Example replay events from an example session. For each event, the top rows are position decoding results and the bottom rows are trial block decoding results. For each event, there are four columns of results each obtained from a different decoding method.

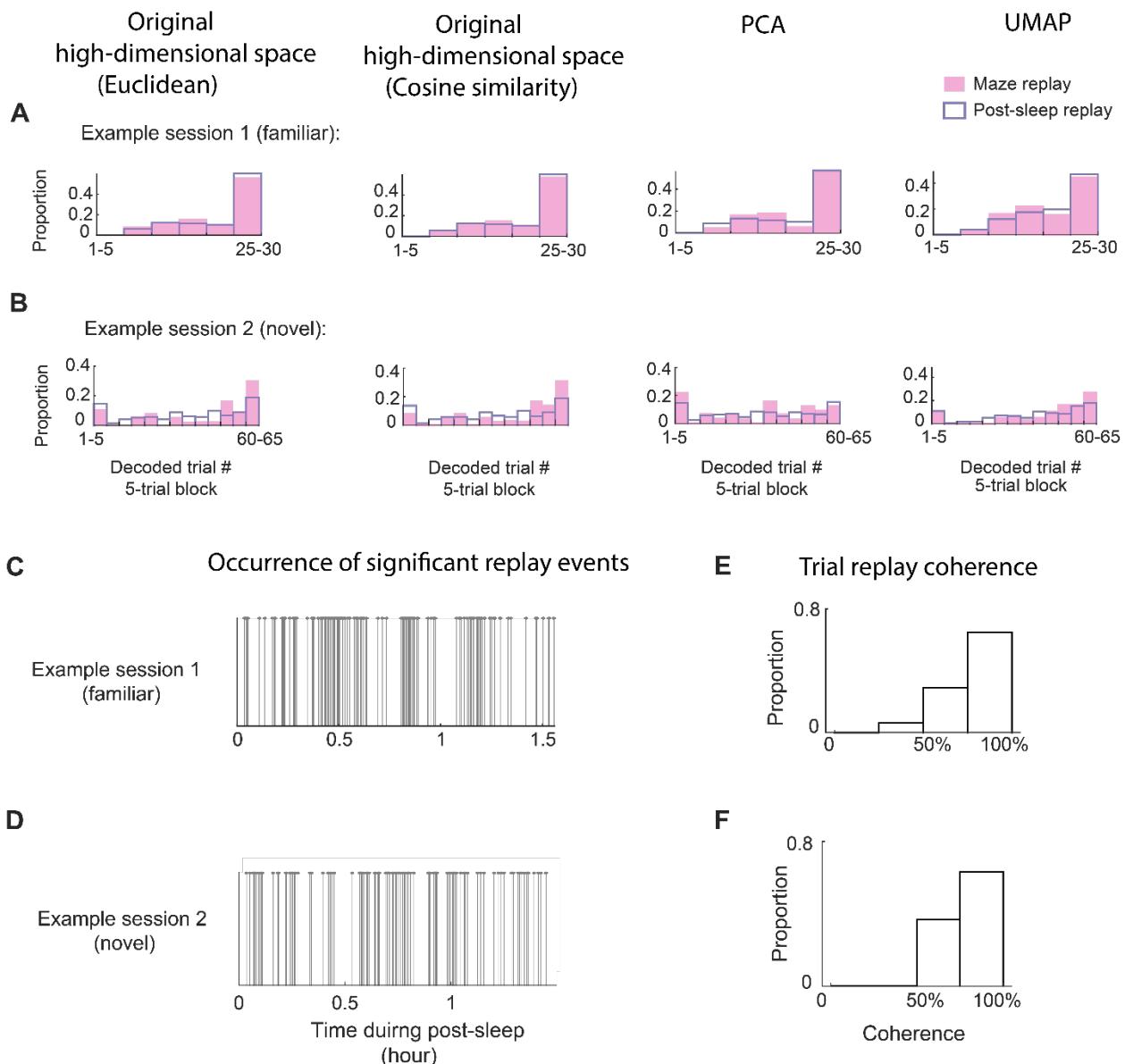


Fig. S17. Significant replay events and their decoding results in two example sessions.

(A)-(B) Trial distribution pattern for awake replay and post-sleep replay for two example sessions using 4 different decoding methods.

(C)-(D) Distribution of significant replay events over the course of post-sleep session (each grey line indicates the occurrence of one significant replay event).

(E)-(F) Distribution of within-event replay coherence for significant replay events. The coherency of each event was measured as the proportion of time bins that shared the same trial block label as the mode for that event. 100% means all time bins of a given replay event were decoded to the same trial block label.

