PNAS
www.pnas.org

Supplementary Information for

Adaptive, locally-linear models of complex dynamics

Antonio C. Costa, Tosif Ahamed and Greg J. Stephens

Greg J. Stephens.
E-mail: g.j.stephens@vu.nl

This PDF file includes:

Supplementary text
Figs. S1 to S10
References for SI reference citations

Supporting Information Text

Adaptive locally-linear segmentation algorithm

We first define a set of candidate windows in which to examine whether there are dynamical breaks. This is done iteratively: we set a minimum window size w_{min} and then increment by $\sim 10\%$ which ensures that larger windows contain a proportionally larger number of observations. The candidate windows range between w_{min} and some w_{max} which corresponds to the value at which the step size is larger or equal to w_{min}. The specific value of w_{min} depends on the dataset and the dimensionality d and we chose w_{min} to be the smallest interval in which the data can be reliably fit. However, simply setting $w_{min} = d$ does not incorporate the possibility of multicollinearity, when two or more components are not linearly independent, which produces an ill-conditioned linear regression. This linear dependence results in a moment matrix $\mathbf{X}^\top\mathbf{X}$ that is not full rank or nearly singular, and therefore small perturbations result in large fluctuations in the estimated linear parameters. In addition, computing the log-likelihood function Eq. (4) requires inverting the covariance matrix of the error Σ. Thus, we require a minimum window size for which both $\mathbf{X}^\top\mathbf{X}$ and Σ are well-conditioned. We compute the condition number of these matrices as a function of window size and choose w_{min} as the smallest window for which the condition numbers are reasonably small. The results for each analyzed dataset are shown in Fig. (S8).

Algorithm 1 Iterative construction of window sizes

```
w = w_min
s = 0
while s < w_min do
  save w
  s = int(w/10)
  if s < w_min then
    w = w + s
  else
    break
  end if
end while
```

Given a set of candidate windows we iterate over pairs of consecutive windows of size w_k and w_{k+1}, estimate the respective model parameters θ_k and θ_{k+1}, and locate a dynamical break if θ_{k+1} performs significantly better than θ_k in fitting the data from the window of size w_{k+1}. We assess significance through a likelihood ratio test and obtain $\Lambda_{k,k+1}$ from Eq. (3). We note that our models are non-nested for which the likelihood ratio would be asymptotically χ^2 distributed. Instead, we take θ_k as a null model for the observations in the window of size w_{k+1} and use a Monte Carlo approach to generate $N = 5000$ surrogate trials of size w_{k+1} from θ_k in order to compute $P_{\text{null}}(\Lambda)$, the distribution of the log-likelihood ratio under the null hypothesis of having no model change. We identify a dynamical break if $\Lambda_{k,k+1} > \Lambda_{\text{thresh}}$ where Λ_{thresh} is defined by the larger solution of $P_{\text{null}}(\Lambda_{\text{thresh}}) = 0.05$. A graphical representation of the technique is shown in Fig. (1) and the algorithm is detailed below. Finally, if the algorithm iterates to the maximum window size w_{max} we automatically assign a break which we then asses through the following procedure: we start with $w_k = w_{min}$ and compare the models found in the intervals $[w_{max} - w_k, w_{max}]$ and $[w_{max} - w_k, w_{max} + (w_{k+1} - w_k)]$ as we increase k until we span the entire set of candidate windows. If none of these tests suggest a break then we simply remove it.

We choose the significance threshold empirically and this choice reflects a tension between model complexity and accuracy; varying $P_{\text{null}}(\Lambda_{\text{thresh}})$ principally changes the number of breaks. While we have found $P_{\text{null}}(\Lambda_{\text{thresh}}) = 0.05$ to be reasonable across multiple datasets we provide additional intuition through a toy segmentation problem illustrated in Fig. (S9). We simulate $N = 100$ two-dimensional systems $\vec{x}_s$ for which we change the model parameters twice: first we apply a small change to the coupling between x_1 and x_2, $A_{12} \to A_{12} + 0.03$, while in the next change we symmetrize the couplings between x_1 and x_2 thus reverting the direction of the oscillation, Fig. (S9-top). Both change points are accurately determined even for a significance level of 1%, Fig. (S9-middle); the dynamical changes are found $\sim 96\%$ of the time, even though the change between the first two models is quantitatively small. In Fig. (S9-bottom) we show the number of true positives (breaks found by the algorithm that are true dynamical changes) and false positives (breaks found by the algorithm that are not true dynamical changes) as a function of the significance level. The fraction of true positives is essentially preserved (even if we stretch to a 1% significance level), indicating that missing true dynamical changes is rare. The results reported in this manuscript do not depend sensitively on the significance threshold.

Algorithm 2 Description of the adaptive locally-linear segmentation of a d-dimensional time series $\vec{x}$, of length T given a set of N_w candidate windows.

