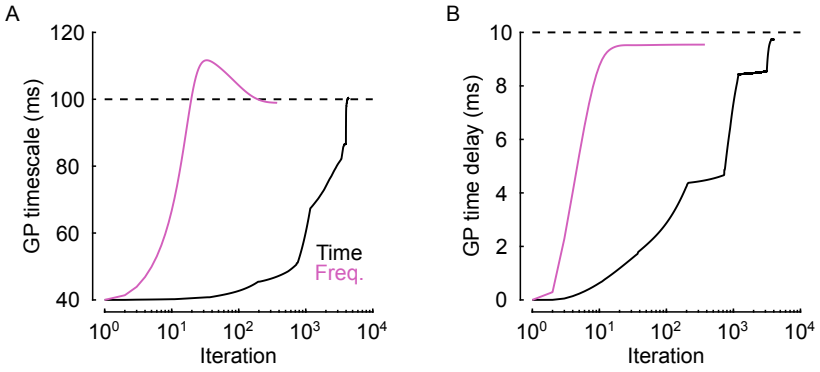
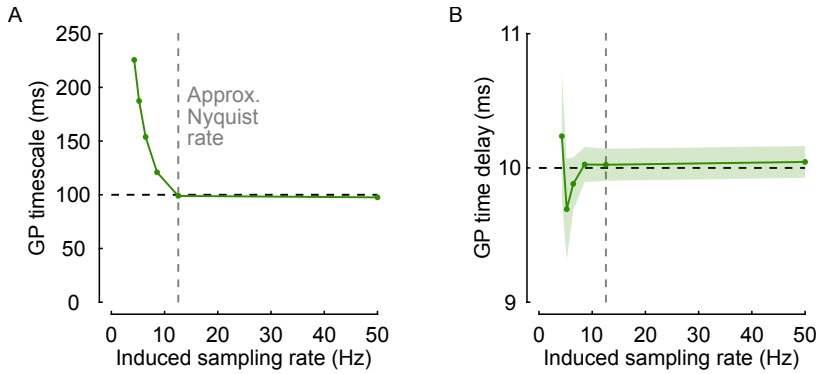
# Supplementary Tables and Figures

Fast Multi-Group Gaussian Process Factor Models

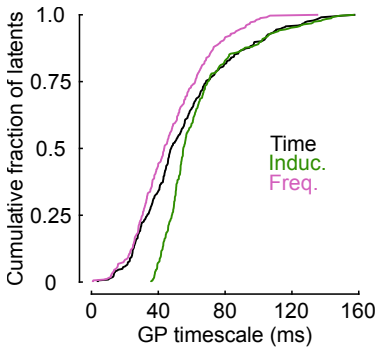
Evren Gokcen, Anna I. Jasper,
Adam Kohn, Christian K. Machens, Byron M. Yu



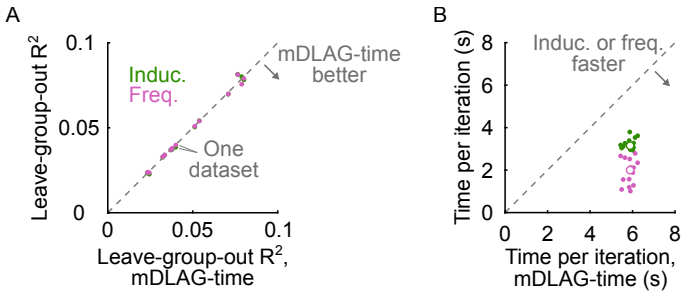

Supplementary Figure 1: Progress of Gaussian process (GP) parameter estimates with each fitting iteration (related to Fig. 3C, an example run with $T = 500$). (A) GP timescale progress with each fitting iteration. (B) GP time delay progress with each fitting iteration. Black: mDLAG-time (Time); magenta: mDLAG-frequency (Freq.). Black dashed lines indicate ground truth parameter values. GP parameter estimates for mDLAG-frequency converge toward the optimum in significantly fewer iterations than the GP parameter estimates for mDLAG-time, partially explaining the faster convergence of the mDLAG-frequency method.



