Automated long-term recording and analysis of neural activity in behaving animals

Ashesh K. Dhawale[1]*, Rajesh Poddar[1]*, Evi Kopelowitz[1], Valentin Normand[1,2], Steffen B. E. Wolff[1] and Bence P. Ölveczky[1]

1. Department of Organismic and Evolutionary Biology and Center for Brain Science, Harvard University, Cambridge MA, USA 02138
2. Present address: École Normale Supérieure, Paris, France 75005

* co-first authors

Summary

Addressing how neural circuits underlie behavior is routinely done by measuring electrical activity from single neurons during experimental sessions. While such recordings yield snapshots of neural dynamics during specified tasks, they are ill-suited for tracking single-unit activity over longer timescales relevant for most developmental and learning processes, or for capturing neural dynamics outside of task context. Here we describe an automated platform for continuous long-term recordings of neural activity and behavior in freely moving animals. An unsupervised algorithm identifies and tracks the activity of single units over weeks of recording, dramatically simplifying the analysis of large datasets. Months-long recordings from motor cortex and striatum made and analyzed with our system revealed remarkable stability in basic neuronal properties, such as firing rates and inter-spike interval distributions. Interneuronal correlations and the representation of different movements and behaviors were similarly stable. This establishes the feasibility of high-throughput long-term extracellular recordings in behaving animals.

Introduction

The goal of systems neuroscience is to understand how neural activity generates behavior. A common approach is to record from neuronal populations in targeted brain areas during experimental sessions while subjects perform designated tasks. Such intermittent recordings provide brief 'snapshots' of task-related neural dynamics, but fail to address how neural activity is modulated outside of task context and across behavioral states[1]. Furthermore, intermittent recordings are ill-suited for reliably tracking the same neurons over time, making it difficult to discern how neural dynamics and properties of individual neurons are shaped by developmental and learning processes that evolve over longer timescales[1].

Addressing such fundamental questions would be greatly helped by recording neural activity and behavior continuously over days and weeks in freely moving animals. Recording from the same neurons over long periods of time would allow their dynamics to be followed over more trials and experimental conditions, thus increasing the power with which inferences about neural function can be made. While in-vivo calcium imaging allows the same neuronal population to be recorded intermittently over longer durations[2–4], photo-bleaching, photo-toxicity, and cyto-toxicity[5,6], as well as the usual requirements for head-restraint in most versions of such experiments[2,3,7], make the method unsuitable for continuous long-term recording. Calcium imaging also has relatively poor temporal resolution[5,8], limiting its ability to resolve precise spike patterns[9,10]. In contrast, extracellular recordings using electrode arrays can measure the activity of many single neurons simultaneously with sub-millisecond resolution[11]. Despite the benefits of long-term electrical recordings, they are not routinely performed. A major reason is the inherently laborious and difficult process of reliably and efficiently isolating single units from large datasets, wherein firing rates of neurons can vary over many orders of magnitude and spike waveforms continuously change over time[12–14].

To address this issue, we designed and deployed a low-cost recording system that enables fully automated long-term continuous extracellular recordings from large numbers of neurons in freely behaving rats engaged in natural behaviors and prescribed tasks. To efficiently parse the large streams a neural data, we developed an unsupervised spike sorting algorithm

that automatically processes the raw data from electrode array recordings and clusters spiking events into putative single units, tracking their activity over long timescales. We used this integrated system to record from motor cortex and striatum continuously over several months. These experiments revealed a remarkable stability in basic neuronal properties, such as firing rates and inter-spike interval distributions. We also found interneuronal correlations and movement tuning across a range of behaviors to be highly stable.

By streamlining and automating the acquisition and analysis of large behavioral and neural datasets, our system makes continuous high-throughput long-term electrical recordings in behaving animals a feasible prospect.

Results

Infrastructure for automated long-term neural recordings in behaving animals

We developed experimental infrastructure for continuous long-term extracellular recordings in behaving rodents. Our starting point was ARTS, a fully Automated Rodent Training System we previously developed[15]. In ARTS, the animal's home-cage doubles as the experimental chamber, making it a suitable platform for continuous long-term recordings.

To ensure that animals remain reliably and comfortably connected to the recording apparatus over months-long experiments, we designed a tethering system that allows animals to move around freely while preventing them from reaching (and chewing) the signal cable (Figure 1A; see Methods for details). Our solution connects the implanted recording device via a cable to a passive commutator attached to a carriage that rides on a low-friction linear slide. The carriage is counterweighted by a pulley, resulting in a small constant upwards force (< 10g) on the cable, keeping it taut and out of the animal's reach without unduly affecting its movements. The recording extension can easily be added to our custom home-cages, allowing animals that have been trained, prescreened, and deemed suitable for recordings to be implanted with electrode drives, and placed back into their familiar training environment (i.e. home-cage) for recordings.

Extracellular signals recorded from behaving animals are amplified and digitized on a custom-designed head-stage (Figure 1B, Methods). To characterize the behavior of animals throughout the recordings, we incorporate a 3-axis accelerometer on the head-stage that records head movements at high temporal resolution (Figure 1A-B). We also record continuous video of the rats' behavior with a wide-angle camera above the cage (Methods). The large volumes of behavioral and neural data (~0.5 TB/day/rat) are streamed to custom-built high-capacity servers.

An automated solution for analyzing neural data from long-term recordings

Extracting single-unit spiking activity from the vast amounts of raw data collected over weeks and months of continuous extracellular recordings presents a significant challenge for which

there is currently no adequate solution. Parsing such large datasets must necessarily rely on automated spike-sorting. Such a method faces three major challenges[16]. First, it must capture the activity of simultaneously recorded neurons whose firing rates can vary over many orders of magnitude[17,18]. Second, it has contend with the spike shapes from recorded units changing over time[12–14]. Third, it must process very large numbers of spikes in a reliable and efficient manner (in our experience $>10^{10}$ spikes per rat over a time span of 3 months for 64 channel recordings).

We developed an unsupervised spike sorting algorithm that meets these challenges and allows single units to be tracked over months-long timescales. Our method comprises two main steps. First, to compress the datasets and normalize for large variations in firing rates between units, we apply 'local clustering' to create a de-noised representation of the spikes in our data. In the second step, we chain together de-noised spikes belonging to the same putative single unit over time using an integer linear programming algorithm[19]. Our modular data processing pipeline is efficient and, depending on overall spike rates, runs two to three times faster than the acquisition rate of the electrophysiological data.

To parse and compress the raw data, we first identify and extract spike events by bandpass filtering and thresholding each electrode channel (Methods, Supplementary Figure 1). Spike clustering is then performed on blocks of 1000 consecutive spike 'snippets' by means of an automated superparamagnetic clustering routine, a step we call 'local clustering'[20,21] (Methods, Figure 1C, Supplementary Figure 2). The block size (1000 spikes) is chosen as a compromise between improving the reliability of the clustering by pooling more observations, reducing the computation time which scales quadratically with the number of spikes, and limiting changes in spike waveforms over time to improve cluster separation. Spikes in a block belonging to the same cluster are represented by their centroid, a step that effectively de-noises and compresses the data by replacing groups of similar spikes with their centroids. However, due to large differences in firing rates across different units, the initial blocks of 1000 spikes will be dominated by high firing rate units. Spikes from more sparsely firing cells that do not contribute at least 15 spikes to a cluster are carried forward to the next round of local clustering, where previously assigned spikes have been removed (Methods, Supplementary Figure 2). Applying this method of pooling and local clustering sequentially four times,

generates a de-noised dataset that accounts for large differences in the firing rates of simultaneously recorded units (Figure 1C, Methods).

The second step of our algorithm is inspired by an automated method ('segmentation fusion') that links similar elements over cross-sections of longitudinal datasets, and does so in a globally optimal manner (Methods, Supplementary Figure 3). This algorithm has been used to trace processes of individual neurons across stacks of two-dimensional serial electron-microscope images[19,22]. We adapted this method to link similar de-noised spikes across consecutive blocks into *chains* containing the spikes of putative single units over longer timescales (Figure 1C). This algorithm allows us to automatically track the same units over days and weeks of recording.

In a final post-processing and verification step, we use a semi-automated method to link 'chains' belonging to the same putative single unit together, and perform visual inspection of each unit. A detailed description of the various steps involved in the automated spike sorting can be found in Methods.

Continuous long-term recordings from motor cortex and striatum

To demonstrate the utility of our experimental platform and analysis pipeline for long-term neural recordings, we implanted tetrode drives (16 tetrodes, 64 channels) into dorsolateral striatum (n=1) or motor cortex (n=1) of rats (Methods). We recorded electrophysiological and behavioral data continuously, or with only brief interruptions, for more than 3 months. We note that our recordings terminated not because of technical issues with the implant or recording apparatus, but because we deemed the experiments to have reached their end points.

We used our automated spike sorting method to cluster spike waveforms into putative single units, isolating a total of 1031 units from motor cortex and 719 units from striatum (Figure 2A). On average, single units were tracked over several days (median holding time of 1.9 days for motor cortex and 2.8 for striatum), with significant fractions of units recorded continuously for more than two weeks (3.0 % and 4.7 % in motor cortex and striatum, respectively) and even a month (0.4% in motor cortex, Figure 2B). Periods of stable recordings

were interrupted by either intentional advancement of the electrode drive or spontaneous events likely related to sudden movement of the electrodes (Figure 2A). On average, we recorded simultaneously from 30.2 ± 15.2 units in motor cortex and 27.2 ± 18.9 units in striatum (mean ± standard deviation), although these numbers varied significantly across animals and over the lifetime of the recordings (Figure 2C).

The quality of single unit isolation was assessed by computing quantitative measures of cluster quality, i.e. cluster isolation distance[23], L-ratio[24], and presence of a clean refractory period (Figure 2D). When we calculated mean cluster quality for units over their recording lifetimes, 87.0% of motor cortical (n=897) and 77.5% of striatal units (n=554) satisfied our conservative criteria (Isolation distance >= 25, L-ratio <= 0.3 and fraction of ISIs below 1 ms <= 1%). However, 98.1% of motor cortical units (n=1011) and 96.2% of striatal units (n=692) met these criteria for at least one hour of recording.

In the absence of any ground truth information with which to benchmark the performance of our unsupervised algorithm[16], we compared clusters identified by our algorithm to those obtained from manual spike sorting within specified time-windows (Methods). Our unsupervised method successfully identified 72.1% of manually sorted spike clusters that exceeded our cluster quality criteria (n = 31 of 43 total), and 90.0 % of high-quality clusters (Isolation distance >= 35, L-ratio <= 0.01, n = 18 of 20 total, Supplementary Figure 4A). For clusters that were matched across the different sorting methods (identified by at least 50% spike overlap), the unsupervised algorithm was able to recover 93.3 ± 4.4% (median ± median absolute deviation) of spikes identified by manual sorting, and 95.5 ± 2.2% for high-quality clusters (Supplementary Figure 4B). Finally, the isolation quality of matched clusters was, on average, slightly better for the unsupervised algorithm as compared to manual clustering (Supplementary Figure 4C). This suggests that our unsupervised method's performance is comparable to manual sorting.

Stability of neural dynamics

We clustered and classified motor cortical and striatal units into putative principal-neurons and interneurons based on their spike shapes and firing rates[25–27]. We putatively identified 264 fast

spiking and 455 medium spiny neuron units in striatum, and 591 fast spiking and 440 regular spiking units in motor cortex (Figure 3A). As previously reported[17,18], average firing rates were log-normally distributed and varied over more than three orders of magnitude (from 0.029 Hz to 37.2 Hz, Figure 3A).

We used our unique datasets to explore the stability of single unit firing rates over time, comparing mean firing rates across 1, 5, or 10-day time lags (Methods). We found that firing rates were very stable even over 10 days of recording (Figures 3B-D), with day-to day differences being significantly lower than the average difference between simultaneously recorded neurons of the same putative unit-type (Figure 3D, $p < 1e-6$). These results suggest that firing rates in cortex and striatum are stable over long timescales, and that individual units within the same putative unit type have their own firing rate set-point that is stably maintained[28,29].

We next examined the stability of second-order properties of neural activity characterized by the inter-spike interval (ISI) distribution, a measure sensitive to a cell's mode of spiking (bursting, tonic etc.) and other intrinsic properties such as refractoriness, spike frequency adaptation. We found that the ISI distribution of single units on different days were virtually identical, suggesting a remarkable stability in the spiking statistics of neurons over time (Figure 3E). Over all recorded units, similarity was far higher between ISI distributions of the same neuron across different time-lags, than between different neurons within the same unit type recorded at the same time (Figure 3F, $p < 1e-6$).

However, measures of single unit activity do not address the stability of the network in which the neurons are embedded, as they do not account for interneuronal correlations[30]. Thus to further address the stability of circuit dynamics, we calculated the cross-correlograms of all simultaneously recorded neuron pairs (Figure 4A, Methods). We found that 34 % of the pairs recorded in the striatum (n=22,134 pairs) and 44% in motor cortex (n=32,567 pairs) were significantly correlated (Methods). Out of these, 19% (striatum) and 31% (motor cortex) had negative correlations (i.e. values below 1 in Figure 3B). The average time-lag of the significantly correlated pairs was 19.1±16.17 ms (median: 8.75 ms) and 23.7±4.7 ms (median: 12 ms) for striatum and motor cortex respectively, suggesting mostly indirect interactions. The pairwise

correlations were remarkable stable, remaining essentially unchanged even after 10 days (Figures 4B,C), consistent with a very stable underlying network[31,32].

Automated classification of behavioral states from continuous behavioral recordings

Our results demonstrated a remarkable long-term stability in the time-averaged statistics of single units (Figure 3) as well as their interactions (Figure 4), both in motor cortex and striatum. However, the brain functions to control behavior, and the activity of both cortical and striatal units can be expected to differ for different behaviors. To better understand the relationship between the firing properties of single units and ongoing behavior, and to assess the degree to which these relationships are stable over time, we analyzed the activity of single units in different behavioral states. To this end, we developed an algorithm for automatically classifying a rats' behavior into 'states' using high-resolution measurements of head acceleration and local field potentials (LFPs). Our method distinguishes grooming, eating, active exploration, task engagement as well as quiet wakefulness, rapid eye movement and slow wave sleep [33,34] (Figure 5A, Methods). We were able to assign ~85% of the recordings to one of these states and benchmarked our method against behavioral classifications made by a human observer scoring videos of the rats' behavior (see Methods).

Consistent with prior studies[33], we found that rats spent most of the time sleeping (49±6%, n=2) or resting (17±8%) (Figure 5B). Sleep occurred at all hours, but was more frequent when lights were on (51±11%) versus when they were off (38±17%), consistent with the nocturnal habits of rats (Figure 5C). We found that, within individual rats, the fraction of time spent in each behavioral state was stable (Figure 5B) as were the circadian patterns (Figure 5C), though rats showed individual differences in behavioral patterns (Figures 5B,C).

Stability and clustering of state dependent firing rates

Our ability to reliably classify ongoing behavior into different active and quiescent 'states' allowed us to examine the extent to which average firing rates of motor cortical and striatal units depend on behavioral states. Not surprisingly, we found that firing rates of both striatal and motor cortical units depended on what the animal was doing (Figure 6A). Interestingly, the

behavioral state modulation of the firing rates was largely stable throughout the units' recording lifetimes (Figure 6A). Indeed, for the population of all recorded units, the similarity of the state-modulated firing rate (or 'behavioral state tuning') for the *same neuron* across days, was significantly higher than for state tuning curves *between neurons* recorded on the same day (Figure 6B, $p < 1e\text{-}6$).

We noticed striking similarities in the behavioral state tuning of groups of neurons. To look at this more systematically, we applied a clustering algorithm (k-means) that classifies neurons based on their state-tuning (Methods). We found that units in both motor cortex and striatum clustered into relatively few classes (Figure 6C). Interestingly, although the clustering was carried out independently for cortical and striatal populations, the functional classes we identified were remarkably similar across the two brain areas (Figure 6C). For example, neurons belonging to cluster-type 3 (cortex) and 4 (striatum) fired predominantly when the rat was eating, while units in cluster-type 5 (cortex) and 6 (striatum) fired in all 'active' states, i.e. grooming, eating, exploration and task execution.

Stability of coupling between neural and behavioral dynamics

Neurons in motor-related areas encode and generate motor output, yet the degree to which the relationship between spiking activity in single neurons and various movement parameters (i.e. a cell's 'motor tuning') is stable over days and weeks has been the subject of recent debate[35–39]. Continuous recordings during natural and learned behaviors over days and weeks constitute unique datasets for characterizing the stability of movement coding at the level of single neurons.