$t = 0$
while $t < T$ **do**
  $k = 0$
  while k $< N_w$ **do**
    $X_k = \vec{x}_t, t \in [t, t + w_k]$
    $X_{k+1} = \vec{x}_t, t \in [t, t + w_{k+1}]$
    Fit θ_k and θ_{k+1} to X_k and X_{k+1}, respectively
    Compute $\Lambda_{k,k+1}$ over X_{k+1}, from θ_k and θ_{k+1}
    Generate N_s time series X^s_{k+1} of size w_{k+1}, using θ_k
    Compute $\Lambda^s_{k,k+1}$ with the newly obtained θ^s_k and θ^s_{k+1}, for each N_s time series, obtaining a distribution $P_{\text{null}}(\Lambda)$
    Estimate Λ_{thresh} as the largest solution of $P_{\text{null}}(\Lambda_{\text{thresh}}) = 0.05$
    if $\Lambda_{k,k+1} \leq \Lambda_{\text{tresh}}$ **then**
      There is no dynamical change
      $k = k + 1$
    else
      save the window $[t, t + w_k]$
      break
    end if
  end while
  $t = t + w_k$
end while

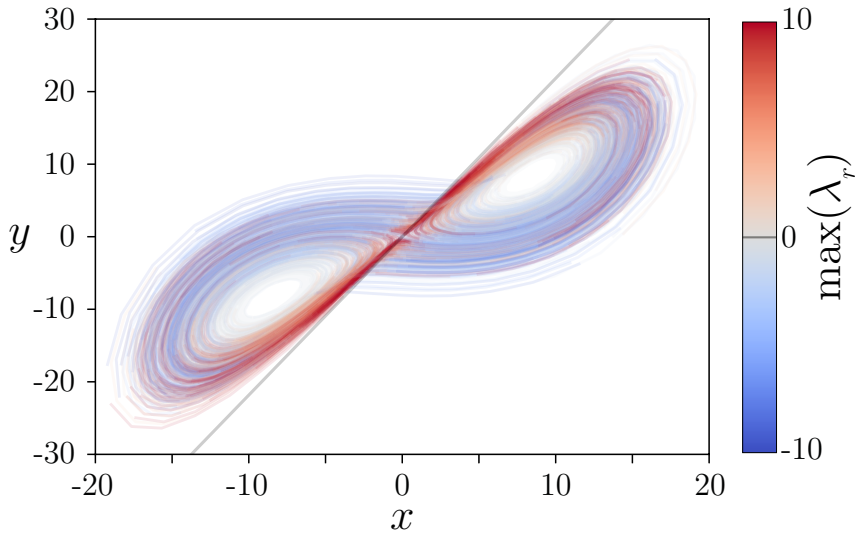

Fig. S1. Unstable linear models in the chaotic regime of the Lorenz system lie along the unstable manifold of the origin. We plot the xy projection of the Lorenz system in the chaotic regime, color coded by the magnitude of the maximum real eigenvalue λ_r of the linear model found in each window resulting from the adaptive segmentation. Most unstable models are found close to the origin, along its $1d$ unstable manifold (gray line).

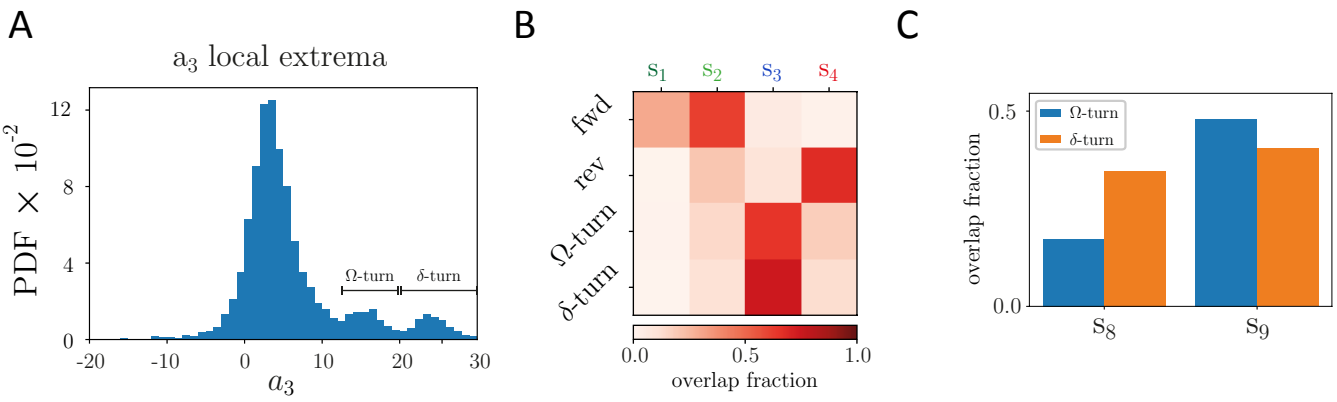

Fig. S2. Overlap between the clusters found in *C. elegans* postural dynamics and behavioral motifs defined phenomenologically. (A) We identified Ω and δ turns using similar criteria as (1). First, we found local extrema using scipy.signal.find_peaks package (2) in Python (3), with a prominence of 0.5. Only peaks without local extrema in a 3 s window around are taken into account. For each of the windows identified with our locally-linear segmentation, we first check whether there is a well defined turn, according to the previous criteria. Then, if the amplitude of the peak is between 12 and 20, we classify it as an Ω-turn; if it is larger than 20, we classify it as a δ-turn. Windows for which there is no well defined turn and the maximum turning amplitude is below 12 are classified as either forward or reversal based on the sign of the phase velocity ω. All remaining windows receive no label. (B) Fraction of windows classified phenomenologically that fall into each of our clusters at a 4-branch level. The sparsity of the overlap matrix indicates that our clusters agree with classical definitions of coarse-grained behaviors. (C) At a 12-branch level, cluster s_8 exhibits more δ that Ω turns, while s_9 exhibits more Ω turns. Nonetheless, there is still some confusion between δ and Ω turns. Our local linear models span time scales that are shorter than a typical turn, thus subdividing it. Therefore it is not surprising that there is some confusion between δ and Ω turns at this level of clustering: the only distinction between these is the height of the a_3 peak, which will correspond to only a small fraction of linear models. In addition, our clustering takes into account the entire dynamical pattern and therefore the value of the peak alone plays a minor role in differentiating between clusters.