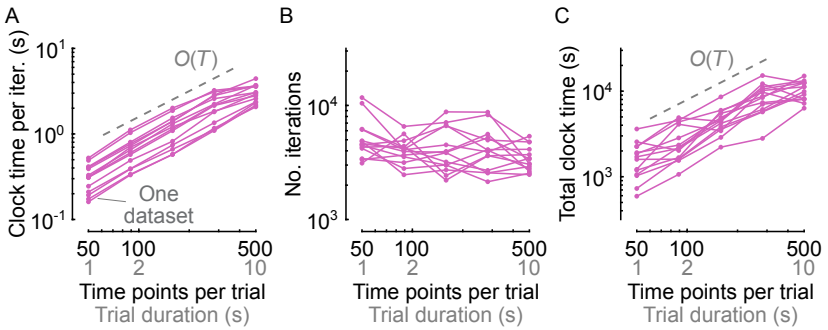

Supplementary Figure 2: Effects of induced sampling rate (determined by the chosen number of inducing points) on mDLAG-inducing performance. For the same simulated data as in Fig. 3, we chose a fixed trial length ($T = 163$ time points, 3.3 s trial duration), and then re-fit mDLAG-inducing models with varying numbers of inducing points ($T_{\text{ind}} \in \{14, 17, 21, 28, 41, 163\}$). Each number of inducing points corresponded to a different induced sampling rate. (A) GP timescale estimates versus induced sampling rate. (B) GP time delay estimates versus induced sampling rate. Horizontal black dashed lines indicate ground truth parameter values. Solid traces indicate the mean of estimates, and shading (where visible) indicates standard error of the mean, computed across 20 runs. Performance degrades below an induced sampling rate of 12-13 Hz, the approximate Nyquist rate of a squared exponential GP with 100 ms timescale (vertical gray dashed lines).



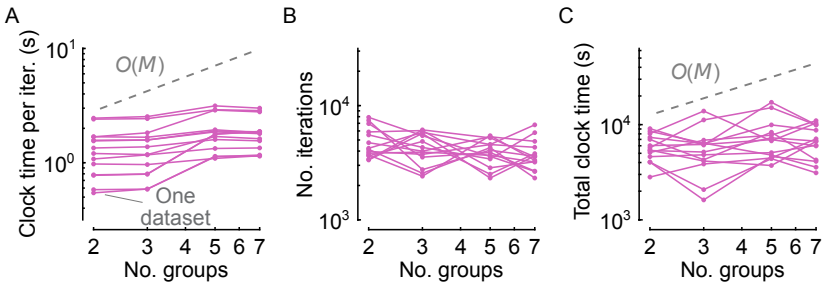

Supplementary Figure 3: Empirical cumulative distribution functions (eCDFs) of estimated GP timescales across all Neuropixels datasets. Black: mDLAG-time (Time); green: mDLAG-inducing (Induc.); magenta: mDLAG-frequency (Freq.). In contrast with mDLAG-time and mDLAG-frequency, mDLAG-inducing (with $T_{\text{ind}} = 20$ inducing points) did not capture fast timescales. For instance, by mDLAG-time estimates, about 25% of the GP timescales encountered across Neuropixels datasets were 35 ms or shorter (34%, according to mDLAG-frequency). No mDLAG-inducing timescale estimates were shorter than 35 ms. This discrepancy in estimated timescales explains the discrepancy in performance between mDLAG-inducing and the other two methods (Fig. 5B). eCDFs are computed over 324, 309, and 323 significant latents for mDLAG-time, mDLAG-inducing, and mDLAG-frequency, respectively. These numbers are pooled across all 14 Neuropixels datasets.




Supplementary Figure 4: Neuropixels results (related to Fig. 5), but with $T_{\text{ind}} = 32$ inducing points for mDLAG-inducing, rather than $T_{\text{ind}} = 20$. (A) Performance (leave-group-out $R^2$) of mDLAG-inducing (green points; Induc.) or mDLAG-frequency (magenta points; Freq.) versus performance of mDLAG-time. Each data point represents one dataset. For each dataset, performance was evaluated on 75 held-out test trials. In Fig. 5B, mDLAG-inducing (with $T_{\text{ind}} = 20$ inducing points) was outperformed by both mDLAG-time and mDLAG-frequency. Here, mDLAG-inducing (with $T_{\text{ind}} = 32$ inducing points) performed similarly to the other methods (one-sided paired sign tests: mDLAG-time better than mDLAG-inducing, $p = 0.3953$; mDLAG-frequency better than mDLAG-inducing, $p = 0.0898$). (B) Mean clock time per iteration, mDLAG-inducing or mDLAG-frequency versus mDLAG-time. As in panel (A), each data point represents one dataset. White-filled circles indicate the median of the magenta points (circle with magenta border) or of the green points (circle with green border). In Fig. 5C-E, mDLAG-inducing (with $T_{\text{ind}} = 20$ inducing points) gave a similar runtime (and speed-up over mDLAG-time) to mDLAG-frequency. Here, while mDLAG-inducing (with $T_{\text{ind}} = 32$ inducing points) still provides a significant speed-up over mDLAG-time, mDLAG-frequency is now 55% faster than mDLAG-inducing per iteration, and 72% faster overall. Median clock time per iteration across datasets: mDLAG-time, 5.88 s; mDLAG-inducing, 3.15 s; mDLAG-frequency, 2.00 s. Median number of iterations across datasets (not shown): mDLAG-time, 30,737; mDLAG-inducing, 3,221; mDLAG-frequency, 3,659; for mDLAG-time, we cut off fitting at 50,000 iterations. Median total clock time across datasets (not shown): mDLAG-time, 50.4 hrs; mDLAG-inducing, 2.9 hrs; mDLAG-frequency, 1.8 hrs.



