PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

Filter inference: A scalable nonlinear mixed effects inference approach for snapshot time series data

David Augustin[1]*, Ben Lambert[2], Ken Wang[3], Antje-Christine Walz[3], Martin Robinson[1], David Gavaghan[1]

1 Department of Computer Science, University of Oxford, Oxford, United Kingdom, **2** College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, United Kingdom, **3** Research and Early Development, F. Hoffmann-La Roche AG, Basel, Switzerland

* david.augustin@cs.ox.ac.uk

OPEN ACCESS

Citation: Augustin D, Lambert B, Wang K, Walz A-C, Robinson M, Gavaghan D (2023) Filter inference: A scalable nonlinear mixed effects inference approach for snapshot time series data. PLoS Comput Biol 19(5): e1011135. https://doi.org/10.1371/journal.pcbi.1011135

Editor: Jason M. Haugh, North Carolina State University, UNITED STATES

Received: November 1, 2022

Accepted: April 26, 2023

Published: May 22, 2023

Data Availability Statement: All data and code used for running experiments, model fitting, and plotting is available on a GitHub repository at https://github.com/DavAug/filter-inference.

Funding: This work was supported by the UK Engineering and Physical Sciences Research Council [grant number EP/S024093/1]; and the Biotechnology and Biological Sciences Research Council [grant number BB/P010008/1]. D.A. acknowledges EPSRC for studentship support via the Doctoral Training Centre in Sustainable

Abstract

Variability is an intrinsic property of biological systems and is often at the heart of their complex behaviour. Examples range from cell-to-cell variability in cell signalling pathways to variability in the response to treatment across patients. A popular approach to model and understand this variability is nonlinear mixed effects (NLME) modelling. However, estimating the parameters of NLME models from measurements quickly becomes computationally expensive as the number of measured individuals grows, making NLME inference intractable for datasets with thousands of measured individuals. This shortcoming is particularly limiting for snapshot datasets, common e.g. in cell biology, where high-throughput measurement techniques provide large numbers of single cell measurements. We introduce a novel approach for the estimation of NLME model parameters from snapshot measurements, which we call filter inference. Filter inference uses measurements of simulated individuals to define an approximate likelihood for the model parameters, avoiding the computational limitations of traditional NLME inference approaches and making efficient inferences from snapshot measurements possible. Filter inference also scales well with the number of model parameters, using state-of-the-art gradient-based MCMC algorithms such as the No-U-Turn Sampler (NUTS). We demonstrate the properties of filter inference using examples from early cancer growth modelling and from epidermal growth factor signalling pathway modelling.

Author summary

Nonlinear mixed effects (NLME) models are widely used to model differences between individuals in a population. In pharmacology, for example, they are used to model the treatment response variability across patients, and in cell biology they are used to model the cell-to-cell variability in cell signalling pathways. However, NLME models introduce parameters, which typically need to be estimated from data. This estimation becomes computationally intractable when the number of measured individuals—be they patients

Approaches to Biomedical Science: Responsible and Reproducible Research, as well as the Clarendon Fund for studentship support. B.L., M.R. and D.G. acknowledge support from the EPSRC Centres for Doctoral Training Programme. D.G. acknowledge support from a Biotechnology and Biological Sciences Research Council project grant. A-C.W. and K.W. are employees of F. Hoffmann La Roche Ltd. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: I have read the journal's policy and the authors of this manuscript have the following competing interests: KW and ACW are employees and shareholders of F. Hoffmann-La Roche Ltd. DA, BL, MR and DG have declared that no competing interests exist.

or cells—is too large. But, the more individuals are measured in a population, the better the variability can be understood. This is especially true when individuals are measured only once. Such snapshot measurements are particularly common in cell biology, where high-throughput measurement techniques provide large numbers of single cell measurements. In clinical pharmacology, datasets consisting of many snapshot measurements are less common but are easier and cheaper to obtain than detailed time series measurements across patients. Our approach can be used to estimate the parameters of NLME models from snapshot time series data with thousands of measured individuals.

Introduction

Variability is an intrinsic property of biological systems and is often the reason for their complex behaviour [1]. Examples are plentiful. One is the evolution of organisms, whereby variability in the genetic material across individuals is one of the key drivers for adaptation of populations [2]. Another is the human adaptive immune system, wherein variability in the antigen binding sites across antibodies is crucial for the defence against a large variety of pathogens [3]. However, variability in the function and regulation of cells is also the cause of many diseases, such as cancer and Alzheimer's disease [4–6]. Quantifying variability is therefore central to understanding many biological systems.

Nonlinear mixed effects (NLME) modelling is a popular approach to model variability in populations [7, 8]. NLME models introduce a set of model parameters which typically need to be estimated from measurements. However, the inference of NLME models from measurements quickly becomes prohibitively expensive when the number of measured individuals increases [9]. This shortcoming is particularly limiting when individual entities can only be measured once, since such 'snapshot' measurements do not capture individual trajectories and are therefore relatively uninformative about the dynamics across individuals, requiring large numbers of snapshot measurements for good inference results. In this article, we introduce a novel inference approach, which we call filter inference. We demonstrate that filter inference provides a scalable inference approach for snapshot time series measurements.

Snapshot measurements are particularly common in cell biology, where experimental techniques, such as single-cell RNA sequencing and flow cytometry, provide high-throughput measurements without the possibility to repeatedly measure individual cells [10–12]. The availability of snapshot measurements paired with the limitations of the NLME inference has led to the development of a variety of inference methods. Hasenauer et al (2011) simulate measurements to construct an approximate likelihood for the model parameters using kernel density estimation (KDE) [9]. To this end, they make explicit assumptions about the population parameter distribution. Dixit et al (2020) use the simulated measurements to fit the histogram of the observed measurements exactly, making no explicit assumptions about the population parameter distribution [13]. Instead, they require that the entropy of the model parameter distribution is maximised. Lambert et al (2021) use exhaustive simulations from a prior distribution of the model parameters to construct a contour volume distribution which enables an efficient inference of the model parameters [14]. Similar to Dixit et al's method, Lambert et al's approach also does not make any explicit assumptions about the population parameter distribution. However, it does require the simplifying assumption that measurement noise is negligible for the inference. Browning et al (2022) use an ABC inference approach to infer NLME models from snapshot measurements, where summary statistics of simulated and observed measurements are compared [15]. A comprehensive treatment of estimating NLME model

parameters with likelihood-free methods, such as ABC and the Bayesian synthetic likelihood (BSL) approach, is provided by Drovandi et al (2022) [16].

With filter inference, we extend Hasenauer et al's inference approach. We show that the simulation-based KDE approximation of the likelihood can be generalised to approximating the likelihood with any distribution based on simulated measurements. The choice of the distribution acts as an information filter for the comparison between simulated and real measurements and influences the quality of the inference results. We show that choosing distributions taking into account the nature of the problem can improve the inference results. The distribution or *filter* choice in filter inference has similarities with the choice of summary statistics in ABC or BSL. We use this similarity to systematically study the properties of filter inference and the consequences of different filter choices on the parameter estimates. We also introduce a differentiable form of the approximate likelihood, making filter inference applicable for state-of-the-art gradient-based sampling algorithms, such as Hamiltonian Monte Carlo (HMC) and the No-U-Turn sampler (NUTS) [17–19]. This improves the inference efficiency, especially for NLME models with many parameters.

The body of this article is divided into two sections: a methods and a results section. In the methods we review the NLME modelling framework and introduce filter inference. In the results we demonstrate the performance of filter inference for two NLME inference problems, which both suppose access to snapshot measurements: 1. for an early cancer growth model; and 2. for an epidermal growth factor (EGF) pathway model. We also use these modelling problems to demonstrate the reduction of the computational costs when using filter inference, and draw comparisons between filter inference, ABC and BSL. We conclude the article by addressing potential sources for information loss and bias. The data, models and scripts used in this article are hosted on https://github.com/DavAug/filter-inference. A user-friendly API for filter inference has been implemented in the open source Python package `chi` [20].

Methods

NLME models account for the dynamics of heterogeneous populations using a hierarchical modelling structure [7, 8]. First, a time series model, $\bar{y}(\psi, t)$, is used to model the dynamics of an individual. Here, $\bar{y}$ denotes a quantity of interest, t denotes the time and ψ denotes the parameters of the model. An example time series model for early cancer growth is illustrated in red in Fig 1, where the quantity of interest,

$$\bar{y}(\psi, t) = y_0 e^{\lambda t}, \tag{1}$$

captures a patient's tumour volume over time. The parameters of the model, $\psi = (y_0, \lambda)$, are the initial tumour volume and the growth rate.

Second, a population model, $p(\psi|\theta)$, with population parameters θ is used to capture the inter-individual variability (IIV) by modelling the distribution of ψ in the population. For example in Fig 1A, the initial tumour volume and the growth rate are normally distributed across patients

$$p(\psi|\theta) = \mathcal{N}(y_0|\mu_{y_0}, \sigma^2_{y_0})\,\mathcal{N}(\lambda|\mu_\lambda, \sigma^2_\lambda), \tag{2}$$

where $\theta = (\mu_{y_0}, \sigma_{y_0}, \mu_\lambda, \sigma_\lambda)$ denotes the population means and standard deviations of ψ. For clarity, we will refer to ψ as individual-level parameters and to θ as population-level parameters. Although equivalent, note that our notation deviates from the standard NLME literature, where the individual-level parameters are decomposed into fixed and random effects, $\psi = \bar{\psi} + \eta$ [7]. In this notation, we recover the population model in Eq 2, by letting the fixed

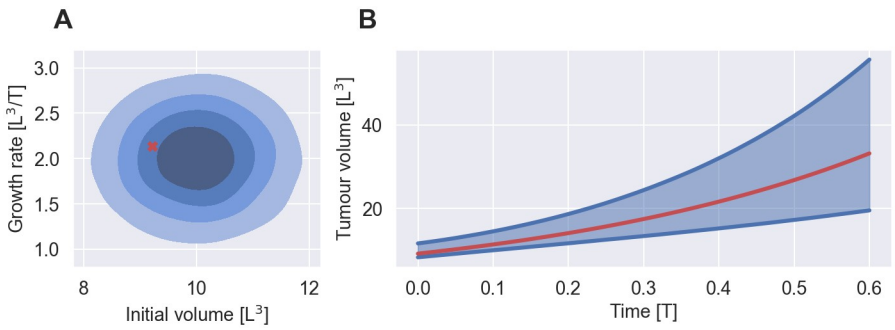

Fig 1. NLME model of early cancer growth. A: Shows the population model, $p(\psi|\theta)$, for $\theta = (10, 1, 2, 0.5)$ in shades of blue and the parameters of a randomly chosen individual in red. The shades of blue indicate the bulk 20%, 40%, 60% and 80% probability of the distribution. B: Shows the distribution of tumour volumes across individuals in the population, $p(y|\theta, t)$, in blue and the tumour volume of a randomly chosen individual over time in red. The blue lines indicate the 5th and 95th percentile of the tumour volume distribution at each time point. The quantities are shown in arbitrary units. L denotes length dimensions and T denotes time dimensions.

https://doi.org/10.1371/journal.pcbi.1011135.g001

effects, $\bar{\psi}$, be equal to the population means, and the random effects, η, be normally distributed with zero means and variances equal to the population variances.

With this hierarchical structure, the variability in the dynamics can be simulated by repeatedly evaluating the time series model for different samples of the individual-level parameters. Each sample of ψ represents an individual in the population. As the number of samples increases, the histogram over the simulations converges to the population distribution of the quantity of interest, which is illustrated in blue in Fig 1B. Formally, this population distribution is defined as

$$p(y|\theta, t) = \int \mathrm{d}\psi\, \delta(y - \bar{y}(\psi, t))\, p(\psi|\theta), \tag{3}$$

where $\delta(x)$ denotes the Dirac delta distribution. In the cancer growth example, $p(y|\theta, t)$ quantifies the probability with which a randomly chosen individual in the population has a tumour of volume y at time t. In general, the spread of $p(y|\theta, t)$ quantifies the IIV of the quantity of interest. NLME models assume that each set of individual-level parameters, ψ, fully characterises the dynamics of an individual. As a result, the heterogeneity of the dynamics in the population arises exclusively from the population model, $p(\psi|\theta)$.

Nonlinear mixed effects inference

The NLME model, as defined in Eq 3, is fully characterised by the population parameters θ. For most biological modelling problems these parameters are unknown and need to be estimated from data. Many algorithms and software packages for the inference of NLME models have been developed and excellent reviews exist [20–23]. A key feature of these inference approaches, henceforth referred to as 'traditional NLME inference', is a hierarchical representation of the data-generating process, where one time series model is calibrated to each measured individual. Here, we will review the Bayesian variant of traditional NLME inference, partly to exposit existing approaches but also to introduce concepts necessary to understand filter inference.