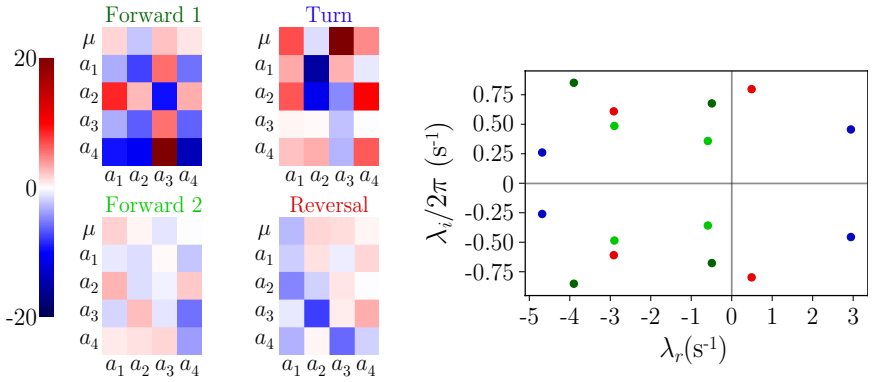

Fig. S3. Distinct behavioral classes in the dendrogram interpreted through model parameters. The mean $\vec{\mu}$ and linear couplings ϕ displayed as a matrix $(\vec{\mu}, \quad \phi)^\top$ (left) and the respective dynamical eigenvalues (right) are shown for a set of example models. From the first to the second forward state, the imaginary eigenvalues shrink, corresponding to a reduction of the oscillation frequencies. The turn state exhibits a higher value of the mean of a_3 and, in this example at the beginning of a turn, we find an unstable oscillation. Finally, in the reversal state, the sign of the coupling between the first two modes is reversed and this signals a change in the sign of the phase velocity.

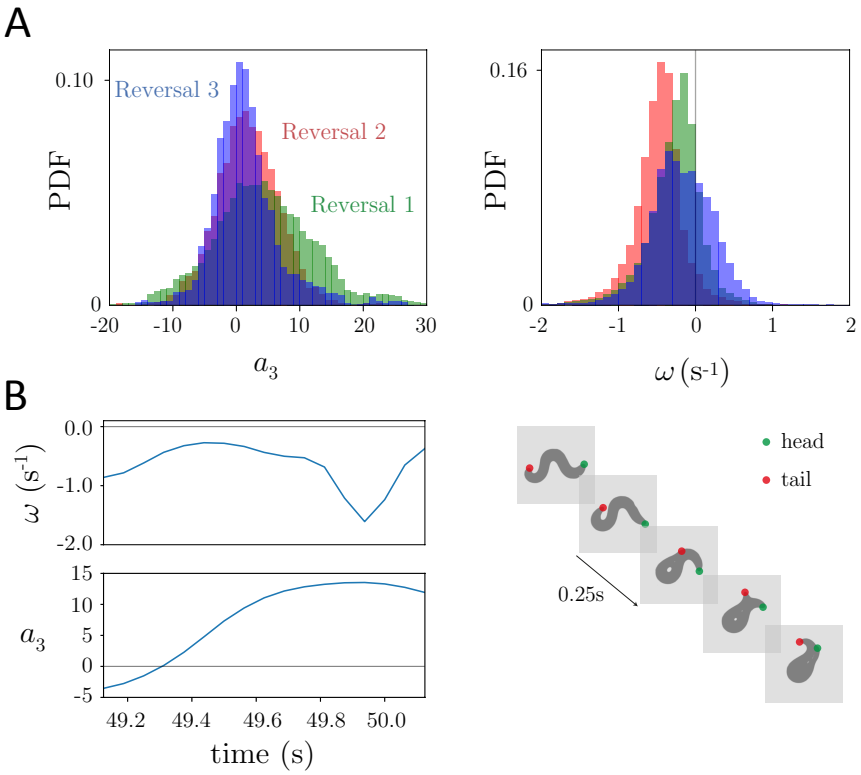

Fig. S4. At a 12-branch level of the dendrogram the reversal branch exhibits a reversal-turn behavioral motif. (A-left) Distribution of the turning amplitudes a_3. The first reversal (Reversal 1 - green) is actually a reversed turn, as noted by the high value of a_3. The other two reversals generally have smaller turning amplitudes. (A-right) Distribution of body wave phase velocities ω. The second reversal branch (Reversal 2 - red) corresponds to faster reversal bouts, while the third reversal (Reversal 3 - blue) includes movements at the start of a reversal when ω is small. (B-left) Example trajectory of a reversal-turn. A negative phase velocity ω is accompanied by a peak in a_3 for which the body is bent as in an Ω-turn. (B-right) Worm images from the example trajectory sampled each 0.25 s. The head and tail are identified with a green and red dot, respectively.