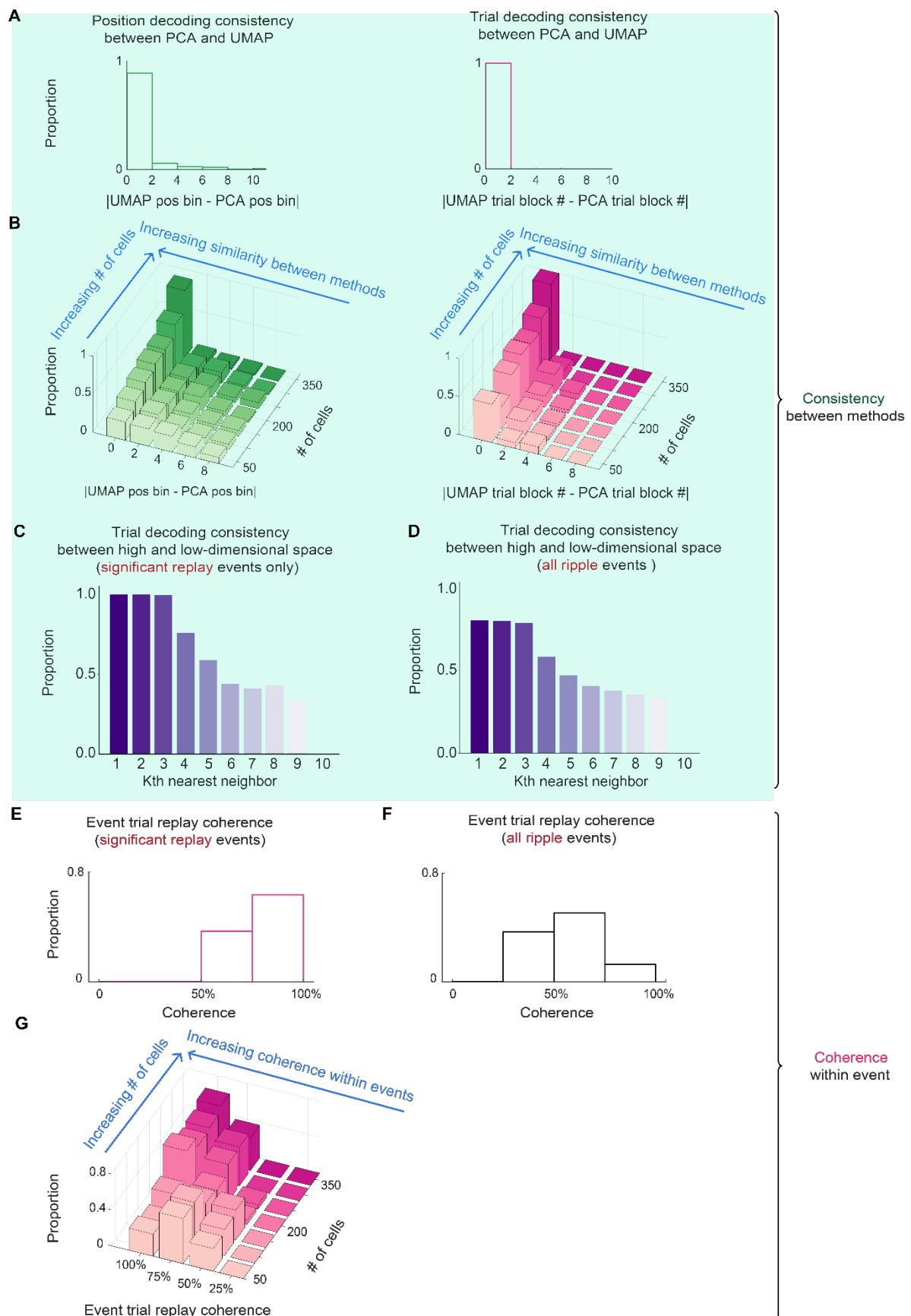


Fig. S18. Quantifying the decoding consistency between methods and within-event replay coherence.

To measure the quality of the replay events, two metrics were used: (1) the decoding consistency between methods (to measure whether different decoding methods are consistent with each other, panels A-D), and (2) within-event trial replay coherence (to measure whether different time bins within each event are coherently decoded to the same trial block, panels E-G).

