



OPEN

A flexible Bayesian framework for unbiased estimation of timescales

Roxana Zeraati ^{1,2}, Tatiana A. Engel ^{3,6} and Anna Levina ^{2,4,5,6}

Timescales characterize the pace of change for many dynamic processes in nature. They are usually estimated by fitting the exponential decay of data autocorrelation in the time or frequency domain. Here we show that this standard procedure often fails to recover the correct timescales due to a statistical bias arising from the finite sample size. We develop an alternative approach to estimate timescales by fitting the sample autocorrelation or power spectrum with a generative model based on a mixture of Ornstein–Uhlenbeck processes using adaptive approximate Bayesian computations. Our method accounts for finite sample size and noise in data and returns a posterior distribution of timescales that quantifies the estimation uncertainty and can be used for model selection. We demonstrate the accuracy of our method on synthetic data and illustrate its application to recordings from the primate cortex. We provide a customizable Python package that implements our framework via different generative models suitable for diverse applications.

Dynamic changes in many stochastic processes occur over typical periods known as timescales. Accurate measurements of timescales from experimental data are necessary to uncover the mechanisms controlling the dynamics of underlying processes and reveal their function^{1–5}. The timescales of a stochastic process are defined by the exponential decay rates of its autocorrelation function. Accordingly, timescales are usually estimated by fitting the autocorrelation of a sample time-series with exponential decay functions^{1,5–15}, or fitting the shape of the sample power spectral density (PSD) with a Lorentzian function^{3,16}.

However, the sample autocorrelation values computed from a finite time-series systematically deviate from the true autocorrelation^{17–22}. This bias is exacerbated by the lack of independence between the samples, especially when the timescales are large and the time-series is short^{23–25}. The magnitude of bias generally depends on the length of the sample time-series, but also on the value of the true autocorrelation at each time-lag. The expected value and variance of the autocorrelation bias can be derived analytically in simple cases (for example, the single-timescale Markov process)^{17,20}, but become intractable for more general processes with multiple timescales or an additional temporal structure. Moreover, as the bias depends on the true autocorrelation itself, which is unknown, it cannot be easily corrected.

The statistical bias deforms the shape of empirical autocorrelations and—according to the Wiener–Khinchin theorem²⁶—PSDs; it can therefore affect the timescales estimated by direct fitting methods. Fitting the sample autocorrelation of an Ornstein–Uhlenbeck (OU) process with an exponential decay function results in systematic errors in the estimated timescale and confidence interval²⁷. It is possible to fit the time-series directly with an autoregressive model to avoid these errors, without using autocorrelation or PSD^{27,28}. However, the advantage of autocorrelation is that factors unrelated to the dynamics of the processes under study (for example, slow activity drifts) can be efficiently removed with resampling methods^{29–32}. By contrast, accounting for irrelevant factors in the

autoregressive model requires the addition of components that must all be fitted to the data. Fitting the summary statistic such as autocorrelation or PSD is therefore attractive, but how the statistical bias affects the estimated timescales was not studied systematically.

We show that large systematic errors in estimated timescales arise from the statistical bias due to a finite sample size. To correct for the bias, we develop a flexible computational framework based on adaptive approximate Bayesian computations (aABCs) that estimate timescales by fitting the autocorrelation or PSD with a generative model. The aABC algorithm approximates the multivariate posterior distribution of parameters of a generative model using population Monte Carlo sampling³³. The posterior distributions can be used for quantifying the estimation uncertainty and model selection. Our computational framework can be adapted to various types of data and can find broad applications in neuroscience, physics and other fields.

Results

Bias in timescales estimated by direct fitting. Timescales of a stochastic process $A(t')$ are defined by the exponential decay rates of its autocorrelation function, that is, the correlation between values of the process at time points separated by a time lag t . For stationary processes, the autocorrelation function only depends on t :

$$AC(t) = \frac{E[(A(t') - \mu)(A(t' + t) - \mu)]_{t'}}{\sigma^2}. \quad (1)$$

Here μ and σ^2 are the mean and variance of the process, respectively, and $E[\cdot]_{t'}$ is the expectation over t' . For empirical data, the true values of μ and σ are unknown. Hence, several estimators of the sample autocorrelation were proposed that use different estimators for the sample mean $\hat{\mu}$ and sample variance $\hat{\sigma}^2$ (see the “Computing sample autocorrelation” section in the Methods)^{17,18}. The sample autocorrelation can also be computed as the inverse Fourier transform of the PSD on the basis of the Wiener–Khinchin theorem²⁶. However, for

¹International Max Planck Research School for the Mechanisms of Mental Function and Dysfunction, University of Tübingen, Tübingen, Germany. ²Max Planck Institute for Biological Cybernetics, Tübingen, Germany. ³Cold Spring Harbor Laboratory, Cold Spring Harbor, NY, USA. ⁴Department of Computer Science, University of Tübingen, Tübingen, Germany. ⁵Bernstein Center for Computational Neuroscience Tübingen, Tübingen, Germany. ⁶These authors contributed equally: Tatiana A. Engel and Anna Levina. ✉e-mail: engel@cshl.edu; anna.levina@uni-tuebingen.de

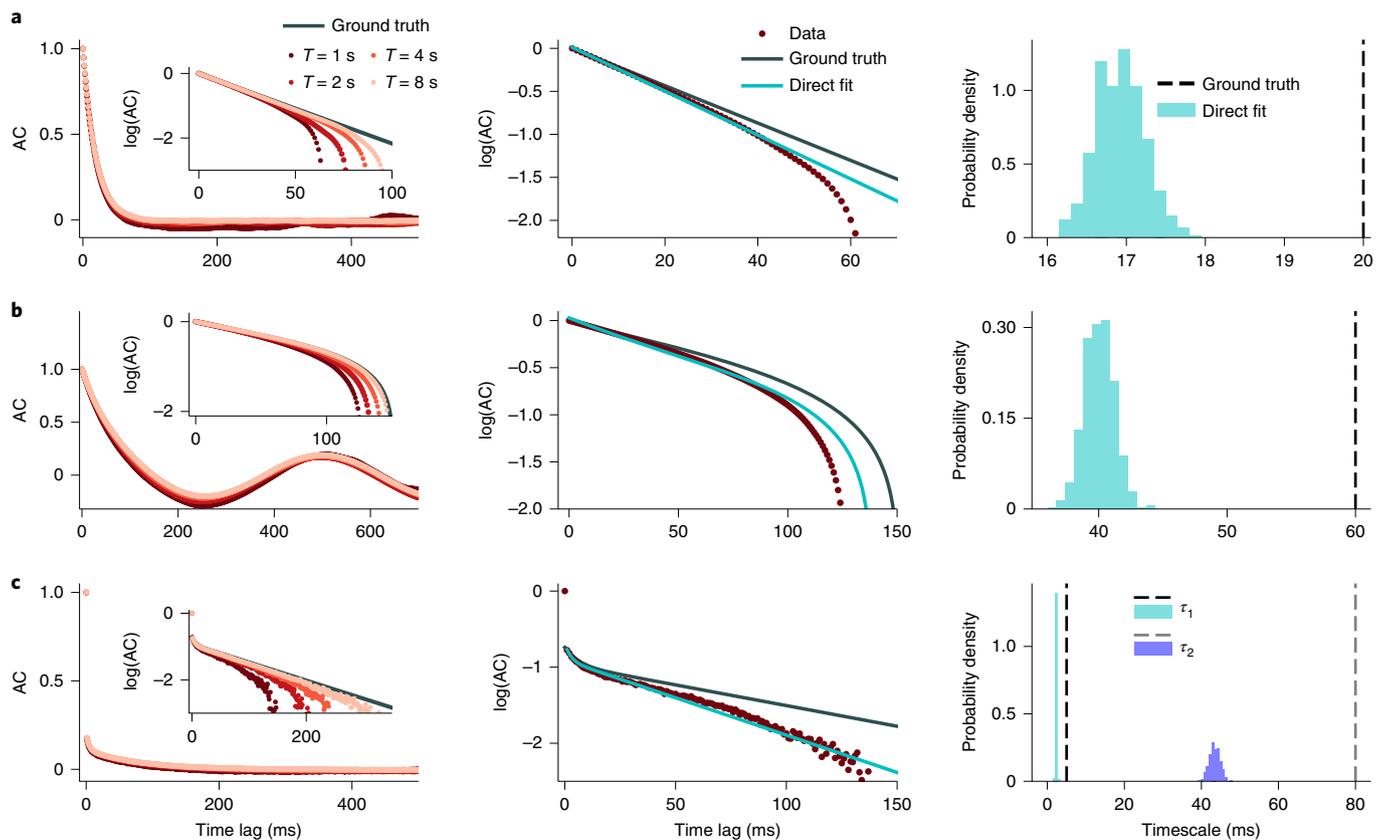


Fig. 1 | Bias in sample autocorrelations and timescales estimated by direct exponential fitting. **a**, Data are generated from an OU process with the timescale $\tau=20$ ms. Left: the sample autocorrelations (colored dots) systematically deviate from the ground-truth autocorrelation. The shape of sample autocorrelation depends on the time-series duration (T) approaching the ground truth with increasing T (the inset shows a close up on a logarithmic-linear scale). Middle: direct fitting of the analytical autocorrelation function to a sample autocorrelation ($T=1$ s) does not recover the ground-truth autocorrelation shape. Right: the ground-truth timescales largely deviate from the distribution of timescales estimated by direct exponential fitting across 500 independent realizations of the same process. **b**, Same as **a** for data from a linear mixture of an OU process with $\tau=60$ ms and an oscillation at frequency $f=2$ Hz. **c**, Same as **a** for data from an inhomogeneous Poisson process, with the instantaneous rate generated from a linear mixture of two OU processes with $\tau_1=5$ ms and $\tau_2=80$ ms. All simulation parameters are provided in Supplementary Table 1.

any of these methods, the sample autocorrelation is a biased estimator: the sample autocorrelation values for a finite-length time-series systematically deviate from the ground-truth autocorrelation^{17–22} (Fig. 1 and Supplementary Fig. 1). This statistical bias deforms the shape of the sample autocorrelation or PSD and may affect the estimation of timescales by direct fitting of the shape.

To investigate the impact of autocorrelation bias on the timescales estimated by direct exponential fitting, we tested how accurately this procedure recovers the correct timescales on synthetic data for which the ground truth is known. We generated synthetic data from several stochastic processes for which the autocorrelation function can be computed analytically (see the “Generative models” section in the Methods). The exponential decay rates of the analytical autocorrelation provide the ground-truth timescales. We considered three ground-truth processes that differed by the number of timescales, an additional temporal structure, and noise: (1) an OU process (Fig. 1a); (2) a linear mixture of an OU process and oscillatory component (Fig. 1b, resembling the firing rate of a neuron modulated by a slow oscillation^{34,35}); and (3) an inhomogeneous Poisson process (often used to model spiking activity of neurons³⁶) with the instantaneous rate modeled as a linear mixture of two OU processes with different timescales (Fig. 1c).