Spike-triggered average (STA) is an analytical tool commonly used in sensory neuroscience to characterize a neuron's 'receptive field' in stimulus space[40,41]. We adapted this method to characterize the 'response fields' in movement space for units recorded in both motor cortex and striatum. For each unit, we computed STAs of the accelerometer power for different active states, i.e. grooming, active exploration, and eating (Methods). The percentage of striatal units that had significant coupling to grooming, exploration and eating as measured by the STAs was 17.1% (n=118), 16.0% (n=111) and 17.5% (n=121), respectively. In motor

cortex, the fractions of units with significant response fields was 28.8% (n=291), 23.6% (n=239) and 32.7% (n=331) for the three active states respectively. Movement STAs varied substantially between neurons and across behavioral states, but were remarkably stable for the same unit when compared over days within a particular behavioral state (Figure 7A). In fact, movement STAs calculated for the same unit were stable for at least 10 days (Figure 7B). In comparison, similarity of STAs between simultaneously recorded units was much lower (Figure 7B, $p < 1e\text{-}6$).

Another measure used to characterize the neural encoding of behavior at fast timescales is the average neural activity around the time of salient behavioral events, or the 'peri-event time histogram' (PETH). We calculated PETHs related to the first lever-press in a learned lever-press sequence[42], or to the entry into a reward port after a successful trial ('nose-poke'). The fraction of units whose activity was significantly locked to the lever-press or nose-poke was 16.9% (n=117) and 19.3% (n=134) in striatum, and 15.9% (n=161) and 23.8% (n=241) in motor cortex (Figure 7C). When comparing PETHs of different units for the same behavioral events, we observed that the time of peak firing was distributed across a range of delays relative to the time of the events (Figure 7C). However, despite the heterogeneity in PETHs across the population of units, the PETHs of individual units were remarkably similar when compared across days (Figure 7D-F). In contrast, the similarity between PETHs for simultaneously recorded units was significantly lower (Figure 7F, $p < 1e\text{-}2$).

Our analysis of the state-dependence of *average* unit activity identified functional clusters of units with similar profiles (Figure 6C). We examined whether neurons belonging to the same functional cluster have similar motor tuning by assessing their movement STAs and PETHs for nose-poke and lever-press events. We found that STAs and PETHs for units within a cluster were as dissimilar from one another as those for units belonging to different clusters (Supplementary Figure 5).

Discussion

Recording from populations of single neurons in behaving animals has traditionally been a laborious undertaking both in terms of experimentation and analysis. Automating this process and extending the recordings over long time-periods increases the efficiency and power of such experiments in multiple ways. First, it eliminates the manual steps of intermittent recordings, such as transferring animals to and from their recording chambers, plugging and unplugging recording cables etc., which besides being time-consuming can be detrimental to the recordings and stressful for the animals. Second, when combined with fully automated home-cage training[15], this approach allows the neural correlates of weeks- and months-long learning processes to be studied in an efficient manner. Indeed, our system should enable a single researcher to supervise tens, if not hundreds, of recordings simultaneously, thus radically improving the ease and throughput with which such experiments can be performed. Third, continuous recordings allow the activity of the same neurons to be tracked over days and weeks, allowing their activity patterns to be pooled and compared for more trials, and across different experimental conditions and behavioral states, thus increasing the power with which the principles of neural function can be inferred[1].

Despite their many advantages, continuous long-term neural recordings are not routinely performed. A major reason is that state-of-the-art methods for processing data from extracellular recordings require significant human input, making the analysis of large datasets prohibitively time-consuming. Our automated spike sorting algorithm overcomes this bottleneck, dramatically increasing the feasibility of continuous and high-throughput long-term recordings of extracellular signals in behaving animals.

Long-term stability of neural dynamics

Though functional calcium imaging allows the activity of hundreds of neurons to be followed across days[1–3], it reports a relatively slow correlate of neural activity, making it difficult to reliably resolve individual spike events and hence to assess fine timescale neuronal interactions[1,5]. Furthermore, calcium indicators function as chelators of free calcium ions[43,44] and could interfere with calcium-dependent plasticity processes[45] and, over the long-term, also

compromise the health of the cells[6]. These variables may have contributed to discrepancies in the findings from longitudinal calcium imaging experiments, with some studies reporting dramatic day-to-day fluctuation in neural activity patterns[2] while others report more stable representations[3], leaving open the question of how stable the brain really is.

While electrical measurements of spiking activity do not suffer from the same drawbacks, intermittent recordings may fail to reliably track the same neurons over multiple days. Attempts at inferring long-term stability of movement-related neural dynamics from such experiments have produced conflicting results, with some studies reporting stability[14,36,37,46], while others finding significant fluctuations even within a single day[35,38]. Our continuous recordings, which allow for the characterization of single neuron properties over long time–periods, suggest a remarkable stability in both spiking statistics (Figures 3), neuronal interactions (Figure 4), and tuning properties (Figures 6,7) of motor cortical and striatal units.

Faced with constant turnover of ion channels, receptor proteins and dendritic spines [47–52], maintaining such long-term stability may require active processes, such as homeostatic regulation of neural activity[29,53–55]. Our findings are consistent with this and suggest that neurons maintain their activity levels with respect to their own individual set-points (Figure 3).

We further observed interneuronal correlations both in motor cortex and the striatum remaining largely unchanged even over the timescales of many days (Figure 4). These observations taken together with studies showing long-term structural stability at the level of synapses[31,32] suggest that neural networks are, over all, highly stable systems, both in terms of their dynamics and connectivity.

Functional cell assemblies in the brain

Interestingly, we found that groups of neurons showed similar firing rate modulation across behavioral states (Figure 6C). Trivially, such groupings may be the consequence of units having similar motor tuning[37]. If so, their characteristic state-modulated firing profiles would simply reflect the fact that the movements they encode are preferentially associated with particular behavioral states (e.g. mastication-related neurons being mapped to the ‘eating state’). However, we did not find any evidence for this; rather, our results show that the motor tunings

of neurons within the same functional grouping were as different as across groupings (Supplementary Figure 5).

Alternatively, such groupings could represent functional cell assemblies[56,57] of neurons that are co-active during specific behaviors or behavioral states. Having dedicated neuronal assemblies encode and generate the animal's motor output in different contexts and for different behaviors could serve to make the acquisition of new behaviors more efficient and the execution of already acquired skills more robust. That the characteristics of these groupings appear similar across both motor cortex and the striatum suggest that such subassemblies may be distributed across large networks in the brain[58].

Future improvements

Improvements in recording technology have the potential to dramatically increase the number of neurons that can be simultaneously recorded from, as well as making the recordings more stable[59]. Our modular and flexible experimental platform can easily incorporate such innovations. Importantly, our novel and automated analysis pipeline solves a major bottleneck downstream of these solutions by allowing increasingly large datasets to be efficiently parsed, thus making continuous high-throughput recordings of large numbers of neurons in behaving animals a feasible prospect.

Methods

Animals

The care and experimental manipulation of all animals were reviewed and approved by the Harvard Institutional Animal Care and Use Committee. Experimental subjects were female Long Evans rats, 3-8 months old at the start of the experiment (n=2, Charles River).

Behavioral training

Before implantation, rats (n=2) were trained twice daily on a timed lever-pressing task[42] using our fully automated home-cage training system[15]. Once animals reached asymptotic performance on the task, they were removed from their home-cages and underwent surgery to implant tetrode drives into dorsolateral striatum (n=1) and motor cortex (n=1) respectively (see below). After 7 days of recovery, rats were placed back into their home-cages, which had been outfitted with an electrophysiology recording extension (Figure 1A). The cage was placed in an acoustic isolation box, and training on the task resumed. Neural and behavioral data was recorded continuously for 16 weeks with only brief interruptions (median time of 0.2 hours) for occasional troubleshooting.

Surgery

Rats underwent surgery to implant a custom-built recording device (see below). Animals were anesthetized with 1-3% isoflurane and placed in a stereotax. The skin was removed to expose the skull, the surface of which was then thoroughly cleaned and swabbed dry of fluids. Five bone screws (MD-1310, BASi, West Lafayette, IN), including one soldered to a 200 µm diameter silver ground wire (786500, A-M Systems, Sequim, WA), were driven into the skull to anchor the implant. A stainless-steel reference wire (50 µm diameter, 790700, AM-Systems, WA) was implanted in the external capsule to a depth of 2.5 mm, at a location posterior and contralateral to the implant site of the electrode array. A 4-5 mm diameter craniotomy was made at a location 2 mm anterior and 3 mm lateral to bregma for targeting electrodes to motor cortex, and 0.5 mm anterior and 4 mm lateral to bregma for targeting dorsolateral striatum. After removing the dura-mater, the pia-mater encircling the implant site was glued to the

surrounding skull with cyanoacrylate glue (Krazy glue, OH) to minimize relative motion between the brain and the skull-affixed electrode implant during subsequent awake recordings. Prior to applying the glue, a drop of ophthalmic ointment (Puralube Vet Ointment) was placed in the center of the craniotomy (approximately 2 mm diameter) to prevent glue from reaching the site of electrode entry[60]. The pia-mater was then weakened using a solution of 20 mg/ml collagenase (Sigma, St Louis, MO) and 0.36 mM calcium chloride in 50 mM HEPES buffer (pH 7.4) in order to minimize dimpling of the brain surface during electrode penetration [60]. The 16-tetrode array was then slowly lowered to the desired depth of 1.85 mm for motor cortex and 4.5 mm for striatum. The microdrive was encased in a protective shield and cemented to the skull by applying a base layer of Metabond luting cement (Parkell, Edgewood NY) followed by dental acrylic (A-M systems, WA).

Histology

At the end of the experiments, animals were anesthetized and anodal current (30 µA for 30 seconds) passed through select electrodes to create micro-lesions at the electrode tips. Animals were transcardially perfused with PBS and subsequently fixed with 4% paraformaldehyde (PFA) in PBS. Brains were removed and post-fixed in 4% PFA. Coronal sections (60 µm) were cut on a Vibratome (Leica), mounted, and stained with cresyl violet to reconstruct the location of implanted electrodes. All electrode tracks were consistent with the recordings having been done in the targeted areas.

Continuous behavioral monitoring

To monitor the rats' head movements continuously, we placed a small 3-axis accelerometer (ADXL 335, Analog Devices, Norwood, MA) on the recording head-stage. The output of the accelerometer was sampled at 7.5 kHz per axis. We also recorded 24x7 continuous video at 30 frames per second with a CCD camera (Flea 3) or a webcam (Agama V-1325R). Videos were compressed by H-264 encoding for storage on disk in the Matroska multimedia container (mkv) format. Video was synchronized to electrophysiological signals by recording TTL pulses from the CCD cameras that signaled frame capture times.

Analysis and automatic classification of behavior

We developed an unsupervised algorithm to classify behaviors based on accelerometer data, LFPs, raw spike data, and task event times [33,34]. Behaviors were classified at 1 second resolution into one of the following 'states': grooming (GRO), eating (EAT), active exploration (AX), task engagement (TSK), quiet wakefulness (RST), rapid eye movement sleep (REM) or slow-wave sleep (SWS).

Processing of accelerometer data: We performed wavelet analysis on the accelerometer signals to reconstruct a 5th level approximation. This reconstruction, which captures information relating to slow movements (similar to low pass filtering) was subtracted from the original signal. Accelerometer power was calculated in 1 ms bins from the 0-50 Hz frequency band. The power had a bimodal distribution with one narrow peak at low values corresponding to immobility and a secondary broad peak corresponding to epochs of movement. The threshold for immobility was set to best discriminate the two peaks.

Processing of the LFP: We computed the spectrogram for the LFP signal on each electrode in 1 second bins. The delta (1-4 Hz) and theta (5-9 Hz) power was normalized by the total power, averaged over all 64 channels, and smoothed with a 10 second boxcar filter. The delta power had a bimodal distribution with a broad peak around the low values corresponding to awake states and a secondary narrow peak with higher values corresponding to epochs of immobility. The threshold for high delta values was set to best discriminate the two peaks and was used to identify SWS epochs (see Figure 5A). Similarly, the theta power had a bimodal distribution with a narrow peak at low values corresponding to REM sleep states and a secondary broad peak at higher values. The threshold for high theta values was set to best discriminate the two peaks and was used to identify REM epochs (see Figure 5A).

Classification of inactive states – REM and slow-wave sleep, and quiet wakefulness. Inactive states, i.e. when rats are immobile, were identified as bins having total accelerometer power below the immobility threshold (see above). Slow-wave sleep (SWS) states were distinguished by high delta power relative to the delta power threshold (see above), and similarly, REM sleep states were distinguished by high theta power relative to the theta threshold (see above)[33,61].

The ratio index delta/(delta+theta) was then used as a secondary measure to distinguish SWS and REM from non-sleep states, with an index >0.4 indicating SWS and an index<0.4 indicating REM[33,61], such that only epochs with high delta power and ratio index > 0.4 were identified as SWS, and only epochs with high theta power and ratio index < 0.4 were identified as REM. The immobile epochs that were not classified as sleeping were labelled as quiet wakefulness.

Classification of active states – grooming, eating, and active exploration. Active states were characterized by the accelerometer power being above the immobility threshold.

Grooming epochs are characterized by repetitive licking, stroking and scratching. To identify these rhythmic behaviors, we calculated the periodogram of the spectrogram in 1 second bins and evaluated the power in ± (5-50 Hz). The distribution of these values had a Gaussian shape with a non-Gaussian long tail towards high values. The threshold for high power, indicating rhythmic movements associated with repetitive grooming, was set to on the 95th percentile of the Gaussian distribution.

Eating epochs were classified based on a strong common-mode oscillatory signal on all electrodes, likely due to electrical interference from mastication. To extract eating epochs, the raw spike data was down-sampled 10-fold (from 30 kHz to 3kHz), filtered by a 2th order Butterworth filter and smoothed with a 0.3 second averaging window. We computed the spectrograms for the filtered data in one second windows and calculated the total power in the 0-50 Hz frequency band. The distribution of these values had a Gaussian shape with a non-Gaussian long tail towards high values. The threshold for eating was set to the 95th percentile of the Gaussian distribution.

Task engagement was identified based on lever-press event times. If less than 2 seconds had elapsed between presses (corresponding to the average trial length) the time between them was classified as task engagement. Active epochs that were not classified as grooming, eating or task engagement were classified as *active exploration*. The algorithm labeled each time bin exclusively, i.e. only one label was possible for each bin. Bins corresponding to multiple states (e.g. eating and grooming) where classified as *unlabeled*, as were times when no data was available due to brief pauses in the recording.

Removal of seizure-like epochs. Long Evans rats are susceptible to absence seizures, characterized by strong ~8Hz synchronous oscillations in neural activity and associated whisker twitching[33,62]. To identify seizure-like episodes, we calculated the autocorrelation of the raw spike data on all electrodes in 2 second windows. To identify peaks associated with seizure-like states, we calculated the periodogram of each autocorrelation and evaluated the power in ± (7-9 Hz). The distribution of these values had a Gaussian shape with a non-Gaussian long tail towards high values. The threshold for seizure-like states was set to on the 95th percentile of the Gaussian distribution. Using this classification, 11% of the time was classified as seizure (10±2% for the DLS rat and 12±3% for the MC rat) which is comparable with previously published reports[63]. These epochs were removed from the behavioral state analysis.

Benchmarking the state classification algorithm. We benchmarked our automated classification algorithm against a human observer scoring 12 hours of videos of the rats' behavior. For every state scored manually (superscript 'm') we calculated the distribution of the algorithm scores of the same time bins (superscript 'a').

	Inactive^a	Grooming^a	Exploring^a	Eating^a	Task^a	Seizure^a	Unlabeled^a
Inactive^m	82.2	0	0.7	0.5	0	5	11.6
Grooming^m	1.7	29.4	43.2	0.3	6.1	1.8	17.5
Exploring^m	3.1	9.4	56.3	0.5	5	4.6	21.1
Eating^m	0	1.5	17.3	79	0	0	2.2
Task^m	0	1	3	0	96	0	0

While the labels largely overlapped, a significant discrepancy was in the identification of 'grooming'. This is because much of the grooming is not repetitive enough to be distinguished from other exploratory behaviors. However the vast majority of grooming episodes picked out by our algorithm were indeed classified as grooming also by the human observer. Thus the more accurate description of grooming in our analysis would be 'repetitive' or 'periodic' grooming.

Analysis of Neural Data

All analysis of neural data was carried out using custom scripts in Matlab (Mathworks, Natick, MA).