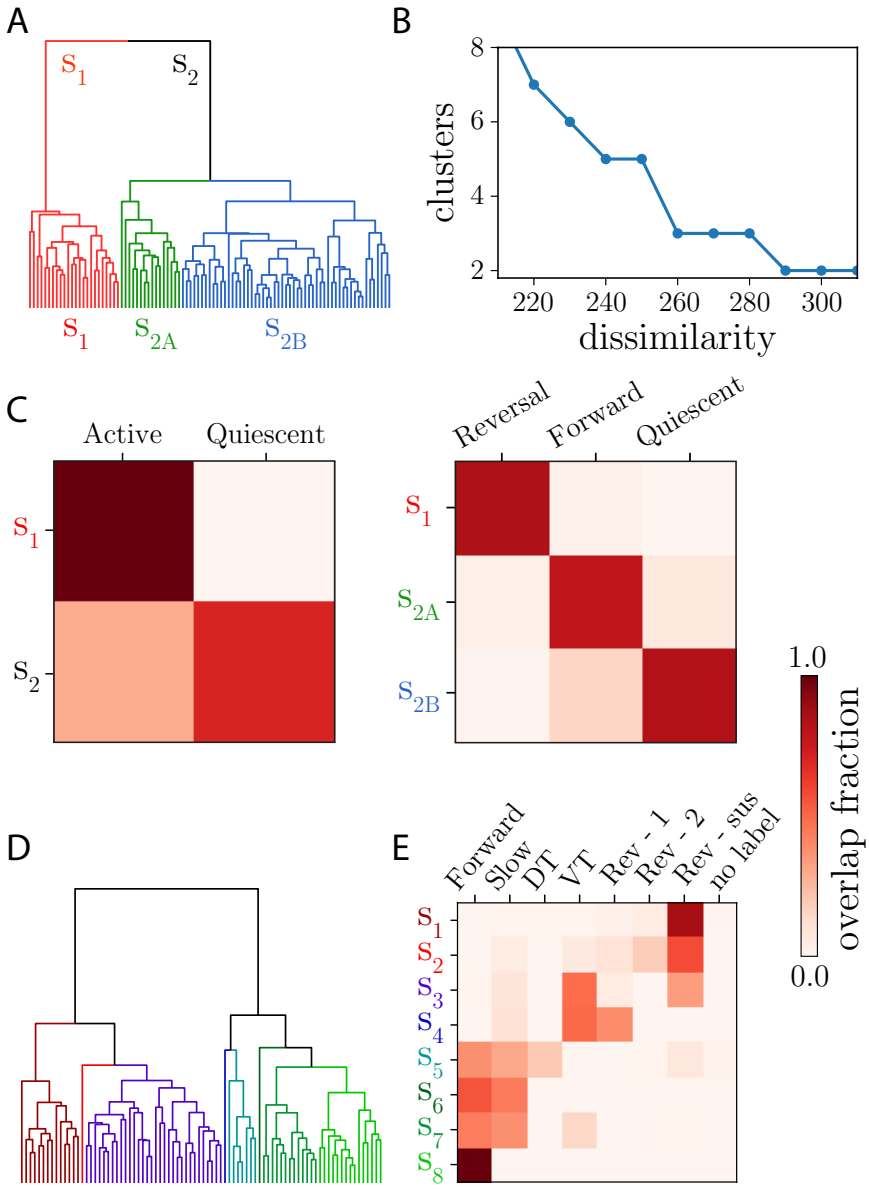

Fig. S5. **Model clusters exhibit approximate correspondence with labeled brain states.** (A) Model clustering dendrogram obtained by segmenting the neural activity of the example worm shown in Fig. (5A). (B) Number of clusters as a function of dissimilarity. The first major splits occur at the two branch and three branch level. (C) Overlap between model clusters and labeled brain states (4). At the two branch level, we find that most of the frames in branch S_1 were labeled as "active", while frames in branch S_2 were labeled as "quiescent". At the three branch level we find a high degree overlap between S_1 and "reversal", S_{2A} and "forward", and S_{2B} and "quiescent". (D) Model clustering dendrogram obtained by segmenting the neural dynamics of an exemplar worm from previous global brain imaging experiments (5). (E) Overlap between model clusters and labeled brain states from previous experiments (5). The sparseness in the matrix indicates a broad match between the states, specially for the forward and reverse states. (DT - Dorsal Turns, VT - Ventral Turns, Rev-Sus - Sustained Reversal)