For all three processes, the sample autocorrelation exhibits a negative bias, that is, the values of the sample autocorrelation are systematically below the ground-truth autocorrelation function

(Fig. 1, left). This bias is clearly visible in the logarithmic-linear scale, in which the ground-truth exponential decay turns into a straight line. The sample autocorrelation deviates from the straight line and even becomes systematically negative at intermediate lags (and hence is not visible on the logarithmic scale) for processes with a strictly positive ground-truth autocorrelation (Fig. 1a,c). The deviations from the ground truth are larger when the timescales are longer or when multiple timescales are involved. The negative bias decreases for longer trial durations (Fig. 1, left inset) but it is still substantial for realistic trial durations such as in neuroscience data.

Due to the negative bias, a direct fit of the sample autocorrelation with the correct analytical function cannot recover the ground-truth timescales (Fig. 1, middle, right; Supplementary Figs. 2 and 3). When increasing the duration of each trial, the timescales obtained from the direct fits become more closely aligned with the ground-truth values (Supplementary Fig. 4). This observation indicates that timescales estimated from datasets with different trial durations cannot be directly compared, as differences in the estimation bias may result in misleading interpretations. Direct fitting of a sample autocorrelation—even with a known correct analytical form—is therefore not a reliable method for measuring timescales in experimental data. Bias-correction methods based on parametric bootstrapping can mitigate the estimation bias of timescales from direct fits³⁷. However, as the amount of bias depends on the ground-truth

timescales, such corrections cannot guarantee accurate estimates in all cases (Supplementary Fig. 5).

Alternatively, we can estimate timescales in the frequency domain by fitting the PSD shape with a Lorentzian function, equation (5), which is the ground-truth PSD for a stochastic process with an exponentially decaying autocorrelation^{3,16}. Comparison of the ground-truth PSD with the sample PSD of an OU process with finite duration reveals that the statistical bias also persists in the frequency domain (Supplementary Fig. 6a). Due to this bias, the estimated timescale deviates from the ground truth by the amount that depends on the fitted frequency range. Although careful choice of the fitted frequency range can improve the estimation accuracy (Supplementary Fig. 6b), without knowing the ground-truth timescale, there is no principled way to choose the correct range for all cases. Slightly changing the fitted frequency range can produce large errors in estimated timescale, especially in the presence of additional noise, for example, spiking activity.

Estimating timescales by fitting generative models with aABC.

As direct fitting methods cannot estimate timescales reliably, we developed an alternative computational framework based on fitting the sample autocorrelation (or PSD) with a generative model. Using a model with known ground-truth timescales, we can generate synthetic data that match the essential statistics of the observed data, that is, with the same duration and number of trials, mean and variance. Hence the sample autocorrelations (or PSDs) of the synthetic and observed data will be affected by a similar statistical bias when their shapes match. We chose a linear mixture of OU processes (one for each estimated timescale) as a generative model, which, if necessary, could be augmented with an additional temporal structure (such as oscillations) and noise. The advantage of using a mixture of OU processes is that the analytical autocorrelation function of this mixture explicitly defines the timescales. We set the number of components in the generative model according to our hypothesis on the autocorrelation in the data, for example, the number of timescales, additional temporal structure and noise (see the “Generative models” section in the Methods). We then optimize parameters of the generative model to match the shape of the autocorrelation (or PSD) between the synthetic and observed data. The timescales of the optimized generative model provide an unbiased estimation of timescales in the data.

Calculating the likelihood for complex generative models can be computationally expensive or even intractable. We therefore optimize the generative model parameters using aABC (Fig. 2)³³, an iterative algorithm that minimizes the distance between the summary statistic of synthetic and observed data (Fig. 2 and the “Optimizing generative model parameters with aABC” section in the Methods). Depending on the application, we can choose a summary statistic in the time (autocorrelation) or frequency (PSD) domains. On each iteration, we draw a set of parameters of the generative model, either from a prior distribution (first iteration) or a proposal distribution, and generate synthetic data using these parameter samples. The sampled parameters are accepted if the distance d between the summary statistic of the observed and synthetic data is smaller than a selected error threshold ϵ . The last set of accepted parameter samples provides an approximation of the posterior distribution. The joint posterior distribution quantifies the estimation uncertainty taking into account the stochasticity of the observed data (Supplementary Fig. 7). We implemented this algorithm in the abcTau Python package including different types of summary statistics and generative models (see the “abcTau Python package” in the Methods). We marginalize the multivariate posterior distribution over all other parameters of the generative model to visualize the posterior distribution of a parameter (such as a timescale).

We illustrate our method on synthetic data from the processes described in the previous section, using the autocorrelation as the

summary statistic (compare Fig. 3 with Fig. 1). For all three processes, the shape of the sample autocorrelation of the observed data is accurately reproduced by the autocorrelation of synthetic data generated using the maximum a posteriori (MAP) estimate of parameters from the joint multivariate posterior distribution (Fig. 3, left). The posterior distributions inferred by aABC include the ground-truth timescales (Fig. 3, middle). The posterior variance quantifies the estimation uncertainty. In our simulations, the number of trials controls the signal-to-noise ratio in sample autocorrelation and consequently the width of the posteriors (Supplementary Fig. 8).

The aABC method also recovers the ground-truth timescales by fitting the sample PSD without tuning the frequency range, even in the presence of multiple timescales, additional spiking noise (Supplementary Fig. 9) or multiple oscillatory components (Supplementary Fig. 10). Moreover, the aABC method can uncover slow oscillations in signals that do not exhibit clear peaks in PSD due to the short duration of the time-series and the low frequency resolution (Supplementary Fig. 9c). The aABC method can be used in combination with any method for computing the PSD (for example, any window functions for removing the spectral leakage), as the exact same method applied to synthetic data would alter their sample PSD in the same way.

Different summary statistics and fitting ranges may be preferred depending on the application. For example, autocorrelations allow for using additional correction methods such as jittering³¹, whereas PSD estimation can be improved with filtering or multitapers³⁸. Furthermore, selecting a smaller maximum time-lag when fitting autocorrelations prevents overfitting to noise in the autocorrelation tail. The choice of summary statistic (metric and fitting range) can influence the shape of the approximated posterior (for example, posterior width), but the posteriors peak close to the ground-truth timescales as long as the same summary statistic is used for observed and synthetic data (Supplementary Fig. 11). The advantage of such behavior is particularly visible when changing the range over which we compute the summary statistics (for example, minimum and maximum frequency in PSD), which affects direct fit but not aABC estimates (Supplementary Fig. 6).

In the aABC algorithm we set a multivariate uniform prior distribution over the parameters of the generative model. The ranges of uniform prior should be broad enough to include the ground-truth values. Selecting broader priors does not affect the shape of posteriors and only slows down the fitting procedure (Supplementary Fig. 12). Hence we can set wide prior distributions when a reasonable range of parameters is unknown. Timescales estimated from the direct exponential fits can be used as a potential lower bound for the priors.

We evaluated the reliability of the aABC method on a wide range of timescales and trial durations and compared the results to direct fitting (Supplementary Fig. 13). The estimation error of direct fitting increases when trial durations become short relative to the timescale, whereas the aABC method always returns reliable estimates. However, the estimation of the full posterior comes at a price of higher computational costs (Supplementary Table 3) than point estimates; thus, the direct fit of the sample autocorrelation may be preferred when the long time-series data are available so that statistical bias does not corrupt the results. To empirically verify whether this is the case, we implemented a pre-processing algorithm in our Python package that uses parametric bootstrapping to return an approximate error bound of the direct fit estimates (see Supplementary Fig. 14 and the “Pre-processing for evaluating quality of direct” section in the Methods).

Fitting generative models with the aABC method provides a principled framework for estimating timescales that can be used with different metrics in time or frequency domain. Furthermore, joint posterior distribution of the inferred parameters allows us to

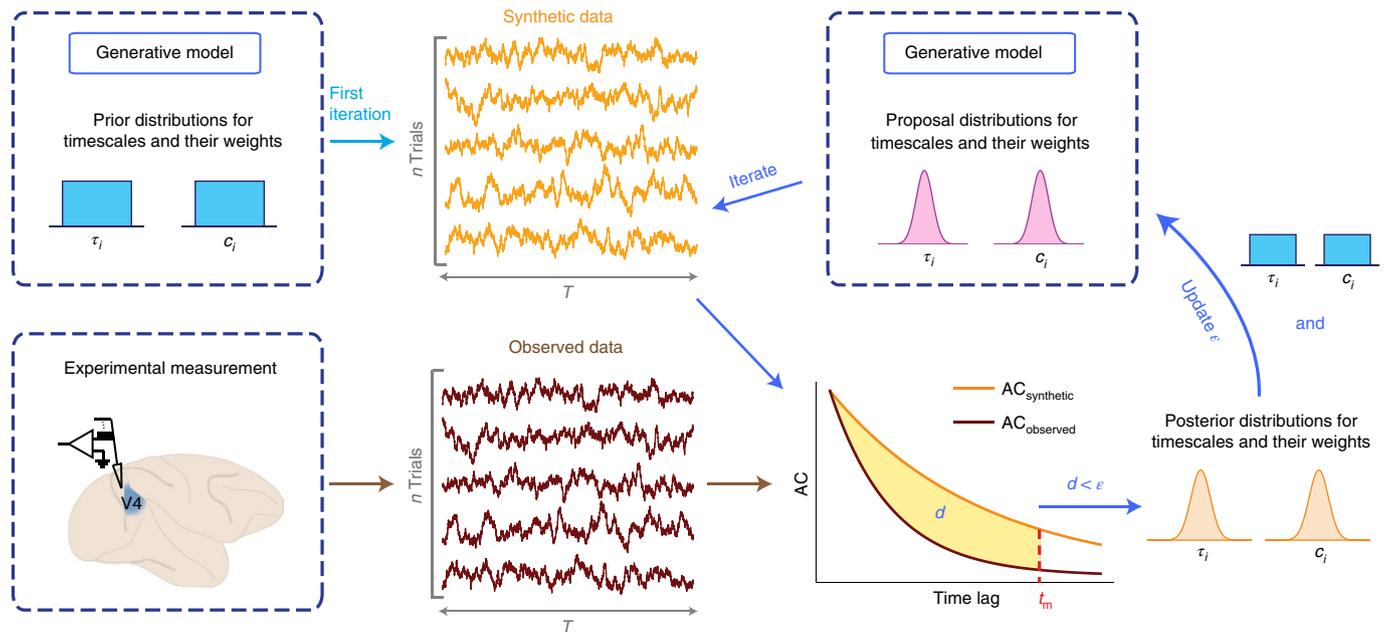


Fig. 2 | Estimation of timescales with adaptive approximate Bayesian computations. aABC estimates timescales by fitting the sample autocorrelation of observed data with a generative model. At the first iteration of the algorithm, parameters of the generative model are drawn from the multivariate prior distribution (for example, a uniform distribution; upper left). Synthetic data are generated from the generative model with these parameters. If the value of d between the autocorrelations of synthetic and observed data (computed up to maximum time-lag t_m) is smaller than a specific ϵ , these parameters are added to the multivariate posterior distribution (lower right). In subsequent iterations, new parameters are drawn from a proposal distribution, which is computed based on the posterior distribution of the previous iteration and the initial prior distribution (upper right, see Methods).

examine correlations between different parameters, find manifolds of possible solutions, and identify a potential degeneracy in the parameter space (Supplementary Fig. 15).

