Supplementary Material

A. Discussion of additional UQ methods

In this manuscript, we evaluate ensemble-based methods, Markov Chain Monte Carlo (MCMC), and Variational Inference (VI) for Uncertainty Quantification (UQ) in Universal Differential Equations (UDEs). We consider these methods due to their positive evaluation results in previous studies and their general scalability to complex models.

Some UQ approaches commonly used in mechanistic modelling, such as Profile Likelihoods (PLs), were not employed in this study. Profile Likelihoods [1] evaluate the shape of the posterior distribution $p(\boldsymbol{\theta}|\mathcal{D})$ by fixing one parameter θ_i to a value z and re-estimating the maximum-likelihood estimate feasible for this fixed value, i.e.,

$$\mathrm{PL}(z) = \max_{\boldsymbol{\theta} \in \{\boldsymbol{\theta} \mid \theta_i = z\}} p(\mathcal{D}|\boldsymbol{\theta}). \tag{A 1}$$

This procedure is performed multiple times, scanning over a series of fixed values z. Profile Likelihoods are a comparatively fast method to evaluate plausible values in one dimension of $\boldsymbol{\theta}$. For predictive uncertainty, all dimensions of $\boldsymbol{\theta}$ have to be evaluated, which is still comparatively fast if the parameters are not correlated. However, this becomes infeasible for higher dimensions of $\boldsymbol{\theta}$ that are highly correlated. Moreover, previous studies have reported limited capabilities of PL calculations for large-scale Ordinary Differential Equation (ODE) models, making them less suitable for the high-dimensional parameter spaces encountered in UDEs. The computational burden and scalability issues associated with PLs render them impractical for models that integrate machine learning components like neural networks. Hence, while Profile Likelihoods can be used to gain a deeper understanding of the uncertainty of the mechanistic parameters of the UDE, they are not suitable for the evaluation of prediction uncertainty of UDEs.

Similarly, methods based on the Fisher Information Matrix (FIM) were not used. FIM-based approaches require the calculation of state sensitivities, which are difficult to assess for high-dimensional models. More importantly, asymptotic confidence intervals via the FIM [2] provide only a local assessment and depend on the inversion of the FIM to obtain the covariance matrix of a maximum likelihood estimate. While effective in settings with identifiable parameters, these methods become unsuitable if the parameters are partially non-identifiable. This case is to be expected in UDEs, as the ANN adds a large number of degrees of freedom. Accordingly, the FIM is likely not invertible, meaning that confidence intervals for parameters and model predictions are ill-determined [3]. Indeed, UDEs share similar issues with neural networks: Overparameterization and the presence of plateaus in the loss landscape are likely, rendering FIM-based methods less applicable.

Given these considerations, we focus on ensemble-based methods, MCMC, and VI for UQ in UDEs. These methods are better suited to handle the challenges posed by the high dimensionality of the parameter space and the complex structure of the loss or posterior function landscape inherent in these models.

B. Problem overview

In the following, we list a few tables that provide an overview over the different problem scenarios, its initial conditions and parameter values. Furthermore, we provide a visualisation of the values of β for the SEIR Waves and SEIR Pulse settings in Figure 1.

Table S1: Overview of the synthetic problem scenarios considered, including information about the time span, number of time points n_t and parameters $n_{\boldsymbol{\theta}}$.

Dynamic model	Noise model	Noise parameter	time span	n_t	$n_{\boldsymbol{\theta}}$
Quadratic Dynamics	Gaussian	$\sigma = 0.01$ $\sigma = 0.05$	$(0.0, 10.0)$	12	22
SEIR Waves	Gaussian	$\sigma = 0.01$ $\sigma = 0.03$ $\sigma = 0.05$	$(0.0, 130.0)$	30	64
	Negative Binomial	$d = 1.2$ $d = 2.2$			
SEIR Pulse	Gaussian	$\sigma = 0.01$ $\sigma = 0.03$	$(0.0, 130.0)$	30	64
	Negative Binomial	$d = 1.2$ $d = 2.2$			

Table S2: Overview of the initial condition for the synthetic problem scenarios considered simulating data.