- (A) Consistency between UMAP and PCA decoding results, measured by the difference in decoding results from the two methods. Left, consistency between UMAP and PCA results for position decoding (measured by the absolute value of the difference between UMAP decoded position bin and PCA decoded position bin). Right, consistency between UMAP and PCA results for trial block decoding (measured by the absolute value of the difference between UMAP decoded trial block and PCA decoded trial block). 0 means PCA and UMAP decoded to the same position bin or trial block. For example, event 1422 in Fig. S16A had completely coherent decoding results between PCA and UMAP decoding (0 difference).
- (B) Neurons used for the decoding analysis were downsampled. Left, consistency between UMAP and PCA for position decoding. Right, consistency between UMAP and PCA for trial block identity decoding. Note that UMAP and PCA were very consistent when the number of neurons used for decoding was high but became inconsistent with a decreasing number of neurons.
- (C) For all significant replay events, the consistency between the original high-dimensional and the reduced low-dimensional space was plotted. For each data point, the decoded trial block membership of its ten nearest neighbors in the high-dimensional space (*trial_member_high*) and the reduced low-dimensional space (*trial_member_low*) were compared. A data point was considered consistent if $trial_member_high - trial_member_low = 0$. For the first three nearest neighbors, the decoded trial block identity between the high and low-dimensional space was 100% consistent.
- (D) Similar to (C), but for all (not just significant) SWP-R events. Including non-significant events reduced the decoding consistency compared to only including the significant events.
- (E) Distribution of within-event coherency for significant replay events. The coherency of each event was measured as the proportion of time bins that share the same trial block label as the mode for that event. 100% means all time bins of a given replay event were decoded to the same trial block label. For instance, event 1422 in Fig. S16A was an example event with 100% coherency (all 4 time-bins were decoded as trial block 4).
- (F) Similar to (E), but for all (not just significant) ripple events.
- (G) As in (B), downsampling analysis with decreasing the number of neurons used for decoding was performed. More neurons included for decoding resulted in higher within-event trial replay coherency (100% coherency means all the time bins of the event decoded to the same trial block). As more units got excluded, the replay event became less coherent.

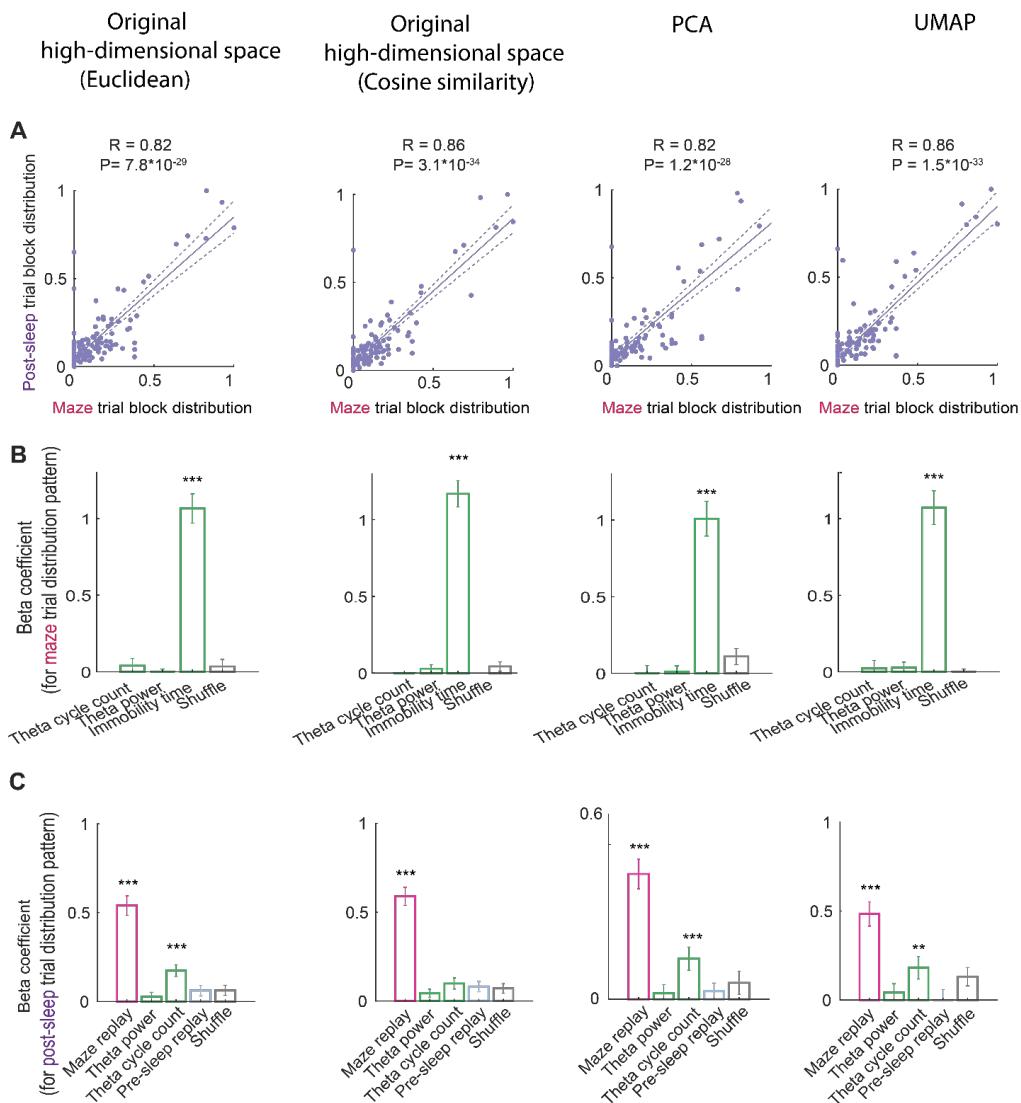


Fig. S19. Correlational and regression summary statistics are consistent across different methods.