Estimating the timescale of activity in a branching network. So far we have demonstrated that our aABC method accurately recovers the ground-truth timescales when the generative model and the process that produced the observed data are the same. However, the inference of timescales using OU processes is broadly applicable when the mechanism that generated the exponential decay of autocorrelation in the data is not an OU process. As an example, we tested our method on discrete-time data from an integer-valued autoregressive model with a known ground-truth autocorrelation function. Specifically, we applied our method to estimate the timescale of the global activity in a branching network model, which is often used to study the operating regime of dynamics in neuronal networks (Fig. 4a)^{39–41}. We simulated the network model to generate the time-series data of the global activity. We then used aABC to fit the sample autocorrelation of these data with a one-timescale OU process as a generative model. The inferred posterior is centered on the theoretically predicted timescale of the branching network, and the MAP estimate parameter accurately reproduces the shape of the sample autocorrelation of the network activity (Fig. 4b,c). These results show that our framework can be used to estimate timescales in diverse types of data.

Model selection with ABC. The correct generative model (for example, the number of timescales) is usually unknown for experimental data. We thus need a procedure to select between alternative hypotheses. Assuming that the autocorrelation or PSD is a sufficient summary statistic for estimating timescales, we can use ABC to approximate the Bayes factor for selecting between alternative models^{42–44}. Model selection based on ABC can produce inconsistent results when the summary statistic is insufficient⁴⁵, but whether

this is likely to be the case can be verified empirically⁴⁴. Specifically, the summary statistic can be used for model selection with ABC if its mathematical expectation is significantly different for the two models.

Based on this empirical procedure⁴⁴, we developed a method to select between two alternative models M_1 and M_2 using their aABC fits (see the “Model selection and Bayes factor approximation with ABC” section in the Methods). We compare models using a goodness of fit measure computed as the distance between the summary statistic of synthetic and observed data. We estimate the Bayes factor, which is the ratio of marginal likelihoods of the two models and accounts for the model complexity⁴⁶, to select between M_1 and M_2 . Assuming both models are a priori equally probable, the Bayes factor can be approximated as the ratio between the cumulative distribution functions (CDFs) of distances for two models at different error thresholds $B_{21}(\epsilon) = \text{CDF}_{M_2}(\epsilon)/\text{CDF}_{M_1}(\epsilon)$.

We evaluated our model selection method using synthetic data from three example processes with known ground truth so that the correct number of timescales is known. We used an OU process with a single timescale (Fig. 5a) and two different examples of an inhomogeneous Poisson process with two timescales: one with well separated ground-truth timescales, such that multiple timescales are evident in the autocorrelation shape (Fig. 5d); and another with similar ground-truth timescales, such that the autocorrelation shape does not clearly suggest the number of timescales in the underlying process (Fig. 5g). For all three example processes, we fitted the data with one- (M_1) and two-timescale (M_2) generative models using aABC and selected between these models by computing the Bayes factors. The one- and two-timescale models were based on a single OU process or a linear mixture of two OU processes, respectively. For the data from inhomogeneous Poisson processes, the generative model also incorporated an inhomogeneous Poisson noise.

For the example OU process with a single timescale, the one- and two-timescale models fitted the shape of the data autocorrelation

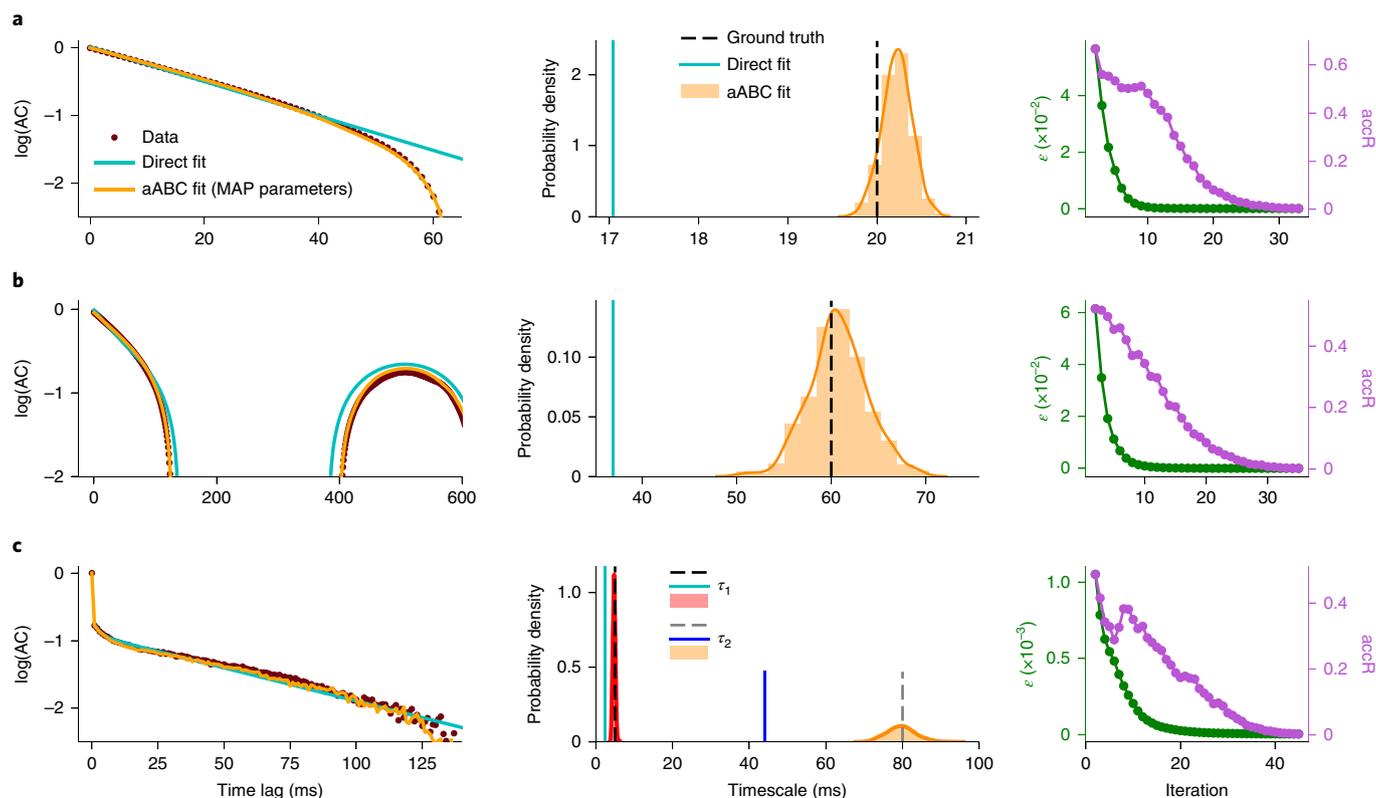


Fig. 3 | Accurate estimation of timescales and uncertainty quantification with the aABC algorithm. The same synthetic data as in Fig. 1 for $T=1$ s. **a**, Data are generated from an OU process with $\tau=20$ ms. Left: the shape of the data autocorrelation is accurately reproduced by the autocorrelation of synthetic data from the generative model with the MAP estimate parameters ($\tau_{\text{MAP}}=20.2$ ms), but it cannot be captured by the direct exponential fit. Middle: the marginal posterior distributions (histograms overlaid with Gaussian kernel smoothing) include the ground-truth timescales, whereas direct exponential fits underestimate the ground-truth timescales. The width of the posteriors indicates the estimation uncertainty. Right: the convergence of the aABC algorithm is defined based on accR, which decreases together with ϵ over the iterations of the algorithm. Data are plotted from the second iteration. The initial error threshold for all fits was set to 1. **b**, Same as **a** for data from a linear mixture of an OU process with $\tau=20$ ms and an oscillation with $f=2$ Hz; $\tau_{\text{MAP}}=60.5$ ms. **c**, Same as **a** for data from an inhomogeneous Poisson process with two timescales: $\tau_1=5$ ms and $\tau_2=80$ ms. $\tau_{1,\text{MAP}}=4.7$ ms and $\tau_{2,\text{MAP}}=80$ ms. Other parameters are provided in Supplementary Tables 1 and 2.

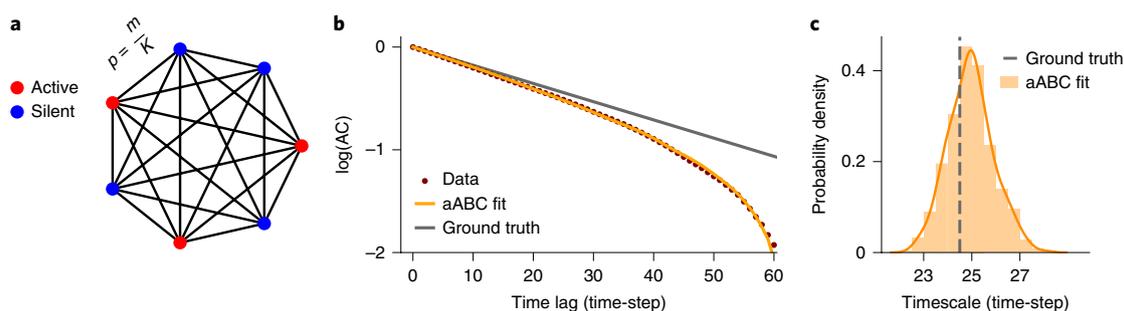


Fig. 4 | aABC inference with a generative model based on the OU process in a branching network. **a**, A schematic of a fully connected branching network with binary neurons. The branching parameter ($m=0.96$) and number of neurons ($K=10^4$) define the probability of activity propagation (p). Each neuron receives an external input with probability $h=10^{-3}$. **b**, The shape of the sample autocorrelation of simulated activity in the branching network deviates from the ground truth, but is accurately reproduced by the autocorrelation of synthetic data generated from a one-timescale OU process with the MAP estimate timescale from the aABC method. $\tau_{\text{MAP}}=24.9$. The data contained 100 trials of 500 time-steps. **c**, The aABC posterior distribution includes the ground-truth timescale. $\tau_{\text{groundtruth}}=24.5$, $\tau_{\text{aABC}}=24.9 \pm 0.9$ (mean \pm s.d.). Fitting parameters are provided in Supplementary Table 2.

almost equally well (Fig. 5a). The marginal posterior distributions of the two timescales estimated by the two-timescale model heavily overlap around their peak values (Fig. 5b), which indicates that the one-timescale model possibly better describes the data. For the two-timescale model we enforce the ordering $\tau_1 < \tau_2$, which generates

the difference between their distributions. We compare the CDFs of distances to select between the two models (Fig. 5c). Although the two-timescale model has more parameters, it has significantly larger distances than the one-timescale model (Wilcoxon rank-sum test, $P=0.002$; mean $d_{M_1} = 6 \times 10^{-5}$; mean $d_{M_2} = 8 \times 10^{-5}$). The