In order to infer parameters from measurements, it is customary to include an error model, $y = \bar{y} + \epsilon$, in the NLME model definition [7]. The error model accounts for both measurement noise and discrepancies between the model and the true process—these two processes are collectively accounted for by ϵ. This extends the deterministic time series model output to a

distribution of measurements, $p(y|\psi, t) = p(y|\bar{y}(\psi, t), \psi)$. For ease of notation, we extend the definition of ψ to also include the parameters of the error model. A common model choice is assuming normally distributed residual errors, $p(y|\psi, t) = \mathcal{N}(y|\bar{y}(\psi, t), \sigma^2)$. The distribution of measurements across individuals in the population can then be defined analogously to Eq 3

$$p(y|\theta, t) = \int \mathrm{d}\psi \, p(y|\psi, t) \, p(\psi|\theta), \tag{4}$$

where $p(y|\psi, t)$ replaces the Dirac delta distribution. Thus, Eq 3 can be interpreted as a special case of Eq 4, where measurements are assumed to capture the value of the time series model output, $\bar{y}$, without any error. Note that Eq 4 implicitly defines a joint probability distribution for measurement and individual-level parameters, $p(y, \psi|\theta, t) = p(y|\psi, t)p(\psi|\theta)$, which is marginalised over ψ on the right hand side of Eq 4: $p(y|\theta, t) = \int \mathrm{d}\psi \, p(y, \psi|\theta, t)$.

Given measurements across individuals, the joint probability distribution, $p(y, \psi|\theta, t)$, can be used to define a hierarchical log-likelihood for the model parameters

$$\log p(\mathcal{D}, \Psi|\theta) = \sum_{ij} \log p(y_{ij}|\psi_i, t_j) + \sum_i \log p(\psi_i|\theta), \tag{5}$$

quantifying the likelihood of parameter values, (θ, Ψ), to capture the observed dynamics, $\mathcal{D} = (Y, T)$. We use $\Psi = (\psi_1, \psi_2, \ldots, \psi_N)$ to denote the individual-level parameters across N measured individuals, and Y to denote the associated matrix of measurements across individuals and times. In particular, the ijth element of Y, y_{ij}, denotes the measurement of individual i at time t_j, where the vector of all unique measurement times is denoted by T. As a result, Y has N rows, and $K = \dim(T)$ columns. Missing measurement values do not contribute to the likelihood. Eq 5 shows that the hierarchical likelihood comprises a term accounting for the likelihoods of individual-level parameters to describe the measurements, and a term accounting for the likelihood of the population parameters to describe the distribution of the individual-level parameters.

An example dataset suitable for the inference of the early cancer growth model is outlined in Table 1. For simplicity, we neglect challenges of the measurement process and assume that it is possible to measure the tumour volume across patients *in vivo*. In practice, it may be more feasible to use the *in vitro* proliferation of cancerous cells from tissue samples as a proxy for

Table 1. Outline of an example tumour volume dataset. The dataset contains (fictitious) time series measurements of tumour volumes across patients. Patients are labelled with unique IDs. The time and tumour volume are presented in arbitrary units. T indicates the time dimension and L the length dimension.

	ID	Time [T]	Tumour volume [L^3]
1	1	1.5	11.00
2	2	1.5	8.30
3	1	2.1	11.52
4	3	2.1	9.80
5	1	4	12.03
6	2	4	8.50
⋮	⋮	⋮	⋮

https://doi.org/10.1371/journal.pcbi.1011135.t001

the tumour growth. For inference, the dataset can be expressed in matrix form

$$Y = \begin{pmatrix} 11.00 & 11.52 & 12.03 & \cdots & y_{1K} \\ 1.5 & \text{NA} & 8.50 & \cdots & y_{2K} \\ \text{NA} & 9.80 & \text{NA} & \cdots & y_{3K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{N1} & y_{N2} & y_{N3} & \cdots & y_{NK} \end{pmatrix} \quad (6)$$

with $T = (1.5, 2.1, 4, \ldots, t_K)$, where the first row contains the measurements of the patient with ID 1, the second row contains the measurements of the patient with ID 2, and so on. NA denotes missing values. An important feature of the dataset is that individuals are not necessarily measured with the same frequency, or at the same time points. In the extreme, the dataset may contain only one measurement per individual, i.e. snapshot measurements.

In a Bayesian inference approach, Bayes' rule is used to translate the hierarchical likelihood into a distribution of parameter values consistent with the observations and prior knowledge, also known as the posterior distribution

$$\log p(\theta, \Psi|\mathcal{D}) = \log p(\mathcal{D}, \Psi|\theta) + \log p(\theta) + \text{constant}, \quad (7)$$

where $p(\theta)$ is the prior distribution of the population-level parameters [24]. $p(\theta)$ is used in Bayesian inference to quantify knowledge about parameter values and is a modelling choice. The model parameters can now be inferred from $p(\theta, \Psi|\mathcal{D})$ using sampling algorithms, such as Markov chain Monte Carlo (MCMC) algorithms [19], see Alg 1. This concludes the review of traditional Bayesian NLME inference.

For simplicity, Alg 1 in Box 1 illustrates the inference using the Metropolis-Hastings (MH) algorithm [25]. However, the large dimensionality of the posterior, $p(\theta, \Psi|\mathcal{D})$, will often limit the sampling efficiency of the MH algorithm in practice, necessitating the use of more advanced MCMC algorithms, such as Hamiltonian Monte Carlo (HMC) or the No-U-Turn sampler (NUTS) [17, 18].

Filter inference

The intractability of traditional NLME inference for snapshot data stems from the increasing cost of evaluating the log-likelihood as the number of measured individuals grows. This is because the evaluation of the log-likelihood defined in Eq 5 requires one evaluation of the time series model for each observed individual, resulting in computational costs that increase at least linearly with the number of measured individuals. This expense renders traditional NLME inference intractable when thousands of individuals are measured, especially when time series models are defined by systems of differential equations that need to be solved numerically.

In theory, this intractability can be avoided by fitting to the measurements on a population-level, removing the need to evaluate the time series model for each individual separately. In particular, using the population distribution of measurements defined in Eq 4, a log-likelihood for the population-level parameters can be defined directly, $\log p(\mathcal{D}|\theta) = \sum_{ij} \log p(y_{ij}|\theta, t_j)$. From this population-level log-likelihood, we can derive a log-posterior, $\log p(\theta|\mathcal{D}) = \log p(\mathcal{D}|\theta) + p(\theta) + \text{constant}$, which can be inferred using MCMC sampling.

Box 1. Algorithm 1: Traditional Bayesian NLME inference using MH MCMC sampling. The details of the proposal and acceptance step are omitted for clarity, but may be found in [25].

```
Input  : 1. Hierarchical log-likelihood: log p(𝒟, Ψ|θ);
         2. Log-prior: log p(θ);
         3. Starting point: (θ^(0), Ψ^(0));
         4. Metropolis-Hastings sampler: Sampler;
         5. Number of iterations: n
Output: Samples from the posterior p(θ, Ψ|𝒟).
samples = []          // Initialise sampling
θ = θ^(0), Ψ = Ψ^(0)
score = log p(𝒟, Ψ|θ) + log p(θ)       // Eval. log p(θ, Ψ|𝒟) up to
const.
for i ← 1 to n do
  θ′, Ψ′ = Sampler.propose(θ, Ψ)       // Propose next sample
  score′ = log p(𝒟, Ψ′|θ′) + log p(θ′)     // Eval. log p(θ′, Ψ′|𝒟) up to
const.
  accepted = Sampler.check(score′, score)       // Accept or
reject
  if accepted then
    θ = θ′, Ψ = Ψ′, score = score′       // Continue from proposal
  end if
  samples.append([θ, Ψ])       // Store current sample
end for
return samples
```

However, in practice, the integral in Eq 4 is too expensive to compute to make $p(y|\theta, t)$ tractable for inference.

To address the computational costs, Browning et al (2022) propose a moment matching algorithm, where $p(y|\theta, t)$ is approximated by its first moments [26]. Using a truncated Taylor expansion, they derive approximate expressions for these moments, making their estimation from just one evaluation of the time series model and its higher order derivatives possible, thereby resolving the computational bottleneck.

Two alternative inference approaches, widely applicable to problems with intractable likelihoods, are ABC and BSL [27, 28]. Here, summary statistics of simulated measurements are compared to summary statistics of the observed measurements in order to construct an approximate log-likelihood. In ABC, the similarity of the summary statistics is quantified using kernel functions whose acceptance scale is defined by manually chosen error margins. In contrast, BSL uses repeated simulation of summary statistics and parametric distributions to construct synthetic likelihoods for the summary statistics [28–30]. For a detailed introduction to NLME inference using ABC or BSL, we refer to [16].

Filter inference is a conceptually related NLME inference approach, with elements from moment matching, ABC and BSL. Similar to moment matching, filter inference approximates the population measurement distribution to estimate the log-likelihood of model parameters. However, instead of using a Taylor expansion, it uses parametrised distributions constructed from simulated measurements to do so

$$\log p(\mathcal{D}|\theta) \approx \sum_{ij} \log p(y_{ij}|\tilde{Y}_j(\theta)). \tag{8}$$

$p(y|\tilde{Y})$ denotes the approximate distribution, which we will refer to as *filter*. $\tilde{Y}_j = (\tilde{y}_{1j}, \ldots, \tilde{y}_{Sj})^t$ denotes S simulated measurements sampled from $p(y|\theta, t_j)$ at time t_j. Filters can be constructed from any summary statistics of the simulated measurements, including their moments. As a result, filter inference may be seen as a sampling-based generalisation of Browning et al's moment matching algorithm.

The filter choice in filter inference has similarities with the choice of summary statistics in ABC or BSL and can be used to capture different information about the measurement distribution. For example, a Gaussian filter, introduced below in Filters, compares the mean and variance of the measurements, while a Lognormal filter compares the median and the scale of the measurements. In contrast to ABC, filter inference does not require manually chosen error margins, while also avoiding BSL's repeated simulation of summary statistics per log-likelihood evaluation (see Relationship to ABC and BSL). The algorithmic details of filter inference are presented in Alg 2 (Box 2).

Alg 2 (Box 2) uses MH MCMC sampling, similar to Alg 1 (Box 1), to infer the posterior distribution. The main difference between the algorithms is the replacement of the hierarchical log-likelihood evaluation by the estimate of the population-level log-likelihood, defined in Eq 8. In particular, we estimate the log-likelihood by simulating measurements from the model, $p(y|\theta, t)$, by first sampling simulated individuals, $\tilde{\Psi}$, from the population model, $p(\psi|\theta)$, see Alg 2 (Box 2) lines 17–21. We then simulate measurements for each simulated individual by sampling from $p(y|\psi_s, t)$ in lines 24–29. Here, we use s to label simulated individuals, instead of i which we reserve for real individuals. Using the simulated measurements, we construct a filter that summarises population-level information of the measurements. The details of this construction are filter-specific and are discussed below. The filter defines a population-level distribution of measurements, which we use to estimate the likelihood of the model parameters, see lines 21–29. From this estimate we can derive an estimate of the posterior which is computationally tractable.

Filter inference makes the number of time series model evaluations independent of the number of observed individuals. In this way, filter inference remains tractable even when millions of snapshot measurements are used for parameter estimation. In particular, in Alg 2 (Box 2) the number of time series model evaluations is determined by the number of simulated individuals, S, and the number of measured time points, see line 27 and its surrounding for-loops. In an optimised implementation, this number can be reduced to a total of S time series model evaluations per log-likelihood estimation, see S1 Text. The dominant computational costs of filter inference therefore do not scale with the number of measured individuals, but instead are set by the number of simulated individuals.

This form of approximate inference was first introduced by Hasenauer et al for a specific filter choice: the lognormal KDE filter introduced below [9]. Alg 2 (Box 2) generalises this approach to a framework, where filters can be chosen specific to the needs of the inference problem. However, Hasenauer et al's algorithm reportedly becomes inefficient for models with more than a few parameters [13]. This is because the approach samples from the posterior using the Metropolis-Hastings (MH) MCMC algorithm, whose sampling efficiency is known to scale poorly with the dimension of the posterior distribution [25].