Dynamic model	Noise model	Initial condition
Quadratic Dynamics	Gaussian	(0.1)
SEIR Waves	Gaussian	$(0.995, 0.004, 0.001, 0.0)$
	Negative Binomial	$(995.0, 4.0, 1.0, 0.0)$
SEIR Pulse	Gaussian	$(0.995, 0.004, 0.001, 0.0)$
	Negative Binomial	$(995.0, 4.0, 1.0, 0.0)$

Table S3: Overview of the mechanistic parameters of the differential equations for the synthetic problem scenarios considered.

Dynamic model	α	β	γ
Quadratic Dynamics	1.0	2.0	-
SEIR Waves	0.9	-	0.1
SEIR Pulse	0.33	-	0.05

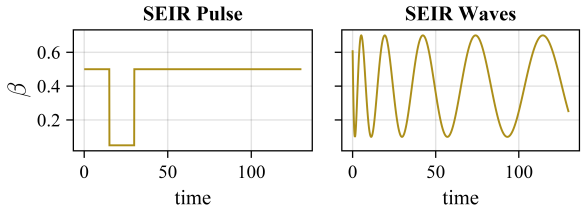

Figure 1: Visualization of the values of β for the data generation process of the SEIR Pulse and SEIR Waves problem scenarios.

C. Quadratic dynamics

This problem is a comparatively small problem with only two non-identifiable mechanistic parameters. While finding local optima in the loss space should be feasible, the identification of distributions of non-identifiable parameters is more complex.

To generate synthetic data for the quadratic dynamic problem, we simulate the differential equation

$$\begin{aligned} \frac{dx}{dt} &= \alpha x - \beta x^2, \\ x(0) &= 0.1 \end{aligned} \tag{A 1}$$

with $t \in (0, 10)$, $\alpha = 1$ and $\beta = 2$. We assume, that the observable mapping is the identity, i.e. x is directly observed. Noise is added to the observable x according to Table S1. We assume that only one of the components of the mechanistic terms in the differential equation is known. Hence, the UDE is defined as

$$\begin{aligned} \frac{d\hat{x}}{dt} &= \hat{\alpha}\hat{x} - f_{\text{net}}(\hat{x}; \boldsymbol{\theta}_{\text{net}}), \\ x(0) &= 0.1 \end{aligned} \tag{A 2}$$

where f_{net} is a fully connected neural network with parameters $\boldsymbol{\theta}_{\text{net}}$. To ensure the positivity of α we parametrize α as $\log(\alpha)$. Both the initial value of $\log(\alpha)$ and that of the noise parameter σ are sampled from a log-uniform distribution with a minimal value of 0.1 and maximum value of 10.0.

D. Implementation details

All experiments were conducted on the Unicorn cluster (CPU cores: 2x AMD EPYC 7F72; 3.2 GHz, 1 TB RAM) at the university of Bonn.

Due to its variety of available solvers and automatic differentiation support, we implemented all experiments in Julia [4]. The ensemble-based parameter estimation was conducted using packages introduced with SciML [5]. For Variational Inference and most of the MCMC sampling algorithms, we used Turing [6]. For parallel tempering, the package Pigeons [7] provided sampling algorithms with an interface to Turing models. A full list of packages is provided in the environment's Manifest and Project files.

As is commonly done in the context of dynamical modelling [8], we transformed the mechanistic parameters for estimation. The standard deviation was optimized in log scale. Furthermore, we implemented a tanh-based transformation for the other mechanistic parameters to ensure consistency with parametric bounds independent of the optimization algorithm (see Appendix (b) for details).

The neural network architecture was the same throughout all reported experiments. Specifically, after a short initial hyperparameter search, we used a feed-forward neural network with 2 hidden layers with 6 neurons each and $\tanh$ activation functions for all layers apart from the output layer.