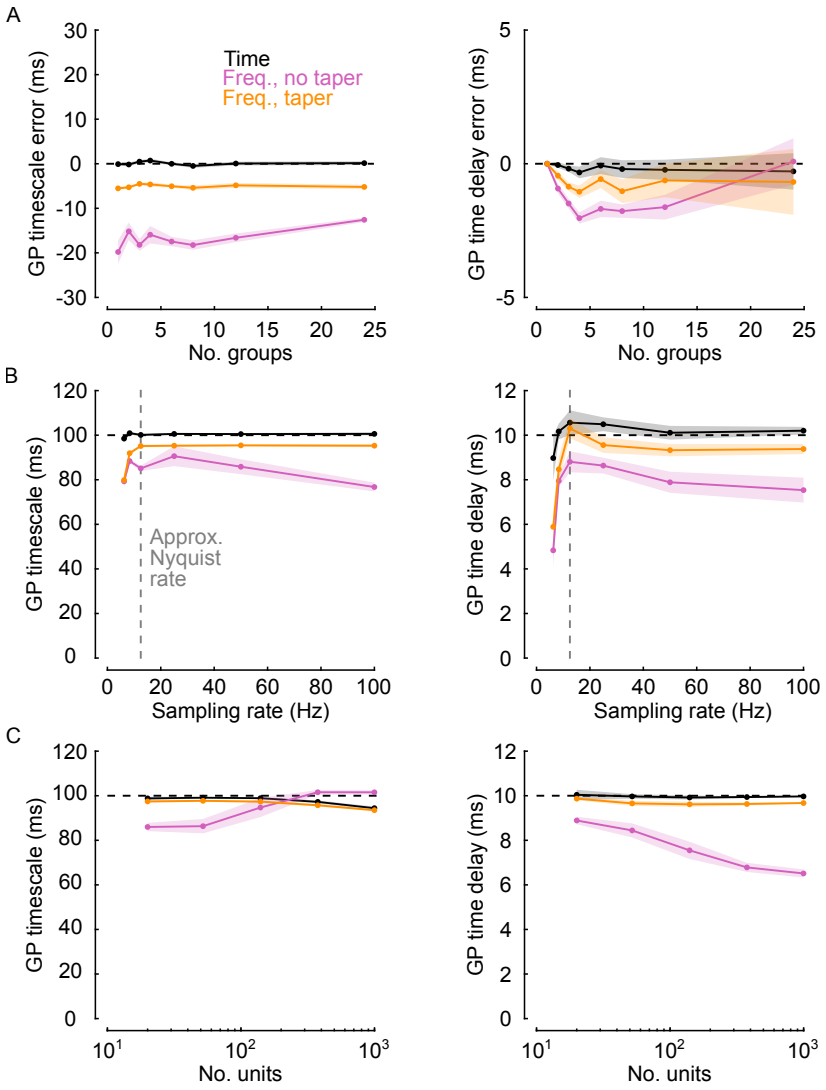

Supplementary Figure 5: Scaling of mDLAG-frequency's runtime with number of time points per trial—Neuropixels recordings. We sought to demonstrate that mDLAG-frequency's scaling with number of time points per trials ($T$) in simulation (Fig. 3) held in the Neuropixels recordings (Fig. 5). Toward that end, we started with all 300 trials within each dataset, normally $T = 50$ time points (1 s in duration) each. We then reshaped these data: we concatenated trials to construct 30 pseudo-trials, each $T = 500$ time points (10 s) in length. Then in a similar manner as in Fig. 3, for each of the 14 Neuropixels datasets, we fit models via mDLAG-frequency to increasingly long pseudo-trial epochs (lengths equally spaced on a log scale from $T = 50$ to $T = 500$ time points per trial). (A) Mean (across fitting iterations) clock time per iteration versus number of time points per pseudo-trial. Each line represents one of the 14 datasets. Lines may occlude one another. The total number of neurons varies from session to session, directly leading to runtime variability from session to session. Gray dashed line: reference slope for a linear scaling law, $O(T)$. (B) Number of iterations to convergence versus number of time points per pseudo-trial. Same conventions as in (A). (C) Total clock time to convergence versus number of time points per pseudo-trial. Same conventions as in (A). mDLAG-frequency's linear runtime scaling with the number of time points per pseudo-trial is indeed consistent with Fig. 3B–D.




Supplementary Figure 6: Scaling of mDLAG-frequency's runtime with number of groups—Neuropixels recordings. We sought to demonstrate that mDLAG-frequency's scaling with group number ($M$) in simulation (Fig. 4) held in the Neuropixels recordings (Fig. 5). Toward that end, our Neuropixels probes spanned the full laminar depth of V1 and V2 (and the