As highlighted for traditional NLME inference, efficient sampling algorithms for high dimensional models exist, such as the Hamiltonian Monte Carlo (HMC) MCMC algorithm and its variants [17, 19]. HMC uses gradient-information to produce better proposals, resulting in a higher sampling efficiency per step. However, the estimate of the log-likelihood from Eq 8 changes non-deterministically with the population-level parameters θ, making its derivatives less useful for the HMC algorithm. In particular, the estimation of the log-likelihood

Box 2. Algorithm 2: Filter inference using MH MCMC sampling. The details of the proposal and acceptance step are omitted for clarity, but may be found in [25].

```
Input  : 1. Measurements: 𝒟 = (Y, T);
         2. Filter: p(y|Ỹ);
         3. Number of simulated individuals: S;
         4. NLME model: p(y|ψ, t), p(ψ|θ);
         5. Log-prior: log p(θ);
         6. Starting point: θ^(0);
         7. Metropolis-Hastings sampler: Sampler;
         8. Number of iterations: n
Output: Samples from the posterior p(θ|𝒟).
samples = []       // Initialise sampling
θ = θ^(0)
estimate = estimateLogLikelihood(θ)
score = estimate + log p(θ)       // Approx. log p(θ|𝒟) up to
const.
for i ← 1 to n do
  θ′ = Sampler.propose(θ)       // Propose next sample
  estimate = estimateLogLikelihood(θ′)
  score′ = estimate + log p(θ′)       // Approx. log p(θ′|𝒟)
  accepted = Sampler.check(score′, score)       // Accept or
reject
  if accepted then
    θ = θ′, score = score′       // Continue from proposal
  end if
  samples.append([θ])       // Store current sample
end for
return samples
define estimateLogLikelihood (θ)
  Ψ̃ = [ ]       // Simulate individuals
  for s ← 1 to S do
    ψ_s ~ p(ψ|θ)       // Sample an individual
    Ψ̃.append(ψ_s)
  end for
  estimate = 0       // Initialise estimate
  n_times = length(T)
  for j ← 1 to n_times do
    Ỹ = [ ]       // Simulate measurements at t_j
    for ψ_s ∈ Ψ̃ do
      ȳ = ȳ(ψ_s, t_j)       // Evaluate time series model
      ỹ_sj ~ p(y|ȳ, ψ_s)       // Sample a measurement
      Ỹ.append(ỹ_sj)
    end for
    f = Σ_i log p(y_ij|Ỹ)       // Compute filter log-likelihood at
t_j
    estimate += f       // Add filter log-likelihood to
estimate
  end for
  return estimate
```

involves random sampling from the population model and from the individual-level measurement distributions, see lines 19 and 28 in Alg 2 (Box 2). As a result, estimates of the log-likelihood will vary due to the stochasticity inherent in both the population-level distribution and the measurement noise distribution, even for fixed population-level parameters.

To make gradient-based methods useful for filter inference, we can recast the log-likelihood estimate from Eq 8 into a hierarchical form that changes deterministically with its input parameters. In particular, we can define a joint distribution of measurements, simulated measurements and simulated individual-level parameters, $p(y, \tilde{Y}, \tilde{\Psi}|\theta, t) = p(y|\tilde{Y})\prod_s p(\tilde{y}_s|\psi_s, t)\, p(\psi_s|\theta)$, making the dependence on the realisation of the simulated measurements and parameters explicit. From this joint distribution, we can define a hierarchical log-likelihood comprising the filter estimate of the population-level log-likelihood, and the log-likelihood of model parameters $(\theta, \tilde{\Psi})$ to describe the simulated measurements, $\tilde{Y}$,

$$\log p(\mathcal{D}, \tilde{Y}, \tilde{\Psi}|\theta) = \sum_{ij} \log p(y_{ij}|\tilde{Y}_j) + \sum_{sj} \log p(\tilde{y}_{sj}|\psi_s, t_j) + \sum_s \log p(\psi_s|\theta). \tag{9}$$

We can use this log-likelihood and Bayes' rule to derive a log-posterior analogously to the hierarchical log-likelihood and the NLME log-posterior in Eqs 5 and 7

$$\log p(\theta, \tilde{Y}, \tilde{\Psi}|\mathcal{D}) = \log p(\mathcal{D}, \tilde{Y}, \tilde{\Psi}|\theta) + \log p(\theta) + \text{constant}. \tag{10}$$

This log-posterior depends deterministically on its parameters, $(\theta, \tilde{Y}, \tilde{\Psi})$. As a result, we can use HMC to efficiently sample from $p(\theta, \tilde{Y}, \tilde{\Psi}|\mathcal{D})$, even for high-dimensional NLME models. Once $(\theta, \tilde{Y}, \tilde{\Psi})$ are inferred, the approximate posterior for the population-level parameters can be obtained by considering only the θ estimates (i.e. marginalisation). The algorithmic details of the approach are presented in Alg 3 (Box 3). The algorithm is illustrated using a MH sampler for easier comparison with Algs 1 and 2. Note that line 11 in Alg 3 (Box 3) already implements the marginalisation over $\tilde{Y}$ and $\tilde{\Psi}$.

Importantly, the posteriors for θ inferred using the stochastic likelihood, Eq 8, and using the deterministic likelihood, Eq 9, are identical. The main difference is that the implementation of the stochastic form uses ancestral sampling to simulate individuals and measurements to estimate the log-likelihood, which makes its estimates change non-deterministically with its input parameters. This stochastic dependence is eliminated in Eq 9 by explicitly formulating a log-likelihood term for each random variable contributing to the overall likelihood of the population-level parameters. In this way, the likelihood becomes a deterministic function of the random variables $(\theta, \tilde{Y}, \tilde{\Psi})$.

Filters

Filters are the central element of filter inference, making inference of NLME models from measurements of thousands of individuals possible. Below we introduce five filters that we have found useful in our experiments.

1. Gaussian filter. A Gaussian filter summarises the population measurement distribution using a Gaussian distribution

$$p(y|\tilde{Y}_j) = \mathcal{N}(y|\tilde{\mu}_j, \tilde{\sigma}_j^2), \tag{11}$$

where $\tilde{\mu}_j$ and $\tilde{\sigma}_j^2$ are given by the empirical mean and variance of the simulated measurements,

Box 3. Algorithm 3: Filter inference (deterministic form) using Metropolis-Hastings MCMC sampling. The details of the proposal and acceptance step are omitted for clarity, but may be found in [25].

```
Input  : 1. Filter log-likelihood: log p(D, Ỹ, Ψ̃|θ);
         2. Log-prior: log p(θ);
         3. Starting point: (θ^(0), Ỹ^(0), Ψ̃^(0));
         7. Metropolis-Hastings sampler: Sampler;
         8. Number of iterations: n
Output: Samples from the posterior p(θ|D).
samples = []        // Initialise sampling
θ = θ^(0), Ỹ = Ỹ^(0), Ψ̃ = Ψ̃^(0),
score = log p(D, Ỹ, Ψ̃|θ) + log p(θ)        // Eval. log p(θ, Ỹ, Ψ̃|D) up to const.
for i ← 1 to n do
  θ′, Ỹ′, Ψ̃′ = Sampler.propose(θ, Ỹ, Ψ̃)        // Propose next sample
  score′ = log p(D, Ỹ′, Ψ̃′|θ′) + log p(θ′)        // Eval. log p(θ′, Ỹ′, Ψ̃′|D) up to const.
  accepted = Sampler.check(score′, score)        // Accept or reject proposal
  if accepted then
    θ = θ′, Ỹ = Ỹ′, Ψ̃ = Ψ̃′, score = score′        // Continue from proposal
  end if
  samples.append([θ])        // Store current sample
end for
return samples
```

$\tilde{\mu}_j = \sum_s \tilde{y}_{sj}/S$ and $\tilde{\sigma}_j^2 = \sum_s (\tilde{y}_{sj} - \tilde{\mu}_j)^2/(S-1)$, where S denotes the number of simulated individuals.

A Gaussian filter is illustrated with other filters in Fig 2, where we simulate $S = 100$ measurements from the early cancer growth model at $t = 1$, using the population model introduced in Early cancer growth model inference. We use the simulated measurements to construct the filters, e.g. for the Gaussian filter we compute the mean and variance of the simulations. A random realisation of each filter is illustrated in red. Repeating this construction 1000 times, we estimate the 5th to 95th percentile of the filter density distribution, illustrated in blue. As a reference for the filter approximations, the figure shows the exact population measurement distribution, $p(y|\theta, t)$, in black.

2. Lognormal filter. A lognormal filter summarises the population measurement distribution using a lognormal distribution

$$p(y|\tilde{Y}_j) = \mathrm{LN}(y|\tilde{\mu}_j, \tilde{\sigma}_j), \tag{12}$$

where the location and scale of the lognormal distribution, $(\tilde{\mu}_j, \tilde{\sigma}_j)$, are given by the empirical mean and standard deviation of the log-transformed simulated measurements.

3. Gaussian mixture filter. A Gaussian mixture filter summarises the population measurement distribution using a Gaussian mixture distribution

$$p(y|\tilde{Y}_j) = \frac{1}{M}\sum_{m=1}^{M} \mathcal{N}(y|\tilde{\mu}_{j,m}, \tilde{\sigma}_{j,m}^2), \tag{13}$$

where M is a hyperparameter and determines the number of Gaussian kernels. For computational efficiency, the mean and the variance of the Gaussians are estimated from the simulated measurements by assigning S/M simulated individuals to each subpopulation. The parameters of the mth kernel are then estimated using the empirical mean and variance of the measurements of the mth subpopulation.

A more general implementation of the Gaussian mixture filter, not explored in this publication, estimates the maximum likelihood estimates of the filter parameters from the simulated measurements using the expectation-maximisation (EM) algorithm [31]. While computationally less efficient, this implementation provides more flexibility, making it possible to approximate the population measurement distribution more faithfully.

4. Gaussian KDE filter. A Gaussian KDE filter summarises the population measurement distribution using a Gaussian kernel density estimation

$$p(y|\tilde{Y}_j) = \frac{1}{S}\sum_{s=1}^{S}\mathcal{N}(y|\tilde{y}_{sj}, b_j^2), \tag{14}$$

where S is the number of simulated individuals. A Gaussian KDE population filter is a Gaussian mixture population filter, where each individual is assigned to its own subpopulation, i.e. $M = S$. The bandwidth of the kernels, $\tilde{b}_j^2$, is a hyperparameter of the population filter. In this article we use the widely used rule of thumb for bandwidth selection, $b_j^2 = (4/3/S)^{2/5}\tilde{\sigma}_j^2$, following Hasenauer et al (2011) [9]. Here, $\tilde{\sigma}_j^2$ is the empirical variance of the simulated measurements at time t_j.

5. Lognormal KDE filter. A lognormal KDE filter summarises the population measurement distribution using a lognormal kernel density estimation

$$p(y|\tilde{Y}_j) = \frac{1}{S}\sum_{s=1}^{S}\mathrm{LN}(y|\tilde{y}_{sj}, b_j), \tag{15}$$

The bandwidth is computed using the rule of thumb, $b_j^2 = (4/3/S)^{2/5}\tilde{\sigma}_j^2$, where $\tilde{\sigma}_j^2$ is the empirical variance of the log-transformed simulated measurements.

Results and discussion

To demonstrate the properties of filter inference, we first infer posterior distributions from snapshot measurements for two modelling problems: 1. early cancer growth; and 2. EGF pathway signalling. We then compare the computational costs of traditional NLME inference and filter inference and quantify the impact of using NUTS on the sampling efficiency. We conclude the section by highlighting the similarities between the filter choice in filter inference and choosing summary statistics in ABC or BSL. We also illustrate how inappropriate choices of filters may result in information loss or bias. Python scripts to reproduce the results are hosted on https://github.com/DavAug/filter-inference. All models are implemented in the open-source Python package `chi` [20], which we have extended to provide a user-friendly API for filter inference. For the inference, we use `pints`' implementations of NUTS and the MH algorithm [32]. The gradients of the log-posterior, needed for NUTS, are automatically computed by `chi`, using the open-source Python package `myokit` [33].