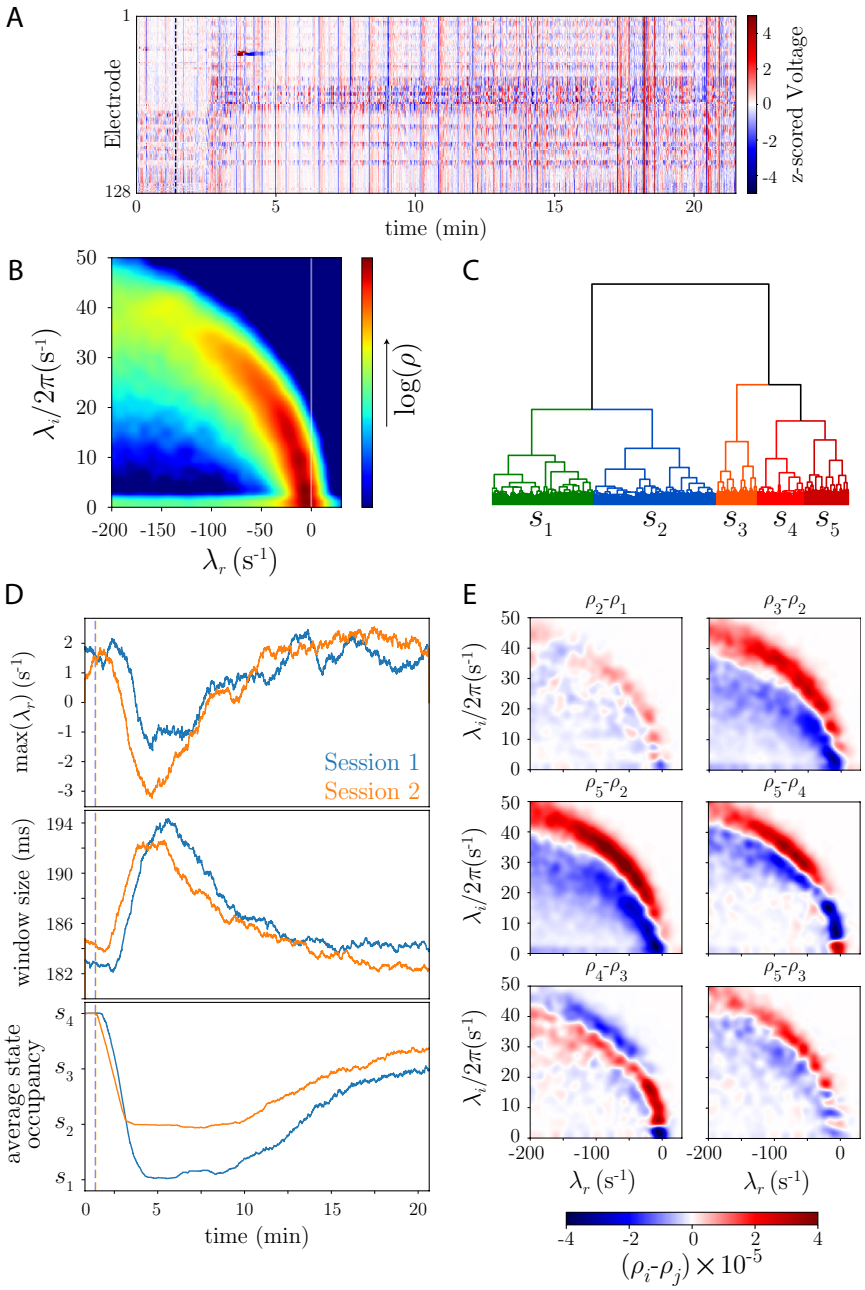

Fig. S6. Locally-linear analysis in the higher-dimensional context of electrocorticography recordings from non-human primates. (A) Example traces from an experiment in which the injection of propofol (dashed line) induces anesthesia in the monkey subject. There are two trials which occurred with the same subject but on different days. We project the time series into a 40-dimensional space through principal component analysis, capturing $\sim 99\%$ of the variance. (B) Eigenvalue spectrum of the collection of linear models found through adaptive segmentation. The distribution of eigenvalues spans a wide range of frequencies and peaks near the instability boundary. (C) Dendrogram of the likelihood clustering of the space of models. (D) Propofol injection (dashed line) induces profound changes in brain dynamics. (top) The injection first results in dynamics that are increasingly unstable, then more deeply stable followed by a slow relaxation towards the instability boundary. (middle) These effects are also present in the window sizes, which increase after the injection of the anesthetic drug, reflecting a period in which the dynamics is less nonlinear. (bottom) Anesthesia also results in an abrupt change in the average state occupancy. The two different sessions differ in their average anesthetized state: while in session 1 the dynamics sits more in s_1, in session 2 we find a higher occupancy of s_2. The curves were smoothed using a 1 s running average. (E) Difference between the eigenvalue distributions of different clusters. In general, the clusters exhibit frequency dependent changes in stability. In s_1, higher frequency states are more damped, while frequencies in the δ band are long-lived. In contrast, s_2 exhibits less damped frequencies (specially in the θ and α bands). Compared to any other cluster however, both s_1 and s_2 have their high frequency dynamics significantly more damped. This loss of power in the β band has been associated with loss of consciousness (6, 7) and is naturally captured by our technique.