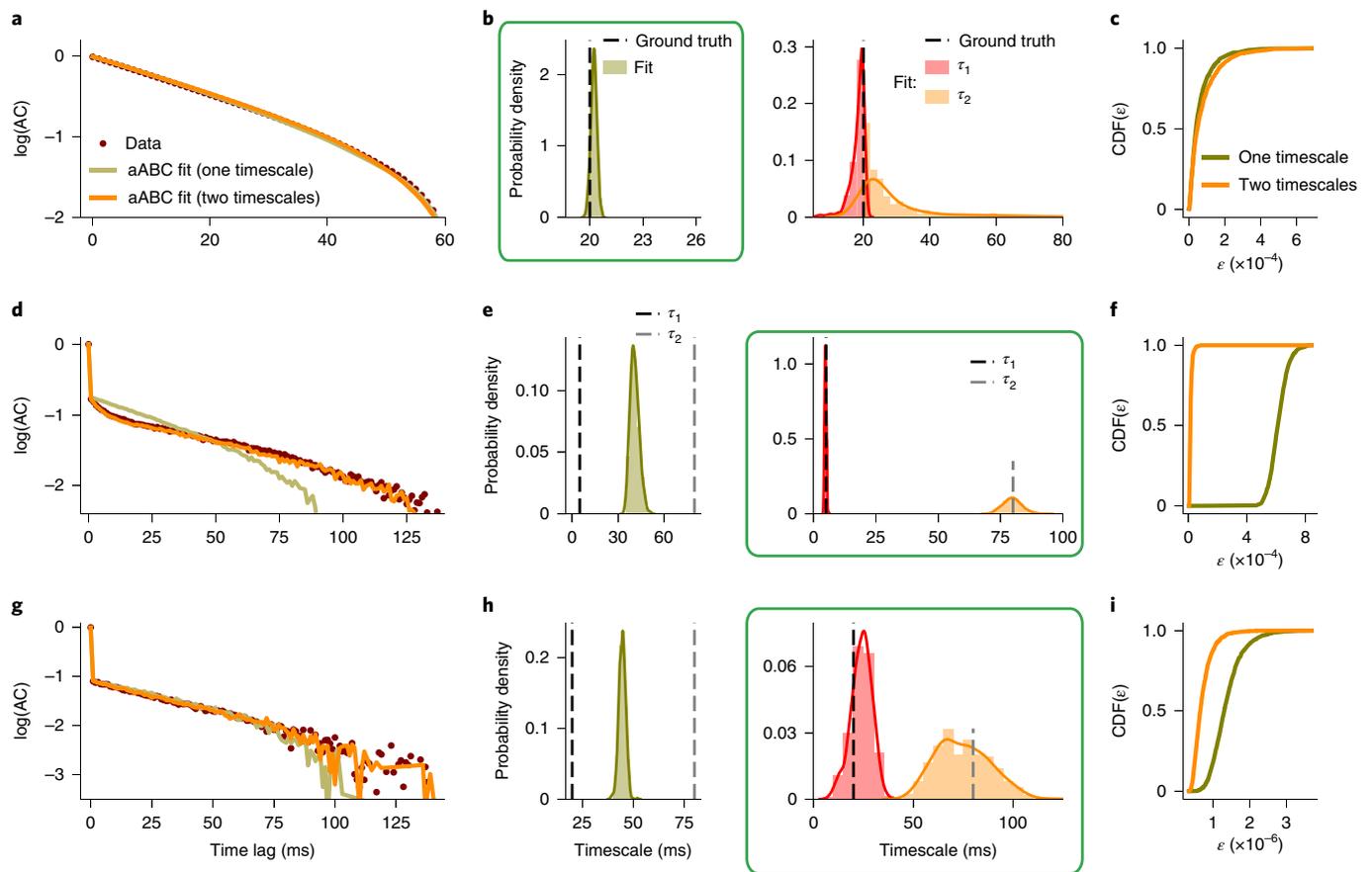


Fig. 5 | Model selection with ABC. **a–c**, Data are generated from a one-timescale OU process with $\tau = 20$ ms. **a**, The data autocorrelation is fitted with one- and two-timescale generative models (sample autocorrelations for MAP estimate parameters are shown). **b**, Marginal posterior distribution of the timescale estimated using the one-timescale model includes the ground truth and has small variance (left). Marginal posterior distributions of two timescales estimated using the two-timescale model heavily overlap and have large variance (right). **c**, Cumulative distributions of d for two models, $CDF_{M_1}(\epsilon)$. As M_1 resulted in smaller distances ($P = 0.002$, $CL = 0.54$, where CL is the effect size) and $CDF_{M_2}(\epsilon) < CDF_{M_1}(\epsilon)$ for all ϵ , the one-timescale model is selected (green box in **b**). **d–f**, Data are generated from an inhomogeneous Poisson process with two timescales: $\tau_1 = 5$ ms and $\tau_2 = 80$ ms. **d**, Same format as **a**. **e**, Marginal posterior distributions for two timescales estimated by the two-timescale model include the ground truth (right). Marginal posterior distributions for the timescale estimated by the one-timescale model falls in between the ground-truth timescales (left). **f**, Cumulative distributions of d for two models. As M_2 resulted in smaller distances ($P < 10^{-10}$, $CL = 1$) and $CDF_{M_2}(\epsilon) > CDF_{M_1}(\epsilon)$ for all ϵ , the two-timescale model is selected (green box in **e**). **g–i**, Same as **d–f** but for $\tau_1 = 20$ ms and $\tau_2 = 80$ ms ($P < 10^{-10}$, $CL = 0.93$). For all fits, $CDF_{M_i}(\epsilon)$ values are computed from $n_1 = n_2 = 1,000$ samples. P -values and CL values are computed from the two-sided Wilcoxon rank-sum test. Simulation and fitting parameters are provided in Supplementary Tables 1 and 2.

two-timescale model has a larger average distance as its posterior distribution has larger variance, which leads to a greater chance to sample a combination of parameters with a larger distance. As $CDF_{M_2}(\epsilon) < CDF_{M_1}(\epsilon)$ (that is, $B_{21}(\epsilon) < 1$) for all ϵ , the one-timescale model is preferred over the two-timescale model, in agreement with the ground-truth generative process.

For both example inhomogeneous Poisson processes with two timescales, the shape of the data autocorrelation is better matched by the two- rather than one-timescale model (Fig. 5d,g; the difference is subtle in Fig. 5g). The marginal posterior distributions of two timescales estimated by the two-timescale model are clearly separated and include the ground-truth values, whereas the timescale estimated by the one-timescale model is in between the two ground-truth values (Fig. 5e,h). The two-timescale model has significantly smaller distances (Fig. 5f, $P < 10^{-10}$, mean $d_{M_1} = 6 \times 10^{-4}$; mean $d_{M_2} = 1.5 \times 10^{-5}$; Fig. 5i, $P < 10^{-10}$, mean $d_{M_1} = 10^{-6}$, mean $d_{M_2} = 7 \times 10^{-7}$). As $CDF_{M_2}(\epsilon) > CDF_{M_1}(\epsilon)$ (that is, $B_{21}(\epsilon) > 1$) for all ϵ , the two-timescale model provides a better description of the data for both examples, in agreement with the ground truth. Our

method thus selects the correct generative model even for a challenging case where the shape of the data autocorrelation does not suggest the existence of multiple timescales. Our method can be used to discriminate between a broad class of models, for example, slow periodic (oscillation) versus aperiodic (exponential decay) components in the time-series (Supplementary Fig. 16).

Estimating timescales of ongoing neural activity. To illustrate an application of our framework to experimental data, we estimated the timescales of ongoing spiking activity in the primate visual cortex during fixation on a blank screen⁴⁷. We computed the autocorrelation of the population spiking activity pooled across 16 recording channels (see the “Neural recordings and analysis” section in the Methods). Autocorrelation of neural activity in several brain regions was previously modeled as an exponential decay with a single timescale¹. To determine whether a single timescale is sufficient to describe the temporal dynamics of neural activity in our data, we fitted M_1 and M_2 using aABC and selected the model that better described the data (Fig. 6a–c). We used a doubly stochastic

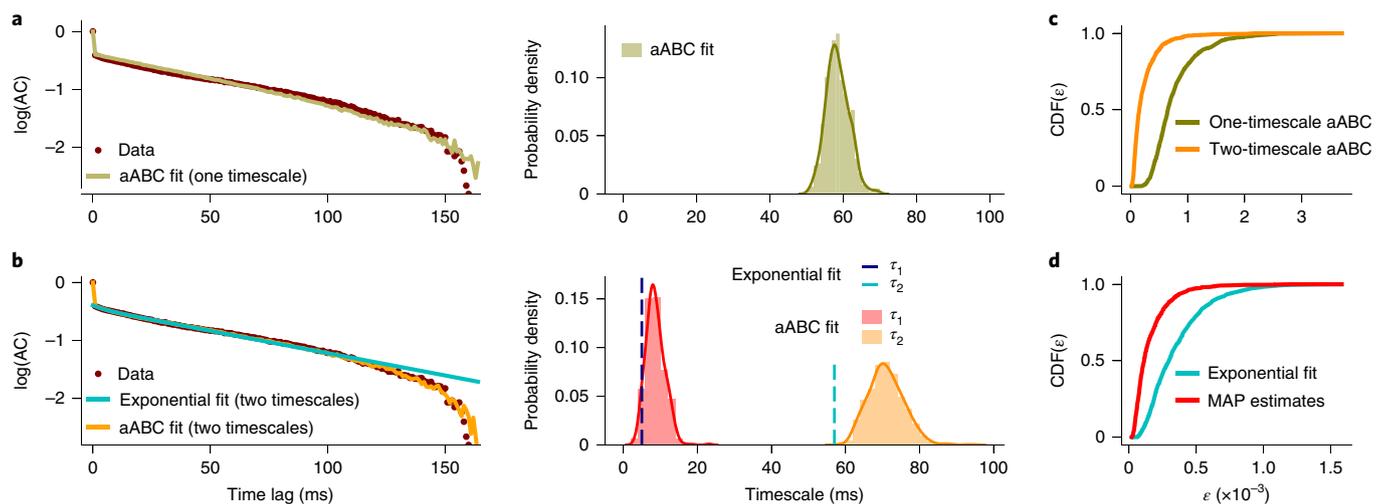


Fig. 6 | Estimating timescales of ongoing neural activity and comparing hypotheses about their number with aABC. **a**, Left: the autocorrelation of neural spike-counts is fitted with a one-timescale doubly stochastic model using aABC (sample autocorrelation for MAP estimate parameters). Right: posterior distribution of the timescale. $\tau_{\text{MAP}} = 58$ ms. **b**, Left: same data as in **a** is fitted directly with a double exponential function and with a two-timescale doubly stochastic model using aABC (sample autocorrelation for MAP estimate parameters). Right: timescales estimated by the direct exponential fit and marginal posterior distributions of timescales inferred with aABC. $\tau_{1,\text{exp}} = 5$ ms and $\tau_{2,\text{exp}} = 57$ ms; $\tau_{1,\text{MAP}} = 8$ ms and $\tau_{2,\text{MAP}} = 70$ ms. **c**, Cumulative distribution of d for M_1 and M_2 . As M_2 resulted in smaller distances ($P < 10^{-10}$, CL = 0.92, $n_1 = n_2 = 1,000$) and $\text{CDF}_{M_2}(\epsilon) > \text{CDF}_{M_1}(\epsilon)$ for all ϵ , the two-timescale model is selected. **d**, Cumulative distributions of d between the autocorrelation of neural data and synthetic data from the two-timescale doubly stochastic model with parameters either from the direct fit or MAP estimate with aABC. The MAP parameters have smaller distances ($P < 10^{-10}$, CL = 0.82, $n_1 = n_2 = 1000$), that is, they describe the autocorrelation of neural data more accurately than the direct fit. P -values and CL values are computed from the two-sided Wilcoxon rank-sum test. Fitting parameters are provided in Supplementary Table 2.

process as a generative model^{36,48}, in which spike-counts in each time bin are generated from an instantaneous firing rate modeled as one OU process (M_1) or a mixture of two OU processes (M_2). To account for the non-Poisson statistics of the spike-generation process⁴⁹, we sampled spike-counts from a gamma distribution (see the “Doubly stochastic process with multiple timescales” section in the Methods). The two-timescale model provided a better description of the data, as it had smaller distances and $\text{CDF}_{M_2}(\epsilon) > \text{CDF}_{M_1}(\epsilon)$ for all error thresholds (Fig. 6c, $P < 10^{-10}$; mean $d_{M_1} = 8 \times 10^{-4}$; mean $d_{M_2} = 2 \times 10^{-4}$).