- (A) Correlation between maze replay events versus the post-experience sleep replay events, for their trial block distribution patterns. The results from four different decoding methods were presented in parallel to highlight the consistency across different methods.
- (B) The predictive relationship between the trial block distribution pattern of maze replay events and other candidate variables, including the trial block distribution patterns of theta cycles, theta power and immobility time (when speed < 2-cm/s), and trial-shuffle data (refer to Fig. S8 for the procedure for generating trial shuffle data). For decoding from the original high-dimensional space with Euclidean distance metric, *** $P < 10^{-25}$ for immobility time; for decoding from high-dimensional space with cosine similarity, *** $P < 10^{-19}$ for immobility time; for PCA, *** $P < 10^{-16}$ for immobility time; for UMAP, *** $P < 10^{-5}$ for immobility time.
- (C) The predictive relationship between trial block distribution pattern of post-experience sleep and other candidate variables. (For decoding from original high-dimensional space with Euclidean distance metric, *** $P < 10^{-16}$ for awake replay, *** $P < 10^{-7}$ for theta cycle number; for decoding from high-dimensional space with cosine similarity, *** $P < 10^{-20}$ for awake replay; for PCA, *** $P < 10^{-12}$ for awake replay, *** $P < 10^{-6}$ for theta cycle number; for

UMAP, *** $P < 10^{-10}$ for awake replay, ** $P < 10^{-3}$ for theta cycle number). The relative predictive power of a given metric was considered non-significant when it overlapped with zero ($n = 16$ sessions from 5 animals).

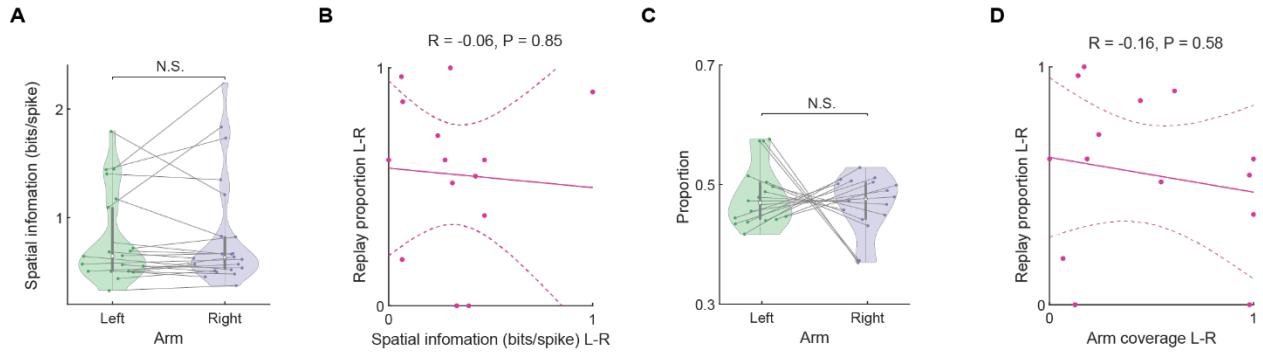


Fig. S20. Replay arm distribution pattern cannot be explained by the difference in decodability of the two side arms of the maze.

- (A)** Mean spatial information for left versus right side arm segments. Each dot in the violin plot represents mean spatial information across all cells from one session (N.S. not significant, $P = 0.76$; unpaired t-test; $n = 13$ sessions from 5 animals).
- (B)** Correlation between spatial information difference (mean spatial information of left arm trials minus mean spatial information of right arm trials) and replay proportion difference between left and right arm (proportion of left arm replay minus proportion right arm replay). This shows that the difference in replay distribution across arms could not be explained by the difference in decodability (measured by the spatial information score) between the two arms (Pearson correlation coefficient, $R = 0.07$, $P = 0.85$; $n = 13$ sessions from 5 animals).
- (C)** Proportion of left versus right arm visit across sessions. Each dot in the violin plot represents one session (N.S. not significant; $P = 0.33$; unpaired t-test; $n = 13$ sessions from 5 animals).
- (D)** Correlation between the difference in arm visit (proportion of left arm visits minus proportion of right arm visits) and difference in replay proportion between left and right arm (proportion of left arm replay minus proportion right arm replay). Pearson correlation coefficient, $R = -0.16$, $P = 0.58$; $n = 13$ sessions from 5 animals.

