Computationally efficient mechanism discovery for cell invasion with uncertainty quantification

Daniel J. VandenHeuvel\(^1\), Christopher Drovandi\(^1\), and Matthew J. Simpson\(^1\)

\(^1\)School of Mathematical Sciences, Queensland University of Technology, Brisbane, Queensland, Australia

October 2, 2022

In this document, all section, table, and equation numbers without the S-prefix refer to the main manuscript.

S1 Simulation studies

In this section we describe two simulation studies used for testing our results. The models will all take the form

\[
\frac{\partial u}{\partial t} = T(t; \alpha) \left[ \frac{\partial}{\partial x} \left( D(u; \beta) \frac{\partial u}{\partial x} \right) + R(u; \gamma) \right].
\]

To simulate data, we start with the data from Jin et al. [1] with an initial density of 10,000 cells per well, taking only the data at \( t = 0 \). We then define the grid \( x_j = x_1 + (j - 1)\Delta x \), where \( \Delta x = (x_{N_p} - x_1)/(N_p - 1) \), \( j = 1, \ldots, N_p \), where \( x_1 = 75 \mu m \), \( x_{N_p} = 1875 \mu m \), and \( N_p = 100 \). We then fit a linear spline over the average cell density data at each point \( x \) at \( t = 0 \) from the data from Jin et al. [1] mentioned, and evaluate this spline at each \( x_j \), \( j = 1, \ldots, N_p \). These splines are fit using the implementation of the Dierckx Fortran library [2] in Julia available from the Dierckx package [3]. We then prescribe \( T(t; \alpha) \), \( D(u; \beta) \), and \( R(u; \gamma) \), along with the respective parameters for each function, and solve the corresponding partial differential equation (PDE). We store the solutions at \( t = 0, 12, 24, 36, 48 \text{ h} \), giving solutions \( \hat{u}_{ij} \) for the density at \( x = x_i \) and \( t = t_j \), with \( x_i = x_1 + (i - 1)\Delta x \) and \( t_j = (j - 1)\Delta t \), \( \Delta t = 12 \), for \( i = 1, \ldots, N_p \) and \( j = 1, \ldots, 5 \). To then add noise to this data, we
replace each $\tilde{u}_{ij}$ with $\tilde{u}_{ij} + |\tilde{u}_{ij}|^{0.45}z_{ij}$, where $z_{ij}$ is a sample from $\mathcal{N}(0, 1)$; to ensure the densities remain positive, any negative values from this replacement are further replaced by zero. Finally, to induce sparsity in the data, we thin the data randomly so that only 190 points remain, with 38 points at each time $t_j$, $j = 1, \ldots, 5$.

All results in these studies are presented in the same units as in Jin et al. [1], but in our implementation we put space and time into units of mm and d, respectively.

S1.1 Study I: Fisher-Kolmogorov model without delay

The first study we consider is a model of the form

$$\frac{\partial u}{\partial t} = \beta_1 \frac{\partial^2 u}{\partial x^2} + \gamma_1 u \left( 1 - \frac{u}{K} \right),$$  \hspace{1cm} (B)

where $K = 1.7 \times 10^{-3}$ cells $\mu m^{-2}$. We use $\beta_1 = 301 \mu m^2 h^{-1}$ and $\gamma_1 = 0.044 h^{-1}$. The data simulated from this model, along with a Gaussian process fit to the data, are shown in Fig A.

![Figure A: Simulated data and fitted Gaussian process.](image)

**Figure A: Simulated data and fitted Gaussian process.** Simulated data according to the model (B) with $D(u; \beta) = \beta_1 + \beta_2 (u/K)$ and $R(u; \gamma) = \gamma_1 u (1 - u/K)$, with $\beta_1 = 301 \mu m^2 h^{-1}$ and $\gamma_1 = 0.044 h^{-1}$. The lines and shaded regions represent a Gaussian process fit to the spatiotemporal data.

To begin with this model, we start with assuming that $T(t; \alpha) = 1$, $D(u; \beta) = \beta_1 + \beta_2 (u/K)$, and $R(u; \gamma) = \gamma_1 u (1 - u/K)$. This nonlinear diffusivity function is misspecified, with the true function corresponding to $\beta_2 = 0$. With this model we start with 10 bootstrap iterations and four optimisation restarts, sampling $\beta_1 \in [250, 1000]$, $\beta_2 \in [250, 1000]$, and $\gamma_1 \in [0.035, 0.05]$; note that $[\beta_1] = \mu m^2 h^{-1}$,
\[ \beta_2 = \mu m^2 h^{-1}, \text{ and } [\gamma_1] = h^{-1}, \text{ where } [y] \text{ denotes the units of } y. \] The results from this procedure are shown in Fig B. We see that we nearly capture the value of \( \beta_1 \) at a 95% level, although \( \beta_2 \) is not zero.

Figure B: Initial simulation results. Initial simulation results for the data from model (B) where \( R(u; \gamma) = \gamma_1 u(1 - u/K) \) and we use the misspecified nonlinear diffusivity model \( D(u; \beta) = \beta_1 + \beta_2 (u/K) \). In the density plots, the dashed red vertical line is at the position of the true parameter value, and in the curve plots the dashed red curve shows the true mechanisms.

To go further, and to make the optimisation more efficient, we rescale our problem so that

\[
D(u; \beta) = 300\beta_1 - 200\beta_2 \left( \frac{u}{K} \right) \quad \text{and} \quad R(u; \gamma) = 0.044\gamma_1 u \left( 1 - \frac{u}{K} \right). \tag{C}
\]

This puts all the parameters on the same scale. With this rescaling we perform \( B = 200 \) bootstrap iterations and four optimisation restarts around parameter values of 1. The revised results after rescaling are given in Fig C. We see that we have captured the true values of \( \beta_1 \) and \( \gamma_1 \), although \( \beta_2 \) is given by the uncertainty interval \((-326.1, -145.4) \mu m^2 h^{-1}\) rather than its true value of \(0 \mu m^2 h^{-1}\). We note that the uncertainty around the nonlinear diffusivity curve in (d) is correct for small densities, and most of the data are around these small values as shown in Fig A. Thus, the mechanism is still correct for small time, and it is not of concern that we have a different mechanism for larger densities given that there is only little data in this region.

Now let us obtain these results when we instead correctly specify \( D(u; \beta) = \beta_1 \). We start with 10 bootstrap iterations and four optimisation restarts as before, sampling \( \beta_1 \in [291.667, 541.666] \) and \( \gamma_1 \in [0.0380, 0.0420] \). The initial results are shown in Fig D, where we have captured \( \gamma_1 \) but only just missed the value of \( \beta_1 \).
Figure C: Final simulation results. Final simulation results after re-scaling for the data from model (B) where \( R(u; \gamma) = \gamma_1 u(1 - u/K) \) and we use the misspecified nonlinear diffusivity model \( D(u; \gamma) = \beta_1 + \beta_2 (u/K) \). The re-scaling used is given in (C). In the density plots, the dashed red vertical line is at the position of the true parameter value, and in the curve plots the dashed red curve shows the true mechanisms.