Synthetic data was created according to the problem description, noise model and ground-truth parameters provided in the sections above. All UQ methods used the same data per problem scenario.

(a) Ensemble-based UQ

The neural networks' initial parameter values are sampled according to the default setting (Glorot uniform [9] for weights, zero for biases) with one exception: We observed a more stable training process with fewer numerical instabilities during the solving process of the dynamic equation

when the initial parameters of the neural network were initialized to values equal to zero. The mechanistic parameters were sampled according to the prior distributions specified in Appendix D (c). Similar to [5], optimization was realised using the optimization algorithms ADAM (for the first 4000 epochs) and then BFGS (up to 1000 epochs). To avoid overfitting, we introduced a small L2 regularization on the neural network parameters (with penalty factor 10^{-5}) and retrospectively stored those parameters per model training that minimized the negative log-likelihood on the respective validation set. The train-validation split was implemented using one individualized random seed per potential ensemble member. The selection of ensemble members from the candidate models was conducted using a significance level of 0.05.

(b) Tanh-based parameter transformation

While box-constraints are available for many optimization algorithms in Julia, this is not the case for the BFGS algorithm. Ensuring that parameters stay within physically plausible bounds is, however, often necessary to define a solvable differential equation. Furthermore, encoding more prior knowledge can narrow down the hypothesis space of the model and, hence, make the exploration of the loss landscape more feasible.

BFGS is a standard optimization algorithm for UDEs [5]. For purely mechanistic dynamical models, primarily other optimizers with customized box-constraint implementations are used [10]. We use a $\tanh$-based transformation of the parameters that allows enforcing box-constraints independent of the optimization algorithm:

Let θ_i^p be the parametrized version of a mechanistic parameter θ_i. By setting $\theta_i = a \cdot \tanh(\theta_i^p - c) + b$ for suitable $a, b, c \in \mathbb{R}$, we can ensure that for any $\theta_i^p \in \mathbb{R}$, θ_i stays within given bounds. The constant c allows for symmetry around $\theta_i^p = 0$.

For the SEIR based problems, the latent period (inverse of α) could reasonably be anywhere from an hour (e.g. for certain foodborne illnesses) to several years (e.g. certain malaria cases), hence we assume $\alpha \in (0, 24)$ to be known. Similarly, we assume that a person stays infectious for at least one day, i.e. $\gamma \in (0, 1)$ and that $\beta(t) \in (0, 3)$. As described in Appendix C, no tanh bounds were used for the quadratic dynamics problem.

(c) Prior definition for the mechanistic and neural network parameters

Table S4 gives an overview of the prior definition of the mechanistic parameters for the different problem scenarios. For Variational Inference and MCMC based sampling, the neural network parameters' prior was defined as $\mathcal{N}(\mathbf{0}, 3 \cdot I)$.

Table S4: Overview of the priors for the mechanistic parameters of the considered problems. Note that the noise parameters σ and d are only defined in a Gaussian and negative Binomial noise setting, respectively. The mentioned Normal distribution is defined by its mean and standard deviation, the (Log-)uniform distribution by its lower and upper bounds and the Beta distribution by its two shape parameters.

Problem	Parameter	Prior on transformed parameter space	Prior
Quadratic Dynamics	α	True	LogUniform$(0.1, 10)$
Quadratic Dynamics	σ^2	True	Uniform(-10,10)
SEIR Waves/Pulse	α	True	Normal(0,1)
SEIR Waves/Pulse	γ	True	Normal(0,1)
SEIR Waves/Pulse	σ^2	True	Uniform(-10,10)
SEIR Waves/Pulse	d	True	Beta(2,2)

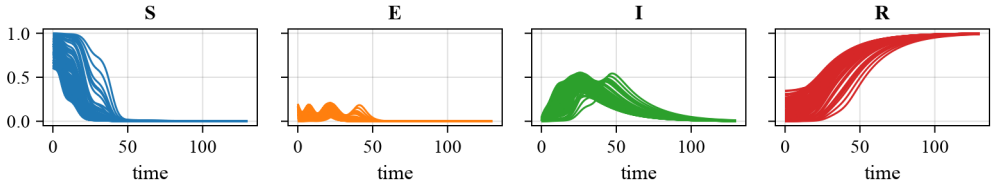

Figure S1: Visualization of trajectories generated from new initial conditions for the SEIR Waves problem.