References and Notes

1. M. Inostroza, J. Born, Sleep for preserving and transforming episodic memory. *Annu. Rev. Neurosci.* **36**, 79–102 (2013). [doi:10.1146/annurev-neuro-062012-170429](https://doi.org/10.1146/annurev-neuro-062012-170429) Medline
2. A. Tambini, N. Ketz, L. Davachi, Enhanced brain correlations during rest are related to memory for recent experiences. *Neuron* **65**, 280–290 (2010). [doi:10.1016/j.neuron.2010.01.001](https://doi.org/10.1016/j.neuron.2010.01.001) Medline
3. B. P. Staresina, A. Alink, N. Kriegeskorte, R. N. Henson, Awake reactivation predicts memory in humans. *Proc. Natl. Acad. Sci. U.S.A.* **110**, 21159–21164 (2013). [doi:10.1073/pnas.1311989110](https://doi.org/10.1073/pnas.1311989110) Medline
4. J. L. McClelland, B. L. McNaughton, R. C. O'Reilly, Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychol. Rev.* **102**, 419–457 (1995). [doi:10.1037/0033-295X.102.3.419](https://doi.org/10.1037/0033-295X.102.3.419) Medline
5. U. Frey, R. G. M. Morris, Synaptic tagging and long-term potentiation. *Nature* **385**, 533–536 (1997). [doi:10.1038/385533a0](https://doi.org/10.1038/385533a0) Medline
6. R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, ed. 2, 2018).
7. S. P. Jadhav, C. Kemere, P. W. German, L. M. Frank, Awake hippocampal sharp-wave ripples support spatial memory. *Science* **336**, 1454–1458 (2012). [doi:10.1126/science.1217230](https://doi.org/10.1126/science.1217230) Medline
8. A. Fernández-Ruiz, A. Oliva, E. Fermino de Oliveira, F. Rocha-Almeida, D. Tingley, G. Buzsáki, Long-duration hippocampal sharp wave ripples improve memory. *Science* **364**, 1082–1086 (2019). [doi:10.1126/science.aax0758](https://doi.org/10.1126/science.aax0758) Medline
9. M. Huelin Gorri, M. Takigawa, D. Bendor, The role of experience in prioritizing hippocampal replay. *Nat. Commun.* **14**, 8157 (2023). [doi:10.1038/s41467-023-43939-z](https://doi.org/10.1038/s41467-023-43939-z) Medline
10. H. Igata, Y. Ikegaya, T. Sasaki, Prioritized experience replays on a hippocampal predictive map for learning. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2011266118 (2021). [doi:10.1073/pnas.2011266118](https://doi.org/10.1073/pnas.2011266118) Medline
11. R. Huszár, Y. Zhang, H. Blockus, G. Buzsáki, Preconfigured dynamics in the hippocampus are guided by embryonic birthdate and rate of neurogenesis. *Nat. Neurosci.* **25**, 1201–1212 (2022). [doi:10.1038/s41593-022-01138-x](https://doi.org/10.1038/s41593-022-01138-x) Medline
12. E. L. Mackevicius, A. H. Bahle, A. H. Williams, S. Gu, N. I. Denisenko, M. S. Goldman, M. S. Fee, Unsupervised discovery of temporal sequences in high-dimensional datasets, with applications to neuroscience. *eLife* **8**, e38471 (2019). [doi:10.7554/eLife.38471](https://doi.org/10.7554/eLife.38471) Medline
13. L. McInnes, J. Healy, J. Melville, UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. [arXiv:1802.03426](https://arxiv.org/abs/1802.03426) [stat.ML] (2020).
14. W. Tang, J. D. Shin, S. P. Jadhav, Geometric transformation of cognitive maps for generalization across hippocampal-prefrontal circuits. *Cell Rep.* **42**, 112246 (2023). [doi:10.1016/j.celrep.2023.112246](https://doi.org/10.1016/j.celrep.2023.112246) Medline