Early cancer growth model inference

To establish that filter inference is a sound approach for the inference of NLME models, we compare the results of filter inference and NLME inference on a common dataset. We

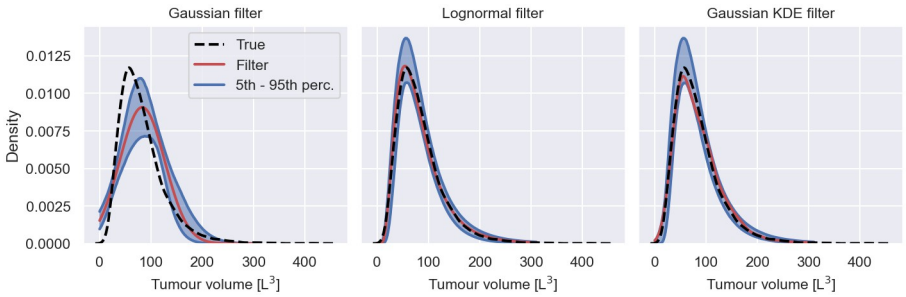

Fig 2. Filters in filter inference. The figure shows a Gaussian filter, a lognormal filter and a Gaussian KDE filter of the early cancer growth model for $S = 100$ simulated individuals at time $t = 1$. Each filter is illustrated by a randomly chosen realisation, illustrated in red, and the 5th to 95th percentile of the filter distribution for different sets of simulated individuals. As a reference, the exact population measurement distribution is illustrated in black.

https://doi.org/10.1371/journal.pcbi.1011135.g002

synthesise snapshot measurements from the early cancer growth model, Eq 1, with a Gaussian error model, $p(y|\psi, t) = \mathcal{N}(y|y_0 e^{\lambda t}, \sigma^2)$, by first sampling individual-level parameters from the population model, $p(\psi|\theta) = \mathcal{N}(y_0|\mu_{y_0}, \sigma^2_{y_0})\,\mathcal{N}(\lambda|\mu_\lambda, \sigma^2_\lambda)\,\delta(\mu_\sigma - \sigma)$. We then measure each individual by sampling from $p(y|\psi, t)$. The population parameters are set to $\theta = (\mu_{y_0}, \sigma_{y_0}, \mu_\lambda, \sigma_\lambda, \mu_\sigma) = (10, 1, 2, 0.5, 0.8)$ for the data-generation. 15 snapshot measurements are synthesised for 6 time points between 0 and 0.6 time units. The resulting dataset with a total of 90 measured individuals is illustrated by scatter points in Fig 3A. This dataset is still tractable for traditional NLME inference. The details of the inference procedure and the convergence assessment are reported in S2 Text and S1 Table.

The inference results show that filter inference with a Gaussian filter and $S = 100$ simulated individuals and NLME inference produce almost identical fits to the measurements (see Fig 3A) and similar posterior distributions (see Fig 3B). Notably, the posterior distributions of

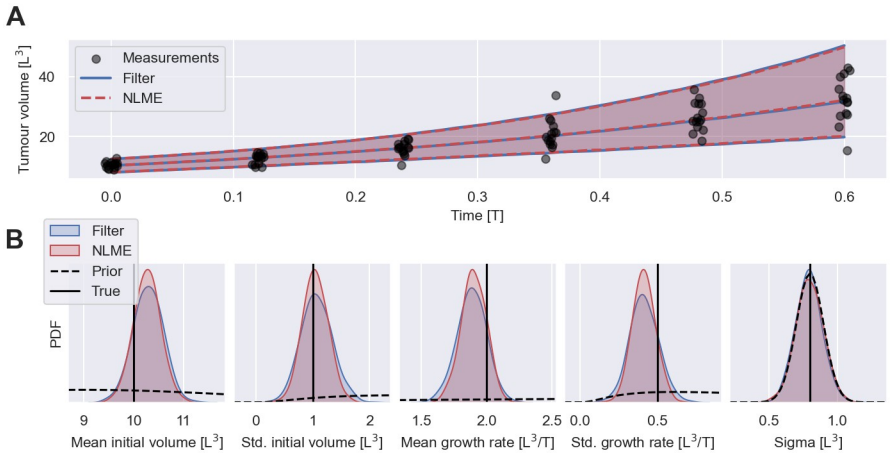

Fig 3. Filter inference versus traditional NLME inference. A: Shows 90 snapshot measurements in arbitrary units, generated from the early cancer growth model, and the fitted NLME models obtained using filter inference with a Gaussian filter (blue) and traditional Bayesian NLME inference (red). The measurements are illustrated with jitter on the time axis. The fitted models are illustrated by the medians and the 5th to 95th percentile range of the inferred measurement distributions, $\mathbb{E}_{\theta|\mathcal{D}}[p(y|\theta, t)]$. The filter is constructed using $S = 100$ simulated individuals. B: Shows the inferred posterior distributions obtained using filter inference (blue) and NLME inference (red). The data-generating parameters (solid lines) as well as the prior distributions used for the inference (dashed lines) are also shown.

https://doi.org/10.1371/journal.pcbi.1011135.g003

both approaches encompass the data-generating parameter values within their main bulk probability mass. The measurements appear highly informative about the population means of the parameters but less so about the variability in the population. Importantly, the measurements are not informative about the noise parameter, μ_σ, since the posterior distributions do not differ substantially from the prior distribution. This is because observed variability is not easily attributed to either IIV or measurement noise when individuals are not measured repeatedly. While the increase of the tumour volume variability over time indicates that IIV is present at least in one of (y_0, λ), since a Gaussian error model cannot capture heteroscedasticity, the measurements in Fig 3A leave room for attributing all observed variability at $t = 0$ to just noise, or just IIV in y_0. Thus, as long as the combined variability is of the same magnitude as the observed variability, each contribution may assume any magnitude between zero and the observed variability at $t = 0$. The variance of the measurements at $t = 0$ is 1.0 ± 0.4 (see S3 Text for the details of the estimation), while the prior of μ_σ focusses on values between 0.5 and 1 as shown in the right-most figure in Fig 3, constraining the noise variance to be at most of order $\mathcal{O}(1)$. As a result, all values from the prior of μ_σ are compatible with the observations, and the posterior is not informed by the measurements.

This lack of IIV-noise identifiability can be overcome when variability contributions from IIV and noise lead to distinct shapes of the measurement distribution, $p(y|\theta, t)$, and sufficiently many measurements are available to resolve such distributional differences. However, traditional NLME inference from large snapshot datasets is computationally intractable and filter inference requires a large number of simulated individuals in order to distinguish IIV and noise, diminishing its computational advantages (see S4 Text for a detailed discussion). The efficient inference from snapshot measurements using filter inference with only a small number of simulated individuals is therefore reliant on informative noise priors. In many applications, such priors may be informed by the specifications of measurement devices. Where possible, repeated measurements of a few individuals may also be used to estimate noise parameters.

The inferred population distributions, represented by the averages over the posterior distributions, $\mathbb{E}_{\theta|\mathcal{D}}[p(\psi|\theta)]$, are illustrated together with the data-generating distribution in the left panel of Fig 4A (see S5 Text for details on the computation of $\mathbb{E}_{\theta|\mathcal{D}}[p(\psi|\theta)]$). The comparison shows that the inference from 90 snapshot measurements with either inference method provides only a crude estimate of the IIV. This inaccuracy is a direct consequence of sampling bias —90 individuals do not faithfully depict the whole population. To test whether this sampling bias can be mitigated by increasing the number of measured individuals, we exponentially

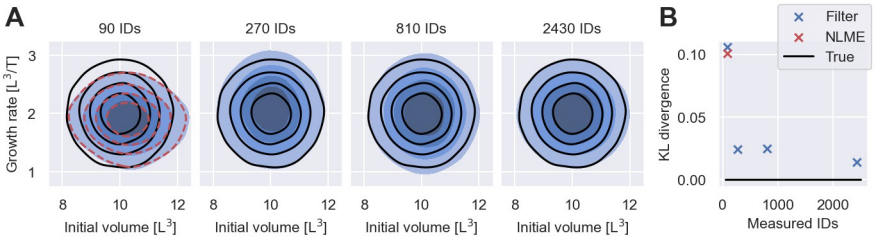

Fig 4. Quality of IIV estimates for varying dataset sizes. A: Shows inferred population distributions from snapshot measurements of varying numbers of individuals (IDs) using filter inference with a Gaussian filter and $S = 100$ simulated individuals in blue. The different shades of blue indicate the bulk 20%, 40%, 60% and 80% probability regions. The distribution inferred from measurements of 90 individuals using NLME inference from Fig 3 is illustrated by red dashed lines. The data-generating distribution is illustrated in black. B: Shows the KL divergences between the data-generating population distribution and the inferred distributions from A.

https://doi.org/10.1371/journal.pcbi.1011135.g004

increase the number of snapshot measurements per time point from 15 to $3 \times 15 = 45$, $3^2 \times 15 = 135$ and $3^3 \times 15 = 405$ snapshot measurements, resulting in datasets totalling $N = 270$, $N = 810$ and $N = 2430$ measurements. The inferred population distributions using filter inference are illustrated in panels 2, 3 and 4 of Fig 4A. The figure demonstrates that the estimation of the IIV improves with the number of measured individuals, as also quantified by the Kullback-Leibler (KL) divergence between the data-generating population distribution and the inferred distributions shown in Fig 4B. Overall, Fig 4 illustrates that many more than 90 snapshot measurements are needed to obtain accurate estimates of the IIV for the early cancer growth model. For higher dimensional NLME model, snapshot measurements from even more individuals are likely required, causing computational challenges for traditional NLME inference. In contrast, the computational costs of filter inference are not dominated by the number of measured individuals, and thus inference remains tractable.

EGF pathway model inference

The EGF pathway plays an important role in regulating the behaviour of epithelial cells and tumours of epithelial origin [34]. Understanding the cell-to-cell variability in the biochemical signalling is therefore of great interest [13]. Here, we demonstrate the ability of filter inference to estimate the parameters of a published EGF signalling pathway model from snapshot measurements.

In particular, we consider a model of inactive and active EGF receptor (EGFR) concentrations [35]

$$\frac{\mathrm{d}c_r}{\mathrm{d}t} = p - k_{\mathrm{on}} c_l c_r + k_{\mathrm{off}} c_a - k_{\mathrm{deg},r} c_r, \quad \text{and} \quad \frac{\mathrm{d}c_a}{\mathrm{d}t} = k_{\mathrm{on}} c_l c_r - k_{\mathrm{off}} c_a - k_{\mathrm{deg},a} c_a. \tag{16}$$

c_r models inactive and c_a active EGFR. The model assumes that inactive EGFR is produced at a constant rate, p. Upon binding of EGF, inactive EGFR is activated at a rate proportional to the surrounding EGF concentration c_l. Once activated, receptors deactivate at a rate k_{off}. Both, active and inactive receptors are assumed to degrade over time at rates $k_{\mathrm{deg,a}}$ and $k_{\mathrm{deg,r}}$, respectively. In our study, the cell-to-cell variability is modelled by varying production and activation rates. The remaining model parameters are fixed across cells. We synthesise two distinct datasets, each comprising 1200 cells and use the collective data from both datasets to perform filter inference. The first dataset contains snapshot measurements of cells exposed to a constant EGF concentration of $c_l = 2$ng/mL, henceforth denoted as 'data (low)'. The second dataset, 'data (high)', contains snapshot measurements of the same experiment with a higher constant EGF concentration of $c_l = 10$ng/mL. The data are simulated using a lognormal error model centred on the model outputs with scale parameter μ_σ. The measurements are generated over a period of 25 min with population parameters $\theta = (\mu_p, \sigma_p, \mu_{k_{\mathrm{on}}}, \sigma_{k_{\mathrm{on}}}, \mu_{k_{\mathrm{off}}}, \mu_{k_{\mathrm{deg},r}}, \mu_{k_{\mathrm{deg},a}}, \mu_\sigma) = (1.7, 0.05, 1.7, 0.05, 8, 0.25, 0.015, 0.05)$. Both receptor concentrations are initialised at 0 ng/mL. The generated datasets are illustrated by black scatter points (data (low)) and grey scatter points (data (high)) in Fig 5A. We infer the model parameters from the synthetic datasets using Gaussian filters with $S = 100$ simulated cells. The noise parameter, μ_σ, is fixed to the data-generating value during the inference. Details on the inference procedure are reported in S6 Text and S2 Table.