Figure D: Initial simulation results. Initial simulation results for the data from model (B) where \( R(u; \gamma) = \gamma_1 u(1 - u/K) \) and we use the correctly specified model \( D(u; \gamma) = \beta_1 \). In the density plots, the dashed red vertical line is at the position of the true parameter value, and in the curve plots the dashed red curve shows the true mechanisms.

The results in Fig D suggest the rescaling

\[
D(u; \beta) = 300\hat{\beta}_1 \quad \text{and} \quad R(u; \gamma) = 0.044\hat{\gamma}_1 u \left(1 - \frac{u}{K}\right). \tag{D}
\]

S4
We perform 200 bootstrap iterations with this rescaled model and use four optimisation restarts. The final results are given in Fig E, where we now see that we have captured the true functional forms.

Figure E: Final simulation results. Final simulation results after re-scaling for the data from model (B) where \( R(u; \gamma) = \gamma_1 u(1 - u/K) \) and we use the correctly specified nonlinear diffusivity model \( D(u; \beta) = \beta_1 \). The rescaling used is \( \beta_1 = 300\beta_1 \) and \( \gamma_1 = 0.044\gamma_1 \). In the density plots, the dashed red vertical line is at the position of the true parameter value, and in the curve plots the dashed red curve shows the true mechanism.

Let us now use our model selection procedure for comparing the two models in Figs C and E. These results are shown in Table A. We see that the \( \beta_1 \) estimates are reasonably similar, and the \( \gamma_1 \) estimates are the same across each model. The probability that the models are in the class of optimal models for the data, \( \mathbb{P}(E_1) \), is 6.0% for the \( D(u; \beta) = \beta_1 + \beta_2(u/K) \) model and 98.5% for the \( D(u; \beta) = \beta_1 \) model. In particular, our model selection procedure has correctly identified the correct model out of the two models 98.5% of the time.

| Model                  | \( \beta_1 + \beta_2 \left( \frac{u}{K} \right) \) | \( \beta_1 \) |
|------------------------|---------------------------------------------------|--------------|
| PDE Error (%)          | (3.624, 5.348)                                    | (3.49, 5.084) |
| \( \beta_1 \) (\( \mu m^2 h^{-1} \)) | (298.591, 385.381)                                | (240.633, 305.532) |
| \( \beta_2 \) (\( \mu m^2 h^{-1} \)) | (−388.67, −108.825)                              | —            |
| \( \gamma_1 \) (\( h^{-1} \)) | (0.044, 0.045)                                    | (0.044, 0.045) |
| \( \mathbb{P}(E_1) \)  | 0.060                                             | 0.985        |
| \( \mathbb{P}(E_2) \)  | 0.065                                             | 0.005        |
| \( \mathbb{P}(E_3) \)  | 0.875                                             | 0.010        |

Table A: Results for the two models considered. Interval estimates, PDE errors, and model selection results for the two nonlinear diffusivity functions considered in this study. The PDE errors are computed using sampled initial conditions. Significance levels for all uncertainty intervals are 95%. For both models we use \( R(u; \gamma) = \gamma_1 u(1 - u/K) \) and \( T(t; \alpha) = 1 \).
S1.2 Study II: Fisher-Kolmogorov model with delay

We now consider simulating data from the model

\[
\frac{\partial u}{\partial t} = \frac{1}{1 + \exp(-\alpha_1 - \alpha_2 t)} \left[ \beta_1 \frac{\partial^2 u}{\partial x^2} + \gamma_1 u \left(1 - \frac{u}{K}\right) \right],
\]

(E)

where \( \alpha_1 = -1.50, \alpha_2 = 0.31 \text{ h}^{-1}, \beta_1 = 571 \mu\text{m}^2\text{h}^{-1}, \) and \( \gamma_1 = 0.081 \text{ h}^{-1}. \) The data simulated from this model, along with a Gaussian process fit to the data, are shown in Fig F.

![Figure F: Simulated data and fitted Gaussian process.](image)

Figure F: Simulated data according to the model (E) with \( \alpha_1 = -1.5, \alpha_2 = 0.31 \text{ h}^{-1}, \beta_1 = 571 \mu\text{m}^2\text{h}^{-1}, \) and \( \gamma_1 = 0.081 \text{ h}^{-1}. \) The lines and shaded regions represent a Gaussian process fit to the spatiotemporal data.

We consider fitting the following five models to this data:

- Model 1: \( T(t; \alpha) = 1, \quad D(u; \beta) = \beta_1, \) (F)
- Model 2: \( T(t; \alpha) = \frac{1}{1 + \exp(-\alpha_1 - \alpha_2 t)}, \quad D(u; \beta) = \beta_1, \) (G)
- Model 3: \( T(t; \alpha) = 1, \quad D(u; \beta) = \beta_2 \left(\frac{u}{K}\right), \) (H)
- Model 4: \( T(t; \alpha) = \frac{1}{1 + \exp(-\alpha_1 - \alpha_2 t)}, \quad D(u; \beta) = \beta_2 \left(\frac{u}{K}\right), \) (I)
- Model 5: \( T(t; \alpha) = \frac{1}{1 + \exp(-\alpha_1 - \alpha_2 t)}, \quad D(u; \beta) = \beta_1 + \beta_2 \left(\frac{u}{K}\right)^{\beta_3}, \) (J)

with \( R(u; \gamma) = \gamma_1 u(1 - u/K) \) for each model. We note that these are the same five models considered in the main body of the manuscript. We do not show the preliminary results used to obtain the scaling for the parameters in each of these models as we did in the previous study, but note that the procedure...
remains the same. For each model, after rescaling, we use 100 bootstrap iterations with no optimiser restarts, starting each parameter at 1. The final results are shown in Table B.

| Model | Model 1 | Model 2 | Model 3 | Model 4 | Model 5 |
|-------|---------|---------|---------|---------|---------|
| PDE Error (%) | (7.978, 10.730) | (2.600, 4.556) | (8.731, 11.598) | (3.864, 6.312) | (2.605, 4.329) |
| $\alpha_1$ | (−1.723, −0.583) | − | − | − | − |
| $\alpha_2$ (h$^{-1}$) | (0.172, 0.320) | − | − | − | − |
| $\beta_1$ (gm$^{-2}$ h$^{-1}$) | (424.904, 567.49) | (483.627, 620.185) | − | − | (0.169, 0.345) |
| $\beta_2$ (gm$^{-2}$ h$^{-1}$) | (1703.240, 2253.560) | (1547.660, 2281.510) | − | − | (130.914, 424.938) |
| $\gamma_1$ (h$^{-1}$) | (0.068, 0.071) | (0.080, 0.083) | (0.069, 0.072) | (0.079, 0.084) | (0.080, 0.085) |
| $\beta_3$ | − | − | − | − | − |
| $\Pi(E_1)$ | 0.000 | 0.000 | 0.000 | 0.000 | 0.560 |
| $\Pi(E_2)$ | 0.000 | 0.660 | 0.000 | 0.000 | 0.180 |
| $\Pi(E_3)$ | 1.000 | 0.180 | 1.000 | 1.000 | 0.260 |

Table B: Results for the five models considered. Interval estimates, PDE errors, and model selection results for the five models (F)–(J) considered for the data generated by (E). 100 bootstrap iterations are used for each model. The PDE errors are computed using sampled initial conditions. Significance levels for all uncertainty intervals are 95%.