Cluster quality. After spike sorting (see below), we computed standard measures of cluster quality, specifically the isolation distance[23], L-ratio[24] and fraction of inter-spike intervals less than 1 ms, for every unit within consecutive one-hour blocks. Isolation distances and L-ratios were calculated with four features for every electrode channel – the waveform peak, waveform energy (L^2 norm), and the projections on the first two principal components of the spike waveform. Only units meeting all of our quality criteria (isolation distance => 25, L-ratio <= 0.3, fraction ISI less than 1 ms <= 0.01) in a particular block were included in further analysis.

Comparison to manual spike sorting. We performed manual spike sorting for 2 separate hour-long blocks from a continuous electrophysiology dataset recorded in the dorsolateral striatum. Manual sorting was performed with M-Clust 3.5 (A. D. Redish et al). The resultant clusters were then compared to those obtained by our unsupervised algorithm as described in the Results and Supplementary Figure 4.

Identification of putative cell types. We used mean spike waveforms and average firing rate[25,26] to separate units into putative cell types: medium spiny neurons (MSNs) and fast spiking neurons (FS) in the striatum, and regular spiking (RS) and fast spiking (FS) neurons in the cortex. Briefly, we performed k-means clustering of units based on their peak-normalized spike-waveform, concatenated with their log-transformed firing rates. The number of clusters (k=2) was chosen by visual inspection of unit waveforms in three dimensions - the first and second principal components of the spike waveform, and unit firing rate.

Firing rate stability. Firing rate similarity compared across different days i and j for the same unit, or when comparing distinct units i and j recorded on the same day, was measured by the following formula:

$$FR\ similarity_{i,j} = 1 - 2\left(\frac{abs(FR_i - FR_j)}{FR_i + FR_j}\right) \qquad \text{(eq.1)},$$

where FR is the firing rate. A firing rate similarity score of 1 means that the two firing rates FR_i and FR_j are identical while a firing rate similarity score of -1 implies maximum dissimilarity in firing rates, such that one of the firing rates is 0. When comparing firing rates for the *same unit across time*, we calculated average firing rate similarity scores for time-lags ranging from 1 to 10 days. When comparing firing rates *across units* and within the same time-bin, we averaged together all similarity scores for pairs of units of the same putative unit-type recorded in that time-bin.

Stability of inter-spike interval (ISI) distribution. We computed ISI histograms using 50 log-spaced time-bins spanning inter-spike intervals from 1 ms to 1000 sec. We used Pearson's correlation coefficient to measure the similarity between ISI distributions on different days for the same unit, and between ISI distributions of distinct units recorded on the same day. For each unit, as with the firing rate similarity measure described above, we averaged the ISI correlations for each time-lag ranging from 1 to 10 days. When comparing across units, we only compared their ISI distributions to other simultaneously recorded units of the same putative unit-type.

Stability of state-dependent firing rates. We used the Pearson's correlation coefficient to estimate the similarity between units' state-dependent firing rate profiles, either across days, or between different units recorded simultaneously.

Spike-triggered averages (STA). For each unit, we computed spike-triggered averages of accelerometer power in three different behavioral states – eating, grooming and active exploration. The accelerometer power P_{acc} was calculated as:

$$P_{acc} = \sqrt{acc_x^2 + acc_y^2 + acc_z^2} \qquad \text{(eq.2)},$$

where acc_i is the accelerometer signal on channel i. The accelerometer power was band-pass filtered between 0.5 Hz and 300 Hz using a 3rd order Butterworth filter and zero-phase filtering. Spike-triggered averages (STA) were computed for each unit in a particular behavioral state by

averaging, over spikes, the accelerometer power in a window of ± 500 ms centered on each spike. Formally,

$$STA_{acc}(\tau) = \frac{1}{n}\sum_{i=1}^{n} P_{acc}(t_i + \tau) \qquad \text{(eq. 3)},$$

where τ is the time-lag, sampled at 3.33 ms resolution, between spike and accelerometer power ranging from -500 to +500 ms, and t_i the arrival times of unit spikes (a total of n spikes) in a particular behavioral state. We only considered 'trigger-spikes' that were at least 500 ms away from a behavioral state-transition.

To measure the significance of the resultant STA kernels, we computed noise STAs after jittering spike times by ± 500 ms. We used the standard deviation of the noise STAs across time-bins to calculate the Z-scored STAs. Any STA having at least 2 bins equal to or exceeding a Z-score of 5 was considered statistically significant and used for further analysis.

Just as with ISI distribution similarity, we calculated the similarity between STAs for the same unit on different days using Pearson's correlation coefficient. We also compared similarity of STAs across simultaneously recorded units, but only within the same behavioral state.

Peri-event time histograms (PETHs). We computed peri-event time histograms (PETHs) of instantaneous firing rates aligned to specific behavioral events during execution of a lever-pressing task [42]. We computed PETHs in windows ±200 ms around the first lever-press event of a trial, as well as at times of nose-pokes into the reward port following successful trials. In order to restrict our analysis of neural activity to periods of stereotyped behavior, we selected only rewarded trials that followed previously rewarded trials (to control for starting position), and these trials' inter-press intervals had to be within 20% of the target inter-press interval of 700 ms (to ensure movement stereotypy). To estimate a unit's instantaneous firing rate on each trial, we convolved its event-aligned spike train with a Gaussian kernel (σ = 20 ms). These firing rates were then averaged over the selected trials from a given day to yield the unit PETH. To determine whether a particular PETH had statistically significant modulation in firing rate over trials, we estimated the p-value and Z-score for each bin in the PETH using a bootstrap

approach. P-values were then pooled across days for all lever-press or nose-poke PETHs for a particular unit using Fisher's method. A unit was deemed to have significant modulation in its time-averaged PETH if at least 2 bins had p-values < 1e-5. At this significance threshold the probability of getting a false positive over 1000 neurons and 20 time-bins is less than 0.01.

Similarity across PETHs for the same unit across days, or between different units on the same day was computed using the Pearson's correlation coefficient, similar to the measurement of similarity between ISI distributions and STAs. Similarities in lever-press and nose-poke PETH similarities were averaged for each unit for a particular time-lag (ranging from 1 to 10 days).

Cross-correlograms. We computed cross-correlograms for all unit pairs recorded simultaneously. Slow-wave sleep was associated with the most prominent correlations, and since this state also does not have stimulus or movement induced correlations, we focused our analysis of network stability on SWS epochs. Correlograms were calculated in 1 ms bins and smoothed with a 3 ms boxcar filter. Correlograms for unit pairs recorded on the same tetrode are, by definition, zero at zero lag since only one spike event is allowed per tetrode in a 1 ms bin. For these pairs, the average of the preceding and following bins was used to approximate the value of the correlogram at zero lag. Correlations were considered significant if two consecutive bins of the correlogram were above a threshold set at 3 standard deviation of a shuffled correlogram computed on the same pair (one spike train was shuffled by adding ±400 ms random jitter to each spike). Only correlograms that had at least 2000 spikes could be considered significant. Each significant correlogram was characterized by its maximum peak (positive correlation) or trough (negative correlation), averaged over 3 bins around the peak and normalized by the average bin height of the shuffled correlogram. Thus a peak >1 indicates a positive correlation whereas peak<1 indicates a negative correlations. The lag was defined as the bin location of the peak in the correlogram.

Stability of correlograms. We used the Pearson's correlation coefficient to quantify the similarity between correlogram distributions on different days for the same significant pairs, or for the same insignificant pairs.

Tests of statistical significance. We used the Kruskal-Wallis one-way analysis of variance test to determine whether differences in within-unit similarity measures of firing rate, ISI distribution, behavioral state modulation of firing rate, spike-triggered averages and PETHs, over time-lags of 1 to 10 were significant when compared to each other or to similarity measures computed across units. P-values were corrected for multiple comparisons by the Tukey-Kramer method. We used an alpha value of 0.05.

Experimental infrastructure for long-term neural recordings in behaving animals

Tethering system. We used extruded aluminum with integrated V-shaped rails for the linear-slide (Inventables, Part #25142-03), with matching wheels (with V-shaped grooves) on the carriage (Inventables, Part #25203-02). The bearings in the wheels were degreased and coated with WD-40 to minimize friction. The carriage plate was custom designed (design files available upon request) and fabricated by laser cutting 3mm acrylic. A low-cost 15-channel commutator (SRC-022, Hangzhou-Prosper Ltd, China) was mounted onto the carriage plate. The commutator was suspended with a counterweight using low friction pulleys (Misumi Part # MBFN23-2.1). We used a 14 conductor cable (Mouser, Part # 517-3600B/14100SF) to connect our custom designed head-stage to the commutator. The outer PVC insulation of the cable bundle was removed to reduce the weight of the cable and to make it more flexible. The same cable was used to connect the commutator to a custom designed data acquisition system. This cable needs to be as flexible as possible to minimize forces on the animals head.

Recording hardware. Our lightweight, low-noise and low-cost system for multi-channel extra-cellular electrophysiology uses a 64-channel head-stage (made of 2 RHD2132 ICs from Intantech), weighs less than 4 grams and measures 18 mm x 28 mm in size. The head-stage filters, amplifies, and digitizes extracellular neural signals at 16 bits/sample and 30 kHz per second per channel. An FPGA (Opal Kelly XEM6010) interfaces the head-stage with a computer that stores and displays the acquired data. Since the neural signals are digitized at the head-stage, the system is low-noise and can support long cable lengths. Design files and complete parts list for all components of the system are available on request. The parts for a complete

64-channel electrophysiology extension, including head-stage, FPGA board, commutator, and pulley system, but excluding the recording computer, cost less than $1500.

Tetrode arrays

Tetrodes were fabricated by twisting together short lengths of four 12.5 µm diameter nichrome wires (Redi Ohm 800, Sandvik-Kanthal, Palm Coast, FL), after which they were bound together by melting their polyimide insulation with a heat-gun. An array of 16 such tetrodes was attached to a custom-built microdrive, advanced by a 0-80 threaded screw (~320 µm per full turn). The wires were electroplated in a gold (5355, SIFCO, Independence, OH) and 0.1% carbon nanotube (Cheap Tubes dispersed MWNTs, 95wt% <8nm, Cambridgeport, VT) solution with 0.1% polyvinylpyrrolidone surfactant (PVP-40, Sigma-Aldrich, St. Louis, MO) to lower electrode impedances to ~100-150 kOhm[64,65]. The prepared electrode array was then implanted into the motor cortex or striatum.

Data storage and computing infrastructure

Hardware setup. Our recordings (64-channels at 30 kHz per channel) generate 1 terabyte (TB) of raw data every 2 days. To cope with these demands, we developed a low-cost and reliable high I/O bandwidth storage solution with a custom lightweight fileserver. Each storage server consists of 24 4TB spinning SATA hard disks connected in parallel to a dual socket Intel server class motherboard via the high bandwidth PCI-E interface. The ZFS file-system (bundled with the open source SmartOS operating system) is used to manage the data in a redundant configuration that allows any two disks in the 24 disk array to simultaneously fail without data loss. Due to the redundancy, each server has 60TB of usable space that can be read at approximately 16 gigabits per second (Gbps). This high I/O bandwidth is critical for data backup, recovery and integrity verification.

Distributed computing software setup. To fully utilize available CPU and I/O resources, we parallelized the data processing[66]. Thread-level parallelization inside a single process is the simplest approach and coordination between threads is orchestrated using memory shared between the threads. However, this approach only works for a single machine and does not

scale to a cluster of computers. The typical approach to cluster-level parallelization is to coordinate the multiple parallel computations running both within a machine and across machines by exchanging messages between the concurrently running processes. The 'map-reduce' abstraction conceptualizes a computation as having two phases: a 'map' phase which first processes small subsets of the data in parallel and a 'reduce' phase which then serially aggregates the results of the 'map' phase. Since much of our data analysis is I/O limited (like computing statistics of spike waveform amplitudes), we developed a custom distributed computing infrastructure for 'map-reduce' like computations for the .NET platform. The major novelty in our framework is that rather than moving the output of the map computation over the network to a central node for performing the reduce computations, it instead moves the reduce computation to the machines containing the output of the map computations in the correct serial order. If the output of the map computation is voluminous compared to the output of each step of the reduce computation then our approach consumes significantly less time and I/O bandwidth. We have used this framework both in a virtual cluster of 10 virtual machines running on the afore-mentioned SmartOS-based storage servers and in Microsoft's commercial cloud computing platform, Windows Azure.

Automated spike sorting

We developed a new and automated spike sorting algorithm that is able to efficiently track single units over long periods of time. Our algorithm first identifies the spikes from the raw recordings then performs two fully automated processing steps.

Spike identification

We first extract spike snippets, i.e. the extracellular voltage traces associated with action potentials, automatically from the raw data. A schematic of this process is shown in Supplementary Figure 1. First, the signal from each channel $s_{ch}(t)$ is partitioned into 15 second blocks with an additional 100 ms tacked onto each end of the block to account for edge effects in filtering. Then, for each block of each channel, the raw data is filtered with a 4^{th} order elliptical band-pass filter (cut-off frequencies 300 Hz and 7500 Hz) first in the forward then the reverse direction to preserve accurate spike times and spike waveforms. For each sample in

each block, the median across all channels is subtracted from every channel to eliminate common mode noise. Finally, a threshold crossing algorithm is used to detect spikes independently for each tetrode (Supplementary Figure 1). In our recordings, we use a threshold of 50μV, which corresponds to about 7 times the median absolute deviation of the signal. After detecting a threshold crossing, we find the sample that corresponds to the local maximum of the event, defined as the maximum of the absolute value across the four channels of a tetrode. A 2 ms (64 sample) snippet of the signal centered on the local maximum is extracted from all channels. Thus, each putative spike in a tetrode recording is characterized by the time of the local maximum and a 256 dimensional vector (64 samples x 4 channels).

Each 15 second block of each tetrode can be processed in parallel. However, since the number of spikes in any given 15 second block is not known in advance, the extracted spike snippets must be serially written to disk. Efficiently utilizing all the cores of the CPUs and simultaneously queuing disk read/write operations asynchronously is essential for keeping this step faster than real-time. In our storage server, the filtering/spike detection step runs 15 times faster than real-time. To extract LFPs, we down-sample the raw data 100-fold (from 30 kHz to 300 Hz) by two applications of a 4^{th} order 5-fold decimating Chebychev filter followed by a single application of a 4^{th} order 4-fold decimating Chebychev filter. After extracting the spike snippets and the LFPs, the raw data is deleted, resulting in a 5-10x reduction in storage space requirements.

A typical week-long recording from 16 tetrodes results in over a billion putative spikes. While most putative spikes are low amplitude and inseparable from noise, the spike detection threshold cannot be substantially increased without losing many cleanly isolatable single units. Assigning many billion putative spikes to clusters corresponding to single units in a largely automated fashion is critical for realizing the full potential of continuous 24x7 neural recordings.

Automatic Processing Step 1. Local clustering and de-noising

This step of the algorithm converts the sequence of all spike waveforms $\{\boldsymbol{x}_i\}_{i=1}^{N}$ from a tetrode to a sequence of averages of spike waveforms $\{\boldsymbol{y}_i\}_{i=1}^{M}$ with each averaging done over a set of approximately 100 raw spike waveforms ($100M \cong N$) with very similar shapes that are highly

likely to be from the same unit. The output of this step of the algorithm is a partitioning of N spike waveforms into M groups. $\boldsymbol{y}_i = \frac{1}{N_i}\sum_{j=1}^{N_i} \boldsymbol{x}_{i_j}$, $\sum_{i=1}^{M} N_i = N$. Local clustering followed by averaging produces de-noised estimates of the spike waveform for each unit during each point in time. The goal in this step is not to reliably find all spike waveforms associated with a single unit, but to be reasonably certain that the waveforms being averaged over are similar enough to be from the same single unit. This results in a dataset of averaged spike waveforms that is about a hundred times smaller than the original dataset greatly aiding in speedily running the remaining half of the spike sorting algorithm and in visualizing the dataset.

Super-paramagnetic clustering. Clustering is inherently a scale-dependent problem, i.e. the 'right' number of clusters in a given dataset depends on the scale being considered. At the coarsest scale, all points can be considered members of a single cluster and at a very fine scale each point belongs to its own cluster. A formal, mathematically precise way of characterizing this tradeoff is to think of clustering as lossy compression[67]. The amount of loss is defined as the amount of information lost by replacing each point in a cluster with their 'average' and the compression comes from characterizing the entire dataset with just the cluster averages. A simple loss measure is the sum of squared distances between each point in a cluster and the cluster centroid, i.e. the within cluster variance. If each point is assigned its own unique cluster then the loss would be zero. Conversely, if all points were assigned the same cluster then the loss would simply be the variance of the entire set of points. For intermediate amount of loss between these two extremes, the fewest number of clusters, i.e. the largest amount of compression, with at most that much loss, can, in principle, be computed. Conversely for a given number of clusters, one can compute the clustering that results in the smallest amount of loss.