We further compared our method with a direct exponential fit of the sample autocorrelation, which is usually employed to infer the timescales of neural activity^{1,5–12}. We fitted the sample autocorrelation with a double exponential function and compared the result with the two-timescale aABC fit (Fig. 6b). Similar to synthetic examples, the direct fit produced timescales systematically smaller than the MAP estimates from aABC. As the ground-truth timescales are not available for biological data, we used a sampling procedure to evaluate whether the data were better described by the timescales from the direct fit or from the MAP estimates (see Fig. 6d and the “Neural recordings and analysis” section in the Methods). The results indicate that MAP estimates better capture the shape of neural data autocorrelation ($P < 10^{-10}$; mean distance of MAP parameters, 10^{-4} ; mean distance of exponential fit parameters, 3×10^{-4}). Our method thus estimates the timescales of neural activity more accurately than a direct exponential fit and allows for comparing alternative hypotheses about the underlying dynamics.

Discussion

Direct fitting the shape of sample autocorrelation or PSD often fails to recover the correct timescales. Although past work²⁷ attributed errors in the estimated timescales to fitting noise in the tail of autocorrelation, we find that the main source of error is the statistical bias in the sample autocorrelation due to finite sample size. This bias arises primarily due to the deviation of the sample mean from

the ground-truth mean. If the ground-truth mean of the process is known, using the true mean for computing the autocorrelation largely eliminates the bias. When the true mean is unknown but assumed to be the same across all trials, the bias is reduced by estimating a single sample mean from the whole dataset instead of estimating a separate mean for each trial; however, this assumption does not always hold. For example, the mean neural activity can change across trials due to changes in the behavioral state of an animal. If the assumption of a constant mean is violated in the data, estimating the sample mean from the whole dataset leads to strong distortions of the autocorrelation shape introducing additional slow timescales⁵⁰. As the bias depends on the duration of time-series, comparing timescales estimated from direct fits of experimental data with different duration can produce misleading interpretations.

We focused on exponentially decaying autocorrelations, which correspond to Lorentzian PSD with the $1/f$ power-law exponent of 2. Defining timescales based on the exponential decay rates of autocorrelation is widely used in the literature^{1,5–15} and has a clear interpretation as a timescale of a generative dynamical system. Many types of data exhibit PSD with $1/f$ exponents deviating from 2 (refs. ^{3,16,51,52}), but there is no universally accepted answer as to what is the definition of timescale for these types of data, and the nature of processes generating this behavior. A prominent hypothesis is that the $1/f$ PSD arises from a mixture of many processes with exponential autocorrelations and different timescales^{51,52}, which can be modeled as a mixture of OU processes. For example, a combination of excitatory and inhibitory synaptic currents with distinct timescales was suggested as a potential mechanism for creating different $1/f$ exponents in neural data⁵¹. However, sometimes we are interested in separating a process with a well-defined timescale from $1/f$ background. We introduced an augmented generative model to handle such cases (see the “Modeling background processes with an arbitrary PSD shape” section in the Methods), which can be used to estimate timescales in the presence of a background process with an arbitrary PSD shape (for example, $1/f$ exponents other than 2). This generative model

is agnostic to the nature of the process generating the background activity and directly models the desired PSD shape. We can use this model to simultaneously estimate the $1/f$ exponent and exponential decay timescales (Supplementary Fig. 17).

The general framework of inferring timescales with aABC using OU processes can be adapted to various data types, different generative models and summary statistics using our Python package. It provides an unbiased estimation of timescales and returns a posterior distribution that quantifies the estimation uncertainty and can be used for hypothesis testing. As estimating the full posterior is computationally expensive, we included a pre-processing algorithm that verifies whether direct-fit estimates are sufficiently reliable for given data and thus may be preferred due to lower computational costs. Furthermore, for processes with complex temporal dynamics, finding a plausible generative model that captures all aspects of the data might be challenging. Some of the challenges can be mitigated by including background processes with an arbitrary PSD shape in the generative model (see the “Modeling background processes with an arbitrary PSD shape” section in the Methods). Our method can select the best model from a proposed set of plausible models. However, the model selection may fail to detect the optimal model when the amount of data is very limited, producing an inconclusive result. This limitation can be addressed by collecting a larger dataset. The modular implementation of our Python package allows users to easily incorporate additional types of dynamics and non-stationarities into customized generative models, or use other types of summary statistics that can be added directly to the package. Our approach is particularly favorable for data organized in short trials or trials of different durations, when direct fitting is unreliable.

Methods

Computing sample autocorrelation. In experiments or simulations, the autocorrelation needs to be estimated from a finite sample of empirical data. A data sample from the process $A(t')$ constitutes a finite time-series measured at discrete times t'_i ($i = 1 \dots N$, where N is the length of the time-series). For example, the sample time-series can be spike counts of a neuron in discrete time bins or a continuous voltage signal measured at a specific sampling rate. Accordingly, the sample autocorrelation is defined for a discrete set of time lags t_j . Several estimators of the sample autocorrelation were proposed using different estimators for $\hat{\mu}$ and $\hat{\sigma}^2$ (refs. 17,18). One possible choice is:

$$\hat{AC}(t_j) = \frac{1}{\hat{\sigma}^2(N-j)} \sum_{i=1}^{N-j} (A(t'_i) - \hat{\mu}_1(j)) (A(t'_{i+j}) - \hat{\mu}_2(j)), \quad (2)$$

with $\hat{\sigma}^2 = \frac{1}{N-1} \sum_{i=1}^N (A(t'_i)^2 - \frac{1}{N} (\sum_{i=1}^N A(t'_i))^2)$ and two different sample means: $\hat{\mu}_1(j) = \frac{1}{N-j} \sum_{i=1}^{N-j} A(t'_i)$ and $\hat{\mu}_2(j) = \frac{1}{N-j} \sum_{i=j+1}^N A(t'_i)$. Different normalizations of autocorrelation are used in literature, but our results do not depend on a specific choice of normalization.

Generative models. We used several generative models based on a linear mixture of OU processes—one for each estimated timescale—sometimes augmented with additional temporal structure (for example, oscillations) and noise.

OU process with multiple timescales. An OU process is defined as

$$\dot{A}(t') = -\frac{A(t')}{\tau} + \sqrt{2D}\xi(t'), \quad (3)$$

where $\xi(t')$ is Gaussian white noise with zero mean and the diffusion parameter D sets the variance $\text{Var}[A_{\text{OU}}(t')] = D\tau$ (refs. 53,54). The autocorrelation of the OU process is an exponential decay function⁵⁵

$$AC(t) = e^{-t/\tau}. \quad (4)$$

Accordingly, the parameter τ provides the ground-truth timescale. The PSD of the OU process is a Lorentzian function

$$\text{PSD}(f) = \frac{c}{f^2 + f_{\text{knee}}^2}. \quad (5)$$

Here f is the frequency and $c = f_{\text{knee}}^2/\pi$ is the normalization constant. From the knee frequency, f_{knee} , we can compute the timescale as $\tau = (2\pi f_{\text{knee}})^{-1}$.

We define an OU process with multiple timescales $A(t')$ as a linear mixture of OU processes $A_k(t')$ with timescales τ_k , $k \in \{1, \dots, n\}$, zero mean and unit variance:

$$A(t') = \sum_{k=1}^n \sqrt{c_k} A_k(t'), \quad \sum_{k=1}^n c_k = 1, \quad c_k \in [0, 1], \quad \tau_1 < \tau_2 < \dots < \tau_n. \quad (6)$$

Here n is the number of timescales in the mixture and c_k are the mixing coefficients that set the relative weights of components without changing the total variance of the process. We simulate each OU process A_k by iterating its time-discrete version using the Euler scheme⁵⁵

$$A_k(t'_{i+1}) = \left(1 - \frac{\Delta t'}{\tau_k}\right) A_k(t'_i) + \sqrt{2D_k \Delta t'} \eta_k(t'_i), \quad (7)$$

where $\Delta t' = t'_{i+1} - t'_i$ is the discretization time-step and $\eta_k(t'_i)$ is a random number generated from a normal distribution. We set the unit variance for each OU process $\text{Var}(A_k) = D_k \tau_k = 1$ by fixing $D_k = 1/\tau_k$. The parameter vector θ for a linear mixture of n OU processes consists of $2n-1$ values: n timescales, τ_k , and $n-1$ coefficients, c_k , in equation (6).

We match the mean and variance of the multi-timescale OU process to $\hat{\mu}$ and $\hat{\sigma}^2$ of the observed data using a linear transformation:

$$A_{\text{trans}}(t') = \hat{\sigma} A(t') + \hat{\mu}. \quad (8)$$

We use the process $A_{\text{trans}}(t')$ as a generative model for data fitting and hypothesis testing (see below).

Multi-timescale OU process with oscillations. To obtain a generative model with an oscillation, we add an oscillatory component with the weight c_{n+1} to a multi-timescale OU process, equation (6):

$$A(t') = \sum_{k=1}^n \sqrt{c_k} A_k(t') + \sqrt{2c_{n+1}} \sin(\phi + 2\pi f t'), \quad \sum_{k=1}^{n+1} c_k = 1, \quad c_k \in [0, 1]. \quad (9)$$

For each trial, we draw the phase ϕ independently from a uniform distribution on $[0, 2\pi]$. We use the linear transformation in equation (8) to match the mean and variance of this generative process to the observed data.

The autocorrelation of this process with a single timescale τ is given by

$$AC(t) = (1 - c_1) e^{-t/\tau} + c_1 \cos(2\pi f t), \quad (10)$$

hence the ground-truth timescale is defined by the OU parameter τ .

For the analysis in Figs. 1b and 3b, we assumed that f is known and only fitted c_{n+1} to show that the bias of the direct fit persists even if we know the correct frequency of oscillation; f can be fitted with aABC as an additional free parameter (Supplementary Figs. 9 and 10).