See that for the model in Table B which is correctly specified, (G), the correct parameter values are captured within the uncertainty intervals for the respective parameters. When we do not include delay but give the correct cell migration mechanism, (F), we nearly recover the correct diffusivity $\beta_1$ but no longer obtain the correct reaction coefficient $\gamma_1$, leading to a significant increase in the PDE error. For the Porous-Fisher model with delay, (I), we see that the delay and reaction mechanisms are recovered, but the error is still slightly larger than for (G). The error is significantly increased in (H) when we do not include delay. Finally, the generalised Porous-FKPP model (J) recovers the correct delay and reaction terms. This model also gives an error that is very similar to (G). Finally, in terms of model selection, we notice that all the models without delay and the Porous-Fisher model with delay are never selected as falling within the class of optimal models for this data. The correctly specified model, (G), is correctly selected 66% of the time, with the next best model being the generalised Porous-FKPP model (J). We show in Fig G the results for the correctly specified model in (G), where we can indeed see that the model fits the data well and all the mechanisms are recovered.

To further understand how model misspecification can present itself from the results, in Fig H we plot the PDE solutions for all the models in Table B. In (a), where only the delay term is missing, we see that the curves are initially over the data, but eventually fall below it — this feature implies the need for delay. In (c), which misspecifies both the delay and the diffusion, we again see the same type of delay where the curves start above and go below. In addition, we see that the cell densities are clearly problematic, namely our model fails to predict how the cells move into the centre of the well. That is, the diffusion term is misspecified. Hence, an incorrect diffusion term can be seen through a poor migration of cell densities compared to the experimental data. This feature is also present in (d), except here in (d) the delay term is correctly specified and so we do not see the curves going below and then over. Finally,
Figure G: Final model results for the optimal model. Results for the fitted model in (G) for the data generated from (E). In the density plots, the dashed red vertical line is at the position of the true parameter value, in the curve plots the dashed red curve shows the true mechanism, and in the curve plots the different shaded regions show various values of the products of the respective mechanisms at the shown times.

1. A missing delay term can be identified by seeing that the learned curves start below (resp. above) the data and then, as time increases, go above (resp. below) the data.

2. A misspecified diffusion term can be identified by inferring from the cell densities the migration behaviour of the cells compared to the data. In our case, we saw that the cells were predicted to migrate into the middle of the well in a way that was not like what we see from the data, thus indicating an incorrect diffusion term.

S2  Basis function approach

In this section we present an alternative to the nonlinear optimisation problem in (35) for problems without delay and whose nonlinear diffusivity and reaction functions can be represented as linear com-
Figure H: All model results for the tested models. All PDE solutions for the fitted models in Table B. The initial conditions are sampled using the Gaussian process. The correctly specified model’s results are those in (b).

Combinations of individual basis functions,

\[ D(u; \beta) = \sum_{k=1}^{d} \beta_k \phi_k(u) \quad \text{and} \quad R(u; \gamma) = \sum_{k=1}^{r} \gamma_k \psi_k(u), \]

where the basis functions \( \phi_k \) and \( \psi_k \), for each \( k = 1, \ldots, d \) and \( k = 1, \ldots, r \), respectively, are differentiable.

With these functions, (32) becomes

\[ \frac{\partial u^*_{ij}}{\partial t} = \sum_{k=1}^{d} \left[ \phi_k'(u^*_{ij}) \left( \frac{\partial u^*_{ij}}{\partial x} \right)^2 + \phi_k'(u^*_{ij}) \frac{\partial^2 u^*_{ij}}{\partial x^2} \right] \beta_k + \sum_{k=1}^{r} \gamma_k \psi_k(u^*_{ij}), \]

where primes denote derivatives in \( u \). We then define the matrices \( A_1 \in \mathbb{R}^{nm \times d} \) and \( A_2 \in \mathbb{R}^{nm \times r} \) such that

\[ [A_1]_{(i,j),k} = \phi_k(u_{ij}) \frac{\partial^2 u^*_{ij}}{\partial x^2} + \phi_k'(u_{ij}) \left( \frac{\partial u^*_{ij}}{\partial x} \right)^2 \quad \text{and} \quad [A_2]_{(i,j),k} = \psi_k(u_{ij}). \]
respectively, where $[A_{\ell}]_{(i,j),k}$ denotes the entry in $A_{\ell}$ in the $k$th column in the row corresponding to the point $(\xi_i, \tau_j)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m$, and $\ell = 1, 2$. Next, define $A = [A_1, A_2] \in \mathbb{R}^{nm \times (d+r)}$ and $\theta = (\beta^T, \gamma^T)^T$. Given this form we obtain the matrix system

$$A \theta = \frac{\partial u^*}{\partial x}.$$  \hspace{1cm} \text{(N)}

Equation (N) can then be solved for $\theta$ using the backslash operator in Julia [4]. This approach could be especially useful for obtaining initial scale estimates for parameters, for example, rather than dealing with the far slower nonlinear optimisation problem on an unscaled problem to start with.

We note that while this approach could be suitable for say a Porous-Fisher PDE,

$$\frac{\partial u}{\partial t} = \beta_1 \frac{\partial}{\partial x} \left( u \frac{\partial u}{\partial x} \right) + \gamma_1 u \left( 1 - \frac{u}{K} \right),$$  \hspace{1cm} \text{(O)}

it would not be suitable if there were any nonlinear delay or if we had for example the PDE

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left[ \beta_1 e^{-\beta_2 u} \frac{\partial u}{\partial x} \right] + \gamma_1 u \left( 1 - \frac{u}{K} \right),$$  \hspace{1cm} \text{(P)}

since the nonlinear diffusivity function cannot be represented as a linear combination of basis functions. We could still approximate this problem such that the functions are linear in the parameters, for example by temporarily fixing $\beta_2$ at a value, say $\beta_2 = 1$, and solving (N) to estimate $\beta_1$. We could then fix $\beta_1$ at this found value and estimate $e^{-\beta_2 u} \approx 1 - \beta_2 u$, assuming $\beta_2 u$ is small, and estimate $\beta_2$ in the same way. This procedure of fitting $\beta_1$ and then $\beta_2$ separately could be repeated if desired. These approximations would give rough scales of the estimates that could be used to rescale the parameters and make the nonlinear optimisation problem in (35) significantly faster.