Fig 5A shows that filter inference is able to infer measurement distributions that capture the dynamics of the observed EGF signalling pathway. Fig 5B shows that the inferred posteriors assign substantial weights to the data-generating parameters. The inferred cell-to-cell variability of the model parameters is of a reasonable magnitude, as the comparison of the data-generating population distribution and the inferred distribution in Fig 5C shows. However,

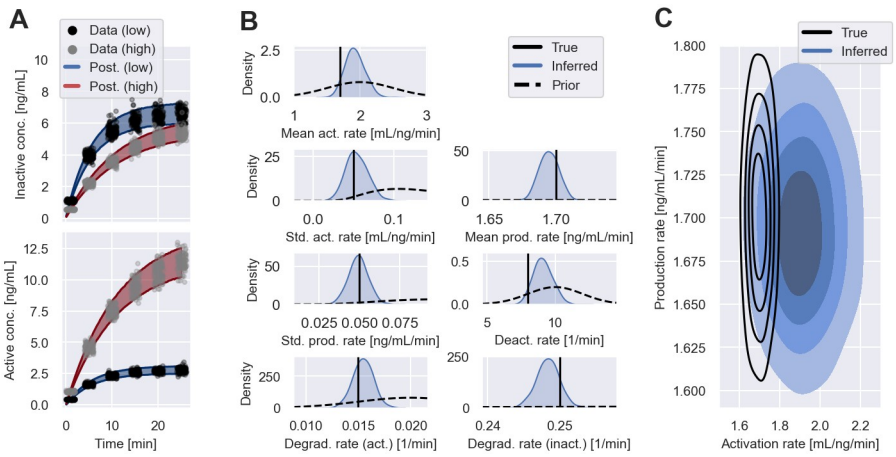

Fig 5. Inference results of EGF pathway model I. A: Shows snapshot measurements of active and inactive EGFR concentrations across cells. The cells are exposed to one of two EGF concentrations: data (low) with $c_l = 2$ng/mL (black scatter points); and data (high) with $c_l = 10$ng/mL (grey scatter points). The shaded areas illustrate the 5th to 95th percentile of the inferred measurement distributions using filter inference with Gaussian filters and $S = 100$ simulated cells. B: Shows the inferred posterior distributions of the model parameters illustrated by blue density plots, together with the data-generating parameter values illustrated by black solid lines. The density of the prior distribution is illustrated by dashed lines for each parameter. C: Shows the inferred population distribution of the production rate and the activation rate in blue. The different shades of blue indicate the bulk 20%, 40%, 60% and 80% probability regions. The probability regions of the data-generating distribution are illustrated by black contours.

https://doi.org/10.1371/journal.pcbi.1011135.g005

the data appears to be more informative about the production rate variability than the activation rate variability. To investigate this further, in Fig 6A, we plot the posterior distribution of the population-level mean activation rate versus the deactivation rate. The strong correlation between the two parameters in the posterior distribution indicates that it is not possible to identify both the activation rate and deactivation rates from the synthesised datasets. In Fig 6B, we show inference results when we fix the deactivation rate to its known data-generating value—in this case, the inferred population distribution is closer to the true one.

Scaling & computational costs

Filter inference is able to infer the parameters of NLME models from thousands of snapshot measurements. This is because the dominant computational costs of the log-posterior evaluation do not scale with the number of measured individuals, as demonstrated in Fig 7. The figure shows the evaluation time of the NLME log-posterior and its gradient (blue lines), defined in Eq 7, and the filter log-posterior and its gradient (red lines), defined in Eq 9, for the early cancer growth model and the EGF pathway model with increasing numbers of measured individuals. The filter log-posterior is defined using a Gaussian filter with varying numbers of simulated individuals. The gradients of the posteriors are automatically computed using `chi` [20]. Details on the estimation of the evaluation times are presented in S7 Text.

The figure shows that the evaluation time of the NLME log-posterior scales linearly with the number of measured individuals, while the cost of the filter log-posterior remains constant. The figure also shows that the computational costs of the filter log-posterior are roughly proportional to the number of simulated individuals, as discussed in the Methods. As a result, the speed up provided by filter inference is of order $\mathcal{O}(N/S)$, where N and S denote the number of measured and simulated individuals, respectively. This reduces the log-posterior evaluation costs 34-fold for the early cancer growth model, and 13-fold for the EGF pathway model for

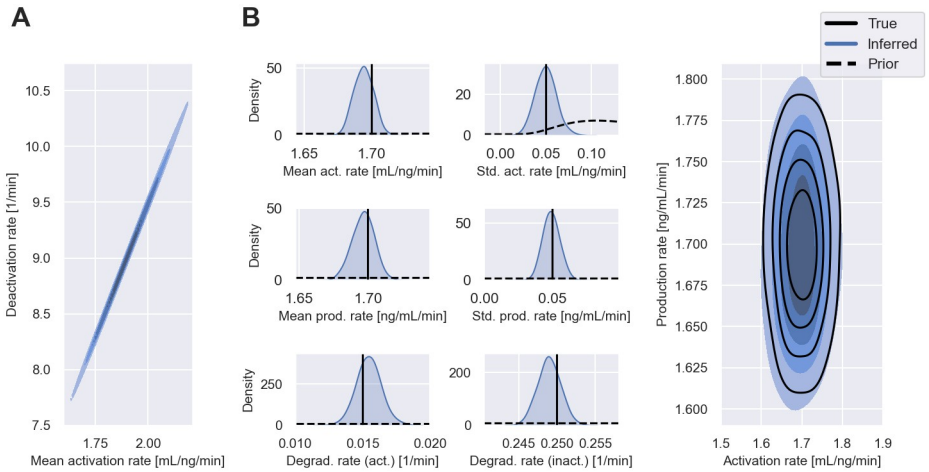

Fig 6. Inference results of EGF pathway model II. A: Shows the joint posterior distribution of the mean activation rate, $\mu_{k_{on}}$, and the deactivation rate, $\mu_{k_{off}}$ from Fig 5B. B: Shows the inferred posterior distribution and the inferred population distribution from a separate inference run, where we fixed the deactivation rate to its data-generating value. All other inference settings, including data and priors, remain unchanged from the inference approach used to generate Fig 5.

https://doi.org/10.1371/journal.pcbi.1011135.g006

datasets with 1000 snapshot measurements and filters with $S = 100$ simulated individuals. This cost reduction increases as the number of measured individuals grows.

However, in addition to the evaluation time of the log-posterior, the computational costs of inference are also determined by the total number of evaluations needed for convergence of the MCMC sampler. Since MCMC is a form of dependent sampling, there are typically auto-correlations between samples, reducing the number of i.i.d. samples drawn from the posterior distribution [36]. In order to determine the total computational costs of the approaches, we thus need to compare the number of log-posterior evaluations of traditional NLME inference and filter inference needed for convergence.

Several metrics for the convergence assessment of MCMC samples exist, such as the $\hat{R}$-metric or the R^*-metric [37, 38]. These metrics could be used to determine the number of log-posterior evaluations needed to reach a certain degree of convergence for different MCMC

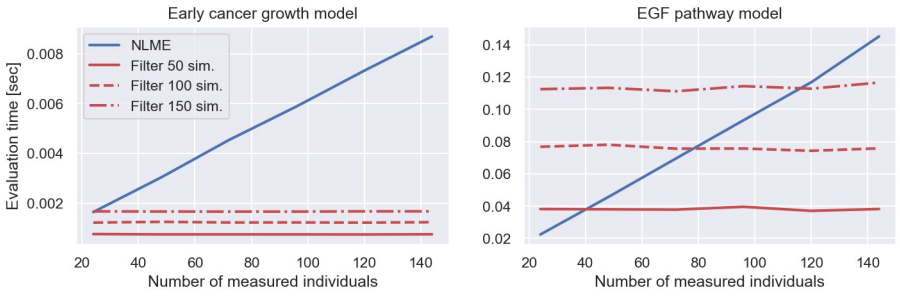

Fig 7. Computational costs of filter inference and traditional NLME inference I—Evaluation time of log-posterior. The figure shows the evaluation time of the traditional NLME log-posterior and its gradient, defined in Eq 7, (blue lines) and the filter inference posterior and its gradient, defined in Eq 10, with a Gaussian filter and $S = 50$, $S = 100$ and $S = 150$ simulated individuals (red lines). The left and right panel illustrate the results for the early cancer growth model and the EGF pathway model, respectively.

https://doi.org/10.1371/journal.pcbi.1011135.g007

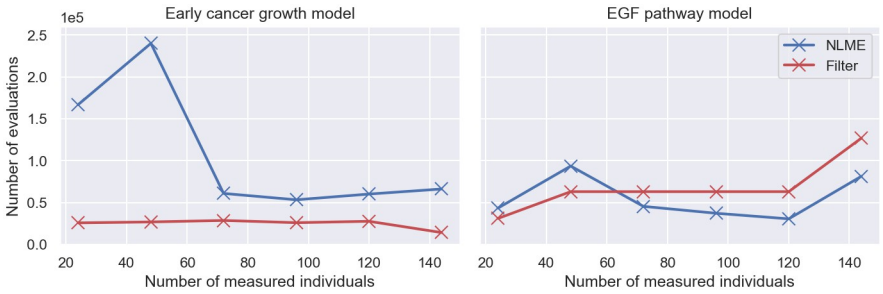

Fig 8. Computational costs of filter inference and traditional NLME inference II—Number of log-posterior evaluations. The figure shows the number of log-posterior evaluations of traditional NLME inference (blue lines) and filter inference, with a Gaussian filter and $S = 100$ simulated individuals, (red lines) for the early cancer growth model and the EGF pathway model using varying sizes of snapshot datasets. Each log-posterior evaluation includes the evaluation of its gradient. The posterior distributions are inferred using 1000 MCMC iterations of NUTS after calibrating the algorithm for 500 iterations.

https://doi.org/10.1371/journal.pcbi.1011135.g008

algorithms. An alternative approach to compare the computational costs of traditional NLME inference and filter inference is to infer the posteriors using an MCMC algorithm that uses an initial calibration phase to adjust the number of log-posterior evaluations per MCMC step in order to maximise the convergence rate across inference problems. Such an algorithm is NUTS [18, 19]. Using its calibration strategy, NUTS converges within 1000 MCMC iterations post calibration for the early cancer growth model and the EGF pathway model. We choose to estimate the total computational costs of traditional NLME inference and filter inference using the latter approach, and run 1500 NUTS iterations for both, the early cancer growth model and the EGF pathway model for different datasets with varying numbers of measured individuals (see Fig 8). The initial 500 iterations of each inference run are used to calibrate the algorithm. Each evaluation includes the evaluation of the log-posterior and its gradient.

The figure shows that the number of log-posterior evaluations of traditional NLME inference and filter inference are of the same order of magnitude for the investigated models. Each log-posterior evaluation includes the evaluation of the log-posterior gradient. For the early cancer growth model, NUTS requires fewer evaluations during filter inference, while for the EGF pathway model NUTS evaluates the log-posterior less often during traditional NLME inference. Overall, the linear cost scaling of traditional NLME with the number of measured individuals, and the comparable total number of evaluations demonstrate that the benefit to using filter inference scales linearly with the number of measured individuals, meaning, for large datasets typical in cell biology, filter inference can be orders of magnitude faster than traditional NLME inference.

Sampling efficiency

Filter inference can reduce the costs of inference in both its stochastic and its deterministic form. In this section, we investigate the degree to which the deterministic version and the use of NUTS improves the efficiency of filter inference.

To this end, we estimate the sampling efficiency of the approaches using the effective sample size (ESS) metric, defined in [39]. The ESS estimates the number of i.i.d. samples drawn from each posterior dimension by an MCMC algorithm. Thus, the larger the ESS for a fixed number of log-posterior evaluations, the better the sampling efficiency of the algorithm. In Fig 9 we show the minimum ESS across dimensions, obtained from inferring the early cancer growth model posterior and the EGF pathway model posterior. We infer the posteriors twice:

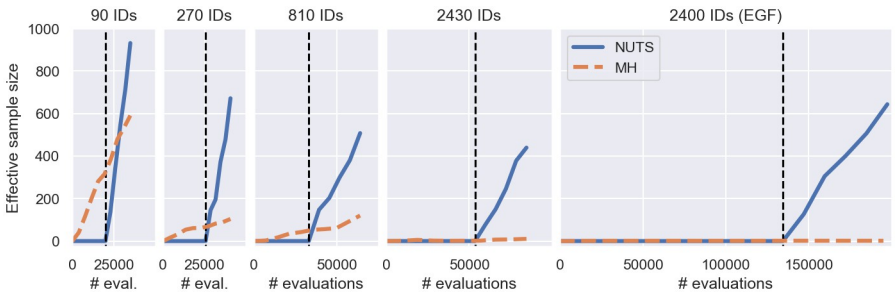

Fig 9. Sampling efficiency of filter inference variants. The figure shows the minimum ESS across dimensions as a function of log-posterior evaluations for different posterior distributions inferred with: 1. NUTS and the deterministic filter posterior (blue); and 2. MH and the stochastic filter posterior (orange). For NUTS, the number of evaluations include evaluations of the log-posterior gradient. For MH, the log-posterior gradient is not evaluated. Panels 1, 2, 3 and 4 show the minimum ESS of the cancer growth model posteriors from Fig 4 and panel 5 shows the EGF pathway model posteriors from Fig 6B.

https://doi.org/10.1371/journal.pcbi.1011135.g009

1. using the MH algorithm and filter inference with a stochastic posterior, defined in Alg 2; and 2. using NUTS and filter inference with a deterministic posterior, defined in Alg 3.