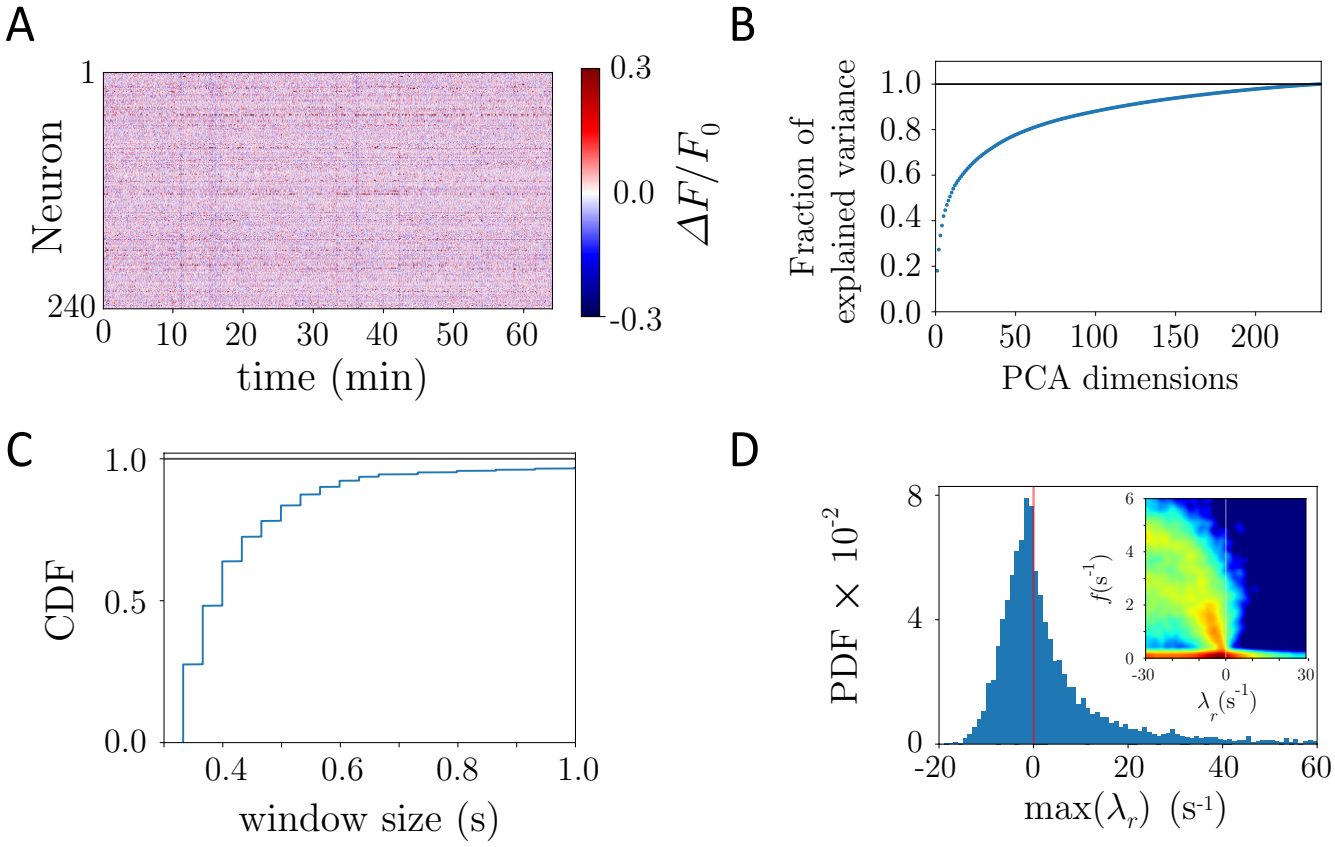

Fig. S7. Locally-linear analysis with regularization applied to recordings of *Mus musculus* visual cortex at single-cell resolution. (A) Time series of 240 neurons from mouse visual cortex under a natural movie stimulus. (B) The neural population does not appear low dimensional: the spectrum of eigenvalues of the covariance matrix indicates that in order to capture most of the variance we would need almost as many principal components as the original number of cells. (C) Extending the original locally-linear analysis to include a regularization step, the inferred window sizes exhibit a wide distribution with heavy tails extending from $0.3\,\mathrm{s}$ to longer than $1\,\mathrm{s}$. Without regularization, the minimum window size is ~ 500 frames ($\sim 16.5\,\mathrm{s}$) in order to ensure a well-conditioned model fit. Here, we have used a condition number threshold of $\kappa_{\mathrm{thresh}} = 10^5$. Further details are given in Methods. (D) The neural dynamics exhibits a wide range of frequencies and dynamics that sit near the instability boundary.

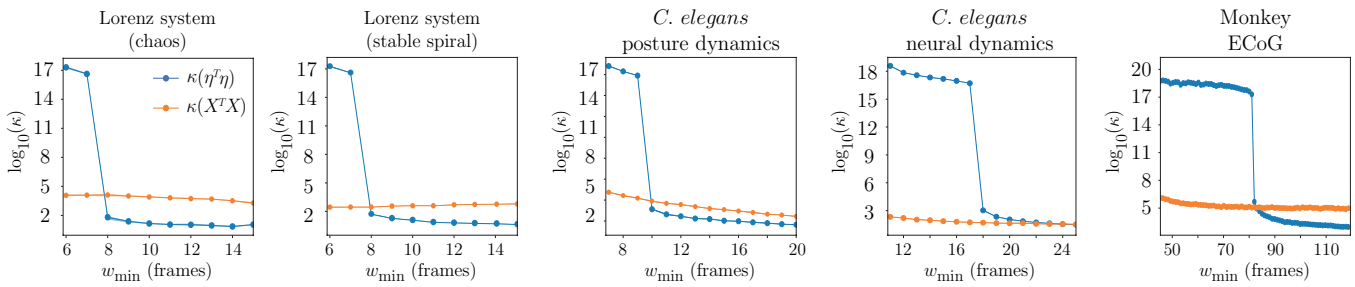

Fig. S8. Condition number of the moment matrix $\mathbf{X}^\top\mathbf{X}$ and the error covariance matrix $\eta^\top\eta$ as a function of window size. We select the minimum window size as the smallest number of frames for which the model fit and log-likelihood estimation are well-conditioned. The condition number of $X^\top X$ and $\eta^\top\eta$ is calculated for different window sizes, and the median is estimated across samples taken randomly at different times in the time series. There is a drastic decrease in the condition number beyond a minimum window whose size depends on the data. Beyond this window, the model fit and log-likelihood estimation are well-conditioned.