Doubly stochastic process with multiple timescales. The doubly stochastic process with multiple timescales is generated in two steps: first generating a time-varying rate and then generating event counts from this rate. To generate the time-varying rate, we scale, shift and rectify a multi-timescale OU process (equation (6)) using the transformation

$$A_{\text{trans}}(t') = \max(\sigma' A(t') + \mu', 0). \quad (11)$$

The resulting rate $A_{\text{trans}}(t')$ is non-negative, and for $\mu' \gg \sigma'$ it has the mean $E[A_{\text{trans}}(t')] \approx \mu'$ and variance $\text{Var}[A_{\text{trans}}(t')] \approx \sigma'^2$. We then draw event counts s for each time bin $[t'_i, t'_{i+1}]$ from an event-count distribution $p_{\text{count}}(s|\lambda(t'_i))$, where $\lambda(t'_i) = A_{\text{trans}}(t'_i)\Delta t'$ is the mean event count and $\Delta t' = t'_{i+1} - t'_i$ is the bin size (in our simulations $\Delta t' = 1$ ms). A frequent choice of $p_{\text{count}}(s|\lambda(t'_i))$ is a Poisson distribution

$$p_{\text{count}}(s|\lambda(t'_i)) = \frac{(\lambda(t'_i))^s}{s!} e^{-\lambda(t'_i)}, \quad \lambda(t'_i) = A_{\text{trans}}(t'_i)\Delta t', \quad (12)$$

which results in an inhomogeneous Poisson process.

To match μ and σ^2 of the doubly stochastic process to the observed data, we need to estimate the generative model parameters μ' , σ'^2 and the variance of the event-count distribution $\sigma_{s|\lambda(t'_i)}^2$. According to the law of total expectation, the mean rate $\mu' = \hat{\lambda}/\Delta t'$, where $\hat{\lambda}$ is the sample mean of the observed event counts. According to the law of total variance⁴⁸, the total variance of event counts σ^2 arises from two contributions: the variance of the rate and the variance of the event-count distribution

$$\sigma^2 = \text{Var}[\lambda(t'_i)] + E[\sigma_{s|\lambda(t'_i)}^2] = (\Delta t')^2 \sigma'^2 + E[\sigma_{s|\lambda(t'_i)}^2]. \quad (13)$$

For the Poisson distribution, the variance of the event-count distribution is equal to its mean: $\sigma_{s|\lambda(t'_i)}^2 = \lambda(t'_i)$. However, the condition of equal mean and variance does not always hold in experimental data⁴⁹. We therefore also use other event-count distributions, in particular a Gaussian and gamma distribution. We define

α as the variance to mean ratio of the event-count distribution, $\alpha = \sigma_{s|\lambda(t_i)}^2/\lambda(t_i)$; $\alpha = 1$ always holds for the Poisson distribution, whereas we assume that α is constant (that is, it does not depend on the rate) for other distributions. With this assumption, the law of total variance, equation (13), becomes

$$\sigma^2 = (\Delta t')^2 \sigma'^2 + \alpha \Delta t' \mu', \quad (14)$$

where $\mu' = E[A_{\text{trans}}(t_i)]$ is the mean rate. From equation (14) we find the rate variance

$$\sigma'^2 = \frac{1}{(\Delta t')^2} (\hat{\sigma}^2 - \alpha \hat{\lambda}), \quad (15)$$

where $\hat{\sigma}^2$ is the sample variance of event counts in the observed data. We find that with the correct values of μ' , σ'^2 and α , both the Gaussian and gamma distributions of event counts produce comparable estimates of timescales (Supplementary Fig. 18).

To calculate σ'^2 with equation (15), we first need to estimate α from the observed data. We estimate α from the drop of autocorrelation of event counts between the time-lags t_0 and t_1 . As event counts in different time bins are drawn independently, this drop mainly reflects the difference between the total variance of event counts and variance of the rate (equation (13)); we neglect a small decrease of the rate autocorrelation between t_0 and t_1 and does not depend on timescales. We thus find α with a grid search that minimizes the distance between the autocorrelation at t_1 of the observed and synthetic data from the generative model with fixed timescales. Alternatively, α can be fitted together with all other parameters of the generative model using aABC. We find that α is almost independent from other parameters, aABC finds the correct value of α first and then fits the rest of parameters. The MAP estimate of α converges to the same value as estimated by the grid search, but aABC requires more iterations to get posterior distributions for estimated timescales with a similar variance (Supplementary Fig. 19). The grid-search method is thus preferred when moderate errors in α are acceptable and an approximate range of ground-truth timescales are known, but for more accurate results it is better to fit α by aABC together with other parameters.

For a doubly stochastic process, we can compute the autocorrelation function analytically based on the autocorrelation of its underlying time-varying rate and the mean and variance of the doubly stochastic process. For the two-timescale inhomogeneous Poisson process (Fig. 1c), the autocorrelation is given by:

$$\text{AC}(t_j) = \begin{cases} 1, & j = 0 \\ \frac{\sigma^2 - \mu}{\sigma^2} (c_1 e^{-t_j/\tau_1} + (1 - c_1) e^{-t_j/\tau_2}), & j > 0. \end{cases} \quad (16)$$

Here τ_1 and τ_2 are the ground-truth timescales defined by the parameters of two OU processes; c_1 is the mixing coefficient; and μ and σ^2 are the mean and variance of the event counts, respectively. The drop of autocorrelation between t_0 and t_1 mainly reflects the variance in the Poisson process. To estimate the timescales with direct exponential fits (Fig. 1c), we assumed that the mean and variance of the event counts are known and only estimated τ_1 , τ_2 and the coefficient c_1 by fitting equation (16) to the sample autocorrelation starting from the lag t_1 . When we later estimate the timescales from the data, we first need to find the correct μ and σ^2 .

Modeling background processes with an arbitrary PSD shape. We can generate random time-series with any desired shape of the PSD. First, we convert the power spectrum $\text{PSD}(f)$ to amplitudes $A(f) = \sqrt{2 \cdot \text{PSD}(f)}$. We then draw random phases $\phi(f)$ from a uniform distribution on $[0, 2\pi]$ and construct a frequency domain signal $Z(f) = A(f)e^{i\phi(f)}$. We transform this signal to the time domain by applying the inverse fast Fourier transform and z-score to the time-series to obtain zero mean and unit variance.

For example, a common class of background processes exhibit power-law decay in $\text{PSD}^{31,6,51,52}$. We model this PSD shape as $\text{PSD}(f) = f^{-\gamma}$, where $f_{\min} < f < f_{\max}$ is the frequency range and γ is the power-law exponent. We can set the lower frequency cut-off at $f_{\min} = 1$ Hz and the upper frequency cut-off f_{\max} is defined by the Nyquist frequency (that is, half of the desired sampling rate). We can combine this process with a time-series generated from an OU process with τ to obtain a PSD shape with both $1/f$ and Lorentzian components (Supplementary Fig. 17). We sum the two time-series with the coefficients c and $1 - c$ and rescale using the linear transformation in equation (8) to match the mean and variance to the observed data. This generative process can be used when the $1/f$ exponent in the data PSD deviates from 2 (that is, a Lorentzian shape). This method can be applied to generate background processes with arbitrary desired PSD shapes and estimate the relevant parameters.

Generating synthetic data for direct fitting. Each synthetic dataset consisted of 500 independent realizations of the process (that is, trials) with a fixed duration. Such trial-based data are typical in neuroscience but usually with a smaller number of trials. We computed the sample autocorrelation for each trial using equation (2), averaged them to reduce the noise and then fitted the average autocorrelation with the correct analytical functional form to estimate the timescale parameters.

We repeated the entire procedure 500 times to obtain a distribution with multiple independent samples of timescales estimated by direct fit (that is, we simulated 500×500 trials for each distribution).

Optimizing generative model parameters with aABC. We optimize parameters of generative models with aABC following the algorithm from ref. 33; aABC is an iterative algorithm to approximate the multivariate posterior distribution of model parameters. It uses population Monte-Carlo sampling to minimize the distance between the summary statistic of the observed and synthetic data from the generative model.

We can use autocorrelations as the summary statistic and define the suitable distance between the autocorrelations of synthetic and observed data, for example, as

$$d(t_m) = \frac{1}{m} \sum_{j=0}^m (\text{AC}_{\text{observed}}(t_j) - \text{AC}_{\text{synthetic}}(t_j))^2, \quad (17)$$

where t_m is the maximum time-lag considered in computing the distance. Alternatively, we can compute distances between the PSDs of synthetic and observed data:

$$d(f_n, f_{n+m}) = \frac{1}{m} \sum_{j=n}^{n+m} (\text{PSD}_{\text{observed}}(f_j) - \text{PSD}_{\text{synthetic}}(f_j))^2, \quad (18)$$

where f_n and f_{n+m} define the frequency range for computing the distance. Distances can also be computed on logarithmic scale. For the figures in the main text, we used the autocorrelation as the summary statistic and compute distances on a linear scale.

First, we choose a multivariate prior distribution over the parameters of the generative model and set an initial ε at a rather large value. On the first iteration of the algorithm, the parameters of the generative model $\theta_r^{(1)}$ are drawn from the prior distribution. We use a multidimensional uniform prior distribution $\pi(\theta)$ over fitted parameters (for example, timescales and their weights). The domain of prior distribution for the timescales is chosen to include a broad range below and above the timescales estimated by the direct exponential fits of data autocorrelation (Supplementary Table 2). For the weights of timescales c_r , we use uniform prior distributions on $[0, 1]$. The model with parameters $\theta_r^{(1)}$ is used to generate synthetic time-series $A(t')$ with the same duration and number of trials as in the observed data. We next compute d between the summary statistics of the synthetic and observed data. If d is smaller than ε (initially set to 1), the parameters are accepted and added to the multivariate posterior distribution. Each iteration of the algorithm is repeated until 500 parameters samples are accepted.

On subsequent iterations, the same steps are repeated but with parameters drawn from a proposal distribution and with an updated error threshold. On each iteration, the error threshold is set at the first quartile of the accepted sample distances from the previous iteration. The proposal distribution is computed for each iteration ξ as a mixture of Gaussian distributions based on the prior distribution and the accepted samples θ_r , $r = 1, \dots, N$ from the previous iteration:

$$\tilde{\pi}_\xi(\theta^{(\xi)}) \propto \sum_{r=1}^N \omega_r^{(\xi-1)} \mathcal{K}_\xi(\theta^{(\xi)} | \theta_r^{(\xi-1)}). \quad (19)$$

Here $\omega_r^{(\xi-1)}$ is the importance weight of the accepted sample $\theta_r^{(\xi-1)}$ from the previous iteration

$$\omega_r^{(\xi-1)} \propto \frac{\pi(\theta_r^{(\xi-1)})}{\tilde{\pi}(\theta_r^{(\xi-1)})}. \quad (20)$$

\mathcal{K}_ξ is the random walk kernel for the population Monte Carlo algorithm, which is a multivariate Gaussian with the mean $\theta_r^{(\xi-1)}$ and the covariance equal to twice the covariance of all accepted samples from the previous iteration $\Sigma = 2\text{Cov}[\theta^{(\xi-1)}]$:

$$\begin{aligned} & \mathcal{K}_\xi(\theta^{(\xi)} | \theta_r^{(\xi-1)}) \\ &= \frac{1}{\sqrt{2\pi^{\kappa} |\Sigma|}} \exp\left(-\frac{1}{2} (\theta^{(\xi)} - \theta_r^{(\xi-1)})^T \Sigma^{-1} (\theta^{(\xi)} - \theta_r^{(\xi-1)})\right). \end{aligned} \quad (21)$$

Here κ is the number of fitted parameters, and $|\Sigma|$ is the determinant of Σ .

In the regular ABC algorithm³⁶, the error threshold is fixed and parameters are sampled from the prior distribution (the same as in the first step of aABC). Updating the error threshold and proposal distribution in successive iterations of aABC allows for a more efficient fitting procedure especially when setting wide prior distributions³³.