superficial and middle layers of V3d). On a subset of our probes within each recording session, we could categorize each of the recorded V1 and V2 populations further into three laminar compartments: superficial, middle, and deep. We could therefore analyze up to $M = 7$ subpopulations (groups) with these recordings (V1-superficial: 25–106 neurons; V1-middle: 11–51; V1-deep: 3–40; V2-superficial: 7–39; V2-middle: 2–22; V2-deep: 6–26; V3d: 8–111). Then to demonstrate mDLAG-frequency's group-wise runtime scaling trend, we combined these subpopulations in different ways to create $M = 2$, 3, 5 and 7 groups while keeping the total number of neurons across all groups fixed. The subpopulation combinations were as follows. $M = 2$: group 1: all of V1; group 2: all of V2 and V3d. $M = 3$: group 1: all of V1; group 2: all of V2; group 3: V3d. $M = 5$: group 1: V1-superficial; group 2: V1-middle and V1-deep; group 3: V2-superficial; group 4: V2-middle and V2-deep; group 5: V3d. $M = 7$: each of the seven subpopulations was treated as a separate group. We fit models via mDLAG-frequency to each of these groupings, for each of the 14 datasets. (A) Mean (across fitting iterations) clock time per iteration versus number of groups. Each line represents one of the 14 datasets. Lines may occlude one another. The total number of neurons varies from session to session, directly leading to runtime variability from session to session. Gray dashed line: reference slope for a linear scaling law, $O(M)$. (B) Number of iterations to convergence versus number of groups. Same conventions as in (A). (C) Total clock time to convergence versus number of groups. Same conventions as in (A). mDLAG-frequency's linear or better runtime scaling with the number of groups is indeed consistent with Fig. 4B–D.

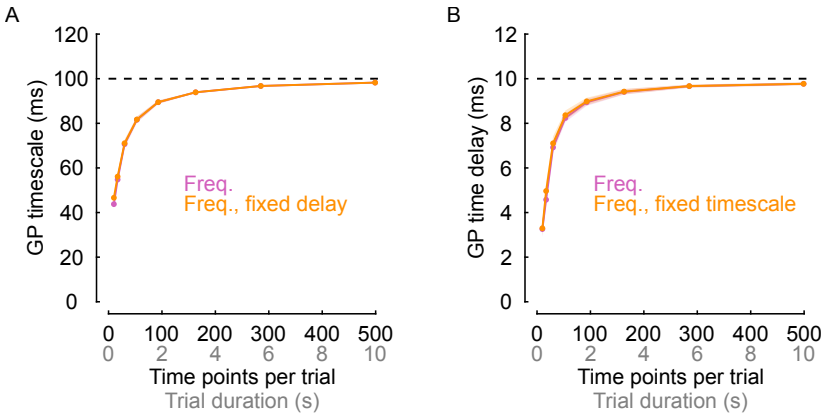
Supplementary Table 1: Summary of Supplementary Simulated Datasets.