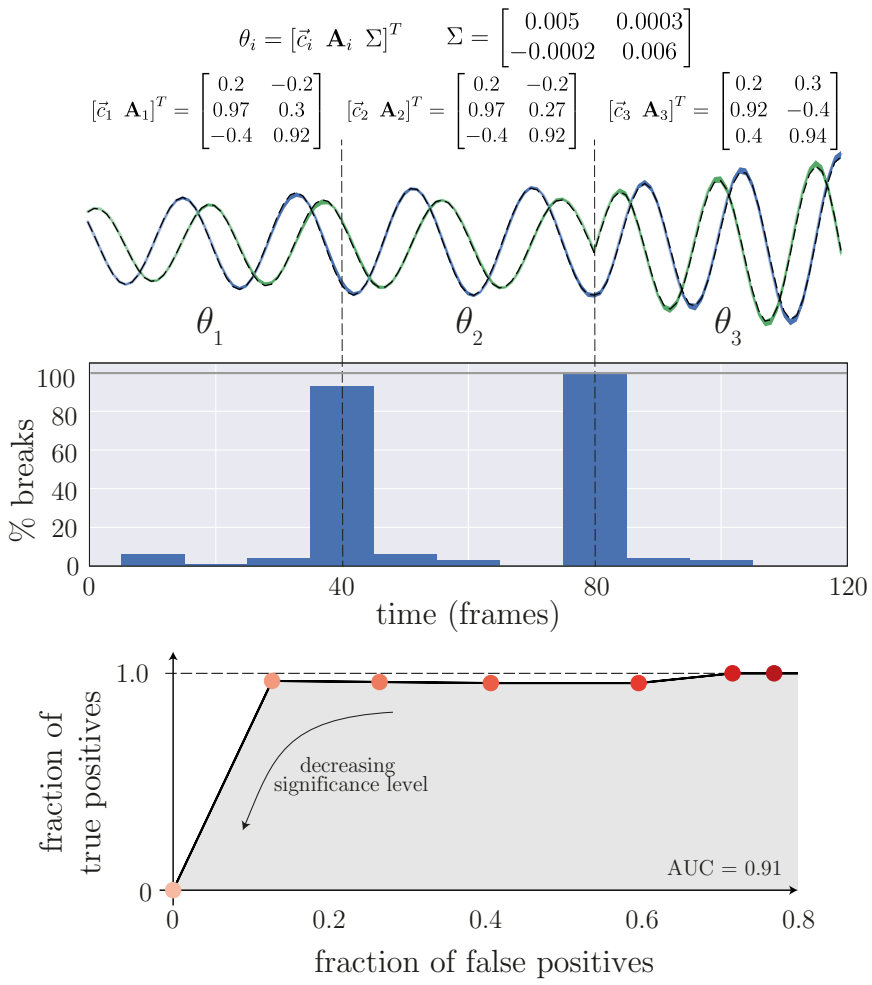

Fig. S9. Accuracy of the adaptive segmentation technique on a 3-state toy example. (top) We generate a set of 100 toy time series, for which there are three dynamical regimes, $\{\theta_1, \theta_2, \theta_3\}$. We plot a sample time series (dashed line) as well as the mean of the simulations made from the models and windows found using the adaptive segmentation technique (blue and green lines represent x_1 and x_2, respectively; shade represents bootstrapped 95% confidence intervals). (middle) We plot the distribution of dynamical breaks, across simulations, for the smallest significance level $P_{\text{null}}(\Lambda_{\text{thresh}}) = 0.01$, for which both breaks are found with high accuracy. (bottom) Fraction of true positives (breaks found by the algorithm that correspond to dynamical changes) and fraction of false positives (breaks that the algorithm found that do not correspond to dynamical changes) as a function of the significance level: $\{60\%, 40\%, 20\%, 10\%, 5\%, 2.5\%, 1\%\}$ (dark red to light red represents decreasing significance levels). At high significance levels, the segmentation algorithm is very sensitive and thus the null hypothesis is rejected easily resulting in a large amount of false positives. As we decrease the significance level, we start rejecting the null hypothesis less, while still capturing the true dynamical changes. For significance levels below 5%, the fraction of false positives drops below 50% while the fraction of true positives remains close to 100%. Indeed, the area under the curve (AUC) is nearly 1 and this is indicative of the quality of the segmentation.