The figure shows that the sampling efficiency is improved when using NUTS. After 35,000 iterations, the MH algorithm generated an ESS of approximately 600 when the posterior was inferred from the cancer growth dataset with 90 measured individuals (see left panel in Fig 9), while NUTS generated an ESS of 932 after 1500 iterations, using the first 500 iterations for calibration. These 1500 iterations translate into 34,985 log-posterior and gradient evaluations (see S3 Table) out of which the majority (20,000 evaluations) are from the calibration phase and therefore do not contribute to the ESS. Evaluating the gradient in addition to the log-posterior using `chi`'s forward sensitivities approximately doubles the evaluation costs for the considered problems (see S4 Fig). As a result, NUTS generated an ESS that is 1.5 times greater with approximately the same ($2 \times 14,985/35,000 \approx 1$) computational costs post calibration. This efficiency advantage becomes more pronounced as the number of measured individuals increases. Despite extensive efforts to tune its hyperparameters (see S8 Text for details), the ESS of the MH algorithm is smaller than 10 after 90,000 MH iterations for both the early cancer growth model and the EGF pathway model, when datasets with thousands of measurements are used for the inference (see panels 4 and 5 in Fig 9). At the same time, NUTS is able to generate an ESS of order 100 for all datasets and models by automatically tuning the number of evaluations per MCMC step during the calibration phase. The number of log-posterior evaluations across all inference runs are reported in S3 Table.

Our study indicates that filter inference with the deterministic posterior and NUTS improves the sampling efficiency across inference problems relative to the MH algorithm and the stochastic variant. This improved efficiency is achieved, despite the increase of the posterior dimension from θ to $(\theta, \tilde{Y}, \tilde{\Psi})$. For example, for the cancer growth problem θ has 5 dimensions, while $(\theta, \tilde{Y}, \tilde{\Psi})$ has 805 dimensions. For more advanced gradient-free sampling methods than the MH algorithm or problems where the gradients cannot be accurately computed, this efficiency advantage may change.

Fig 9 also reveals that sampling from filter posteriors appears to be more challenging when more individuals are measured (see ESS per evaluation). While NUTS is still able to achieve good sampling efficiencies by adaptively using more log-posterior evaluations during the calibration phase, we were not able to achieve comparable sampling rates with the MH algorithm and the stochastic filter posterior. To understand this behaviour, we investigate the relationship between filter inference, ABC and BSL in the next section.

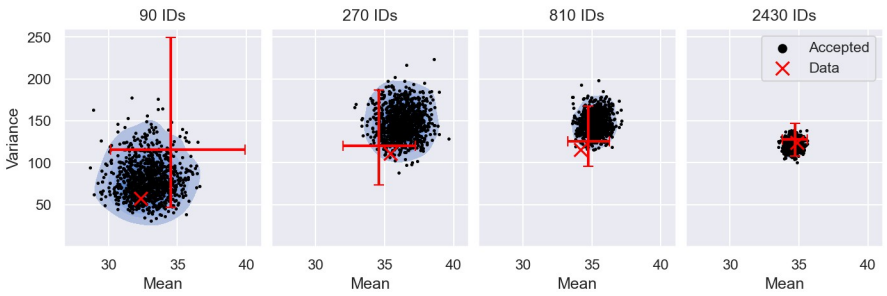

Fig 10. ABC interpretation of filter inference. The figure shows the accepted means and variances of the Gaussian filter at $t = 0.6$ from the inference results in Fig 4, where filter inference is performed on datasets with 90, 270, 810 and 2430 snapshot measurements of the cancer growth model. The accepted summary statistics are illustrated as black scatter points with the corresponding KDE plots shown in blue. The summary statistics of each dataset is illustrated by a red scatter point. The sample variation of the dataset summary statistics is represented by the 5th to 95th percentile of the summary statistic distribution (red bars), estimated from 1000 realisations of each dataset. The exact mean and variance of the data-generating distribution is close to the intersection of the bars.

https://doi.org/10.1371/journal.pcbi.1011135.g010

Relationship to ABC and BSL

In the Methods, we highlighted that the filter choice in filter inference has similarities with the summary statistics choice in ABC or BSL. In this section we illustrate this similarity and use it to shed light on the result from the previous section that the deterministic filter posterior appears to be better suited for the inference from large numbers of snapshot measurement than the stochastic variant.

ABC is a hugely successful inference strategy across fields, such as population genetics [40], epidemiology [41] and climate modelling [42]. A common criticism of ABC is the need for manually choosing error margins in order to quantify the distance between summary statistics of real and simulated measurements, although methods for automatically tuning the error margin exist [43, 44]. Too large error margins reduce the quality of the inference results, while too narrow error margins lead to high rejection rates and, thus, to a poor sampling efficiency.

Filter inference does not require manually chosen error margins, despite some filters, such as the Gaussian filter, being defined by summary statistics. We will denote such filters henceforth as summary statistics-based (SS-based) filters. In Fig 10 we show that error margins arise naturally for SS-based filters in filter inference using Gaussian filters as an example.

The figure revisits the filter inference results from Fig 4 and shows the simulated means and variances for $t = 0.6$ (black points), accepted during the inference. The first panel illustrates the inference results for the dataset with 90 snapshot measurements. Panels 2, 3 and 4 correspond to the datasets with 270, 810 and 2430 snapshot measurements, respectively. As a reference, we visualise the mean and variance of each dataset (red cross) and the true mean and variance of the data-generating process (red bars). We estimate the true mean and variance of the data-generating process repeatedly by sampling 1000 realisations of each dataset. For each dataset size, we then compute the means and the variances of the dataset realisations and plot the 5th to 95th percentile interval in the corresponding panel (red bars). The true mean and variance of the data-generating distribution are approximately equal to the median, i.e. the value marked by the intersection of the red bars. As the true data-generating mean and variance are independent of the estimation procedure, the bars intersect approximately at the same point across the panels. This estimation procedure provides an estimate of data-generating mean and variance, as well as the sampling variation of the data summary statistics, illustrated by the size of the 5th to 95th percentile intervals.

The figure shows that filter inference accepts simulated summary statistics within an error margin around the data summary statistics (blue area), notably without explicitly computing the data summary statistics during the inference. The magnitude of the error margins scales with the number of measured individuals and is comparable to the 5th to 95th percentile interval of the data summary statistics across dataset sizes.

These observations suggest that filter inference with a Gaussian filter is similar to a special variant of ABC based on the mean and variance, where the error margin is automatically scaled with the uncertainty of the summary statistic estimates. This similarity is a consequence of the Gaussian filter construction, defined in Eq 11. The Gaussian filter likelihood assigns high likelihood only to simulated means and variances that are compatible with the data. When the dataset contains few observations, the filter likelihood is flat and permits large deviations from the mean and variance of the data, while datasets containing more observations become more restrictive. This leads to an automatic scaling of the error margin with the uncertainty of the data summary statistics. In S9 Text, we provide a proof that in the limit $N \to \infty$ filter inference with a Gaussian filter is equivalent to ABC based on the mean and variance with vanishing error margins. In S10 Text, we further prove that this equivalence also extends to other SS-based filters, provided the filter likelihood has a unique and identifiable maximum and the maximum likelihood estimates (MLEs) converge to the summary statistics of the data. We will henceforth refer to these filters as sufficient filters.

Revisiting the results in Figs 9 and 10 also provides an explanation of why the number of measured individuals reduces the ESS for the stochastic approach. Alg 2 (Box 2) simulates measurements, and thus summary statistics, for each proposal randomly. The estimation error of these simulated summary statistics is determined by the number of simulated individuals. The error margin around the data summary statistics, on the other hand, is set by the number of measured individuals. As a result, proposals close or equal to the data-generating parameters may still be randomly rejected when the error margin is smaller than the estimation error of the simulated summary statistics.

The scaling behaviour of the error margin in filter inference explains why for a fixed number of simulated individuals ($S = 100$), the rejection rate becomes larger as the number of measured individuals increases (see Fig 9). Especially, the ESS of the inference using the stochastic variant of the filter posterior deteriorates quickly with the dataset size. In contrast, the deterministic filter posterior performs significantly better, suggesting that random rejections due to ancestral sampling can be avoided by giving the sampling responsibility of all random variables involved in the posterior estimation to the MCMC algorithm.

A complementary way to understand the sampling efficiency of filter inference is provided by the BSL literature. For BSL, low sampling efficiencies have been reported when too few realisations are used to reliably estimate the mean and the covariance of the simulated summary statistics [30]. Interpreting each simulated measurement as a 'summary statistic' of the data-generating process, filter inference with a Gaussian filter is a BSL approach and therefore displays the same sampling behaviour.

Information loss and bias

In contrast to traditional NLME inference, filter inference is an approximate inference approach. This approximation may result in information loss and bias. The potential for inaccurate inference results is common to all approximate methods, including ABC and BSL, and in this case comes from the filter approximation of the population-level log-likelihood, Eq 8.

Filters construct a noisy estimate of the likelihood from measurements of a small number of simulated individuals. The fewer individuals are simulated, the lower the costs of the log-

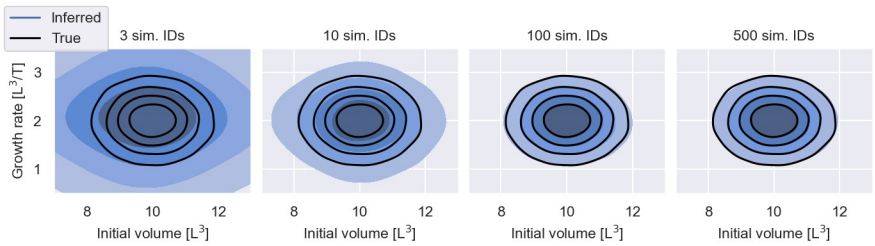

Fig 11. Information loss I—Number of simulated individuals. The figure shows inferred population distributions from 2430 snapshot measurements of the early cancer growth model, using filter inference with a Gaussian filter and $S = 3$, $S = 10$, $S = 100$ and $S = 500$ simulated individuals. The inferred distributions are visualised in blue. The different shades of blue indicate the bulk 20%, 40%, 60% and 80% probability regions. The data-generating distribution is illustrated by black contours.

https://doi.org/10.1371/journal.pcbi.1011135.g011

likelihood evaluation. This provides an incentive to reduce the number of simulated individuals as much as possible. However, as illustrated in Fig 11, there is a trade-off between computational costs and the accuracy of the IIV estimation. The figure shows inference results for the cancer growth dataset with 2430 snapshot measurements using filter inference with Gaussian filters with different numbers of simulated individuals. The fewer individuals are simulated, the more information contained in the data is lost. In this case the information loss manifests itself in an overestimation of the IIV.

In our experiments we achieve reasonable inference results with $S = 100$ simulated individuals across models. However, the number of simulated individuals that achieves an optimal tradeoff between computational costs and information loss will likely vary between problems and modelling rationales.

A second source for information loss and bias is the choice of the filter itself. To illustrate this effect we modify the cancer growth model to two patient subpopulations with different variants of cancer: an aggressive variant and a moderate variant. This is implemented by defining a covariate-dependent mean growth rate, $\mu_\lambda(\chi) = \mu_{\lambda,m} + \chi\Delta\mu_\lambda$, where $\chi = 0$ indicates patients with the moderate variant and $\chi = 1$ patients with the aggressive variant, resulting in a multi-modal population distribution

$$p(\psi|\theta,\chi) = \mathcal{N}(y_0|\mu_{y_0},\sigma^2_{y_0})\,\mathcal{N}(\lambda|\mu_\lambda(\chi),\sigma^2_\lambda)\,\delta(\mu_\sigma - \sigma). \tag{17}$$

We synthesise two datasets: one dataset with snapshot measurements from 120 individuals; and one with snapshot measurements from 3000 individuals. In both cases, half of the individuals have the aggressive cancer variant and the other half the moderate variant. The data are synthesised with $(\mu_{y_0}, \sigma_{y_0}, \mu_{\lambda,m}, \Delta\mu_\lambda, \sigma_\lambda, \mu_\sigma) = (10, 1, 2, 2, 0.5, 0.8)$.