Table S5: Evaluation of the performance of the MCMC and ensemble-based methods on 100 unseen initial conditions. Shown are the area covered by the posterior distribution (approximated using an Euler scheme calculation) and the squared distance between the ground truth and posterior mean per compartment.

	Area				**Squared Distance** (10^{-4})			
	S	**E**	**I**	**R**	**S**	**E**	**I**	**R**
MCMC	10.44	8.29	7.77	6.20	5.79	7.52	2.77	1.32
Ensemble	15.07	17.91	11.79	11.08	52.91	6.21	29.36	30.19

E. Evaluation of MCMC and ensemble-based methods based on new initial conditions

To evaluate the predictive performance of the MCMC and ensemble based methods on new initial conditions of the SEIR Waves problem, we sampled 100 initial conditions (see Figure S1) and calculated the area covered by the posterior distribution and the squared distance between posterior mean and ground truth without retraining the models. The initial conditions were sampled according to

$$\begin{aligned} S_0 &\sim \mathcal{U}([0.6, 0.999]) \\ E_0 &\sim \mathcal{U}([0.001, \min(0.2, 1 - S_0)]) \\ I_0 &\sim \mathcal{U}([0, \min(0.05, 1 - (S_0 + I_0))]) \\ R_0 &= 1 - (S_0 + E_0 + I_0). \end{aligned} \tag{A 1}$$

Our analysis of these ICs revealed that MCMC-based uncertainty quantification yields a smaller posterior area as well as a smaller squared distance to the ground truth than ensemble-based uncertainty quantification; see Table S5. For the hidden state E, the performance of the MCMC and ensemble-based methods are comparable. This study indicates that there is a clear trend for MCMC to perform better than the ensemble-based method on new initial conditions.

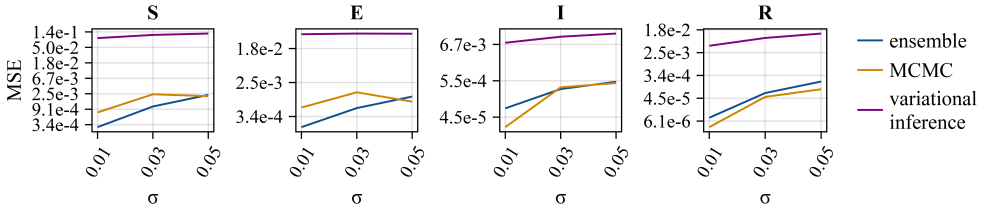

Figure S2: Mean squared error (MSE) between ground truth and posterior mean in dependence of the noise.

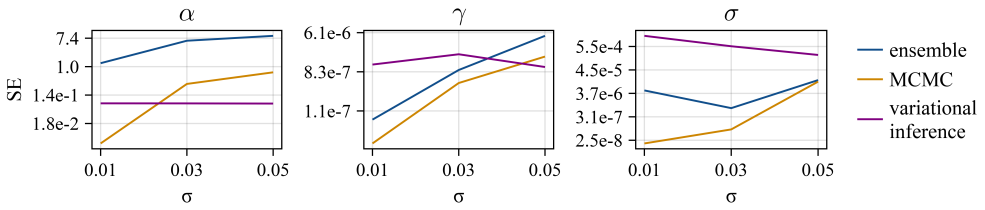

Figure S3: Squared error (SE) between reference value and posterior mean in dependence of the noise.

F. Influence of σ on the accuracy of the posterior distributions

To account for the potential impact of noise on method performance, we compared the uncertainty quantification methods on the SEIR Waves problem with Gaussian noise and standard deviation values 0.01, 0.03 and 0.05. The mean squared error between the ground truth and posterior mean of the three methods increases with increasing noise; see Figure S2. Variational inference struggles even for a noise setting of 0.01, while MCMC and the ensemble-based method perform comparatively well.