15. W. Guo, J. J. Zhang, J. P. Newman, M. A. Wilson, Latent learning drives sleep-dependent plasticity in distinct CA1 subpopulations. *bioRxiv* 2020.02.27.967794 [Preprint] (2020); <https://doi.org/10.1101/2020.02.27.967794>.
16. A. Tsao, J. Sugar, L. Lu, C. Wang, J. J. Knierim, M.-B. Moser, E. I. Moser, Integrating time from experience in the lateral entorhinal cortex. *Nature* **561**, 57–62 (2018).
[doi:10.1038/s41586-018-0459-6](https://doi.org/10.1038/s41586-018-0459-6) [Medline](#)
17. A. D. Grosmark, G. Buzsáki, Diversity in neural firing dynamics supports both rigid and learned hippocampal sequences. *Science* **351**, 1440–1443 (2016).
[doi:10.1126/science.aad1935](https://doi.org/10.1126/science.aad1935)
18. C. Sun, W. Yang, J. Martin, S. Tonegawa, Hippocampal neurons represent events as transferable units of experience. *Nat. Neurosci.* **23**, 651–663 (2020). [doi:10.1038/s41593-020-0614-x](https://doi.org/10.1038/s41593-020-0614-x) [Medline](#)
19. S. Schneider, J. H. Lee, M. W. Mathis, Learnable latent embeddings for joint behavioural and neural analysis. *Nature* **617**, 360–368 (2023). [doi:10.1038/s41586-023-06031-6](https://doi.org/10.1038/s41586-023-06031-6) [Medline](#)
20. A. A. Fenton, R. U. Muller, Place cell discharge is extremely variable during individual passes of the rat through the firing field. *Proc. Natl. Acad. Sci. U.S.A.* **95**, 3182–3187 (1998). [doi:10.1073/pnas.95.6.3182](https://doi.org/10.1073/pnas.95.6.3182) [Medline](#)
21. J. D. Monaco, G. Rao, E. D. Roth, J. J. Knierim, Attentive scanning behavior drives one-trial potentiation of hippocampal place fields. *Nat. Neurosci.* **17**, 725–731 (2014).
[doi:10.1038/nn.3687](https://doi.org/10.1038/nn.3687) [Medline](#)
22. M. R. Mehta, M. C. Quirk, M. A. Wilson, Experience-dependent asymmetric shape of hippocampal receptive fields. *Neuron* **25**, 707–715 (2000). [doi:10.1016/S0896-6273\(00\)81072-7](https://doi.org/10.1016/S0896-6273(00)81072-7) [Medline](#)
23. I. Lee, G. Rao, J. J. Knierim, A double dissociation between hippocampal subfields: Differential time course of CA3 and CA1 place cells for processing changed environments. *Neuron* **42**, 803–815 (2004). [doi:10.1016/j.neuron.2004.05.010](https://doi.org/10.1016/j.neuron.2004.05.010) [Medline](#)
24. I. Lee, A. L. Griffin, E. A. Zilli, H. Eichenbaum, M. E. Hasselmo, Gradual translocation of spatial correlates of neuronal firing in the hippocampus toward prospective reward locations. *Neuron* **51**, 639–650 (2006). [doi:10.1016/j.neuron.2006.06.033](https://doi.org/10.1016/j.neuron.2006.06.033) [Medline](#)
25. C. E. Schoonover, S. N. Ohashi, R. Axel, A. J. P. Fink, Representational drift in primary olfactory cortex. *Nature* **594**, 541–546 (2021). [doi:10.1038/s41586-021-03628-7](https://doi.org/10.1038/s41586-021-03628-7) [Medline](#)
26. C. H. Vanderwolf, Hippocampal electrical activity and voluntary movement in the rat. *Electroencephalogr. Clin. Neurophysiol.* **26**, 407–418 (1969). [doi:10.1016/0013-4694\(69\)90092-3](https://doi.org/10.1016/0013-4694(69)90092-3) [Medline](#)
27. K. Diba, G. Buzsáki, Forward and reverse hippocampal place-cell sequences during ripples. *Nat. Neurosci.* **10**, 1241–1242 (2007). [doi:10.1038/nn1961](https://doi.org/10.1038/nn1961) [Medline](#)
28. D. J. Foster, M. A. Wilson, Reverse replay of behavioural sequences in hippocampal place cells during the awake state. *Nature* **440**, 680–683 (2006). [doi:10.1038/nature04587](https://doi.org/10.1038/nature04587) [Medline](#)
29. G. Buzsáki, Two-stage model of memory trace formation: A role for “noisy” brain states. *Neuroscience* **31**, 551–570 (1989). [doi:10.1016/0306-4522\(89\)90423-5](https://doi.org/10.1016/0306-4522(89)90423-5) [Medline](#)