| Dataset | Sampling rate | Scaling, $q$ |
|---|---|---|
| Independent runs | 10 | 5 |
| Training trials | 100 | 100 |
| Time points | 7 – 100 | 100 |
| Samp. period (ms) | 10 – 160 | 20 |
| Trial duration (s) | 1 | 2 |
| Groups | 2 | 2 |
| Units per group | 12 | 10 – 500 |
| Latents | 1 | 1 |
| GP timescales (ms) | 100 | 100 |
| GP time delays (ms) | 10 | 10 |
| Signal-to-noise ratio | 0.2 | 0.2 |

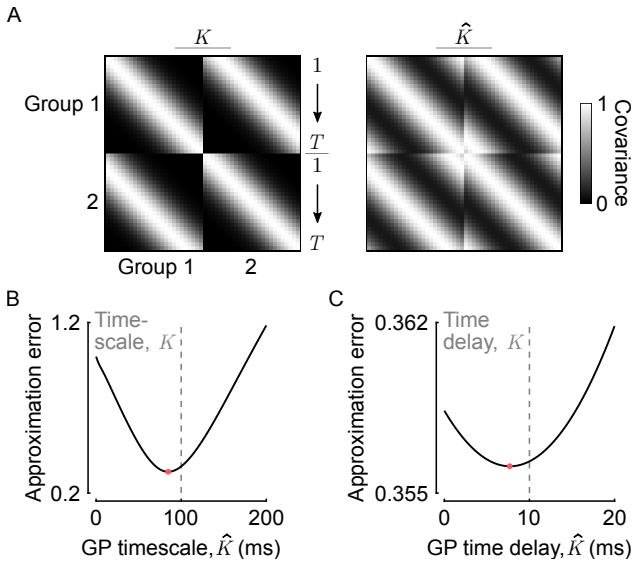



Supplementary Figure 7: Additional characterizations of bias in Gaussian process (GP) parameter estimation. (A) Estimation bias as a function of number of groups (same data as in Fig. 4; Table 1, Scaling, $M$). Left: GP timescale estimation error versus number of groups. Right: GP time delay estimation error versus number of groups. Black dashed lines indicate zero error. Solid traces indicate the mean of estimates, and shading (where visible) indicates standard error of the mean, computed across 20 runs. Black: mDLAG-time (Time); magenta: mDLAG-frequency without tapering (Freq., no taper); gold: mDLAG-frequency with tapering (Freq., taper). Group number has no consistent effect on estimation bias. Tapering (gold) was an effective bias mitigation strategy. (B) Estimation bias as a function of sampling rate. We generated 10 additional datasets (Supplementary Table 1, Sampling rate; see also Section 6.1),

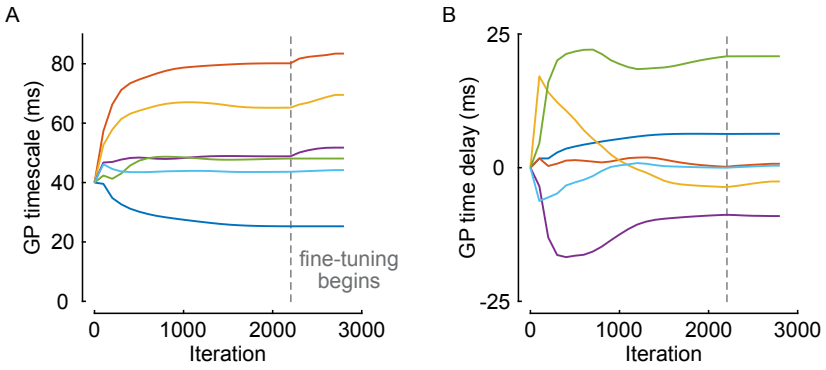
where each trial was $1,000$ ms in length, sampled at 10 ms resolution. We then fit mDLAG models to these data, downsampled to different rates (6.25 – 100 Hz). Regardless of downsampling, trials always spanned $1,000$ ms. Left: GP timescale estimates versus sampling rate. Right: GP time delay estimates versus sampling rate. Solid traces indicate the mean of estimates, and shading (where visible) indicates standard error of the mean, computed across 10 runs. Below the approximate Nyquist rate for the ground truth latent signals (vertical gray dashed lines; 12 – 13 Hz for a squared exponential GP with 100 ms timescale), all methods exhibit a decline in performance, as sampling rate decreases, due to aliasing. mDLAG-frequency (magenta) increasingly underestimated both GP timescales and time delays as sampling rates significantly surpassed the approximate Nyquist rate. Increasingly fast sampling rates likely exacerbate edge effects due to finite trial length and the periodic boundary conditions of the frequency domain approach. Discontinuities between the beginning and ends of each trial could introduce spurious high-frequency content. Tapering (gold), which attenuates observations at the beginning and ends of each trial toward zero, was an effective bias mitigation strategy, eliminating this trend with increasing sampling rate. (C) Estimation bias as a function of number of units (neurons). We generated 5 additional datasets (Supplementary Table 1, Scaling, $q$; see also Section 6.1), comprising $M = 2$ groups with $q_m = 500$ units each. We then fit mDLAG models to observations from increasing subsets of units per group (10 to 500). Left: GP timescale estimates versus number of units (the total across both groups). Right: GP time delay estimates versus number of units. Solid traces indicate the mean of estimates, and shading (where visible) indicates standard error of the mean, computed across 5 runs. Increasing the number of units had distinct effects on GP timescale estimation and GP time delay estimation. GP timescales were underestimated for smaller numbers of units (20 – 140), but slightly overestimated for larger number of units (188 – 500). GP time delays were increasingly underestimated with increasing number of units. It has been reported previously that methods that incorporate Whittle quasi-likelihoods can experience increased bias in high-dimensional regimes (Guillaumin et al., 2022). Tapering (gold) was an effective bias mitigation strategy.