Let us compare the results from this approach to those given in Section S1.1, writing

$$D(u; \beta) = \beta_1 \phi_1(u) \quad \text{and} \quad R(u; \gamma) = \gamma_1 \psi_1(u),$$  \hspace{1cm} \text{(Q)}

where $\phi_1(u) \equiv 1$ and $\psi_1(u) \equiv u(1 - u/K)$. The final results are shown in Fig I with 200 bootstrap iterations. The results are very similar to those in Fig E.

We note that these similarities make sense if we recall that the solution to (N) is equivalent to solving
Figure I: Results for the Fisher-Kolmogorov model without delay using the basis function approach. Results for parameters, functional forms, and PDE solutions for the model equation (B) using the basis function approach. In the density plots, the dashed red line is at the true parameter value, and in the curve plots the dashed red curve shows the true curve.

the least squares problem

$$\hat{\theta} = \arg\min_{\theta \in \mathbb{R}^{d+r}} \mathcal{L}(\theta),$$

$$\mathcal{L}(\theta) = \sum_{i=1}^{n} \left\{ \frac{\partial u^*_i}{\partial t} - \left( \sum_{k=1}^{d} \left[ \phi_k'(u^*_i) \left( \frac{\partial u^*_i}{\partial x} \right)^2 + \phi_k(u^*_i) \frac{\partial^2 u^*_i}{\partial x^2} \right] \beta_k + \sum_{k=1}^{r} \gamma_k \psi_k(u^*_i) \right) \right\}^2,$$

where the summand is the difference between the terms on both sides of (L). See that this optimisation problem is the same as minimising only $\mathcal{L}_{PDE}$ in (33) and not considering $\mathcal{L}_{GLS}$ in (34).

We note that for more complicated diffusivity models, such as the quadratic model $D(u; \beta) = \beta_1 + \beta_2(u/K) + \beta_3(u/K)^2$, we find results with increased PDE errors despite similarly learned mechanisms relative to the nonlinear least squares approach. We explain this by noting that, in this approach, the parameter estimation is focusing only on our sampled values without any direct consideration for the actual data as would be accomplished through $\mathcal{L}_{GLS}$. This failure to capture the data during the bootstrapping procedure using the basis function approach demonstrates one downside.

The results in this section allow us to make the following conclusions around this method. The approach is extremely fast relative to the main approach which requires solving (35). Nevertheless, the approach can lead to greater error in the PDE solutions due to not directly considering the data when estimating the parameters, depending on the complexity of the model. We therefore recommend this method as an efficient way to obtain scales for the parameters when the problem is linear, or for example as a quick way to see if delay may be needed in the problem. Once these scales are found, the coefficients...
should be rescaled and the nonlinear optimisation problem in (35) should be used to give the final results. Future work could for example augment the loss function \( S \) with the loss function \( L_{\text{PDE}} \) or some other term that regularises the loss function in a way that considers the data in a more direct manner, although this may not lead to a simple matrix system as in (N).

S3 Results for the scratch assay data

In this section we present plots for a delayed Fisher-Kolmogorov model and a delayed generalised Porous-FKPP model, as presented in the main body of the manuscript, for each data set from Jin et al. [1]. We first show the final results for the delayed Fisher-Kolmogorov model and then we present those for the delayed generalised Porous-FKPP model.

S3.1 Delayed Fisher-Kolmogorov models

We start by presenting the plots for the delayed Fisher-Kolmogorov model, where \( T(t; \alpha) = \frac{1}{1 + \exp(-\alpha_1 - \alpha_2 t)} \), \( D(u; \beta) = \beta_1 \), and \( R(u; \gamma) = \gamma_1 u(1 - u/K) \), for each data set from Jin et al. [1]. The plots are shown in Figs J–O for the 10,000, 12,000, 14,000, 16,000, 18,000, and 20,000 CPW data sets, respectively. The density plots in (a), (b), (d), (e) for each of these plots appear to be unimodal with well-defined peaks, implying that for these models the parameters have been well identified. For the delay curves in (c) of each plot, we see that the uncertainty around the curve is similar for each data set, although the uncertainty in Fig N(c) is a bit wider than the other plots and in Fig O(f) it is significantly smaller than the other plots. This issue in Fig O(f) is not surprising since we know in Fig O(h), and from the discussion in the manuscript, that there are some issues with this model for the 20,000 CPW data set. The reaction curves in (i) have very little uncertainty for large time, although for small time the uncertainty intervals are wider as the delay takes time to start. The opposite is true for the nonlinear diffusivity curves in (f), where we instead have little uncertainty initially, with increasing uncertainty as the cells start to proliferate at greater rates. For the PDE solutions, we see that while the PDE error is smaller in (g) of each plot where we use a spline initial condition, the uncertainty intervals do not typically match the data as well as they do in (h) where we use a sampled initial conditions. Overall, these results show good fits to the data and that we have identified the parameter coefficients reliably in each case, with the exception of the 20,000 CPW data set as discussed in the manuscript.
Figure J: Final model results for the optimal model with 10,000 cells per well. Results for the Fisher-Kolmogorov model with delay for the 10,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].

Figure K: Final model results for the optimal model with 12,000 cells per well. Results for the Fisher-Kolmogorov model with delay for the 12,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure L: Final model results for the optimal model with 14,000 cells per well. Results for the Fisher-Kolmogorov model with delay for the 14,000 cells per well dataset from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].

Figure M: Final model results for the optimal model with 16,000 cells per well. Results for the Fisher-Kolmogorov model with delay for the 16,000 cells per well dataset from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure N: Final model results for the optimal model with 18,000 cells per well. Results for the Fisher-Kolmogorov model with delay for the 18,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].