30. J. Widloski, D. J. Foster, Flexible rerouting of hippocampal replay sequences around changing barriers in the absence of global place field remapping. *Neuron* **110**, 1547–1558.e8 (2022). [doi:10.1016/j.neuron.2022.02.002](https://doi.org/10.1016/j.neuron.2022.02.002) [Medline](#)
31. J. O'Neill, T. Senior, J. Csicsvari, Place-selective firing of CA1 pyramidal cells during sharp wave/ripple network patterns in exploratory behavior. *Neuron* **49**, 143–155 (2006). [doi:10.1016/j.neuron.2005.10.037](https://doi.org/10.1016/j.neuron.2005.10.037) [Medline](#)
32. C. Drieu, R. Todorova, M. Zugardo, Nested sequences of hippocampal assemblies during behavior support subsequent sleep replay. *Science* **362**, 675–679 (2018). [doi:10.1126/science.aat2952](https://doi.org/10.1126/science.aat2952) [Medline](#)
33. A. V. Samsonovich, G. A. Ascoli, A simple neural network model of the hippocampus suggesting its pathfinding role in episodic memory retrieval. *Learn. Mem.* **12**, 193–208 (2005). [doi:10.1101/lm.85205](https://doi.org/10.1101/lm.85205) [Medline](#)
34. D. Draaijers, N. Kafkafi, Y. Benjamini, G. Elmer, I. Golani, Rats and mice share common ethologically relevant parameters of exploratory behavior. *Behav. Brain Res.* **125**, 133–140 (2001). [doi:10.1016/S0166-4328\(01\)00290-X](https://doi.org/10.1016/S0166-4328(01)00290-X) [Medline](#)
35. D. L. Kramer, R. L. McLaughlin, The Behavioral Ecology of Intermittent Locomotion. *Am. Zool.* **41**, 137–153 (2001).
36. G. Buzsáki, Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning. *Hippocampus* **25**, 1073–1188 (2015). [doi:10.1002/hipo.22488](https://doi.org/10.1002/hipo.22488) [Medline](#)
37. H. R. Joo, L. M. Frank, The hippocampal sharp wave-ripple in memory retrieval for immediate use and consolidation. *Nat. Rev. Neurosci.* **19**, 744–757 (2018). [doi:10.1038/s41583-018-0077-1](https://doi.org/10.1038/s41583-018-0077-1) [Medline](#)
38. S. A. Josselyn, S. Tonegawa, Memory engrams: Recalling the past and imagining the future. *Science* **367**, eaaw4325 (2020). [doi:10.1126/science.aaw4325](https://doi.org/10.1126/science.aaw4325) [Medline](#)
39. D. Bendor, M. A. Wilson, Biasing the content of hippocampal replay during sleep. *Nat. Neurosci.* **15**, 1439–1444 (2012). [doi:10.1038/nn.3203](https://doi.org/10.1038/nn.3203) [Medline](#)
40. B. Rasch, C. Büchel, S. Gais, J. Born, Odor cues during slow-wave sleep prompt declarative memory consolidation. *Science* **315**, 1426–1429 (2007). [doi:10.1126/science.1138581](https://doi.org/10.1126/science.1138581) [Medline](#)
41. J. D. Rudoy, J. L. Voss, C. E. Westerberg, K. A. Paller, Strengthening individual memories by reactivating them during sleep. *Science* **326**, 1079 (2009). [doi:10.1126/science.1179013](https://doi.org/10.1126/science.1179013) [Medline](#)
42. L. Roux, B. Hu, R. Eichler, E. Stark, G. Buzsáki, Sharp wave ripples during learning stabilize the hippocampal spatial map. *Nat. Neurosci.* **20**, 845–853 (2017). [doi:10.1038/nn.4543](https://doi.org/10.1038/nn.4543) [Medline](#)
43. I. Gridchyn, P. Schoenenberger, J. O'Neill, J. Csicsvari, Assembly-Specific Disruption of Hippocampal Replay Leads to Selective Memory Deficit. *Neuron* **106**, 291–300.e6 (2020). [doi:10.1016/j.neuron.2020.01.021](https://doi.org/10.1016/j.neuron.2020.01.021) [Medline](#)
44. G. Girardeau, K. Benchenane, S. I. Wiener, G. Buzsáki, M. B. Zugardo, Selective suppression of hippocampal ripples impairs spatial memory. *Nat. Neurosci.* **12**, 1222–1223 (2009). [doi:10.1038/nn.2384](https://doi.org/10.1038/nn.2384) [Medline](#)