Supplementary Figure 8: Biases in Gaussian process (GP) timescale estimation and GP time delay estimation are decoupled phenomena. (A) GP timescale estimates versus number of time points per trial. Magenta: mDLAG-frequency estimates with all parameters freely estimated (Freq.; same as in Fig. 7A). Gold: mDLAG-frequency estimates with GP time delays fixed at the ground truth value (Freq., fixed delay). (B) Magenta: mDLAG-frequency estimates with all parameters freely estimated (Freq.; same as in Fig. 7B). Gold: mDLAG-frequency estimates with GP timescales fixed at the ground truth value (Freq., fixed timescale). Across panels (A) and (B), black dashed lines indicate ground truth parameter values. Solid traces indicate the mean of estimates, and shading (where visible) indicates standard error of the mean, computed across 20 runs. Regardless of which parameter was held fixed, the bias remained in estimates of the other parameter (gold curves largely coincide with magenta curves), suggesting that these biases are decoupled effects.




Supplementary Figure 9: Understanding biases in Gaussian process (GP) parameter estimation as a covariance matrix approximation problem. (A) Left: An example GP covariance matrix ($K$), corresponding to the mDLAG-time model. Right: The corresponding circulant approximation of $K$ ($\hat{K}$), arising implicitly from the mDLAG-frequency model. Both $K$ and $\hat{K}$ shown here were generated with the same squared exponential GP timescale (100 ms) and time delay (80 ms), for trial length $T = 25$ time points (with 20 ms sampling period) and $M = 2$ groups. (B) Approximation error between the covariance matrix $K$ and its circulant approximation $\hat{K}$ as a function of the GP timescale used to generate $\hat{K}$. Gray dashed line: The "ground truth" GP timescale used to generate $K$. Red point: The GP timescale of the circulant approximation $\hat{K}$ that minimizes approximation error. (C) Approximation error between the covariance matrix $K$ and its circulant approximation $\hat{K}$ as a function of the GP time delay used to generate $\hat{K}$. Gray dashed line: The "ground truth" GP time delay used to generate $K$. Red point: The GP time delay of the circulant approximation $\hat{K}$ that minimizes approximation error. For panels (B) and (C), matrices were generated with $T = 25$ time points and $M = 2$ groups. Approximation error was computed as the normalized Frobenius norm between $K$ and $\hat{K}$, $\|K - \hat{K}\|_{\mathrm{F}} / \|K\|_{\mathrm{F}}$.




Supplementary Figure 10: Using mDLAG-time to fine-tune a model fit by mDLAG-frequency (example Neuropixels dataset). (A) GP timescale progress with each fitting iteration. (B) GP time delay progress with each fitting iteration. Each colored trace corresponds to one latent variable. Colors are consistent across panels, so that purple represents the same latent variable in panel (A) and (B), and so on. For visual clarity, we show only latents found to be significantly shared between areas V1 and V2. Vertical dashed lines indicate the iteration at which mDLAG-frequency converged and fine-tuning with mDLAG-time began. Fine-tuning then took place for another 500 iterations, at which point GP parameters settled into slightly different optimal values. After 500 iterations of fine-tuning with mDLAG-time, GP timescales and time delays settled into slightly different values from the values at which mDLAG-frequency converged. Fine-tuning is a straightforward strategy to mitigate parameter estimation biases.