Figure O: Final model results for the optimal model with 20,000 cells per well. Results for the Fisher-Kolmogorov model with delay for the 20,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
S3.2 Delayed Generalised Porous-FKPP models

In this section we show the results for the delayed generalised Porous-FKPP models, where \( T(t; \alpha) = 1/[1 + \exp(-\alpha_1 - \alpha_2 t)] \), \( D(u; \beta) = \beta_1 + \beta_2 (u/K)\beta_3 \), and \( R(u; \gamma) = \gamma_1 u(1 - u/K) \), for each data set from Jin et al. [1]. This is the same model considered by Lagergren et al. [5]. The plots are shown in Figs P–U for the 10,000, 12,000, 14,000, 16,000, 18,000, and 20,000 CPW data sets, respectively. For these plots, the modes in the density plots in (a)–(f) of each plot are not as well-defined as they were for the delayed Fisher-Kolmogorov models, especially for \( \beta_1 \) and \( \beta_3 \), where the peak of the modes are typically very flat, showing issues with identifiability. Most of the plots are still unimodal. The behaviour of the uncertainty around the delay curves in (g) of each plot is similar to those for the delayed Fisher-Kolmogorov model. The nonlinear diffusivity curves in (h) of each plot show behaviour that is alike \( D(u; \beta) = \beta_1 \) for smaller densities, especially for the 10,000 CPW data set. As the initial cell density increases, the range of values for \( u \) where the curve is completely flat is decreasing, where we see that the curves start to decrease sooner and the rate of decreasing increases with initial cell density, and in Fig U(h) we see that the nonlinear diffusivity curve is immediately decreasing. This observation supports the discussion in the manuscript where nutrients are running out for the cells when we increase the initial cell density, with greater cell densities leading to a greater need for these nutrients, or at least a different noise model for the Gaussian process to handle these differences. The reaction curves in (i) of each plot show no issues.

S4 Data thresholding

In this section we discuss a method for improving the quality of data used for estimating the parameters \( \theta \) by removing data points that violate certain conditions. We suppose we have some data \( u^* \) and corresponding derivatives \( \partial u^*/\partial t \), \( \partial u^*/\partial x \), and \( \partial^2 u^*/\partial x^2 \). We then define some threshold parameters \( 0 \leq \delta u, \delta t < 1 \), where 0 corresponds to no thresholding and 1 means all points are removed. To motivate what conditions we should use to discard data, let us consider the space-time diagrams in Fig 5. We see that there are some regions where there are very little cells so that the density is almost zero, and there are some regions where the density evolves very slowly in time. These values would pollute the larger sum in (33), and so it would be useful to remove them as they do not contribute any valuable information. Thus, we define the following three conditions that must be satisfied for a data point to be used, where \( u_{ij}^* \) is an estimate for the cell density from the Gaussian process at \( (\xi_i, \tau_j) \) for \( i = 1, \ldots , n \) and \( j = 1, \ldots , m \):
Figure P: Comparisons to Lagergren et al.’s results with 10,000 cells per well. Results for the generalised Porous-FKPP model with delay for the 10,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure Q: Comparisons to Lagergren et al.’s results with 12,000 cells per well. Results for the generalised Porous-FKPP model with delay for the 12,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure R: Comparisons to Lagergren et al.'s results with 14,000 cells per well. Results for the generalised Porous-FKPP model with delay for the 14,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure S: Comparisons to Lagergren et al.’s results with 16,000 cells per well. Results for the generalised Porous-FKPP model with delay for the 16,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure T: Comparisons to Lagergren et al.’s results with 18,000 cells per well. Results for the generalised Porous-FKPP model with delay for the 18,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
Figure U: Comparisons to Lagergren et al.’s results with 20,000 cells per well. Results for the generalised Porous-FKPP model with delay for the 20,000 cells per well data set from Table 1. In the curve plots, the different shaded regions show various values of the products of the respective mechanisms at the shown times. The shaded regions in the PDE plots represent the uncertainty at each point around the mean curve shown, and the points shown are the data from Jin et al. [1].
1. \( \min_{k=1,...,n, \ell=1,...,m} |u^*_{k\ell}| \leq \left( \max_{k=1,...,n, \ell=1,...,m} |u^*_{k\ell}| \right) (1 - \delta u) \);

2. \( \min_{k=1,...,n, \ell=1,...,m} |\partial_t u^*_{k\ell}| \leq |\partial_t u^*_{ij}| \leq \left( \max_{k=1,...,n, \ell=1,...,m} |\partial_t u^*_{k\ell}| \right) (1 - \delta u) \);

3. \( u^*_{ij} \geq 0 \).

These conditions are only applied for the Gaussian process samples; none of the terms used for computing \( \mathcal{L}_{GLS} \) are removed. We note that this last condition is needed since some sampled values may go below zero. With the first two conditions we can control for low density regions and for cells that are approaching the carrying capacity density, as discussed above. Those data points that do not satisfy all of these conditions are not included when computing (33).

To now explore these threshold conditions, we explore its impact on the models in (B) and (E). We use 50 bootstrap iterations for each pair \( (\delta u, \delta t) \) and no optimiser restarts, and we scale the parameters such that the true parameter values are 1.0. The errors that we report from these experiments are based on the PDE solutions with the sampled initial conditions, and we only report the mean error. The results from this experiment are shown in Table C. We see that most of the minimum errors are for non-zero threshold tolerances \( \delta u, \delta t \), with most corresponding to a 5% tolerance for the time derivatives. The global minimum in the table is 3.977, corresponding to \( (\delta u, \delta t) = (0.25, 0.02) \). These results suggest that, for this model, it is more important to control for low and high density values than for the rate of evolution of these densities. This result makes sense if we consider the data in Fig A for this simulation. We see that many of the density values are near zero, and thus there are many terms in the sum defining (33) that are essentially the same. We do see that some minima are also at \( \delta t = 5\% \), suggesting that we might still want to consider discarding some densities that are evolving too slowly or too fast. Overall, we see that this thresholding is worthwhile to consider in this study, but a better method would be required to more accurately estimate the optimal \( (\delta u, \delta t) \) for it to be worth configuring these values given the time it takes to produce these errors.

Now let us repeat this experiment for the simulation study in Section S1.2. We now only use 10 bootstrap iterations for each pair \( (\delta u, \delta t) \) and no optimiser restarts, and the parameters are scaled such that the true parameter values are 1.0. Additionally, we choose \( n = m = 30 \). The results for this study are shown in Table D. We see that when we introduce delay, relative to the results in Table C, thresholding is more important for the density values than for their rates of change, although there are some \( \delta u \) values that have a high optimal \( \delta t \) value, such as for the \( (\delta u, \delta t) = (0.15, 0.15) \) pair. The global minimum in Table D is at \( (\delta u, \delta t) = (0, 0.10) \) with an error of 3.025, showing this importance of thresholding density values, but also demonstrating that even slightly controlling for slowly/highly
Table C: Results for data thresholding. Results for pairs of \((\delta u, \delta t)\). The errors reported are mean percentage errors, rounded to three decimal places, when simulating the Fisher-Kolmogorov model \(B\) as described in Section S1.1. The bold values show the minimum error in the column, and the underlined values show the minimum error in the row. The blue cell shows the minimum error in the whole table.

Table D: Results for data thresholding. Results for pairs of \((\delta u, \delta t)\). The errors reported are mean percentage errors when simulating the Fisher-Kolmogorov model with delay as described in Section S1.2. The bold values show the minimum error in the column, and the underlined values show the minimum error in the row. The blue cell shows the minimum error in the whole table.