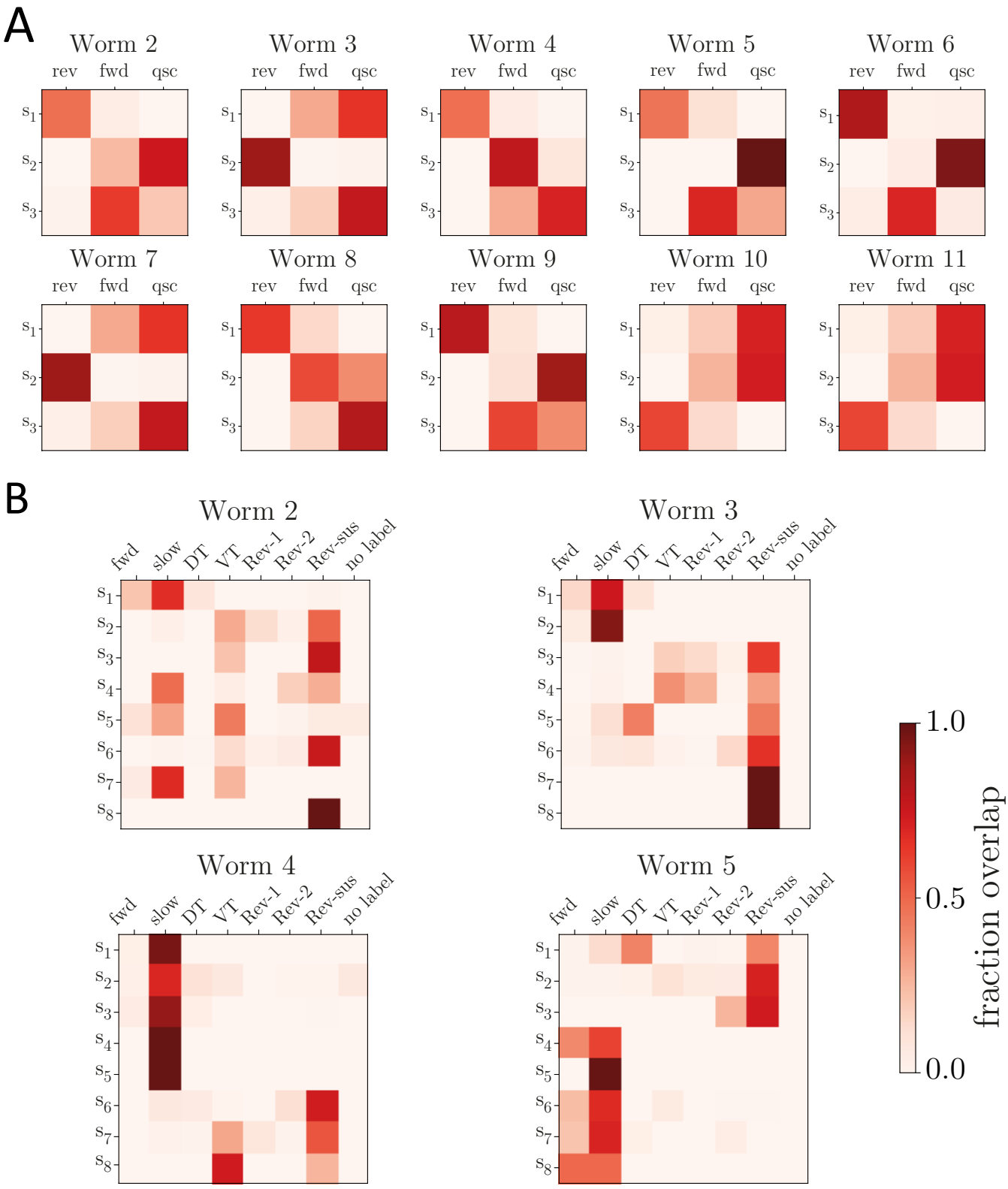

Fig. S10. Confusion matrices for all the worms not shown in Fig. (S5). (A) - Induced quiescence experiments (4). As with the worm in Fig. (S5), we had to remove an outlier in worm 4. (rev - reversal, fwd - forward, qsc - quiescent) (B) - No stimulus experiments (5). The sparsity of the confusion matrices indicates a large degree of overlap. (fwd - forward, slow - slow forward, DT - dorsal turn, VT - ventral turn, Rev-1 - reversal 1, Rev-2 - reversal 2, Res-sus - sustained reversal)

References

1. Broekmans OD, Rodgers JB, Ryu WS, Stephens GJ (2016) Resolving coiled shapes reveals new reorientation behaviors in *C. elegans*. *eLife* 5(e17227).
2. Jones E, Oliphant T, Peterson P, et al. (2001–) SciPy: Open source scientific tools for Python.
3. Rossum G (1995) Python reference manual, (Amsterdam, The Netherlands, The Netherlands), Technical report.
4. Nichols ALA, Eichler T, Latham R, Zimmer M (2017) A global brain state underlies c. elegans sleep behavior. *Science* 356(6344).
5. Kato S, et al. (2015) Global brain dynamics embed the motor command sequence of Caenorhabditis elegans. *Cell* 163:1–14.
6. Ishizawa Y, et al. (2016) Dynamics of Propofol-Induced Loss of Consciousness Across Primate Neocortex. *Journal of Neuroscience* 36(29):7718–7726.
7. Chauvette S, Crochet S, Volgushev M, Timofeev I (2011) Properties of slow oscillation during slow-wave sleep and anesthesia in cats. *Journal of Neuroscience* 31(42):14998–15008.