45. Y. Zhang, L. Cao, V. Varga, M. Jing, M. Karadas, Y. Li, G. Buzsáki, Cholinergic suppression of hippocampal sharp-wave ripples impairs working memory. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2016432118 (2021). [doi:10.1073/pnas.2016432118](https://doi.org/10.1073/pnas.2016432118) Medline
46. A. C. Singer, M. F. Carr, M. P. Karlsson, L. M. Frank, Hippocampal SWR activity predicts correct decisions during the initial learning of an alternation task. *Neuron* **77**, 1163–1173 (2013). [doi:10.1016/j.neuron.2013.01.027](https://doi.org/10.1016/j.neuron.2013.01.027) Medline
47. A. D. Grosmark, F. T. Sparks, M. J. Davis, A. Losonczy, Reactivation predicts the consolidation of unbiased long-term cognitive maps. *Nat. Neurosci.* **24**, 1574–1585 (2021). [doi:10.1038/s41593-021-00920-7](https://doi.org/10.1038/s41593-021-00920-7) Medline
48. T. Schreiner, C. F. Doeller, O. Jensen, B. Rasch, T. Staudigl, Theta Phase-Coordinated Memory Reactivation Reoccurs in a Slow-Oscillatory Rhythm during NREM Sleep. *Cell Rep.* **25**, 296–301 (2018). [doi:10.1016/j.celrep.2018.09.037](https://doi.org/10.1016/j.celrep.2018.09.037) Medline
49. A. Berners-Lee, T. Feng, D. Silva, X. Wu, E. R. Ambrose, B. E. Pfeiffer, D. J. Foster, Hippocampal replays appear after a single experience and incorporate greater detail with more experience. *Neuron* **110**, 1829–1842.e5 (2022). [doi:10.1016/j.neuron.2022.03.010](https://doi.org/10.1016/j.neuron.2022.03.010) Medline
50. W. Ramadan, O. Eschenko, S. J. Sara, Hippocampal sharp wave/ripples during sleep for consolidation of associative memory. *PLOS ONE* **4**, e6697 (2009). [doi:10.1371/journal.pone.0006697](https://doi.org/10.1371/journal.pone.0006697) Medline
51. N. Axmacher, C. E. Elger, J. Fell, Ripples in the medial temporal lobe are relevant for human memory consolidation. *Brain* **131**, 1806–1817 (2008). [doi:10.1093/brain/awn103](https://doi.org/10.1093/brain/awn103) Medline
52. B. Giri, H. Miyawaki, K. Mizuseki, S. Cheng, K. Diba, Hippocampal Reactivation Extends for Several Hours Following Novel Experience. *J. Neurosci.* **39**, 866–875 (2019). [doi:10.1523/JNEUROSCI.1950-18.2018](https://doi.org/10.1523/JNEUROSCI.1950-18.2018) Medline
53. S. Cheng, L. M. Frank, New experiences enhance coordinated neural activity in the hippocampus. *Neuron* **57**, 303–313 (2008). [doi:10.1016/j.neuron.2007.11.035](https://doi.org/10.1016/j.neuron.2007.11.035) Medline
54. M. F. Carr, S. P. Jadhav, L. M. Frank, Hippocampal replay in the awake state: A potential substrate for memory consolidation and retrieval. *Nat. Neurosci.* **14**, 147–153 (2011). [doi:10.1038/nn.2732](https://doi.org/10.1038/nn.2732) Medline
55. A. S. Gupta, M. A. A. van der Meer, D. S. Touretzky, A. D. Redish, Hippocampal replay is not a simple function of experience. *Neuron* **65**, 695–705 (2010). [doi:10.1016/j.neuron.2010.01.034](https://doi.org/10.1016/j.neuron.2010.01.034) Medline
56. A. Rubin, N. Geva, L. Sheintuch, Y. Ziv, Hippocampal ensemble dynamics timestamp events in long-term memory. *eLife* **4**, e12247 (2015). [doi:10.7554/eLife.12247](https://doi.org/10.7554/eLife.12247) Medline
57. G. Buzsáki, A. Bragin, J. J. Chrobak, Z. Nádasdy, A. Sik, M. Hsu, A. Ylinen, “Oscillatory and Intermittent Synchrony in the Hippocampus: Relevance to Memory Trace Formation” in *Temporal Coding of the Brain* (Springer, 1994), pp. 145–172.
58. B. Shahbaba, L. Li, F. Agostinelli, M. Saraf, K. W. Cooper, D. Haghverdian, G. A. Elias, P. Baldi, N. J. Fortin, Hippocampal ensembles represent sequential relationships among an extended sequence of nonspatial events. *Nat. Commun.* **13**, 787 (2022). [doi:10.1038/s41467-022-28057-6](https://doi.org/10.1038/s41467-022-28057-6) Medline

59. M. E. Montchal, Z. M. Reagh, M. A. Yassa, Precise temporal memories are supported by the lateral entorhinal cortex in humans. *Nat. Neurosci.* **22**, 284–288 (2019).
[doi:10.1038/s41593-018-0303-1](https://doi.org/10.1038/s41593-018-0303-1) [Medline](#)
60. M. W. Howard, M. J. Kahana, A Distributed Representation of Temporal Context. *J. Math. Psychol.* **46**, 269–299 (2002). [doi:10.1006/jmps.2001.1388](https://doi.org/10.1006/jmps.2001.1388)
61. P. Dayan, L. F. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems* (The MIT Press, 2001).
62. G. Alain, A. Lamb, C. Sankar, A. Courville, Y. Bengio, Variance Reduction in SGD by Distributed Importance Sampling. [arXiv:1511.06481](https://arxiv.org/abs/1511.06481) [stat.ML] (2013).
63. T. Schaul, J. Quan, I. Antonoglou, D. Silver, *Prioritized Experience Replay* (ICLR, 2016).
64. P. Zhao, T. Zhang, Stochastic Optimization with Importance Sampling. *Proc. Mach. Learn. Res.* **37**, 1–9 (2015).
65. C. Sun, W. Yang, T. Jiralerspong, D. Malenfant, B. Alsbury-Nealy, Y. Bengio, B. Richards, Contrastive introspection to identify critical steps in reinforcement learning. [arXiv:2210.05845](https://arxiv.org/abs/2210.05845) [cs.LG] (2022).
66. D. Levenstein, D. Tingley, B. Watson, A. Fernandez-Ruiz, M. Valero, P. Petersen, L. Sjulson, S. Mckenzie, R. Huszar, K. McClain, winniewyangwannan/buzcode, version 1, Zenodo (2024); <https://doi.org/10.5281/zenodo.10685428>.
67. W. Yang, winniewyangwannan/Selection-of-experience-for-memory-by-hippocampal-sharp-wave-ripples, Zenodo (2024); <https://doi.org/10.5281/zenodo.10685490>.