We use the super-paramagnetic clustering (SPC) algorithm[20] in our spike sorting pipeline partly because it parameterizes the loss-compression tradeoff discussed above with a 'temperature' parameter. At low temperatures, the algorithm assigns all points to a single cluster. As the temperature parameter is increased new clusters appear until, at very high temperatures, each point is assigned its own cluster. Units with large spike waveforms or very distinctive spike shapes appear at relatively low temperatures. However, other units often

appear at relatively high temperatures and clusters at higher temperatures often do not include spikes in the periphery of the cluster. In existing uses of this algorithm for spike sorting the 'right' temperature for each cluster is selected manually[21]. Often several clusters at a higher temperature need to be manually merged as they all correspond to the same single unit.

The SPC algorithm also requires a distance measure between pairs of points. In previous approaches to spike sorting, a small number of features (on the order of 10) are extracted from the full 256 dimensional dataset and the Euclidean distance between points in this new feature space is used as the distance measure for clustering[21]. In our experience the number of coefficients that are necessary to adequately capture the distinction between similar, but well isolated units varies substantially depending on the number of units being recorded on a tetrode and the signal-to-noise ratio of the recording. We find that simply using the Euclidean distance in the raw 256 dimensional space avoids this problem without being computationally prohibitive.

Iterative multi-scale clustering. Two considerations determine the size of the batch for local clustering. First, SPC requires computing distances between every pair of points, making the algorithm quadratic in the number of points being clustered in one batch. In a typical desktop-class PC, a window size of 10,000 spikes runs 8 times slower than real-time, whereas 1,000 spike batches run faster than real-time. Second, changes in the spike waveform of a unit over time means that the space occupied by points belonging to a single cluster increases with the size of the temporal window (i.e. batch size), which in turn decreases the separation between clusters. Both of these considerations favor clustering relatively fewer spikes in a batch, and our algorithm does it in batches of 1000 spikes.

However, different neuron types can have very different firing rates (Figure 3). Therefore, a 1000 spike window might contain just a few or even no spikes from low firing rate units. To overcome this problem, we developed a multi-resolution approach for the local clustering step. It identifies clusters corresponding to units with high firing rates, then removes them from the dataset and then re-clusters the remaining spikes, repeating this process iteratively (Supplementary Figure 2).

Detailed algorithm for local clustering and de-noising. The detailed steps of the algorithm for multi-resolution local clustering are described below and a schematic of the whole process is presented in Supplementary Figure 2.

1. The set of all spike waveforms is partitioned into blocks of 1000 consecutive points. Therefore, the first block contains points $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{1000}\}$, the second block contains $\{\boldsymbol{x}_{1001}, \ldots, \boldsymbol{x}_{2000}\}$ and so on.
2. An SPC cluster tree is generated for each block independently. This is computed by first clustering the set of 1000 points at a range of temperatures $T_i = 0.01i, 0 \leq i \leq 15$. This process assigns a cluster label to each point at each temperature. This matrix of cluster labels is then converted to a tree where each node in the tree corresponds to a cluster at some temperature, i.e. a subset of the 1000 points. The root node of the tree (depth 0) corresponds to a single cluster containing all 1000 points. The children of the root node (depth 1 nodes) correspond to a partition of the set of 1000 points based on the cluster labels at temperature 0.01. For each node in depth 1, the children of that node (depth 2 nodes) correspond to a partition of the points associated with that node based on the cluster labels of those points at temperature 0.02. This is repeated for all temperatures to construct the full tree with depth equal to the number of temperature increments.
3. Each cluster tree is collapsed into a partition (a clustering) of the set of 1000 points (Supplementary Figure 2B). The simplest technique for getting a partition from an SPC cluster tree is to use all the nodes at a fixed depth, i.e. clustering at a fixed temperature. In practice, this approach suffers from major drawbacks. The lowest temperature at which a cluster first separates from its neighbors varies from unit-to-unit and depends on the signal-to-noise ratio of the spike waveform, how distinct the spike waveform of that unit is, etc. Also, when units appear at relatively high temperatures, the clusters corresponding to single units at those temperatures don't include many spikes at the periphery of those clusters. Therefore, instead of using a single temperature we recursively merge leaves of the cluster tree based on the loss-compression tradeoff discussed above to generate a partition. This is done by recursively collapsing the cluster tree one level at a time. Specifically,

a. For each leaf node in the cluster tree, the similarity between the leaf node and its parent is first calculated. Let $L = \{i_1, \ldots, i_N\}$ be the leaf node which is specified by the indices of the subset of the 1000 points belonging to that node. Similarly, let $P = \{j_1, \ldots, j_M\}$ be the parent of the leaf node. Note that $L \subseteq P$. Let $\boldsymbol{l} = \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{x}_{i_k}$ be the average spike waveform of the leaf node and $\boldsymbol{p}$ be the average spike waveform of its parent. Let $d_L = \sum_{k=1}^{N} ||\boldsymbol{x}_{\boldsymbol{i_k}} - \boldsymbol{l}||$ be the total distance of points in the leaf node from their average and $d_P = \sum_{k=1}^{N} ||\boldsymbol{x}_{\boldsymbol{i_k}} - \boldsymbol{p}||$ be the distance from the parent node. The difference $d_P - d_L = a_L$ measures how well the parent node approximates the leaf node, i.e. how much additional loss is incurred in approximating the points in the leaf node with the cluster corresponding to the parent node.

b. Let $L = \{L_i\}$ be the set of all N leaf nodes sharing the same parent node P. The set of leaf nodes that are poorly approximated by their parent (the well-isolated nodes I) are considered distinct clusters. $I \subseteq L$, where $L_i \in I$ if $a_{L_i} > a$. This encodes the intuition that if a cluster at a given temperature splits into multiple sub-clusters at the next higher temperature that are each quite similar to the parent cluster, then treating each of these sub-clusters as distinct clusters is inappropriate. The parameter a provides an intuitive tradeoff between missing distinct clusters that appear at high temperatures and classifying spikes in the periphery of a single cluster into multiple distinct clusters. Let M be the number of elements in I. If any of the remaining $N - M$ nodes are well approximated by one of the well-isolated nodes then they are merged together. For $L_i \in L/I$, if $\min_{L_j \in I} d_{L_j} - d_{L_i} < a$, i.e. if node L_j approximates node L_i well then they are merged. Merging a set of nodes corresponds to creating a node containing all the points from each of the nodes being merged. This yields a set of augmented well-isolated nodes. Any remaining nodes, i.e. non-well-isolated nodes that are also not well-approximated by any of the well isolated nodes, are merged with each other. Therefore, this step results in converting the set of N leaf nodes sharing a parent into a set of Mor $M + 1$ nodes formed by merging some of them together.

c. A depth D tree is converted into a depth $D-1$ tree by replacing all the leaf nodes and their parents with the merged nodes derived in the previous step.

d. Step a – c are repeated recursively until the tree is of depth 1. The leaf nodes of this tree which are typically vastly fewer in number than the total number of leaf nodes of the original tree correspond to a partition of the set of 1000 points of each block.

4. The centroid of each cluster from the previous step containing at least 15 points contributes one element to the output of the local clustering step, the set of averaged spike waveforms $\{\boldsymbol{y}_i\}$ (Supplementary Figure 2C). In our datasets, clusters with at least 15 spikes in a 1000 spike window correspond to firing rates of approximately 1 Hz or greater, on average. The points belonging to the remaining clusters, i.e. ones with fewer than 15 points, are all pooled together, ordered by their spike time and become the new $\{\boldsymbol{x}_i\}$. The number of spikes in this new subset is approximately 10% of the original. Steps 1-4 are used to locally cluster this new subset of spikes and produce a second set of averaged spike waveforms $\{\boldsymbol{y}_i\}$. This process of re-clustering the low-density clusters is repeated two more times. The averaged spike waveforms from all four scales are then grouped together to form the full set $\{\boldsymbol{y}_i\}_{i=1}^{M}$ and ordered by the median spike time of set of spikes that were averaged to generate each $\boldsymbol{y}_i$. This process results in an assignment of over 98% of the original set of Nspikes to a cluster with at least 15 spikes in one of the 4 scales. The firing rate of units in clusters with at least 15 spikes at the fourth scale is about 0.01Hz in the sample dataset.

Automatic Processing Step 2. Sorting and tracking de-noised spike waveforms

This step takes the sequence of averaged spike waveforms $\{\boldsymbol{y}_i\}_{i=1}^{M}$ computed in the previous step and identifies the subsets of $\{\boldsymbol{y}_i\}$ that correspond to the same putative single unit. A subset of $\{\boldsymbol{y}_i\}$ is considered to be the same single unit if distances between $\boldsymbol{y}_i$ and $\boldsymbol{y}_{i+1}$ are sufficiently small for the entire subset. As in the previous step, the set $\{\boldsymbol{y}_i\}_{i=1}^{M}$ is first partitioned into blocks of 1000 consecutive points and an SPC cluster tree is independently computed for each block, this time for the temperature range $T_i = 0.01i, 0 \leq i \leq 10$. Then, we use a binary

linear programming algorithm inspired by a computer vision problem called segmentation fusion[19] to identify the nodes in each cluster tree that correspond to the same single unit.

Binary linear programming allows units to be tracked over time. Tracking multiple single units over time from a sequence of cluster trees requires first selecting a subset of the nodes of each cluster tree that correspond to distinct units, followed by matching nodes from adjacent cluster trees that correspond to the same unit. Doing these steps manually is infeasible because of the large volumes of data. In our datasets, the local-clustering step results in ~100 million averaged spikes from the original set of ~10 billion spikes per rat. This produces a set of ~100,000 cluster trees making manual tracking impossibly labor intensive. We instead adapted the segmentation fusion algorithm, invented to solve the problem of reconstructing the full 3D structure of axons, dendrites and soma present in a volume of neuropil from a stack of 2D electron microscopy sections[19,68]. A cluster tree is analogous to a multi-resolution segmentation of the 2D image. Identifying nodes across cluster trees that correspond to the same single unit is thus analogous to identifying the same segment across 2D sections as a neurite courses through the neuropil.

The algorithm finds a set of sequences of nodes from the sequence of cluster trees, where each sequence of nodes corresponds to a well isolated single unit. This is done by first enumerating all the nodes in all the cluster trees and all possible links between nodes in adjacent cluster trees. Then a constrained optimization problem is solved to find a subset of nodes and links that maximize a score that depends on the similarity between nodes represented by a link and the 'quality' of a node. This maximization is constrained to disallow assigning the same node to multiple single units and to ensure that if a link is selected in the final solution then so are the nodes on each side of the link.

Details of the binary linear programming algorithm. Each step of the binary linear programming algorithm is detailed below (see also Supplementary Figure 3).

1. The sequence of cluster trees is grouped into blocks of 10 consecutive trees with an overlap of 5 cluster trees. Solving the binary linear program for blocks larger than 10 trees is computationally prohibitive.

2. The segmentation fusion algorithm is run independently for each block of 10 cluster trees (Supplementary Figure 3). Let $\left\{\left\{C_j^i\right\}_{j=1}^{N_i}\right\}_{i=1}^{10}$ be the set of binary indicator variables representing all nodes in all 10 cluster trees where the cluster tree indexed by i contains a total of N_i nodes. The total number of nodes is $N = \sum_{i=1}^{10} N_i$. Let $\left\{\left\{L_{jk}^i\right\}_{j,k=1,1}^{N_i,N_{i+1},}\right\}_{i=1}^{9}$ be the variables representing the set of all links between adjacent cluster trees. Link L_{jk}^i connects clusters C_j^i and C_k^{i+1}. The total number of links is $M = \sum_{i=1}^{9} N_i N_{i+1}$. Solving the linear program requires choosing a $\{0,1\}$ value for each of the $N + M$ binary variables that maximizes the objective function $\sum_{ij} C_j^i\, \theta_j^i + \sum_{ijk} L_{jk}^i \left(\theta_{jk}^i - 0.02\right)$. The objective function is a weighted linear sum of all the binary variables where the cluster weights θ_j^i represent the 'quality' of the cluster and the link weights θ_{jk}^i represent the similarity of the clusters joined by the link. The link weights are numbers in the range $(0,1)$. The threshold of 0.02 serves to give negative weight to links between sufficiently dissimilar clusters, effectively constraining the value of the variables representing those links to 0. This objective function is to be optimized subject to three sets of constraints. The first, $\sum_k L_{jk}^i \leq C_j^i$, enforces the constraint that if the node variable C_j^i is assigned a value of 1 then out of all the outgoing links from the node $\left\{L_{jk}^i\right\}_{k=1}^{N_{i+1}}$, at most one is chosen (Supplementary Figure 3C). Similarly, the second set of constraints , $\sum_j L_{jk}^i \leq C_k^{i+1}$, enforces the requirement that at most one incoming link to a node is chosen (Supplementary Figure 3B). The third set of constraints enforces the requirement that for each of the 1000 points in a cluster tree at most one of the nodes containing that point is chosen (Supplementary Figure 3D). This translates to inequalities $\sum_{k \in I_j} C_k^i \leq 1$ where the set of indices I_j represents nodes in the path from the root of cluster tree i to the j^{th} leaf node of the cluster tree. Therefore, the total number of constraints of this type for each cluster tree is the number of leaf nodes in that cluster tree. The link weight θ_{jk}^i is the Euclidean distance between the average spike waveform of clusters C_j^i and C_k^{i+1} non-linearly scaled by a sigmoid function to fall in the range $(0,1)$. If d is the distance then $\theta = \frac{a}{1+a}, \mathrm{a} = \exp\left(\frac{-(d-k)}{s}\right), \mathrm{s} = 0.005, \mathrm{k} = 0.03$. The parameter s

controls the steepness of the sigmoid and the parameter k sets the distance d at which $\theta = 0.5$. The cluster weight θ_j^i gives preference to clean well-isolated clusters, i.e. clusters that appear at low temperatures and retain most of their points across a large temperature range. Let $N^{(0)}$ be the number of points in the cluster corresponding to C_j^i. Let C_k^i be the largest cluster amongst the child nodes of C_j^i and let $N^{(1)}$ be the number of points in C_k^i. Similarly let $N^{(2)}$ be the number of points in the largest cluster among the child nodes of C_k^i. Given the sequence of cluster sizes $N^{(0)}, N^{(1)}, \ldots, N^{(a)}$ where $N^{(a)}$ is the number of points in a leaf node of cluster tree, θ_j^i is defined as $N^{(0)} / \left(N^{(0)} + \cdots + N^{(a)}\right)$. This measure of cluster quality penalizes clusters that split into smaller clusters at higher temperatures and clusters that only appear at high temperatures.

3. The results of the previous step, i.e. the subset of the M links of each block that maximizes the objective function, are finally combined to produce a sorting that tracks single units over long time periods despite gradually changing waveforms. Links that are part of two instances of the segmentation fusion procedure due to the overlap mentioned in step 1 are only included in the final solution if both linear programs include them. The set of links chosen by the segmentation fusion algorithm are chained together to get long chains of clusters. For instance if links L_{jk}^i, L_{kl}^{i+1}, L_{lm}^{i+2} are assigned values of 1 in the solution to the segmentation fusion linear program then all points in clusters $C_j^i, C_k^{i+1}, C_{jl}^{i+2}, C_m^{i+3}$ belong to the same chain and hence the same single unit. Each point that does not belong to any chain is assigned to the chain containing points most similar to it (as measured using the sigmoidal distance of step 2) as long as the similarity $\theta > 0.02$ (again the same threshold as used in step 2).

Merging the chain of nodes and links identified by the binary linear programming algorithm. Often, spike waveforms have multiple equal amplitude local extrema. Since the waveforms are aligned to the local extrema with the largest amplitude during the spike identification phase, different waveforms from the same unit can be aligned to different features of the waveform. This results in multiple chains for the same single unit since the Euclidean distance between waveforms aligned to different features is very large. This is remedied by merging chains that contain spikes from the same recording interval if the translation-invariant distance between

the spike waveforms of the chains is sufficiently low in the overlap region. The translation invariant distance is computed by first calculating the distance between a pair of spike waveforms for a range of relative shifts between the pair and then taking the minimum of this set of distances. Overlapping chains with the smallest translation-invariant distance are first merged. This is done recursively until either no overlap between chains remains or overlapping chains have distinct spike waveforms and hence correspond to different simultaneously recorded single units.

Visualization and manual verification

The final output of the unsupervised spike sorting algorithm consisted of long chains (median length = 8.5 hours) corresponding to single-unit spikes linked over time. However, our algorithm did not link spike chains over discontinuities in recording time (i.e. across recording files), or in the case of rapid changes in spike shape that occasionally occurred during the recording, or when spike amplitudes drifted under 75 μV for brief periods. In such cases, we had to link, or 'merge' chains across these discontinuities.