S5 PDE truncation error

In this section we analyse how different choices of spatial discretisation, and the associated truncation error, affects the inferences made from the bootstrapping procedure. We perform this analysis by considering the results when using \(N_p = 500, 250, 125,\) or 50 equally-spaced spatial gridpoints; the temporal discretisation is chosen adaptively and hence no analysis of it will be presented here. We list below four tables, Table E–H, that compare results with \(N_p = 500, 250, 125,\) and 50, respectively. Note that Table
E is the same as Table 1 to facilitate the comparison. We see from these tables that the results are essentially the same across each grid size, with the same model selected each time. For this best model, model 1, we show the confidence intervals for all the coefficients and for the PDE errors for each $N_p$ and each data set in Fig V. This figure shows that there are virtually no differences in the results. Most remarkably, we have the following runtimes for producing these results:

- $N_p = 500$: 31 hours and 40 minutes.
- $N_p = 250$: 4 hours and 37 minutes.
- $N_p = 125$: 1 hour and 33 minutes.
- $N_p = 50$: 56 minutes and 7 seconds.

These simulation times are produced on a Windows 10 64-bit machine with an i7-8700K CPU @ 3.70 GHz, 32 GB of RAM, and a GTX 1070 Ti, and no parallel computing is used for bootstrapping. We could therefore produce the required results in just under an hour, for all data sets and for all models, without exploiting parallelism. Hence, we see that the net error introduced by solving these PDEs across a wide range of parameter sets is relatively insensitive to the truncation error introduced through spatial discretisation. In practice, then, we could find the results with $N_p = 50$ and $N_p = 125$ and, if there is little improvement in the $N_p = 125$ case, we could be satisfied with the $N_p = 50$ results rather than be concerned about the impact of the spatial truncation.
### Table E: Model selection results for the five models considered with \( N_p = 500 \). Interval estimates, PDE errors, and model selection results for the models in (2)-(6) when applied to each data set from Jin et al. [1]. See the paper for more details. The time taken to produce these results is 31 hours and 40 minutes.
Table F: Model selection results for the five models considered with \( N_p = 250 \). Interval estimates, PDE errors, and model selection results for the models in (2)–(6) when applied to each data set from Jin et al. [1]. See the paper for more details. The time taken to produce these results is 4 hours and 37 minutes.
Table G: Model selection results for the five models considered with $N_p = 125$. Interval estimates, PDE errors, and model selection results for the models in (2)–(6) when applied to each data set from Jin et al. [1]. See the paper for more details. The time taken to produce these results is 1 hour and 33 minutes.
| Models | Model 1 | Model 2 | Model 3 | Model 4 | Model 5 |
|--------|--------|--------|--------|--------|--------|
| PDE Error (%) | (20.841, 32.359) | (14.099, 22.278) | (21.9, 32.748) | (14.408, 24.144) | (13.8, 24.641) |
| $\alpha_1$ | $\alpha_2$ (h^{-1}) | (0.085, 0.39) | $\beta_1$ (ums^{-1}) | (71.496, 368.451) | $\gamma_1$ (h^{-1}) | (0.045, 0.052) |
| $\beta_2$ (ums^{-1}) | (2.836, -0.248) | ($\gamma_3$ (h^{-1}) | (0.045, 0.052) | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.04 | 0.24 | 0.24 | 0.24 | 0.24 |
| $\gamma_4$ (h^{-1}) | (1.0) | (0) | 0.00 | 1.00 | 0.90 | 0.72 |

| PDE Error (%) | (15.566, 21.498) | (9.952, 14.49) | (15.914, 22.565) | (10.463, 15.341) | (9.853, 14.737) |
| $\alpha_1$ | $\alpha_2$ (h^{-1}) | $\beta_1$ (ums^{-1}) | (119.06, 226.051) | (1.344, 3.036) | (0.16, 0.255) |
| $\beta_2$ (ums^{-1}) | (253.376, 592.556) | (231.861, 547.584) | ($\gamma_4$ (h^{-1}) | (0.044, 0.049) | (0.044, 0.048) | (0.054, 0.061) | (0.054, 0.062) |
| $\gamma_5$ (h^{-1}) | (0.00, 0.00) | (0.00, 0.00) | 0.00 | 0.00 | 0.00 | 0.17 |
| $\gamma_6$ (h^{-1}) | (0.00, 0.00) | (0.00, 0.00) | 0.00 | 0.00 | 0.00 | 0.47 |
| $\gamma_7$ (h^{-1}) | (1.00, 1.00) | (1.00, 1.00) | (0.95, 0.98) | (0.60, 0.60) |

| PDE Error (%) | (12.043, 14.48) | (10.111, 14.74) | (10.932, 15.963) | (10.517, 15.917) | (10.205, 16.984) |
| $\alpha_1$ | $\alpha_2$ (h^{-1}) | $\beta_1$ (ums^{-1}) | (453.057, 763.85) | (1.019, 0.401) | (0.194, 0.412) | (0.236, 0.331) |
| $\beta_2$ (ums^{-1}) | (852.533, 2022.92) | (382.357, 761.589) | ($\gamma_5$ (h^{-1}) | (0.047, 0.052) | (0.047, 0.052) | (0.054, 0.061) | (0.054, 0.061) |
| $\gamma_6$ (h^{-1}) | (0.00, 0.00) | (0.00, 0.00) | 0.00 | 0.00 | 0.00 | 0.00 |
| $\gamma_7$ (h^{-1}) | (0.00, 0.00) | (0.00, 0.00) | 0.01 | 0.01 | 0.01 | 0.20 |
| $\gamma_8$ (h^{-1}) | (1.00, 1.00) | (1.00, 1.00) | (0.95, 0.98) | (0.60, 0.60) |

| PDE Error (%) | (10.642, 14.6) | (8.664, 12.528) | (10.425, 15.873) | (9.365, 14.321) | (8.452, 13.372) |
| $\alpha_1$ | $\alpha_2$ (h^{-1}) | $\beta_1$ (ums^{-1}) | (434.754, 728.258) | (1.01, 0.334) | (0.194, 0.342) | (0.236, 0.331) |
| $\beta_2$ (ums^{-1}) | (810.515, 1541.17) | (381.943, 732.84) | ($\gamma_6$ (h^{-1}) | (0.049, 0.055) | (0.048, 0.054) | (0.055, 0.065) | (0.052, 0.064) |
| $\gamma_7$ (h^{-1}) | (0.00, 0.00) | (0.00, 0.00) | 0.00 | 0.00 | 0.00 | 0.16 |
| $\gamma_8$ (h^{-1}) | (1.00, 1.00) | (1.00, 1.00) | (0.93, 0.63) | (0.63, 0.63) |