The inference results for different choices of filters and $S = 100$ simulated individuals are illustrated in Fig 12. Where tractable, traditional NLME inference is used to infer the exact posterior distribution. Otherwise, the data-generating distribution is used as a reference for the inference results. The figure shows that the quality of the results varies substantially with the choice of the filter. The Gaussian and lognormal filters yield reasonable approximations of the overall individual-level variability, but are not able to resolve the multi-modal structure of the growth rate when only 120 patients are measured. For 3000 measured patients, both filters begin to distinguish the moderate and aggressive cancer growth subpopulations. In comparison, inference with a Gaussian mixture filter with two kernels resolves the multi-modal population structure for both numbers of measured individuals (see middle panel in Fig 12). Inference with a Gaussian KDE filter or a lognormal KDE filter similarly resolves the multi-

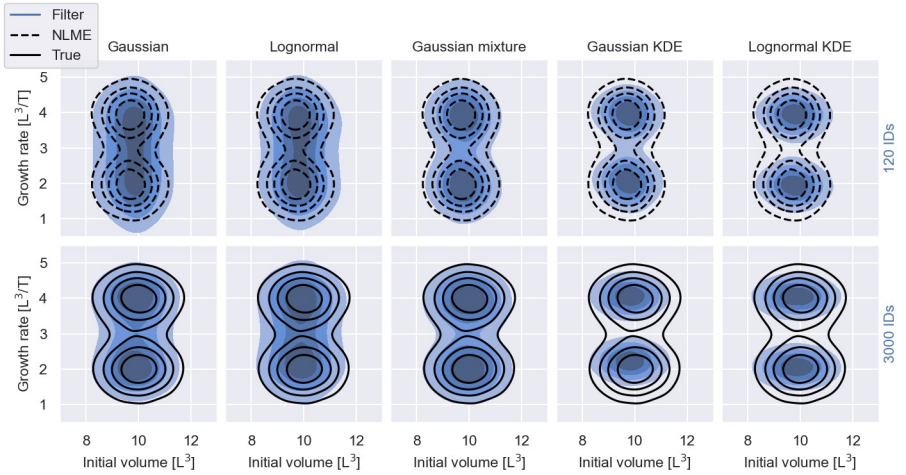

Fig 12. Information loss II—Choice of filter. The figure shows inferred population distributions from 120 (top row) and 3000 (bottom row) snapshot measurements, using filter inference with different filter choices and $S = 100$ simulated individuals. The inferred distributions, $\mathbb{E}_{\theta|\mathcal{D}}[p(\psi|\theta)]$, are visualised in blue. The different shades of blue indicate the bulk 20%, 40%, 60% and 80% probability regions. The data-generating distribution is illustrated by black solid lines. The posterior inferred with NLME inference is illustrated by black dashed lines.

https://doi.org/10.1371/journal.pcbi.1011135.g012

modal population structure, but, here, the filter posteriors underestimate the IIV for both numbers of measured individuals.

The reasons for the observed information loss and biases are different for SS-based filters and KDE-based filters. Intuitively, it is not surprising that SS-based filters with a single mode, such as the Gaussian filter and the lognormal filter, may produce inaccurate inference results when the true measurement distribution is multi-modal. We nevertheless observe in Fig 12 that both filters resolve the multi-modal structure in the population distribution when 3000 individuals are measured.

To develop an understanding for this behaviour we refer to the ABC literaure: ABC also infers an approximate posterior distribution based on summary statistics. This posterior converges to the exact posterior distribution in the limit where 1. the error margin of the summary statistics goes to zero; 2. the summary statistics are sufficient; and 3. the number of simulated measurements is the same as the number of observed measurements [45, 46]. Filter inference is equivalent to ABC for certain SS-based filters in the limit where $N \to \infty$ and the error margin of ABC vanishes (see Relationship to ABC and BSL). The filter posterior therefore also converges to the true posterior in an analogous limit where 1. the number of measured individuals goes to infinity; 2. the filters are sufficient; and 3. the number of simulated measurements is the same as the number of observed measurements. While we have no formal proof that the Gaussian filter or the lognormal filter are sufficient for the cancer growth model, the convergence of filter inference for $N \to \infty$ may provide an explanation why the Gaussian and the lognormal filters start to resolve the multi-modal population structure for 3000 measured individuals, but not for 120 measured individuals.

We support this intuition by comparing the histogram over accepted simulated measurements to the data-generating distribution in Fig 13. The closer the histogram approximates the data-generating distribution, the more accurate the inference results. The left panel shows the data-generating distribution (black) and the histogram over accepted simulations at $t = 0.6$ (blue) during the Gaussian filter inference from Fig 12 for the dataset with 3000 measured

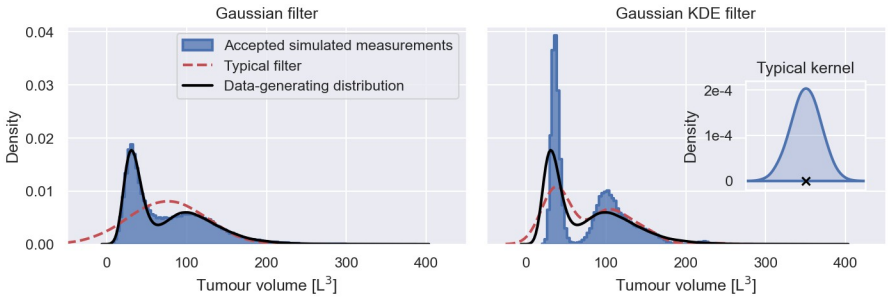

Fig 13. Accepted simulated measurements during filter inference. The figure shows the histograms over all accepted simulated measurements at $t = 0.6$ (blue) during inference with a Gaussian filter (left panel) and a Gaussian KDE filter (right panel) from the bottom panel in Fig 12. The data-generating measurement distribution is illustrated in black. A typical realisation of each filter is illustrated in red. The inset figure in the right panel shows a typical kernel (blue) placed on a simulated measurement (black cross) during the construction of the Gaussian KDE filter. The simulated measurement is placed at a tumour volume of 350 for illustration purposes. The scale of the kernel is taken from the filter realisation.

https://doi.org/10.1371/journal.pcbi.1011135.g013

individuals. The histogram over the accepted measurements approximates the multi-modal data-generating distribution well, despite the use of the unimodal Gaussian filter (see red curve for a typical Gaussian filter sampled during the inference). This indicates that for this model the means and the variances at the measured time points are sufficient statistics for the inference.

KDE-based filters do not have the same convergence behaviour as SS-based filters, as illustrated in Fig 12. In fact, the posterior distributions appear to become less accurate for the Gaussian KDE filter and the lognormal KDE filter as the number of snapshot measurements increases. In the right panel of Fig 13, we repeat the analysis of the histogram over the accepted simulations during the inference for the dataset with 3000 measured individuals for the Gaussian KDE filter. The panel shows that the histogram provides a less faithful approximation of the data-generating distribution than the histogram in the left panel, despite typical Gaussian KDE filters providing more accurate approximations of the data-generating distribution than the Gaussian filter.

To develop an understanding for this behaviour, note that the filter log-likelihood is maximised when the KL divergence between the filter and the observed measurement distribution is minimised

$$\mathrm{KL}(q \,||\, p) = \sum_j \int \mathrm{d}y \, q_j(y)(\log q_j(y) - \log p(y|\tilde{Y}_j)) \approx \text{constant} - \sum_{ij} \log p(y_{ij}|\tilde{Y}_j). \quad (18)$$

Here, $\mathrm{KL}(q||p)$ denotes the KL divergence and q_j denotes the data-generating distribution at time t_j, which we approximate by the observed measurements on the right hand side. We identify the last term on the right as the negative log-likelihood of the filter (see Eq 8). Consequently, maximising the filter likelihood is equivalent to approximating the observed measurement distribution as closely as possible.

For KDE-based filters, the objective of closely approximating the measurement distribution leads to a mismatch between the observed measurement distribution and the simulated measurement distribution. This mismatch is a direct consequence of the filter construction. KDE filters are constructed by averaging the densities of S kernels centered at the simulated measurements (see Eqs 14 and 15). Each kernel carries $1/S$ of the total probability density and has a finite width. For example for the inference in the right panel in Fig 13, typical kernels extend

50 tumour volume units in both directions and carry 1/100 of the total porbability density (see inset). As a result, simulated measurements never have to directly occupy regions of low probability density in order to approximate those regions well, as long as their kernels are wide enough to cover them. The finite probability density of the kernels makes it even unfavourable to occupy low probability density regions of the observed measurement distribution with much less than $1/S$ probability density, resulting in a bias of rejecting simulated measurements in low density regions. This explains the underrepresentation of accepted simulations in the low density regions of the data-generating distribution in the right panel of Fig 13. The absence of simulated measurements in low density regions results in an underestimation of the width of the simulated measurement distribution which, in turn, leads to an underestimation of the IIV (see Fig 12).

As the number of simulated measurements tends to infinity, the packaging of the probability density becomes more granular (see Eqs 14 and 15) and the bias towards high density regions vanishes. If the observed measurement distribution is identical to the data-generating distribution, this implies convergence to the data-generating parameters. But, if the observed measurement distribution is not representative for the whole population, KDE-based filters will overfit the observed distribution, leading to the underestimation of both, the variability in the population and the uncertainty in the parameter estimates.

In practice, inference is performed on datasets with a finite number of measured individuals using a finite number of simulated individuals. In this context, SS-based filters appear to provide a better accuracy-cost trade-off, especially when informed summary statistic choices are possible, as the middle column of Fig 12 demonstrates. Here, we infer the population distribution from the synthesised datasets using a Gaussian mixture filter with $S = 100$ simulated individuals and two kernels. Two Gaussian kernels are able to represent the bimodal structure of the observed measurement distribution more faithfully, resulting in inference results with negligible information loss.

Conclusion

Filter inference is an efficient and scaleable inference approach for NLME models, enabling the study of variability from previously intractable datasets, for example, snapshot measurements of potentially thousands of individuals. However, filter inference also introduces new challenges, such as the potential for information loss and bias, which currently can only be understood with the help of repeated synthetic data-generation and inference cycles. The efficiency and scalability of filter inference may also depend on the availability of gradients, which can be difficult to obtain for large systems of differential equations.

Supporting information

S1 Text. Optimised implementation of filter inference.
(PDF)

S2 Text. Estimation of early cancer growth model parameters.
(PDF)

S3 Text. Estimation of variance estimation error.
(PDF)

S4 Text. IIV-noise distinguishability.
(PDF)

S5 Text. Estimation of posterior average of population distribution.
(PDF)

S6 Text. Estimation of EGF pathway model parameters.
(PDF)

S7 Text. Evaluation time estimation.
(PDF)

S8 Text. Hyperparameter tuning of MH algorithm.
(PDF)

S9 Text. Equivalence of filter inference with a Gaussian filter and ABC based on the mean and variance.
(PDF)

S10 Text. Equivalence of filter inference with identifiable summary statistics-based filters and ABC based on the same summary statistics.
(PDF)

S1 Table. Convergence statistics of MCMC chains during the parameter estimation of the early cancer growth model.
(PDF)

S2 Table. Convergence statistics of MCMC chains during the parameter estimation of the EGF pathway model.
(PDF)

S3 Table. Number of log-posterior evaluations.
(PDF)

S4 Table. Grid search results: ESS of MH.
(PDF)

S5 Table. Variances of filter posteriors: Early cancer model.
(PDF)

S6 Table. Variances of filter posterior: EGF pathway model.
(PDF)

S1 Fig. IIV-noise identifiability of early cancer growth model I. The figure shows inference results of the early cancer model using NLME inference and filter inference with a Gaussian filter and $S = 100$ simulated individuals. The parameters are estimated from snapshot datasets with measurements from $N = 90$, $N = 270$, $N = 810$, $N = 2430$ individuals. The measurements are generated for 6 time points between 0 and 0.6 as described in Early cancer growth model inference. The first row shows the posterior distributions inferred from measurements of $N = 90$ individuals. The second, third and fourth row analogously show the results for $N = 270$, $N = 810$ and $N = 2430$. The filter posteriors are illustrated in blue and the NLME posteriors are illustred in red. The prior distributions are illustrated by black dashed lines and the data-generating parameter values are depicted by solid black lines.
(PDF)

S2 Fig. Early cancer growth measurement distribution. The figure is an extension of S1 Fig and shows the measurement distribution $p(y|\theta, t)$ of the early cancer growth model at $t = 0$, $t = 0.3$ and $t = 0.6$ for three different sets of parameter values. The measurement distribution corresponding to the data-generating parameters,

$\theta = (\mu_{y_0}, \sigma_{y_0}, \mu_\lambda, \sigma_\lambda, \mu_\sigma) = (10, 1, 2, 0.5, 0.8)$, is shown in black. A measurement distribution that overestimates the IIV contributions to the measurement variability, θ = (10, 1.195, 2, 0.5, 0.47), is illustrated in blue, and a measurement distribution that overestimates the noise contributions to the measurement variability, θ = (10, 0.7, 2, 0.5, 1.127), is illustrated in red.
(PDF)

S3 Fig. IIV-noise identifiability of early cancer growth model II. The figure is an extension of S1 Fig and shows filter inference results for the early cancer growth model from 60 000 snapshot measurements using Gaussian filters with S = 5000 simulated individuals. The prior distributions are illustrated by black dashed lines and the data-generating parameter values are depicted by solid black lines.
(PDF)

S4 Fig. Computational costs of log-posterior evaluation with and without gradients. The figure is an extension of Fig 7 and shows the evaluation time of the filter log-posterior with gradients in units of the evaluation time of the filter log-posterior without gradients for different numbers of measured individuals. The evaluation times are estimated according to S7 Text. The left panel shows the results for the early cancer growth model and the right panel the results for the EGF pathway model for S = 50 (blue), S = 100 (red) and S = 150 (green) simulated individuals.
(PDF)

Author Contributions

Conceptualization: David Augustin.