In order to visualize, merge and manually inspect the unsupervised chains, we developed a MATLAB program with a graphical user interface (GUI). This allows users to semi-automatically merge chains belonging to the same unit across discontinuities based on end-to-beginning similarity in their spike waveforms and inter-spike interval distributions. We perform these merging events only if the time-gap between the end of one chain and the start of the next is less than 5 hours of recording time, or up-to 24 hours in the case of gaps in recording time (which in the two experiments we report on were, on average, 2 hours). In occasional cases of inaccurate spike sorting by the unsupervised algorithm, either when spike amplitude was too close to noise or when chains' spike waveforms were too similar, we manually split the clusters using MClust (MClust 4.3, A. D. Redish et al).

Acknowledgements. This work was supported by a McKnight Scholars Award (BPÖ), HSFP and EMBO fellowships (SBEW), and by the Swartz Foundation (EK). A.K.D. is an Ellison Medical Foundation fellow of the Life Sciences Research Foundation. We thank Michelle Choi for help with behavioral scoring from videos of rats.

Author Contributions. BPÖ, RP and AKD designed the study with input from all authors. RP designed and implemented the algorithm for automated spike sorting with the help of AKD and VN. AKD performed the experiments. AKD analyzed the neural data with help from BPÖ and SBEW. EK wrote the software for automated behavioral analysis and analyzed the behavior together with AKD.

References

1. Lütcke, H., Margolis, D. J. & Helmchen, F. Steady or changing? Long-term monitoring of neuronal population activity. *Trends in Neurosciences* **36,** 375–384 (2013).
2. Huber, D. *et al.* Multiple dynamic representations in the motor cortex during sensorimotor learning. *Nature* **484,** 473–478 (2012).
3. Peters, A. J., Chen, S. X. & Komiyama, T. Emergence of reproducible spatiotemporal activity during motor learning. *Nature* **510,** 263–267 (2014).
4. Ziv, Y. *et al.* Long-term dynamics of CA1 hippocampal place codes. *Nat Neurosci* **16,** 264–266 (2013).
5. Grienberger, C. & Konnerth, A. Imaging Calcium in Neurons. *Neuron* **73,** 862–885 (2012).
6. Looger, L. L. & Griesbeck, O. Genetically encoded neural activity indicators. *Current Opinion in Neurobiology* **22,** 18–23 (2012).
7. Dombeck, D. A., Khabbaz, A. N., Collman, F., Adelman, T. L. & Tank, D. W. Imaging Large-Scale Neural Activity with Cellular Resolution in Awake, Mobile Mice. *Neuron* **56,** 43–57 (2007).
8. Vogelstein, J. T. *et al.* Spike Inference from Calcium Imaging Using Sequential Monte Carlo Methods. *Biophysical Journal* **97,** 636–655 (2009).

9. Vogelstein, J. T. *et al.* Fast Nonnegative Deconvolution for Spike Train Inference From Population Calcium Imaging. *Journal of Neurophysiology* **104,** 3691 –3704 (2010).
10. Yaksi, E. & Friedrich, R. W. Reconstruction of firing rate changes across neuronal populations by temporally deconvolved Ca2+ imaging. *Nat Meth* **3,** 377–383 (2006).
11. Buzsáki, G. Large-scale recording of neuronal ensembles. *Nature Neuroscience* **7,** 446–451 (2004).
12. Dickey, A. S., Suminski, A., Amit, Y. & Hatsopoulos, N. G. Single-Unit Stability Using Chronically Implanted Multielectrode Arrays. *Journal of Neurophysiology* **102,** 1331–1339 (2009).
13. Emondi, A. A., Rebrik, S. P., Kurgansky, A. V. & Miller, K. D. Tracking neurons recorded from tetrodes across time. *Journal of Neuroscience Methods* **135,** 95–105 (2004).
14. Fraser, G. W. & Schwartz, A. B. Recording from the same neurons chronically in motor cortex. *Journal of Neurophysiology* **107,** 1970–1978 (2012).
15. Poddar, R., Kawai, R. & Ölveczky, B. P. A Fully Automated High-Throughput Training System for Rodents. *PLoS ONE* **8,** e83171 (2013).
16. Rey, H. G., Pedreira, C. & Quian Quiroga, R. Past, present and future of spike sorting techniques. *Brain Research Bulletin* **119,** 106–117 (2015).
17. Hromádka, T., DeWeese, M. R. & Zador, A. M. Sparse Representation of Sounds in the Unanesthetized Auditory Cortex. *PLoS Biol* **6,** e16 (2008).
18. Mizuseki, K. & Buzsáki, G. Preconfigured, Skewed Distribution of Firing Rates in the Hippocampus and Entorhinal Cortex. *Cell Reports* **4,** 1010–1021 (2013).
19. Vazquez-Reina, A. *et al.* Segmentation fusion for connectomics. in *2011 IEEE International Conference on Computer Vision (ICCV)* 177–184 (2011). doi:10.1109/ICCV.2011.6126240
20. Blatt, M., Wiseman, S. & Domany, E. Superparamagnetic Clustering of Data. *Phys. Rev. Lett.* **76,** 3251–3254 (1996).

21. Quiroga, R. Q., Nadasdy, Z. & Ben-Shaul, Y. Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering. *Neural Computation* **16,** 1661–1687 (2004).
22. Kasthuri, N. *et al.* Saturated Reconstruction of a Volume of Neocortex. *Cell* **162,** 648–661 (2015).
23. Harris, K. D., Henze, D. A., Csicsvari, J., Hirase, H. & Buzsáki, G. Accuracy of Tetrode Spike Separation as Determined by Simultaneous Intracellular and Extracellular Measurements. *Journal of Neurophysiology* **84,** 401–414 (2000).
24. Schmitzer-Torbert, N., Jackson, J., Henze, D., Harris, K. & Redish, A. D. Quantitative measures of cluster quality for use in extracellular recordings. *Neuroscience* **131,** 1–11 (2005).
25. Barthó, P. *et al.* Characterization of Neocortical Principal Cells and Interneurons by Network Interactions and Extracellular Features. *Journal of Neurophysiology* **92,** 600–608 (2004).
26. Berke, J. D., Okatan, M., Skurski, J. & Eichenbaum, H. B. Oscillatory Entrainment of Striatal Neurons in Freely Moving Rats. *Neuron* **43,** 883–896 (2004).
27. Connors, B. W. & Gutnick, M. J. Intrinsic firing patterns of diverse neocortical neurons. *Trends in Neurosciences* **13,** 99–104 (1990).
28. Hengen, K. B., Lambo, M. E., Van Hooser, S. D., Katz, D. B. & Turrigiano, G. G. Firing Rate Homeostasis in Visual Cortex of Freely Behaving Rodents. *Neuron* **80,** 335–342 (2013).
29. Marder, E. & Goaillard, J.-M. Variability, compensation and homeostasis in neuron and network function. *Nat Rev Neurosci* **7,** 563–574 (2006).
30. Singer, W. Neuronal Synchrony: A Versatile Code for the Definition of Relations? *Neuron* **24,** 49–65 (1999).
31. Grutzendler, J., Kasthuri, N. & Gan, W.-B. Long-term dendritic spine stability in the adult cortex. *Nature* **420,** 812–816 (2002).
32. Yang, G., Pan, F. & Gan, W.-B. Stably maintained dendritic spines are associated with lifelong memories. *Nature* **462,** 920–924 (2009).

33. Gervasoni, D. *et al.* Global Forebrain Dynamics Predict Rat Behavioral States and Their Transitions. *J. Neurosci.* **24,** 11137–11147 (2004).

34. Venkatraman, S., Jin, X., Costa, R. M. & Carmena, J. M. Investigating Neural Correlates of Behavior in Freely Behaving Rodents Using Inertial Sensors. *Journal of Neurophysiology* **104,** 569–575 (2010).

35. Carmena, J. M., Lebedev, M. A., Henriquez, C. S. & Nicolelis, M. A. L. Stable Ensemble Performance with Single-Neuron Variability during Reaching Movements in Primates. *J. Neurosci.* **25,** 10712–10716 (2005).

36. Chestek, C. A. *et al.* Single-Neuron Stability during Repeated Reaching in Macaque Premotor Cortex. *J. Neurosci.* **27,** 10742–10750 (2007).

37. Ganguly, K. & Carmena, J. M. Emergence of a Stable Cortical Map for Neuroprosthetic Control. *PLoS Biol* **7,** e1000153 (2009).

38. Rokni, U., Richardson, A. G., Bizzi, E. & Seung, H. S. Motor Learning with Unstable Neural Representations. *Neuron* **54,** 653–666 (2007).

39. Stevenson, I. H. *et al.* Statistical assessment of the stability of neural movement representations. *Journal of Neurophysiology* **106,** 764–774 (2011).

40. Marmarelis, P. Z. & Naka, K.-I. White-Noise Analysis of a Neuron Chain: An Application of the Wiener Theory. *Science* **175,** 1276–1278 (1972).

41. Meister, M., Pine, J. & Baylor, D. A. Multi-neuronal signals from the retina: acquisition and analysis. *Journal of Neuroscience Methods* **51,** 95–106 (1994).

42. Kawai, R. *et al.* Motor Cortex Is Required for Learning but Not for Executing a Motor Skill. *Neuron* **86,** 800–812 (2015).

43. Hires, S. A., Tian, L. & Looger, L. L. Reporting neural activity with genetically encoded calcium indicators. *Brain Cell Biol* **36,** 69–86 (2008).

44. Tsien, R. Y. New calcium indicators and buffers with high selectivity against magnesium and protons: design, synthesis, and properties of prototype structures. *Biochemistry* **19,** 2396–2404 (1980).

45. Zucker, R. S. Calcium- and activity-dependent synaptic plasticity. *Current Opinion in Neurobiology* **9,** 305–313 (1999).

46. Greenberg, P. A. & Wilson, F. A. W. Functional Stability of Dorsolateral Prefrontal Neurons. *Journal of Neurophysiology* **92,** 1042–1055 (2004).

47. Attardo, A., Fitzgerald, J. E. & Schnitzer, M. J. Impermanence of dendritic spines in live adult CA1 hippocampus. *Nature* **523,** 592–596 (2015).

48. Ehlers, M. D. Activity level controls postsynaptic composition and signaling via the ubiquitin-proteasome system. *Nat Neurosci* **6,** 231–242 (2003).

49. Holtmaat, A. J. G. D. *et al.* Transient and Persistent Dendritic Spines in the Neocortex In Vivo. *Neuron* **45,** 279–291 (2005).

50. Malinow, R. & Malenka, R. C. Ampa Receptor Trafficking and Synaptic Plasticity. *Annual Review of Neuroscience* **25,** 103–126 (2002).

51. Staub, O. *et al.* Regulation of stability and function of the epithelial Na+ channel (ENaC) by ubiquitination. *EMBO J* **16,** 6325–6336 (1997).

52. Xu, T. *et al.* Rapid formation and selective stabilization of synapses for enduring motor memories. *Nature* **462,** 915–919 (2009).

53. Keck, T. *et al.* Synaptic Scaling and Homeostatic Plasticity in the Mouse Visual Cortex In Vivo. *Neuron* **80,** 327–334 (2013).

54. Otchy, T. M. *et al.* Acute off-target effects of neural circuit manipulations. *Nature (in press)*

55. Turrigiano, G. G. & Nelson, S. B. Homeostatic plasticity in the developing nervous system. *Nat Rev Neurosci* **5,** 97–107 (2004).

56. Buzsáki, G. Neural Syntax: Cell Assemblies, Synapsembles, and Readers. *Neuron* **68,** 362–385 (2010).

57. Singer, W. *et al.* Neuronal assemblies: necessity, signature and detectability. *Trends in Cognitive Sciences* **1,** 252–261 (1997).

58. Buzsáki, G. & Draguhn, A. Neuronal Oscillations in Cortical Networks. *Science* **304,** 1926–1929 (2004).

59. Alivisatos, A. P. *et al.* Nanotools for Neuroscience and Brain Activity Mapping. *ACS Nano* **7,** 1850–1866 (2013).

60. Kralik, J. D. *et al.* Techniques for Chronic, Multisite Neuronal Ensemble Recordings in Behaving Animals. *Methods* **25,** 121–150 (2001).

61. Eschenko, O., Mölle, M., Born, J. & Sara, S. J. Elevated Sleep Spindle Density after Learning or after Retrieval in Rats. *J. Neurosci.* **26,** 12914–12920 (2006).

62. Nicolelis, M. A., Baccala, L. A., Lin, R. C. & Chapin, J. K. Sensorimotor encoding by synchronous neural ensemble activity at multiple levels of the somatosensory system. *Science* **268,** 1353–1358 (1995).

63. Shaw, F.-Z. Is Spontaneous High-Voltage Rhythmic Spike Discharge in Long Evans Rats an Absence-Like Seizure Activity? *Journal of Neurophysiology* **91,** 63–77 (2003).

64. Keefer, E. W., Botterman, B. R., Romero, M. I., Rossi, A. F. & Gross, G. W. Carbon nanotube coating improves neuronal recordings. *Nature Nanotechnology* **3,** 434–439 (2008).

65. Ferguson, J. E., Boldt, C. & Redish, A. D. Creating low-impedance tetrodes by electroplating with additives. *Sensors and Actuators A: Physical* **156,** 388–393 (2009).

66. Dean, J. & Ghemawat, S. MapReduce: Simplified Data Processing on Large Clusters. *Commun. ACM* **51,** 107–113 (2008).

67. Slonim, N., Atwal, G. S., Tkačik, G. & Bialek, W. Information-based clustering. *Proceedings of the National Academy of Sciences of the United States of America* **102,** 18297–18302 (2005).

68. Kaynig, V. *et al.* Large-scale automatic reconstruction of neuronal processes from electron microscopy images. *Med Image Anal* **22,** 77–88 (2015).

Figure Legends

Figure 1: Experimental infrastructure and analysis pipeline for long-term continuous neural and behavioral recordings in behaving rodents.

- **A.** We adapted our automated rodent training system for long-term electrophysiology. Rats engage in natural behaviors and prescribed motor tasks in their home-cages, while neural data is continuously acquired from implanted tetrode arrays. The tethering cable connects the head-stage to a commutator mounted on a carriage that moves along a low-friction linear slide. The commutator-carriage is counterweighted to eliminate slack in the tethering cable. Behavior is continuously monitored and recorded using a camera and a 3-axis accelerometer.
- **B.** Example of a recording segment showing high-resolution behavioral and neural data simultaneously acquired from a head-mounted 3-axis accelerometer and a tetrode implanted in the motor cortex, respectively.
- **C.** Overview of our novel spike sorting algorithm to identify single units in long-term continuous extracellular recordings (see Methods). (Top, Left) Spike amplitudes on two electrodes of an example tetrode for a 1 hour-long recording segment. Clusters of spikes with similar waveforms are identified and tracked over time. An iterative local-clustering step (Step 1) compresses and de-noises the dataset. De-noised spike clusters (Top, right) are linked across time by a segmentation fusion algorithm (Step 2) to yield cluster-chains corresponding to single units (middle). In the final step, we visually inspect the output of the automated sorting and merge similar chains across time (Step 3) to yield the final single-unit clusters (bottom). Insets (bottom) show average spike waveforms of six example units 44 hours apart.

Figure 2: Single units isolated from long-term recordings from dorsolateral striatum (DLS) and motor cortex (MC).

- **A.** Temporal profile of units recorded in DLS (, left) and MC (right) over a period of ~3 months. Unit recording times are indicated by black bars, and are sorted by when they were first identified in the recording. Black triangles and dotted lines indicate times at

which the electrode array was intentionally advanced into the brain by turning the micro-drive. Open triangles indicate times at which the population of recorded units changed spontaneously.

B. Holding times for units recorded in the DLS (left, green) and MC (right, orange), sorted by duration.

C. Number of simultaneously recorded units in the DLS (left, green) and MC (right, orange) as a function of time in the recording.

D. Cumulative distributions of average cluster isolation quality for all units recorded in DLS (left, green) and MC (right, orange). Cluster quality was measured by the isolation distance (top), L-ratio (middle), and fraction of inter-spike intervals under 1 millisecond (bottom). Dotted lines mark the quality thresholds for each of these measures. Shaded regions denote acceptable values.