The convergence of the algorithm is defined based on the acceptance rate, accR , which is the number of accepted samples divided by the total number of drawn samples on each iteration. The algorithm terminates when the acceptance rate reaches accR_{\min} , which is set to $\text{accR}_{\min} = 0.003$ in our simulations. A smaller accR_{\min} leads to a smaller final error threshold (Fig. 3, right) and a better

approximation of the posterior distributions, but requires longer fitting time. To find the MAP estimates, we smooth the final joint posterior distribution with a multivariate Gaussian kernel and find its maximum with a grid search.

Model selection and Bayes factor approximation with ABC. We compare M_1 and M_2 using a goodness of fit measure that describes how well each model fits the data. The goodness of fit can be measured by the distance between the summary statistic (for example, autocorrelation or PSD) of synthetic and observed data; that is, residual errors (equations (17) and (18)). For a fair comparison, the same summary statistics and fitting range should be used for fitting both models. As d is a noisy measure due to the finite sample size and uncertainty in the model parameters, we compare the distributions of distances generated by two models with parameters sampled from their posterior distributions. To approximate the distributions of distances, we generate multiple samples of synthetic data from each model with parameters drawn from its posterior distribution and compute d for each sample. If the distributions of distances are significantly different (that is, expectations of the summary statistic for two models are significantly different⁴⁴) then we continue with the model selection, otherwise the summary statistic is insufficient to distinguish these models.

Using the distributions of distances we estimate the Bayes factor to select between the two models. The Bayes factor is the ratio of marginal likelihoods of the two models and accounts for the model's complexity⁴⁶. Assuming both models are a priori equally probable $p(M_1) = p(M_2)$, the Bayes factor can be approximated using the models' acceptance rates for a specific ϵ (refs. ^{42,45,57})

$$B_{21}(\epsilon) = \frac{\text{accR}_{M_2}(\epsilon)}{\text{accR}_{M_1}(\epsilon)}. \quad (22)$$

$B_{21}(\epsilon) > 1$ indicates that the model M_2 is more likely to explain the observed data and vice versa. To eliminate the dependence on a specific error threshold, we compute the acceptance rates and the Bayes factor with a varying error threshold. For each ϵ , the acceptance rate is given by the CDF of the distances $\text{CDF}_{M_i}(\epsilon) = p_{M_i}(d < \epsilon) = \text{accR}_{M_i}(\epsilon)$ ($i = 1, 2$). Hence, the ratio between CDFs of two models gives the value of the Bayes factor for every error threshold $B_{21}(\epsilon) = \text{CDF}_{M_2}(\epsilon) / \text{CDF}_{M_1}(\epsilon)$. We select the model M_2 if $B_{21}(\epsilon) > 1$, that is, if $\text{CDF}_{M_2}(\epsilon) > \text{CDF}_{M_1}(\epsilon)$ for all ϵ and vice versa. For experimental data, it is often reasonable to put an upper bound on ϵ for computing the Bayes factor (for example, the largest median(ϵ) between the two models), as only small values of ϵ indicate a well-fitted model.

abcTau Python package. We developed the abcTau Python package implementing our aABC framework for estimation of timescales from autocorrelations or PSDs of various types of experimental data, and the Bayesian model comparison to select between different hypotheses⁵⁸. We also provided tutorials as Jupyter Notebooks and example Python scripts to make our framework easily accessible for researchers in different fields.

The minimal requirements for using this package are Python v.3.7.1, Numpy v.1.15.4 and Scipy v.1.1.0. For visualization, Matplotlib $>=$ v.3.0.2 and Seaborn $>=$ v.0.9.0 are required. The basis of aABC algorithm in the package is adopted from a previous implementation originally developed in Python 2 (<https://github.com/rcmorehead/simpleabc>). As all parameters of our generative models are positive and sometimes subject to additional conditions (for example, $\tau_2 > \tau_1$), we introduced constraints on sampling from proposal distributions. Moreover, we enhanced the algorithm for parallel processing required for analyzing large datasets.

The abcTau package includes various types of generative models that are relevant for different types data and various methods for computing the autocorrelation or PSD. Using this functionality, users can apply our framework to their time-series data, supplied in a Numpy array structured as trials times time-points. The object oriented implementation of the package allows users to easily replace any function, including generative models, summary statistic computations, distance functions and so on with their customized functions to better describe statistics of the data. Users can also add their customized generative models directly to the package to create a larger database of generative models available for different applications.

The package also includes a module for Bayesian model selection. This module computes the cumulative distribution of distances from estimated posterior distributions and a Bayes factor for every error threshold, runs the statistical tests and suggests the best hypothesis describing the underlying processes in data.

Pre-processing for evaluating quality of direct fit. As Bayesian inference of a full posterior distribution can be computationally expensive, we implemented a fast pre-processing function that uses parametric bootstrapping to determine whether the direct exponential fit provides satisfactory estimates of timescales (Supplementary Fig. 14). In this function, a generative model (for example, based on a mixture of OU processes) with parameters obtained from the direct exponential fit can be used to generate multiple synthetic datasets, each with the same amount of data as in the original data. For each synthetic dataset, timescales are estimated by direct exponential fitting. The obtained distribution of timescales

from this bootstrapping procedure can be compared to the initial direct-fit estimate from the original data. The error between the mean of bootstrapping distribution and the initial direct fit is used to approximately evaluate the direct fit quality. If the error is small enough, the direct exponential fit may be sufficiently accurate. The accuracy of timescale estimates with the direct fit can be further improved by empirical bias-correction using the measured deviation between the mean of the parametric bootstrap distribution and the direct-fit estimates (Supplementary Fig. 5). However, this method does not guarantee an accurate bias correction as the deviation of the direct fit from the ground truth can be larger than the observed deviation between the bootstrap and direct fit. Hence, we recommend users to be conservative with the decision to rely on the direct-fit estimates if accurate estimates of timescales are desired.

Branching network model. A branching network consists of K interconnected binary neurons, each described by the state variable $x_i \in \{0, 1\}$, where $x_i = 1$ indicates that neuron i is active and 0 that it is silent. We considered a fully connected network. Each active neuron can activate other neurons with the probability $p = m/K$ and then, if not activated by other neurons, it becomes inactive again in the next time-step. Furthermore, at every time-step, each neuron can be activated with a probability h by an external input. For a small input strength h , the state of the network's dynamics is governed by a branching parameter m ($m = 1$ corresponds to the critical state). The autocorrelation function of the global activity $A(t') = \sum_i x_i(t')$ in this network is known analytically $\text{AC}(t) = \exp(t \ln(m))$ ⁵. Thus the ground-truth timescale of this activity is given by $\tau = -1/\ln(m)$.

Neural recordings and analysis. Experimental procedures and data pre-processing were described previously⁴⁷. Experimental procedures were in accordance with NIH Guide for the Care and Use of Laboratory Animals, the Society for Neuroscience Guidelines and Policies, and Stanford University Animal Care and Use Committee.

In brief, a monkey was trained to fixate a central dot on a blank screen for 3 s on each trial. Spiking activity was recorded with a 16-channel micro-electrode array inserted perpendicularly to the cortical surface to record from all layers in the visual area V4. For fitting, we used a recording session with 81 trials. We pooled the activity across all channels and calculated the population spike-counts in 1 ms bins. First, we subtracted the trial-averaged activity (Peristimulus time histogram, PSTH) from spike-counts to remove the slow trends locked to the trial onset¹. We then computed the autocorrelation of spike-counts using equation (2) and averaged the autocorrelations across all trials.

To estimate the timescales from direct fitting, we fitted the average autocorrelation of spike counts with a double exponential function

$$\text{AC}(t) = c_1 e^{-t/\tau_1} + (1 - c_1) e^{-t/\tau_2} \quad (23)$$

up to the same $t_m = 150$ ms as used for the aABC fit. Including all of the time lags in exponential fitting results in a larger bias in estimated timescales. We used a sampling method based on parametric bootstrapping to compare the goodness of direct-fit estimates with the MAP estimates from the aABC fit. We generated multiple samples of synthetic data using the two-timescale doubly stochastic generative model with parameters from either the direct fit or MAP estimates from aABC. For each sample, we measured the distance between the autocorrelation of synthetic and neural data to obtain the distribution of distances for both types of fits. We then used two-sided Wilcoxon rank-sum test to compare the distributions.

Parameters of simulations and aABC fits in figures. The initial error threshold was set to $\epsilon = 1$ for all fits. The aABC iterations continued until $\text{accR} \leq 0.003$. All datasets (except for the branching network) consisted of 500 trials, each of 1 s duration. The dataset for the branching network (Fig. 4) consisted of 100 trials with 500 time-steps. The parameters for simulations and aABC fits are given in Supplementary Tables 1 and 2, respectively.

Statistics and reproducibility. As the first step of the model selection with ABC, we used two-sided Wilcoxon rank-sum test (also known as Mann-Whitney U test) to compare the distribution of distances of the two models. For this non-parametric test, we can compute the common language effect size⁵⁹ $\text{CL} = U/(n_1 n_2)$ using U statistics from the rank-sum test and number of samples n_1 and n_2 from the two models. For computing the effect size, we took the model with the longer average distance as the reference point. Hence, $\text{CL} = 1$ is the largest effect size. We truncated P -values smaller than 10^{-10} and rounded the remaining P -values to the third decimal place. We rounded the effect size values to the second decimal place.

For all model comparisons performed in this paper, there were always a significant difference between the distances of the two models (that is, $P < 0.05$). The detailed results of the statistical analysis can be found in the captions of Figs. 5 and 6. We reported P -values, effect sizes and sample sizes (test statistic U can be computed directly from the effect size and sample sizes as $U = \text{CL} \cdot n_1 \cdot n_2$).

Data availability

Electrophysiological recordings from primate area V4 were performed at Stanford University and presented in ref.⁴⁷. The raw electrophysiological data (session

2013-06-18, blank screen condition) are available on Figshare at <https://doi.org/10.6084/m9.figshare.19077875.v1> (ref. ⁴⁰). Processed data and data from example aABC fits and model selections are available on GitHub at <https://github.com/roxana-zeraati/abcTau> and on Zenodo at <https://doi.org/10.5281/zenodo.5949117> (ref. ⁵⁸). Source Data are provided with this paper.

Code availability

The abcTau Python package together with tutorials and Jupyter Notebooks for reproducing figures are available on GitHub at <https://github.com/roxana-zeraati/abcTau> and Zenodo at <https://doi.org/10.5281/zenodo.5949117> (ref. ⁵⁸).