| PDE Error (%) | (8.232, 11.244) | (6.793, 9.299) | (8.6, 12.424) | (7.125, 10.417) | (6.767, 10.259) |
| $\alpha_1$ | $\alpha_2$ (h^{-1}) | $\beta_1$ (ums^{-1}) | (470.122, 922.102) | (1.426, 0.056) | (1.248, -0.001) | (1.069, -0.451) |
| $\beta_2$ (ums^{-1}) | (667.259, 1591.49) | (532.904, 1639.44) | ($\gamma_7$ (h^{-1}) | (0.054, 0.061) | (0.054, 0.061) | (0.063, 0.092) | (0.065, 0.087) |
| $\gamma_6$ (h^{-1}) | (0.054, 0.061) | (0.054, 0.061) | 0.00 | 0.00 | 0.00 | 0.32 |
| $\gamma_7$ (h^{-1}) | (0.02, 0.02) | (0.00, 0.00) | 0.00 | 0.00 | 0.00 | 0.08 |
| $\gamma_8$ (h^{-1}) | (1.09, 1.09) | (1.09, 1.09) | (0.93, 0.63) | (0.60, 0.60) |

| PDE Error (%) | (8.891, 11.326) | (7.697, 9.311) | (11.118, 14.583) | (8.136, 11.768) | (7.153, 10.406) |
| $\alpha_1$ | $\alpha_2$ (h^{-1}) | $\beta_1$ (ums^{-1}) | (446.431, 850.315) | (-4.476, -1.94) | (-4.242, -1.758) | (-3.052, -0.532) |
| $\beta_2$ (ums^{-1}) | (646.442, 1202.99) | (636.312, 1463.91) | ($\gamma_6$ (h^{-1}) | (0.069, 0.077) | (0.072, 0.084) | (0.086, 0.101) | (0.082, 0.098) |
| $\gamma_7$ (h^{-1}) | (0.064, 0.079) | (0.073, 0.081) | 0.00 | 0.00 | 0.00 | 0.24 |
| $\gamma_8$ (h^{-1}) | (0.00, 0.00) | (0.00, 0.00) | 0.00 | 0.00 | 0.00 | 0.13 |
| $\gamma_9$ (h^{-1}) | (1.00, 1.00) | (1.00, 1.00) | (0.93, 0.63) | (0.60, 0.60) |

Table H: Model selection results for the five models considered with $N_p = 50$. Interval estimates, PDE errors, and model selection results for the models in (2)–(6) when applied to each data set from Jin et al. [1]. See the paper for more details. The time taken to produce these results is 56 minutes and 7 seconds.
Figure V: Coefficient comparisons for different grid sizes. Comparing coefficients from model 2 for each data set, using different values for $N_p$. The lines shown are the 95% confidence intervals for the corresponding coefficient. The coefficients are all scaled by a value $\tilde{\theta}$, e.g., $\alpha_1$ is scaled by $\tilde{\alpha}_1$, where $\tilde{\theta}$ is the mean value of the coefficient from the results using $N_p = 500$ for that data set, and similarly for the PDE errors.
S6  Notation

In this section we present tables of notation used throughout the manuscript.

| Notation | Description |
|----------|-------------|
| $(x_1, \ldots, x_N)$ | The spatial grid for the experiment, where $x_1 < \cdots < x_N$. Also used for the PDE grid, although the PDE grid goes up to $x_{N_p}$. |
| $(t_1, \ldots, t_M)$ | The temporal grid for the experiment, where $t_1 < \cdots < t_M$. |
| $(x_1^*, \ldots, x_n^*)$ | New points placed between $[x_1, x_N]$ used for bootstrapping. |
| $(t_1^*, \ldots, t_m^*)$ | New points placed between $[t_1, t_M]$ used for bootstrapping. |
| $(\xi_1, \ldots, \xi_n)$ | Scaled version of $(x_1^*, \ldots, x_n^*)$ such that the $\xi_i$ are all in $[0, 1]$. |
| $(\tau_1, \ldots, \tau_m)$ | Scaled version of $(t_1^*, \ldots, t_m^*)$ such that the $\tau_j$ are all in $[0, 1]$. |
| $u_{ij}$ | The experimentally measured density value at $(x_i, t_j)$. |
| $u^*_{ij}$ | The estimate for the cell density from the Gaussian process. |
| $\tilde{u}_{ij}$ | The estimate for the cell density from the numerical solution of the differential equations, given over the experimental grid $(x_i, t_j)$. |
| $\tilde{u}_{ij}^{(b)}$ | Estimates for the cell densities from the numerical solution of the differential equations corresponding to the $b$th bootstrap iteration at $(x_i, t_j)$. |
| $N$ | The number of spatial grid points for the experiment. |
| $M$ | The number of temporal grid points for the experiment. |
| $n$ | The number of spatial grid points for the bootstrapping grid. |
| $m$ | The number of temporal grid points for the bootstrapping grid. |
| $N_p$ | The number of spatial grid points for the discretisation of the PDE. |

Table I: Grid and density notation. Notation used for several grids and for labelling the density values.

References

[1] Jin W, Shah ET, Penington CJ, McCue SW, Chopin LK, Simpson MJ. Reproducibility of scratch assays is affected by the initial degree of confluence: Experiments, modelling and model selection. Journal of Theoretical Biology. 2016; 390:136–145.

[2] Dierckx P. Curve and surface fitting with splines. Oxford: Oxford University Press; 1993.

[3] Barbary K. Dierckx.jl. GitHub Repository. 2014 [cited 2022 May 10]. Available from: https://github.com/kbarbary/Dierckx.jl.

[4] Bezanson J, Edelman A, Karpinski S, Shah VB. Julia: A fresh approach to numerical computing. SIAM Review. 2017; 59:65–98.

[5] Lagergren JH, Nardini JT, Baker RE, Simpson MJ, Flores KB. Biologically-informed neural networks guide mechanistic modeling from sparse experimental data. PLoS Computational Biology. 2020; 16:e1008462.
\( u \in \mathbb{R}^{NM \times 1} \) The experimental density values, \( u = (u_{11}, u_{21}, \ldots, u_{N1}, u_{12}, \ldots, u_{NM})^T \).

\( X \in \mathbb{R}^{2 \times NM} \) The design matrix for the Gaussian process. The first row is \((x_1, \ldots, x_N)\) repeated \(M\) times, and the second row is \((t_1, \ldots, t_1, t_2, \ldots, t_M)\), where each \(t_i\) is repeated \(N\) times followed by the next \(t_{i+1}, i = 1, \ldots, M\).

\( u^* \in \mathbb{R}^{nm \times 1} \) A random variable for the cell densities from the Gaussian process, defined such that \( u = u^* + \sigma_n z \) or \( u^* | X, u, X^* \sim \mathcal{N}(\bar{u}^*, \text{Cov}(u^*)) \).

\( X^* \in \mathbb{R}^{2 \times nm} \) Similarly to \( X \), except instead of \((x_1, \ldots, x_N)\) we use \((\xi_1, \ldots, \xi_n)\) and for \((t_1, \ldots, t_M)\) we use \((\tau_1, \ldots, \tau_m)\).