Data curation: David Augustin.

Formal analysis: David Augustin.

Funding acquisition: David Gavaghan.

Investigation: David Augustin.

Methodology: David Augustin.

Project administration: David Augustin.

Resources: David Augustin.

Software: David Augustin.

Supervision: Ben Lambert, Ken Wang, Antje-Christine Walz, Martin Robinson, David Gavaghan.

Validation: David Augustin.

Visualization: David Augustin.

Writing – original draft: David Augustin.

Writing – review & editing: David Augustin, Ben Lambert, Ken Wang, Antje-Christine Walz, Martin Robinson, David Gavaghan.

References

1. Hallgrímsson B, Hall BK. Variation and variability: central concepts in biology. In: Hallgrímsson B, Hall BK, editors. Variation. Academic Press; 2005. pp. 1–7.
2. Mueller L. Conceptual breakthroughs in evolutionary ecology. 1st ed. Academic Press; 2019.

3. Abbas AK, Lichtman AH, Pillai S. Cellular and molecular immunology. 10th ed. Elsevier Health Sciences; 2021.
4. Hanahan D, Weinberg RA. The hallmarks of cancer. Cell. 2000; 100(1):57–70. https://doi.org/10.1016/S0092-8674(00)81683-9 PMID: 10647931
5. Hanahan D, Weinberg RA. Hallmarks of cancer: the next generation. Cell. 2011; 144(5):646–674. https://doi.org/10.1016/j.cell.2011.02.013 PMID: 21376230
6. Kumar A, Singh A, et al. A review on Alzheimer's disease pathophysiology and its management: an update. Pharmacological reports. 2015; 67(2):195–203. https://doi.org/10.1016/j.pharep.2014.09.004 PMID: 25712639
7. Lavielle M. Mixed Effects Models for the Population Approach: Models, Tasks, Methods and Tools. 1st ed. Chapman and Hall/CRC; 2014.
8. Ribba B, Holford N, Magni P, Trocóniz I, Gueorguieva I, Girard P, et al. A Review of Mixed-Effects Models of Tumor Growth and Effects of Anticancer Drug Treatment Used in Population Analysis. CPT: Pharmacometrics & Systems Pharmacology. 2014; 3(5):113. https://doi.org/10.1038/psp.2014.12 PMID: 24806032
9. Hasenauer J, Waldherr S, Doszczak M, Radde N, Scheurich P, Allgöwer F. Identification of models of heterogeneous cell populations from population snapshot data. BMC bioinformatics. 2011; 12(1):1–15. https://doi.org/10.1186/1471-2105-12-125 PMID: 21527025
10. Luecken MD, Theis FJ. Current best practices in single-cell RNA-seq analysis: a tutorial. Molecular systems biology. 2019; 15(6):e8746. https://doi.org/10.15252/msb.20188746 PMID: 31217225
11. Adan A, Alizada G, Kiraz Y, Baran Y, Nalbant A. Flow cytometry: basic principles and applications. Critical reviews in biotechnology. 2017; 37(2):163–176. https://doi.org/10.3109/07388551.2015.1128876 PMID: 26767547
12. Hughes AJ, Spelke DP, Xu Z, Kang CC, Schaffer DV, Herr AE. Single-cell western blotting. Nature methods. 2014; 11(7):749–755. https://doi.org/10.1038/nmeth.2992 PMID: 24880876
13. Dixit PD, Lyashenko E, Niepel M, Vitkup D. Maximum entropy framework for predictive inference of cell population heterogeneity and responses in signaling networks. Cell systems. 2020; 10(2):204–212. https://doi.org/10.1016/j.cels.2019.11.010 PMID: 31864963
14. Lambert B, Gavaghan DJ, Tavener SJ. A Monte Carlo method to estimate cell population heterogeneity from cell snapshot data. Journal of Theoretical Biology. 2021; 511:110541. https://doi.org/10.1016/j.jtbi.2020.110541 PMID: 33271182
15. Browning AP, Ansari N, Drovandi C, Johnston AP, Simpson MJ, Jenner AL. Identifying cell-to-cell variability in internalization using flow cytometry. Journal of the Royal Society Interface. 2022; 19 (190):20220019. https://doi.org/10.1098/rsif.2022.0019 PMID: 35611619
16. Drovandi C, Lawson B, Jenner AL, Browning AP. Population Calibration using Likelihood-Free Bayesian Inference. arXiv:2202.01962v1 [Preprint]. 2022 [cited 2023 May 1]. Available from: https://arxiv.org/abs/2202.01962v1.
17. Neal RM. MCMC Using Hamiltonian Dynamics. In: Brooks S, Gelman A, Jones GL, Meng XL, editors. Handbook of Markov Chain Monte Carlo. Chapman & Hall/CRC; 2010. pp. 113–162.
18. Hoffman MD, Gelman A, et al. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. J Mach Learn Res. 2014; 15(1):1593–1623.
19. Betancourt M. A conceptual introduction to Hamiltonian Monte Carlo. arXiv:1701.02434v2 [Preprint]. 2018 [cited 2023 May 1]. Available from: https://arxiv.org/abs/1701.02434v2.
20. Augustin D. Chi—An open source python package for treatment response modelling; 2021 [cited 2023 May 1]. Available from: https://github.com/DavAug/chi.
21. Stegmann G, Jacobucci R, Harring JR, Grimm KJ. Nonlinear mixed-effects modeling programs in R. Structural Equation Modeling: A Multidisciplinary Journal. 2018; 25(1):160–165. https://doi.org/10.1080/10705511.2017.1396187
22. Bauer RJ. NONMEM Tutorial Part I: Description of Commands and Options, With Simple Examples of Population Analysis. CPT: Pharmacometrics & Systems Pharmacology. 2019; 8(8):525–537. https://doi.org/10.1002/psp4.12404 PMID: 31056834
23. Rackauckas C, Ma Y, Noack A, Dixit V, Mogensen PK, Byrne S, et al. Accelerated predictive healthcare analytics with pumas, a high performance pharmaceutical modeling and simulation platform. biorxiv:10.1101/2020.11.28.402297v2 [Preprint]. 2022 [cited 2023 May 1]. Available from: https://www.biorxiv.org/content/10.1101/2020.11.28.402297v2.
24. Lunn DJ, Best N, Thomas A, Wakefield J, Spiegelhalter D. Bayesian Analysis of Population PK/PD Models: General Concepts and Software. Journal of Pharmacokinetics and Pharmacodynamics. 2002; 29(3):271–307. https://doi.org/10.1023/A:1020206907668 PMID: 12449499

25. Chib S, Greenberg E. Understanding the metropolis-hastings algorithm. The american statistician. 1995; 49(4):327–335. https://doi.org/10.1080/00031305.1995.10476177
26. Browning AP, Drovandi C, Turner IW, Jenner AL, Simpson MJ. Efficient inference and identifiability analysis for differential equation models with random parameters. PLOS Computational Biology. 2022; 18(11):e1010734. https://doi.org/10.1371/journal.pcbi.1010734 PMID: 36441811
27. Beaumont MA. Approximate Bayesian Computation. Annual Review of Statistics and Its Application. 2019; 6(1):379–403. https://doi.org/10.1146/annurev-statistics-030718-105212
28. Wood SN. Statistical inference for noisy nonlinear ecological dynamic systems. Nature. 2010; 466 (7310):1102–1104. https://doi.org/10.1038/nature09319 PMID: 20703226
29. Price LF, Drovandi CC, Lee A, Nott DJ. Bayesian synthetic likelihood. Journal of Computational and Graphical Statistics. 2018; 27(1):1–11. https://doi.org/10.1080/10618600.2017.1302882
30. Frazier DT, Nott DJ, Drovandi C, Kohn R. Bayesian inference using synthetic likelihood: asymptotics and adjustments. Journal of the American Statistical Association. 2022; 1:28. Available from: https://www.tandfonline.com/doi/full/10.1080/01621459.2022.2086132.
31. Moon TK. The expectation-maximization algorithm. IEEE Signal processing magazine. 1996; 13(6):47–60. https://doi.org/10.1109/79.543975
32. Clerx M, Robinson M, Lambert B, Lei CL, Ghosh S, Mirams GR, et al. Probabilistic Inference on Noisy Time Series (PINTS). Journal of Open Research Software. 2019; 7(1):23. https://doi.org/10.5334/jors.252
33. Clerx M, Collins P, de Lange E, Volders PGA. Myokit: A simple interface to cardiac cellular electrophysiology. Progress in Biophysics and Molecular Biology. 2016; 120(1–3):100–114. https://doi.org/10.1016/j.pbiomolbio.2015.12.008 PMID: 26721671
34. Herbst RS. Review of epidermal growth factor receptor biology. International Journal of Radiation Oncology* Biology* Physics. 2004; 59(2):S21–S26. https://doi.org/10.1016/j.ijrobp.2003.11.041 PMID: 15142631
35. Dixit PD, Lyashenko E, Niepel M, Vitkup D. Maximum entropy framework for inference of cell population heterogeneity in signaling networks. biorxiv:10.1101/137513v4 [Preprint]. 2019 [cited 2023 May 1]. Available from: https://www.biorxiv.org/content/10.1101/137513v4.
36. Van Ravenzwaaij D, Cassey P, Brown SD. A simple introduction to Markov Chain Monte–Carlo sampling. Psychonomic bulletin & review. 2018; 25(1):143–154. https://doi.org/10.3758/s13423-016-1015-8 PMID: 26968853
37. Vehtari A, Gelman A, Simpson D, Carpenter B, Bürkner PC. Rank-Normalization, Folding, and Localization: An Improved $\hat{R}$ for Assessing Convergence of MCMC (with Discussion). Bayesian Analysis. 2021; 16(2):667–718. https://doi.org/10.1214/20-BA1221
38. Lambert B, Vehtari A. R*: A robust MCMC convergence diagnostic with uncertainty using decision tree classifiers. Bayesian Analysis. 2022; 17(2):353–379. https://doi.org/10.1214/20-BA1252
39. Gelman A, Carlin JB, Stern HS, Rubin DB. Bayesian Data Analysis. 2nd ed. Chapman and Hall/CRC; 2004.
40. Sjödin P, E Sjöstrand A, Jakobsson M, Blum MG. Resequencing data provide no evidence for a human bottleneck in Africa during the penultimate glacial period. Molecular biology and evolution. 2012; 29 (7):1851–1860. https://doi.org/10.1093/molbev/mss061 PMID: 22319141
41. McKinley TJ, Vernon I, Andrianakis I, McCreesh N, Oakley JE, Nsubuga RN, et al. Approximate Bayesian computation and simulation-based inference for complex stochastic epidemic models. Statistical science. 2018; 33(1):4–18. https://doi.org/10.1214/17-STS618
42. Holden PB, Edwards NR, Hensman J, Wilkinson RD. ABC for climate: dealing with expensive simulators. In: Handbook of approximate Bayesian computation. Chapman and Hall/CRC; 2018. p. 569–595.
43. Silk D, Filippi S, Stumpf MP. Optimizing threshold-schedules for sequential approximate Bayesian computation: applications to molecular systems. Statistical applications in genetics and molecular biology. 2013; 12(5):603–618. https://doi.org/10.1515/sagmb-2012-0043 PMID: 24025688
44. Prangle D. Adapting the ABC distance function. Bayesian Analysis. 2017; 12(1):289–309. https://doi.org/10.1214/16-BA1002
45. Joyce P, Marjoram P. Approximately Sufficient Statistics and Bayesian Computation. Statistical Applications in Genetics and Molecular Biology. 2008; 7(1). https://doi.org/10.2202/1544-6115.1389 PMID: 18764775
46. Wilkinson RD. Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. Statistical Applications in Genetics and Molecular Biology. 2013; 12(2):129–141. https://doi.org/10.1515/sagmb-2013-0010 PMID: 23652634