Figure 3: Long-term stability of single unit dynamics

A. Histograms of average firing rates for units recorded in DLS (left) and MC (right). Putative cell-types, medium spiny neurons (MSN, blue) and fast-spiking interneurons (FSI, green) in DLS, and regular spiking (RS, brown) and fast spiking (FS, red) neurons in the MC, were classified based on spike shape and firing rate (Methods). The continuous traces are log-normal fits to the firing rate distributions of each putative cell-type. Insets show average peak-normalized waveform shapes for MSNs (left-bottom) and FSIs (left-top), and RS (right-bottom) and FS (right-top) neurons. Shading represents the standard deviation of the spike waveforms.

B. Firing rates of DLS (left) and MC (right) units for their recording duration. The color scale indicates firing rate on a log-scale, calculated in one-hour blocks. Units have been sorted by average firing rate.

C. Scatter plots of unit firing rates over time-lags of 1 (left), 5 (middle) and 10 (right) days for DLS (top) and MC (bottom). The dashed lines indicate equality. Every dot is a comparison of a unit's firing from a baseline day to 1, 5 or 10 days later. The color of the dot indicates putative cell-type as per (A). Each unit may contribute multiple data points, depending on the length of the recording. Day 1: n = 2024 comparisons for

striatum and n = 2225 for cortex; Day 5: n = 851 comparisons for striatum and n = 897 for cortex; Day 10: n = 255 comparisons for striatum and n = 350 for cortex.

D. Stability of unit firing rates over time. The firing rate similarity (see Methods) was measured across time-lags of 1 to 10 days for the same unit (within-unit, solid lines), or between simultaneously recorded units (across-unit, dashed lines) in DLS (left) and MC (right). Colored shaded regions indicate the standard deviation of within-unit firing rate similarity, over all units. Grey shaded regions indicate standard deviation of across-unit firing rate similarity, over all time-bins.

E. Inter-spike interval (ISI) histograms for example units in DLS (left, green) and MC (right, orange) across two weeks of continuous recordings. Each line represents the normalized ISI histogram measured on a particular day.

F. Stability of unit ISI distributions over time. Correlations between ISI distributions (see Methods) were measured across time-lags of 1 to 10 days for the same unit (within-unit, solid lines), or between simultaneously recorded units (across-unit, dashed lines) in MC (left) and DLS (right). Colored shaded regions indicate the standard deviation of within-unit ISI similarity, over all units. Grey shaded regions indicate standard deviation of across-unit ISI similarity, over all time-bins.

Figure 4: Stability of network dynamics

A. Correlograms for example unit pairs from DLS (right) and MC (left) over 10 days. Each line represents the normalized correlogram measured on a particular day.

B. Scatter plots of correlation peaks compares the peak correlation of a unit pair on a baseline day to the same pairs peak correlation 1, 5 or 10 days later, for DLS (top) and MC (bottom). The black line indicates equality. Values >1 (<1) correspond to positive (negative) correlations (Methods). Day 1: n=3846 comparisons for DLS and n=3650 for MC. Day 5: n=1218 comparisons for DLS and n=797 for MC. Day 10: n=198 comparisons for DLS and n=190 for MC.

C. Stability of correlations over time. The correlation similarity (see Methods) was measured across time-lags of 1 to 10 days for the same pairs (solid lines), or for the

same pairs with non-significant correlations (dashed lines) in DLS (top) and MC (bottom). Color shaded regions indicate the standard deviation of the correlation similarity between significant pairs. Grey shaded regions indicate standard deviation of correlation similarity between insignificant pairs.

Figure 5: Automated classification of behavioral states.

- **A.** (Top) Spectrograms of the accelerometer signal (top) and the LFP (middle) for an hour-long window of recording from the striatum. Below is the color-coded output of our automated classification algorithm. UL-unlabeled; SW-Slow wave sleep; RM-REM sleep; QW- quiet wakefulness, GR-grooming, AX-active exploration, ET-eating.
- **B.** Ethograms showing the proportion of time spent in each of the behavioral states over 40 consecutive days for the two rats we recorded from. The ethograms do not include absence seizure-like states or unlabeled states and were normalized to one (Methods).
- **C.** Circadian profiles of behavioral states over 40 days of recording for active and inactive states for the two rats in B. Behavioral states are color-coded as in A, with grey representing inactive (or active) and unlabeled states.

Figure 6: Stability of unit activity across different behavioral states

- **A.** Stability of average firing rates in different behavioral states across several days for example units from the DLS (left) and MC (middle and right). The color-scale corresponds to the firing rate of a unit normalized by its average firing rate. The state abbreviations correspond to slow-wave sleep (SW), REM sleep (RM), quiet wakefulness (QW), grooming (GR), active exploration (AX), eating (ET) and task execution (TK).
- **B.** Stability of state-dependent firing rates over time. Correlations between state-modulated firing rates were measured across time-lags of 1 to 10 days for the same unit (within-unit, solid lines), or between simultaneously recorded units (across-unit, dashed lines) in MC (left) and DLS (right). Colored shaded regions indicate the standard deviation of within-unit state firing rate similarity, over all units. Grey shaded regions indicate standard deviation of across-unit state firing similarity, over all time-bins.

C. Clustering of units into functional 'grouping' based on similarities in their firing rate profiles across different behavioral states for DLS (left, 6 groups) and MC (right, 5 groups) units. Each row represents the average firing rate modulation across behavioral states for a particular unit, normalized by its average firing rate. Units are clustered into groups by k-means clustering, and sorted, within types, by their maximum normalized firing rates.

Figure 7: Stability of behavioral representations in single units.

A. Spike-triggered average (STAs) accelerometer power calculated daily in three different behavioral states – grooming (left), active exploration (middle) and eating (right). Shown are four example units recorded from DLS (top two rows) and MC (bottom two rows).

B. Stability of STAs over time. Correlations between STAs were measured across time-lags of 1 to 10 days for the same unit (within-unit, solid lines), or between simultaneously recorded units (across-unit, dashed lines) in MC (left) and DLS (right). Colored shaded regions indicate the standard deviation of within-unit STA similarity (averaged over the 3 behavioral states in 'A'), over all units. Grey shaded regions indicate standard deviation of across-unit STA similarity, over all time-bins.

C. Peri-event time histograms (PETHs) of DLS (left) and MC (right) unit activity, aligned to the timing of a lever-press or nose-poke during the execution of a skilled motor task. Plotted are the PETHs of units that had significant modulations in their firing rate in a time-window ± 200 ms around the time of the behavioral event (Methods). The color scale indicates Z-scored firing rate. Units are sorted based on the peaks in their PETHs.

D. Spike raster of an example DLS unit over 12 days, aligned to the time of a nose-poke event. Each dot represents a spike-time on a particular trial. The color of the dot indicates day of recording.

E. PETHs computed over several days for example DLS (top) and MC (bottom) units to lever-press (left) and nose-poke (right) events in our task.

F. Stability of task PETHs over time. Correlations between PETHs were measured across time-lags of 1 to 10 days for the same unit (within-unit, solid lines), or between

simultaneously recorded units (across-unit, dashed lines) in DLS (left) and MC (right). Colored shaded regions indicate the standard deviation of within-unit PETH similarity (averages across lever-press and nose-poke events), over all units. Grey shaded regions indicate standard deviation of across-unit PETH similarity (averaged across lever-press and nose-poke events), over all time-bins.

Supplementary Figure 1: Algorithm for identifying spikes from the raw data.

- **A.** Each input channel s_{ch} is split into two streams, one containing the low frequency components lfp_{ch} and one containing the high frequency ones, sf_{ch}. The median of sf_{ch} across all channels is subtracted from each channel resulting in $\overline{sf}_{ch}(t)$. Spike times st_i^j and spike waveforms $\boldsymbol{x}_i^j$ from each tetrode are then extracted. The LFPs and spikes extracted from the raw data is saved to disk resulting in a 5-10x 'compression' of the raw data.
- **B.** Algorithm for detecting spikes. If the absolute value of the filtered signal exceeds 50µV in any channel of a tetrode then a spike is 'detected'. The spike is considered to have 'ended' if all channels remain within 20 µV for 8 consecutive samples.
- **C.** Example 1 s long raw recordings from a tetrode. The red lines mark the ±50µV spike detection threshold.
- **D.** Examples of 2 ms wide spike snippets (64 samples) extracted from the data in **C**. Snippets from all 4 electrodes detected using the state machine of **B** are aligned to the peak of the spike waveform and concatenated to produce the 256 sample spike waveforms $\boldsymbol{x}_i^j$.

Supplementary Figure 2: Algorithm for local clustering and de-noising.

- **A.** Raw spike waveforms $\{\boldsymbol{x}_i\}$ are locally clustered and split into low- and high-density clusters (details in panels B and C). The spikes from low-density clusters are further split into two streams in the same manner 3 more times. The centroids of high density clusters from all 4 stages are pooled together to form the output $\{\boldsymbol{y}_j\}$.

B. Local clustering of each 1000 spike block. Super-paramagnetic clustering generates a cluster tree (ii) from the spike waveforms (i), the leaves of which are recursively merged (iii and iv) to generate a clustering of the 1000 points (v). The dotted blue lines show which leaves of the tree in (ii) are merged to produce the tree in (iii). The nodes marked red in (ii) correspond to 'distinct clusters', i.e. clusters that are very different from the parent nodes. The leaves of (iii) are similarly merged to produce the tree in (iv). The colored leaves correspond to high-density clusters, i.e. clusters with more than 15 points and the black leaves correspond to low-density clusters.

C. Schematic illustrating splitting of spikes into low-density and high-density clusters. The set of input spike waveforms $\{\boldsymbol{x}_i^n\}$ is split into blocks of 1000 spikes (3 blocks shown in the figure) with each block split into low (colored black) and high density clusters (colored blue and red) using the procedure shown in panel **B**. The spikes from the low density clusters are pooled together to form $\{\boldsymbol{x}_i^{n+1}\}$. The centroid of the high density clusters form $\{\boldsymbol{y}_i^n\}$.

D. Number of spike waveforms in a 30 minute period from one tetrode in various stages of the local clustering and de-noising algorithm.

Supplementary Figure 3: Algorithm for linking cluster trees to track single units over time.

A. The output of the previous step (averaged spikes waveforms – see Supplementary Figure 3) is split into a sequence of 1000 spike blocks and converted into a sequence of cluster trees (5 trees shown in the figure). A subset of all possible links between adjacent cluster trees is chosen by maximizing the total similarity between linked nodes subject to the constraints depicted in panels **B**, **C**, and **D**. The subset of chosen nodes and links are highlighted in color. Three sets of nodes connected by links, one in red and two in green, are shown. The two green chains are merged to produce a final sorting containing two units (red and green).

B. The constraint shown ensures that none of the 4 incoming links ($L_1 - L_4$) are chosen if the node marked C is not chosen. It also ensures that if C is chosen, at most one of the incoming links is chosen.

C. Same as **B** but for outgoing links.

D. These constraints ensure that if a node is chosen then none of its parents or child nodes are.

Supplementary Figure 4: Benchmarking the performance of our unsupervised spike sorting algorithm by comparing its output to manual sorting.

A. Fraction of clusters matched after independent spike sorting of the same dataset by manual sorting and our unsupervised sorting algorithm. The number of clusters overlapping (defined as sharing at least 50% of the same spikes) after sorting by the two methods is shown as a fraction of the total number of manual clusters (blue) or clusters identified by the automatic algorithm (red). The x-axes indicate different cluster quality criteria, in terms of isolation distance (bottom) and L-ratio (top), used to eliminate low-quality clusters from both auto and manual sorting results. The dotted line indicates the quality criteria typically used in our recordings.

B. Fraction of spikes overlapping in matched clusters identified by both manual sorting and our unsupervised method. The number of overlapping spikes is shown as a fraction of total spikes in the manual cluster (blue) or the cluster identified by our unsupervised spike sorting method (red). The x-axes indicate different cluster quality criteria, in terms of isolation distance (bottom) and L-ratio (top), used to eliminate low-quality clusters from both auto and manual sorting results. The dotted line indicates the quality criteria typically used in our recordings. Shaded regions indicate standard error of the mean, across clusters.

C. Comparison of cluster quality, measured in terms of L-ratio (top) and isolation distance (bottom), for clusters matched between manual and automatic spike sorting, after elimination of low quality clusters using our typical quality criteria (dotted lines in A-B). Error bars indicate standard error of the mean.

Supplementary Figure 5: Similarity of motor representations in neurons with similar changes in firing rate across behavioral states.

A. Histogram of STA similarity between all pairs of neurons belonging to the same (black) or different types (blue) in the DLS (left, 6 types) and MC (right, 5 types). Clustering of neurons in types is based on similarity in their changes in firing rate across different behavioral states, as shown in Figure 5C.

B. Histogram of PETH similarity between all pairs of neurons belonging to the same (black) or different types (blue) in the DLS (left, 6 types) and MC (right, 5 types).