\( K(P, Q) \) A matrix whose \((i, j)\) value gives the value of the kernel function at \( p_i \) and \( q_j \), \( K_{ij} = k(p_i, q_j) \).

\( k(p_i, q_j) \) The squared-exponential kernel, \( k(p_i, q_j) = \sigma_f^2 \exp \left\{ -\frac{1}{2} (p_i - q_j)^T \Lambda^{-1} (p_i - q_j) \right\} \), where \( \Lambda = \text{diag}(\ell_1^2, \ell_2^2) \).

\( I_N \) The \(N\)-square identity matrix.

\( \bar{u}^* \in \mathbb{R}^{nm \times 1} \) The mean of the Gaussian process, \( \bar{u}^* = K(X^*, X) [K(X, X) + \sigma_n^2 I_{NM}]^{-1} u \).

\( \text{Cov}(u^*) \in \mathbb{R}^{nm \times nm} \) The covariance matrix of the Gaussian process, \( \text{Cov}(u^*) = K(X^*, X^*) - K(X^*, X) [K(X, X) + \sigma_n^2 I_{NM}]^{-1} K(X, X^*) \).

\( z \) The random variable \( z \sim \mathcal{N}(0, I) \).

\( \sigma_n \) The standard deviation of the noise in the data.

\( \sigma_f \) The standard deviation of the noise-free data.

\( \ell_1 \) The spatial length scale.

\( \ell_2 \) The temporal length scale.

\( w \) The number of optimiser restarts.

\( \mathcal{N}(\mu, \Sigma) \) A normal distribution with mean \( \mu \) and covariance matrix \( \Sigma \).

\( \mu \) The mean of the random variable \((u^*, \partial u^*/\partial \tau, \partial u^*/\partial \xi, \partial^2 u^*/\partial \xi^2) | X, u, X^* \).

\( \Sigma \) The covariance matrix of the same random variable above.

\( U \) A random sample, \( U = \mu + L z \).

\( L \) The Cholesky decomposition of \( \Sigma \).

\( \Sigma_{ii} \) The diagonal components of \( \Sigma \), \( \text{diag}(\Sigma) = \text{diag}(\Sigma_{11}, \Sigma_{22}, \Sigma_{33}, \Sigma_{44}) \), where each block is of size \( nm \times nm \).

\( \eta, \eta_i, \eta \) Nugget terms, and \( \eta = (\eta_1 I_{nm}, \eta_2 I_{nm}, \eta_3 I_{nm}, \eta_4 I_{nm}) \).

**Table J: Gaussian process notation.** Notation used for the Gaussian processes.
| Notation | Description |
|----------|-------------|
| $T(t; \alpha)$ | The delay function with parameters $\alpha = (\alpha_1, \ldots, \alpha_a)^T$. |
| $D(u; \beta)$ | The nonlinear diffusivity function with parameters $\beta = (\beta_1, \ldots, \beta_d)^T$. |
| $R(u; \gamma)$ | The reaction function with parameters $\gamma = (\gamma_1, \ldots, \gamma_r)^T$. |
| $K$ | The measured carrying capacity density from Jin et al. [1], given as $K = 1.7 \times 10^{-3}$ cells $\mu$m$^{-2}$. |
| $\theta$ | The parameters $\theta = (\alpha^T, \beta^T, \gamma)^T$. |
| $L_{PDE}(\theta)$ | The loss function over the errors in the assumed form the PDE. |
| $L_{GLS}(\theta)$ | The loss function between the data and the numerical solution of the PDE, using the scaled residuals $(\tilde{u}_{ij} - u_{ij})/\sigma_n$. |
| $\mathcal{L}(\theta)$ | The loss function $\mathcal{L}(\theta) = \log[L_{PDE}(\theta)] + \log[L_{GLS}(\theta)]$. |
| $\hat{\theta}$ | An estimate for the parameters $\hat{\theta} = (\hat{\alpha}^T, \hat{\beta}^T, \hat{\gamma}^T)^T$, given as $\hat{\theta} = \arg\min_{\theta \in \mathbb{R}^{a+d+r}} \mathcal{L}(\theta)$. |
| $a$ | The number of delay parameters. |
| $d$ | The number of diffusion parameters. |
| $r$ | The number of reaction parameters. |
| $\theta_i$ | The $i$th parameter in $\theta$. |
| $B$ | The number of bootstrap iterations. |
| $\mathbf{z}^{(b)}$ | A sample from $\mathcal{N}(\mathbf{0}, \mathbf{I})$. |
| $\mathbf{U}^{(b)}$ | A sample from the joint Gaussian process, $\mathbf{U}^{(b)} = \mathbf{\mu} + \mathbf{Lz}^{(b)}$. |
| $\mathbf{AIC}$ | The Akaike information criterion. |
| $\mathbf{AICc}$ | The Akaike information criterion, corrected for a small sample size. |
| $|\mathcal{M}|$ | The number of proposed models. |
| $\mathcal{M}_i$ | The $i$th candidate model, $i = 1, \ldots, |\mathcal{M}|$. |
| $\mathbf{AICc}_i$ | The $\mathbf{AICc}$ of $\mathcal{M}_i$. |
| $\mathbf{AICc}_{\text{min}}$ | The minimum $\mathbf{AICc}$ over a set of $\mathbf{AICcs}$ from a set of candidate models. |
| $\Delta_i$ | The $\mathbf{AICc}$ difference, $\Delta_i = \mathbf{AICc}_i - \mathbf{AICc}_{\text{min}}$. |
| $\mathcal{M}_i^{(b)}$ | The $b$th realisation of the model $\mathcal{M}_i$, where $i = 1, \ldots, |\mathcal{M}|$ and $j = 1, \ldots, B$. |
| $\mathbf{AICc}_i^{(b)}$ | The $\mathbf{AICc}$ of $\mathcal{M}_i^{(b)}$. |
| $\mathbf{P} \in \mathbb{R}^{3 \times |\mathcal{M}|}$ | The matrix for probabilistic interpretations for model selection from the $\mathbf{AICc}$ results. |
| $\phi_k$ | The $k$th basis function for the nonlinear diffusivity function for the basis function approach. |
| $\psi_k$ | The $k$th basis function for the reaction function for the basis function approach. |
| $\mathbf{A} \in \mathbb{R}^{nm \times (d+r)}$ | The coefficient matrix for the system $\Delta \mathbf{\theta} = \partial \mathbf{u}_x / \partial x$ for the basis function approach. |
| $\delta u$ | The threshold tolerance for the density values. |
| $\delta t$ | The threshold tolerance for the time derivatives of the density values. |

**Table K: Equation learning, bootstrapping, and model selection notation.** Notation used for the equation learning, bootstrapping, and model selection.