Received: 25 May 2021; Accepted: 15 February 2022;

Published online: 24 March 2022

References

- Murray, J. D. et al. A hierarchy of intrinsic timescales across primate cortex. *Nat. Neurosci.* **17**, 1661 (2014).
- Watanabe, T., Rees, G. & Masuda, N. Atypical intrinsic neural timescale in autism. *eLife* **8**, e42256 (2019).
- Gao, R., van den Brink, R. L., Pfeffer, T. & Voytek, B. Neuronal timescales are functionally dynamic and shaped by cortical microarchitecture. *eLife* **9**, e61277 (2020).
- Zeraati, R. et al. Attentional modulation of intrinsic timescales in visual cortex and spatial networks. Preprint at <https://www.biorxiv.org/content/10.1101/2021.05.17.444537v1> (2021).
- Wilting, J. & Priesemann, V. Inferring collective dynamical states from widely unobserved systems. *Nat. Commun.* **9**, 1–7 (2018).
- Cavanagh, S. E., Wallis, J. D., Kennerley, S. W. & Hunt, L. T. Autocorrelation structure at rest predicts value correlates of single neurons during reward-guided choice. *eLife* **5**, e18937 (2016).
- Siegle, J. H. et al. Survey of spiking in the mouse visual system reveals functional hierarchy. *Nature* **592**, 86–92 (2021).
- Stringer, C. et al. Inhibitory control of correlated intrinsic variability in cortical networks. *eLife* **5**, e19695 (2016).
- Fascianelli, V., Tsujimoto, S., Marcos, E. & Genovesio, A. Autocorrelation structure in the macaque dorsolateral, but not orbital or polar, prefrontal cortex predicts response-coding strength in a visually cued strategy task. *Cerebral Cortex* **29**, 230–241 (2019).
- MacDowell, C. J. & Buschman, T. J. Low-dimensional spatiotemporal dynamics underlie cortex-wide neural activity. *Curr. Biol.* **30**, 2665–2680.e8 (2020).
- Kim, R. & Sejnowski, T. J. Strong inhibitory signaling underlies stable temporal dynamics and working memory in spiking neural networks. *Nat. Neurosci.* **24**, 129–139 (2021).
- Ito, T., Hearne, L. J. & Cole, M. W. A cortical hierarchy of localized and distributed processes revealed via dissociation of task activations, connectivity changes, and intrinsic timescales. *NeuroImage* **221**, 117141 (2020).
- Strey, H., Peterson, M. & Sackmann, E. Measurement of erythrocyte membrane elasticity by flicker eigenmode decomposition. *Biophys. J.* **69**, 478–488 (1995).
- Rohrbach, A., Meyer, T., Stelzer, E. H. & Kress, H. Measuring stepwise binding of thermally fluctuating particles to cell membranes without fluorescence. *Biophys. J.* **118**, 1850–1860 (2020).
- Liu, K. et al. Hydrodynamics of transient cell–cell contact: the role of membrane permeability and active protrusion length. *PLoS Comput. Biol.* **15**, e1006352 (2019).
- Donoghue, T. et al. Parameterizing neural power spectra into periodic and aperiodic components. *Nat. Neurosci.* **23**, 1655–1665 (2020).
- Marriott, F. & Pope, J. Bias in the estimation of autocorrelations. *Biometrika* **41**, 390–402 (1954).
- Sastry, A. S. R. Bias in estimation of serial correlation coefficients. *Indian J. Stat.* **11**, 281–296 (1951).
- Huitema, B. E. & McKeen, J. W. Autocorrelation estimation and inference with small samples. *Psychol. Bull.* **110**, 291 (1991).
- White, J. S. Asymptotic expansions for the mean and variance of the serial correlation coefficient. *Biometrika* **48**, 85–94 (1961).
- Lomnicki, Z. & Zaremba, S. On the estimation of autocorrelation in time series. *Ann. Math. Stat.* **28**, 140–158 (1957).
- Kendall, M. G. Note on bias in the estimation of autocorrelation. *Biometrika* **41**, 403–404 (1954).
- Afyouni, S., Smith, S. M. & Nichols, T. E. Effective degrees of freedom of the Pearson's correlation coefficient under autocorrelation. *NeuroImage* **199**, 609–625 (2019).
- Bartlett, M. S. On the theoretical specification and sampling properties of autocorrelated time-series. *J. R. Stat. Soc.* **8**, 27–41 (1946).
- Cliff, O. M., Novelli, L., Fulcher, B. D., Shine, J. M. & Lizier, J. T. Assessing the significance of directed and multivariate measures of linear dependence between time series. *Phys. Rev. Res.* **3**, 013145 (2021).
- Khintchine, A. Korrelationstheorie der stationären stochastischen prozesse. *Math. Ann.* **109**, 604–615 (1934).
- Strey, H. H. Estimation of parameters from time traces originating from an Ornstein–Uhlenbeck process. *Phys. Rev. E* **100**, 062142 (2019).
- Spitmaam, M. M., Seo, H., Lee, D. & Soltani, A. Multiple timescales of neural dynamics and integration of task-relevant signals across cortex. *Proc. Natl Acad. Sci. USA* **117**, 22522–22531 (2020).
- Brody, C. D. Slow covariations in neuronal resting potentials can lead to artefactually fast cross-correlations in their spike trains. *J. Neurophysiol.* **80**, 3345–3351 (1998).
- Ventura, V., Cai, C. & Kass, R. E. Trial-to-trial variability and its effect on time-varying dependency between two neurons. *J. Neurophysiol.* **94**, 2928–2939 (2005).
- Amarasingham, A., Harrison, M. T., Hatsopoulos, N. G. & Geman, S. Conditional modeling and the jitter method of spike resampling. *J. Neurophysiol.* **107**, 517–531 (2012).
- Cohen, M. R. & Kohn, A. Measuring and interpreting neuronal correlations. *Nat. Neurosci.* **14**, 811 (2011).
- Beaumont, M. A., Cornuet, J.-M., Marin, J.-M. & Robert, C. P. Adaptive approximate bayesian computation. *Biometrika* **96**, 983–990 (2009).
- Kelly, R. C., Smith, M. A., Kass, R. E. & Lee, T. S. Local field potentials indicate network state and account for neuronal response variability. *J. Comput. Neurosci.* **29**, 567–579 (2010).
- Ecker, A. S. et al. State dependence of noise correlations in macaque primary visual cortex. *Neuron* **82**, 235–248 (2014).
- Genkin, M. & Engel, T. A. Moving beyond generalization to accurate interpretation of flexible models. *Nat. Mach. Intell.* **2**, 674–683 (2020).
- Neophytou, D., Arribas, D., Levy, R., Park, I. M. & Oviedo, H. V. Recurrent connectivity underlies lateralized temporal processing differences in auditory cortex. Preprint at <https://www.biorxiv.org/content/10.1101/2021.04.14.439872v1> (2021).
- Babadi, B. & Brown, E. N. A review of multitaper spectral analysis. *IEEE Trans. Biomed. Eng.* **61**, 1555–1564 (2014).
- Haldeaman, C. & Beggs, J. M. Critical branching captures activity in living neural networks and maximizes the number of metastable states. *Phys. Rev. Lett.* **94**, 058101 (2005).
- Zierenberg, J., Wilting, J., Priesemann, V. & Levina, A. Description of spreading dynamics by microscopic network models and macroscopic branching processes can differ due to coalescence. *Phys. Rev. E* **101**, 022301 (2020).
- Zierenberg, J., Wilting, J., Priesemann, V. & Levina, A. Tailored ensembles of neural networks optimize sensitivity to stimulus statistics. *Phys. Rev. Res.* **2**, 013115 (2020).
- Grelaud, A. et al. ABC likelihood-free methods for model choice in gibbs random fields. *Bayesian Anal.* **4**, 317–335 (2009).
- Didelot, X., Everitt, R. G., Johansen, A. M. & Lawson, D. J. et al. Likelihood-free estimation of model evidence. *Bayesian Anal.* **6**, 49–76 (2011).
- Marin, J.-M., Pillai, N. S., Robert, C. P. & Rousseau, J. Relevant statistics for Bayesian model choice. *J. R. Stat. Soc. B* **76**, 833–859 (2014).
- Robert, C. P., Cornuet, J.-M., Marin, J.-M. & Pillai, N. S. Lack of confidence in approximate Bayesian computation model choice. *Proc. Natl Acad. Sci. USA* **108**, 15112–15117 (2011).
- Bishop, C. M. *Pattern Recognition and Machine Learning* (Springer, 2006).
- Engel, T. A. et al. Selective modulation of cortical state during spatial attention. *Science* **354**, 1140–1144 (2016).
- Churchland, A. K. et al. Variance as a signature of neural computations during decision making. *Neuron* **69**, 818–831 (2011).
- Goris, R. L., Movshon, J. A. & Simoncelli, E. P. Partitioning neuronal variability. *Nat. Neurosci.* **17**, 858 (2014).
- Brody, C. D. Correlations without synchrony. *Neural Comput.* **11**, 1537–1551 (1999).
- Gao, R., Peterson, E. J. & Voytek, B. Inferring synaptic excitation/inhibition balance from field potentials. *NeuroImage* **158**, 70–78 (2017).
- Wagenmakers, E.-J., Farrell, S. & Ratcliff, R. Estimation and interpretation of 1/α noise in human cognition. *Psych. Bull. Rev.* **11**, 579–615 (2004).
- Uhlenbeck, G. E. & Ornstein, L. S. On the theory of the Brownian motion. *Phys. Rev.* **36**, 823 (1930).
- Risken, H. in *The Fokker–Planck Equation* 63–95 (Springer, 1996).
- Lindner, B. in *Stochastic Methods in Neuroscience* Vol. 1 (Oxford Univ. Press, 2009).
- Sunnåker, M. et al. Approximate Bayesian computation. *PLoS Comput. Biol.* **9**, e1002803 (2013).
- Toni, T. & Stumpf, M. P. Simulation-based model selection for dynamical systems in systems and population biology. *Bioinformatics* **26**, 104–110 (2010).
- Zeraati, R., Engel, T. A. & Levina, A. *Roxana-zeraati/abcTau: A flexible Bayesian Framework for Unbiased Estimation of Timescales* (Zenodo, 2022); <https://doi.org/10.5281/zenodo.5949117>

59. McGraw, K. O. & Wong, S. P. A common language effect size statistic. *Psychol. Bull.* **111**, 361 (1992).
60. Steinmetz, N. & Moore, T. *Dataset of Linear-Array Recordings From Macaque V4 During a Fixation Task* (Figshare, 2022); <https://doi.org/10.6084/m9.figshare.19077875.v1>

Acknowledgements

This work was supported by a Sofja Kovalevskaja Award from the Alexander von Humboldt Foundation, endowed by the Federal Ministry of Education and Research (R.Z. and A.L.), the SMARTSTART2 program provided by Bernstein Center for Computational Neuroscience and Volkswagen Foundation (R.Z.), an NIH grant (grant no. R01 EB026949, T.A.E.), the Pershing Square Foundation (T.A.E.) and the Alfred P. Sloan Foundation Research Fellowship (T.A.E.). We acknowledge the support from the BMBF through the Tübingen AI Center (FKZ grant no. 01IS18039B) and International Max Planck Research School for the Mechanisms of Mental Function and Dysfunction (IMPRS-MMFD). We thank N. A. Steinmetz and T. Moore for sharing the electrophysiological data⁴⁷.

Author contributions

R.Z., T.A.E. and A.L. designed the research, discussed the results and wrote the paper. R.Z. wrote the computer code, performed simulations and analyzed the data.

Funding

Open access funding provided by Max Planck Society.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s43588-022-00214-3>.

Correspondence and requests for materials should be addressed to Tatiana A. Engel or Anna Levina.

Peer review information *Nature Computational Science* thanks Richard Gao and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Handling editor: Ananya Rastogi, in collaboration with the *Nature Computational Science* team.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022