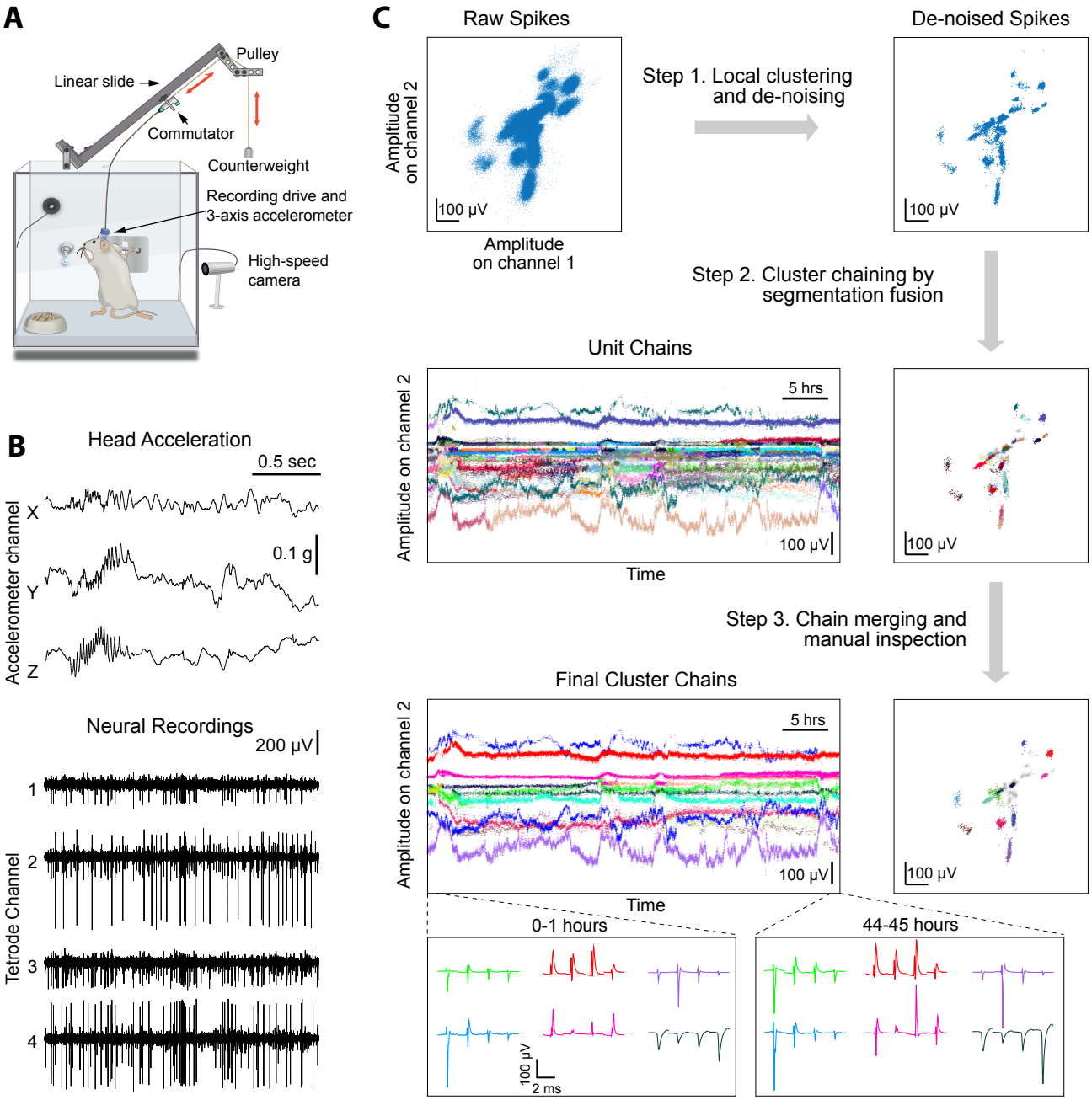


Figure 1

Dhawale et al. 2015

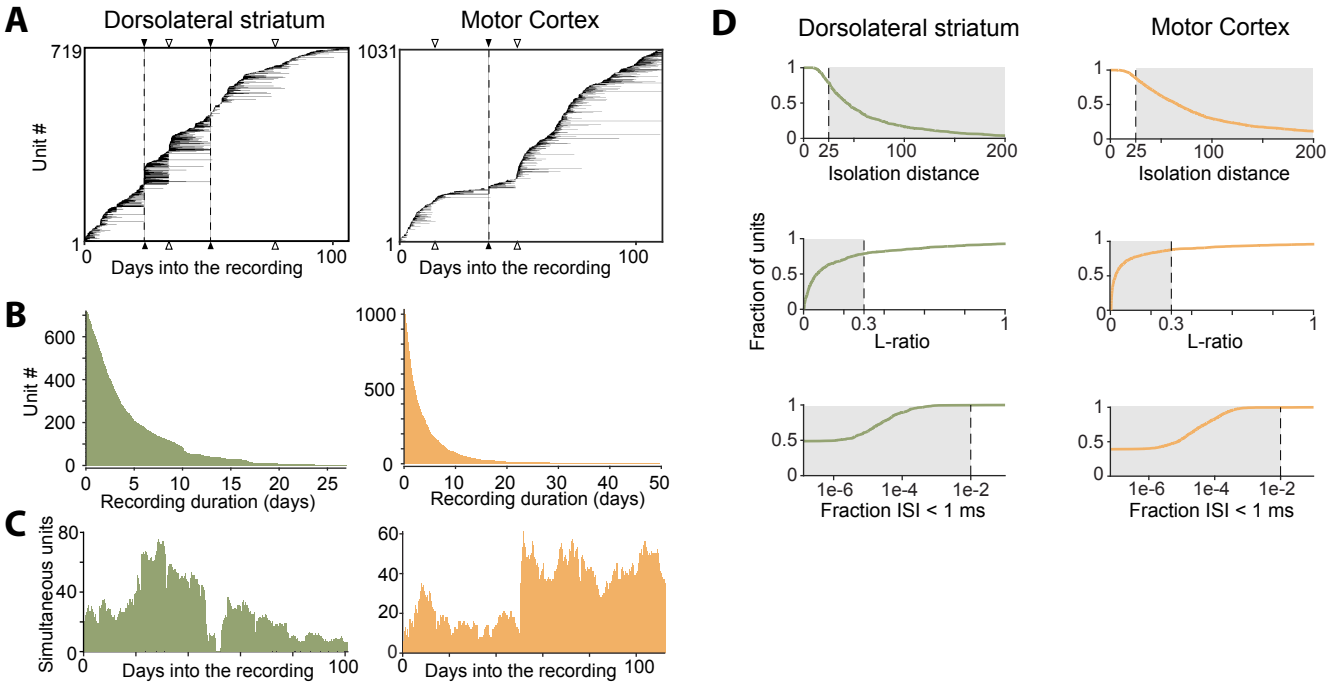


Figure 2
Dhawale et al. 2015

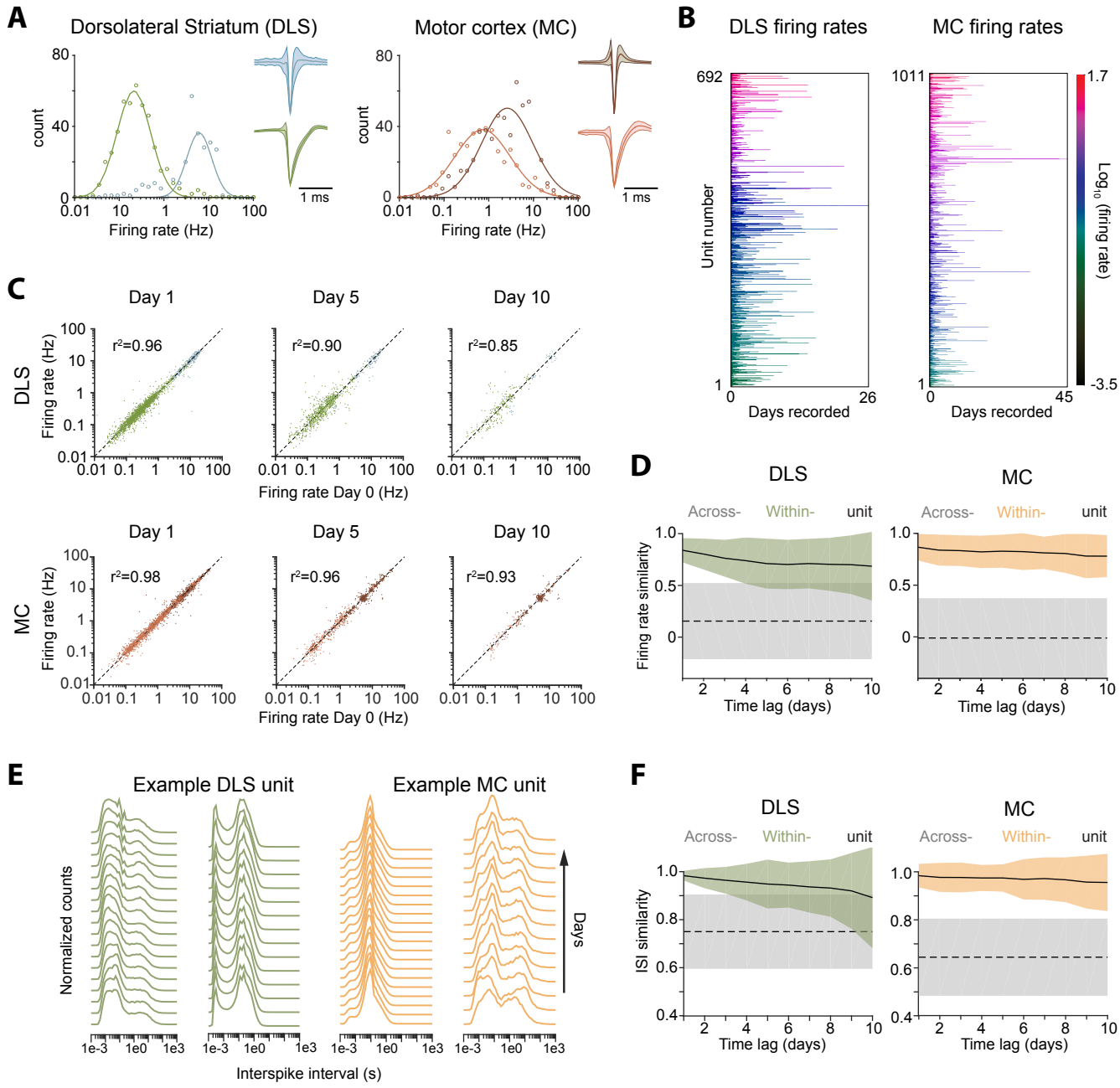


Figure 3

Dhawale et al. 2015

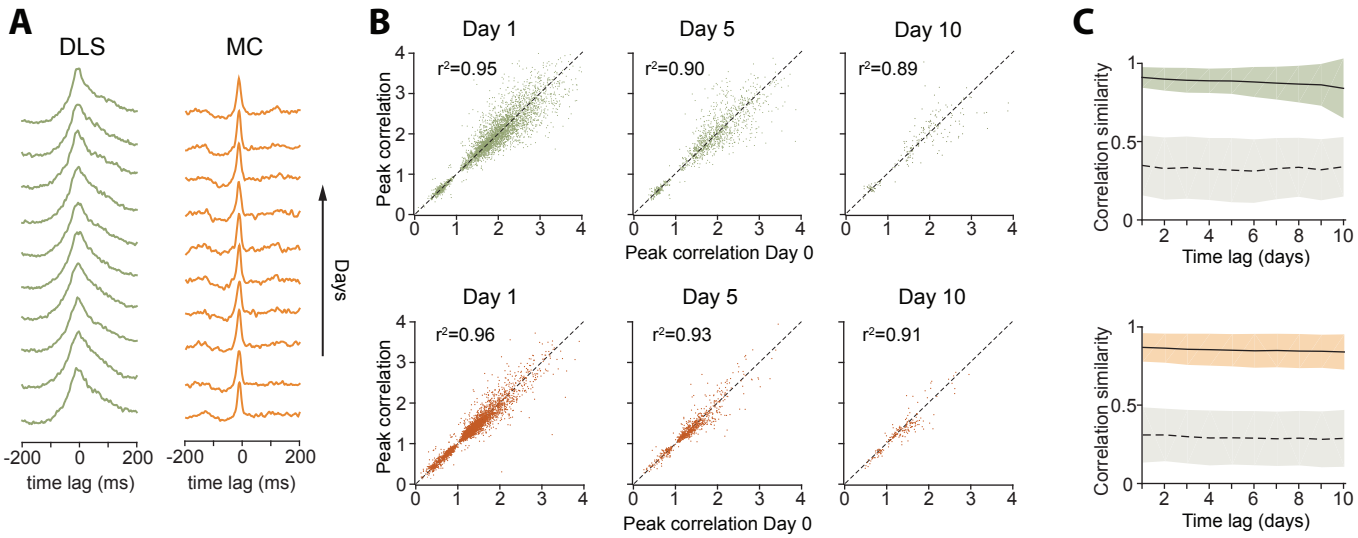


Figure 4

Dhawale et al. 2015

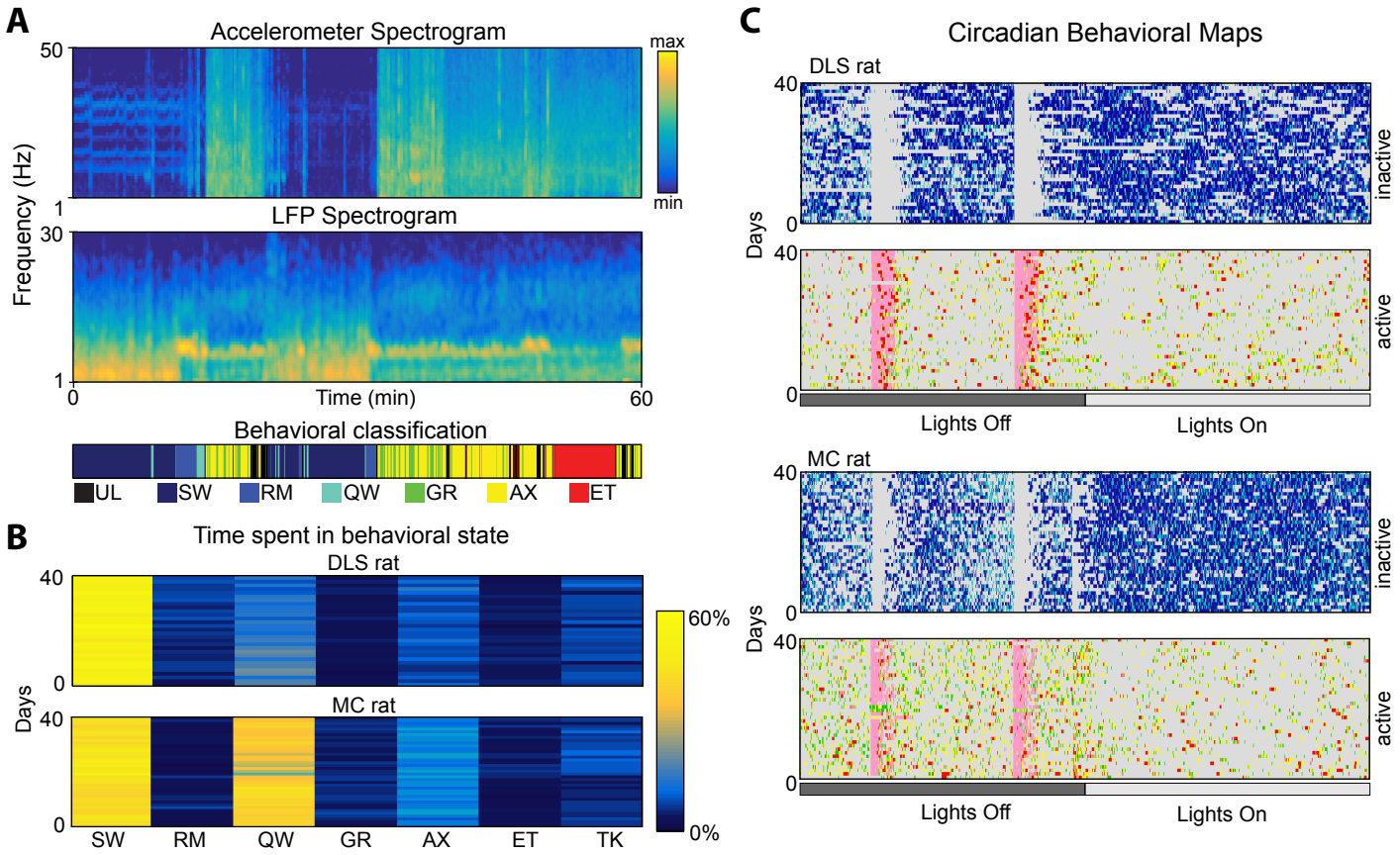


Figure 5

Dhawale et al. 2015

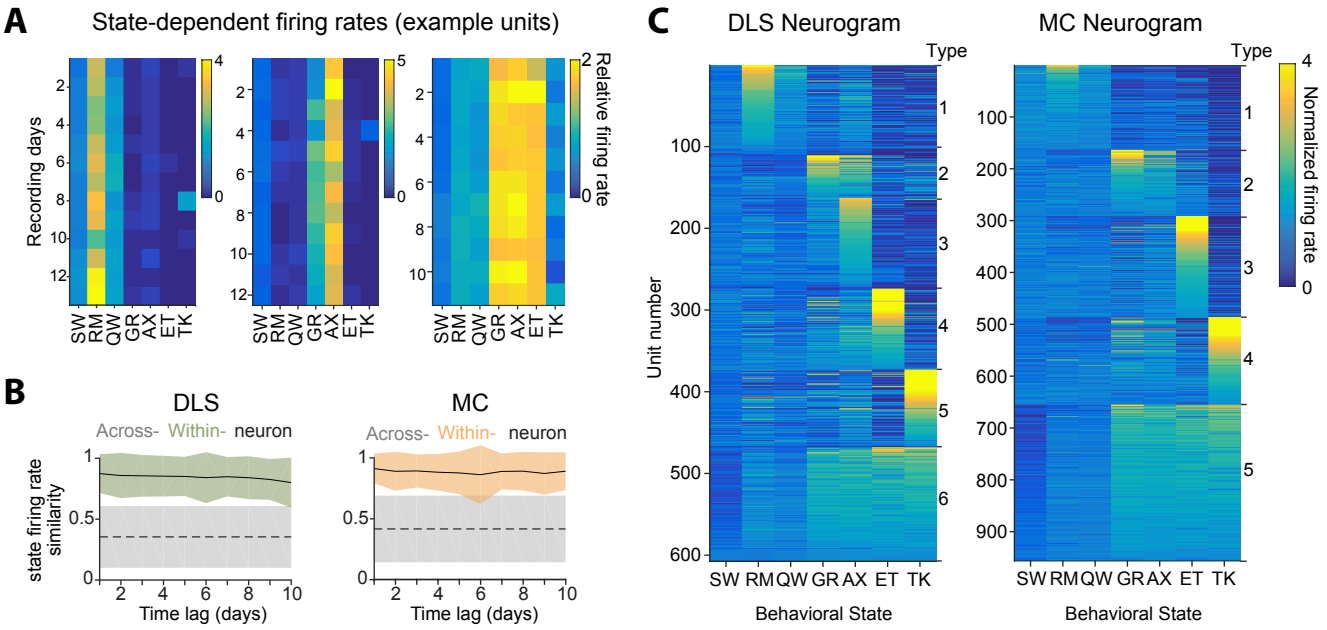


Figure 6

Dhawale et al. 2015

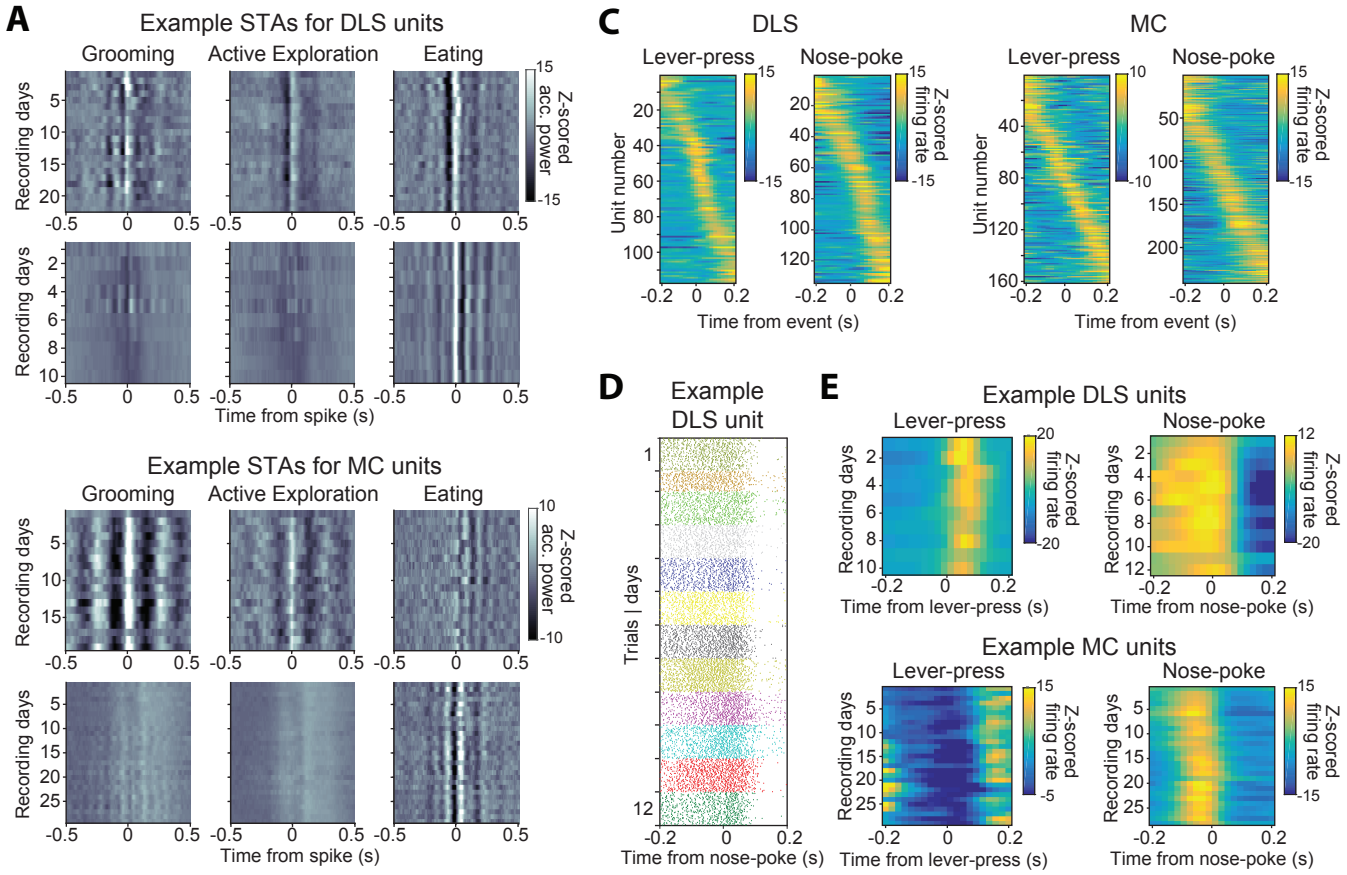
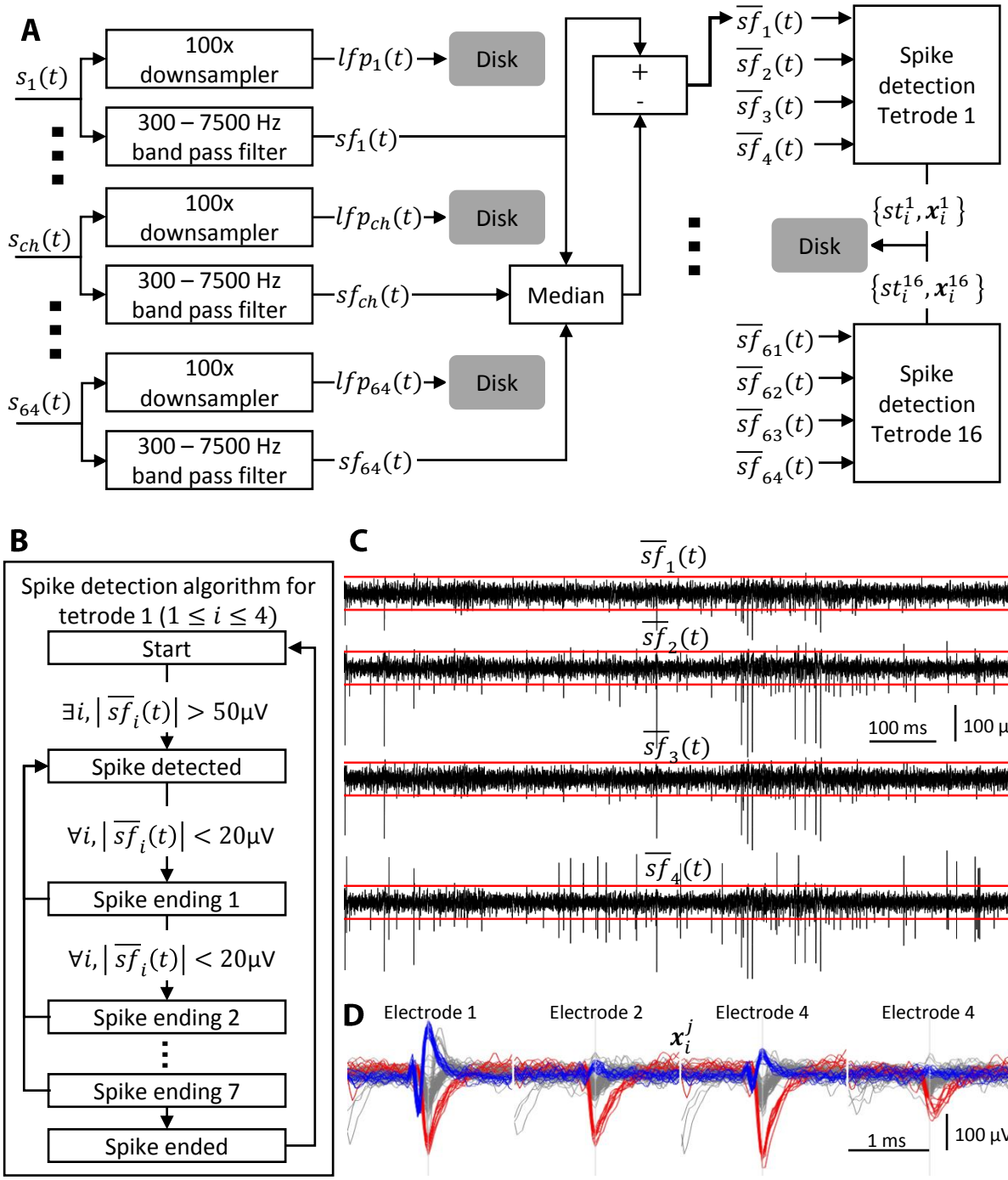
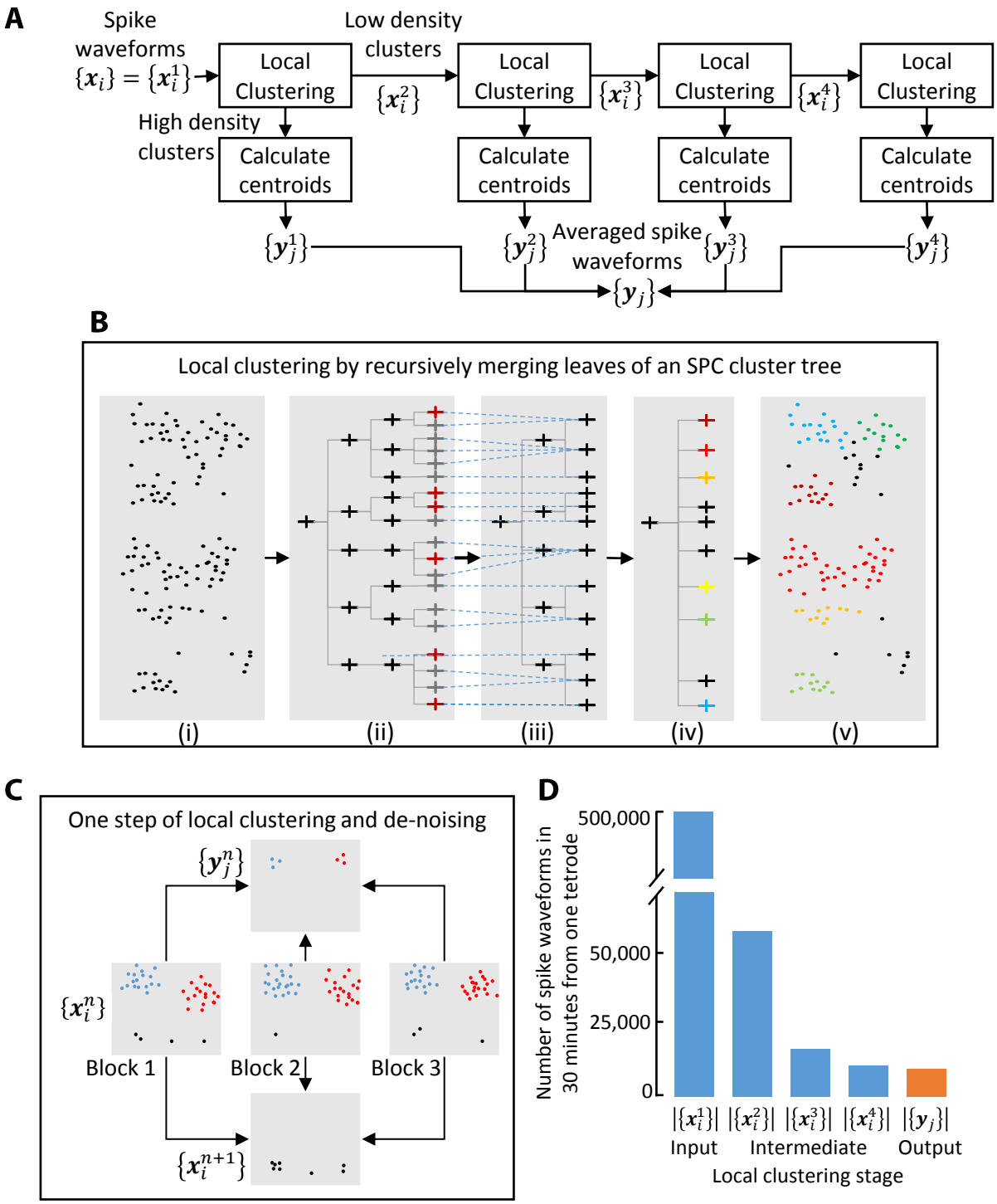


Figure 7
Dhawale et al. 2015

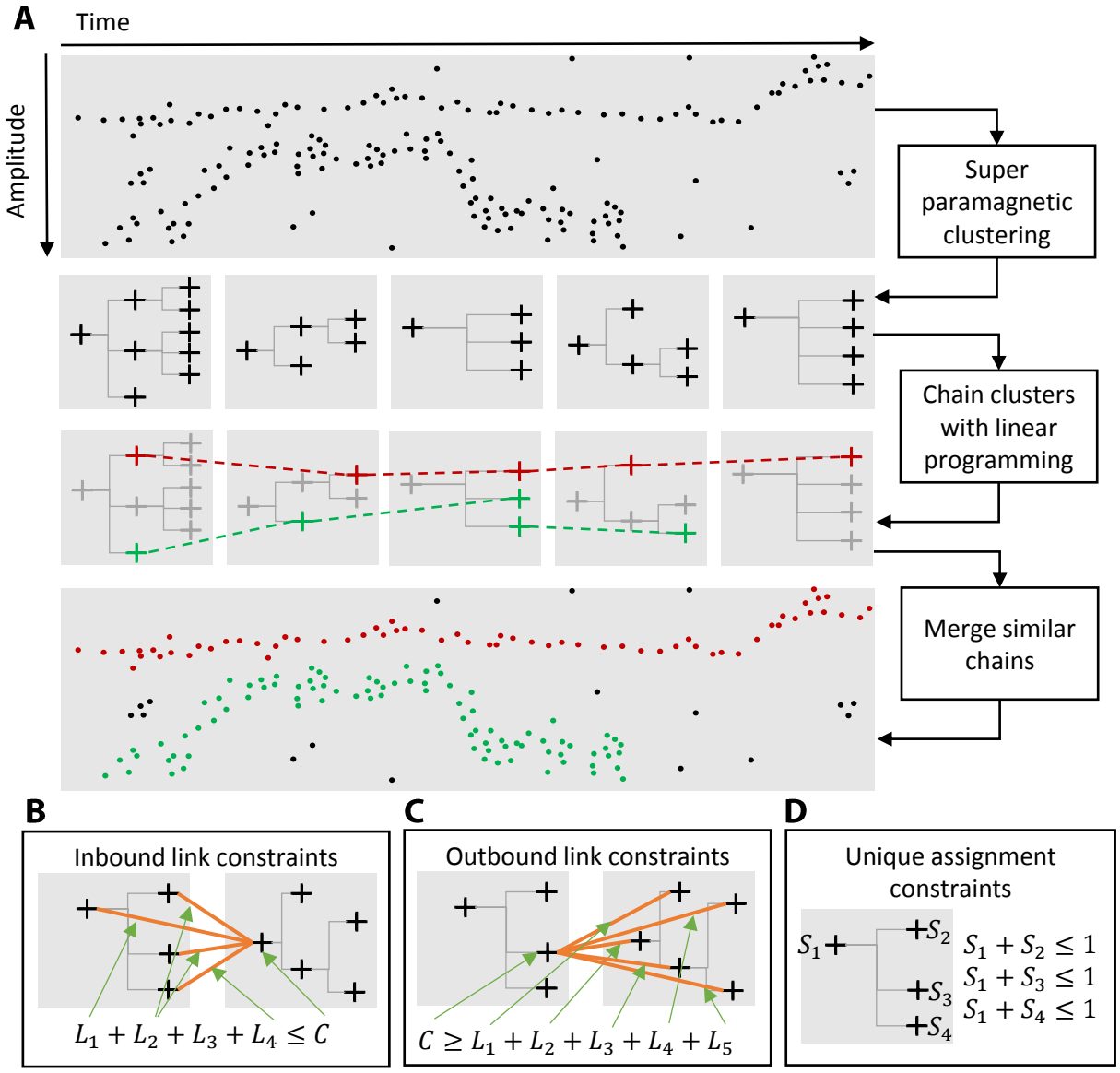
Supplementary Figure 1



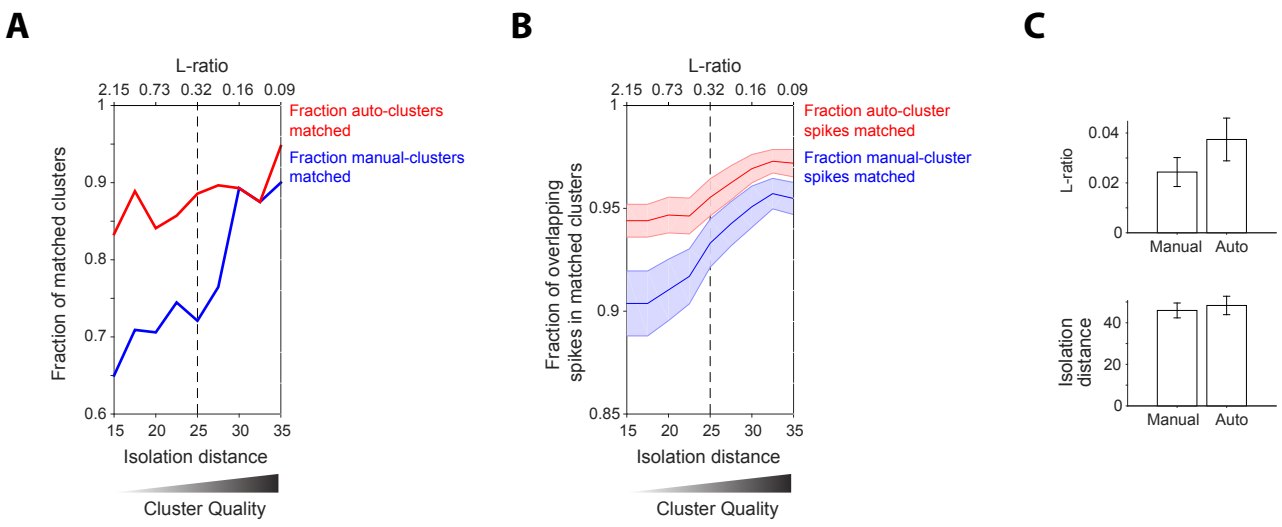
Supplementary Figure 2



Supplementary Figure 3



Supplementary Figure 4



Supplementary Figure 5