The squared error between mean posterior parameter values and reference parameter values tends to favour variational inference for the parameter α and the other methods for σ and γ, see Figure S3. However, the approximation of the posterior distribution of α obtained using variational inference does not cover the ground truth (as is exemplary shown in Figure 2 for a noise value of 0.01). While the respective ensemble- and MCMC-based posteriors indicate a higher uncertainty, the distributions cover the ground truth value.

We believe the mean-field assumption of the applied variational inference method to be too simple to capture the properties of the posterior distributions of dynamical systems. In future work, more complex assumptions of the variational distribution should be investigated.

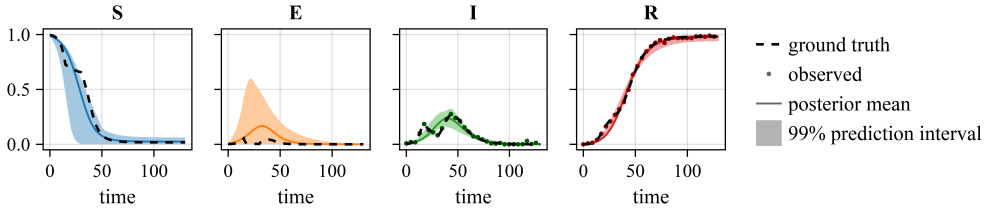

Figure S4: Prediction of the ensemble-based approach given mechanistic knowledge of $\beta(t)$.

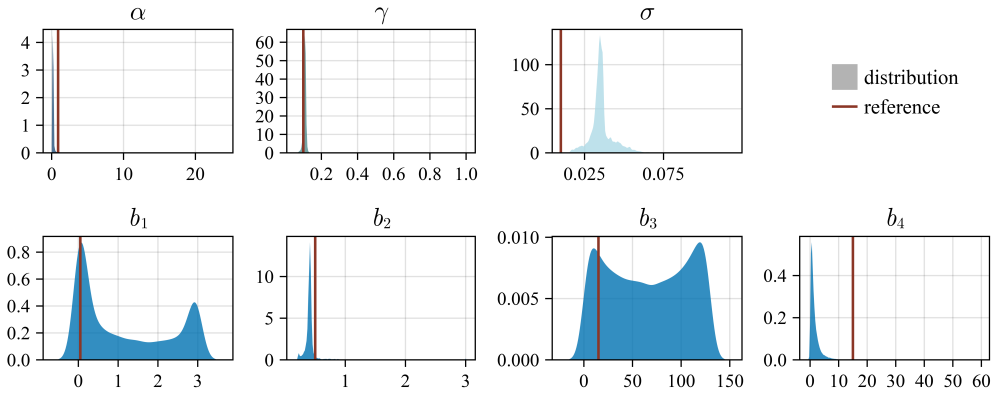

Figure S5: Posterior of the mechanistic parameters of the ensemble-based approach given mechanistic knowledge of $\beta(t)$.

G. Investigation of a mechanistic formulation for β

For all presented problems scenarios, we assumed only partial knowledge of the mechanistic formulation used to generate the data. In the following, we showcase the performance of the ensemble-based approach when - instead of simply using a neural network for β - some mechanistic knowledge is available. Let

$$\beta(t) = \begin{cases} b_1 & \text{if } b_3 < t < b_3 + b_4, \\ b_2 & \text{else.} \end{cases} \tag{A 1}$$

where $b_1, b_2 \in [0, 3]$, $b_3 \in [0, 130]$ and $b_4 \geq 0$. With this, $\beta(t)$ has the same image as previous parametrisations and the data generation process described in Equation 4.2 can be described by the equation above given suitable parameter values. Using data generated for the SEIR Pulse problem (Gaussian noise, $\sigma = 0.01$), the ensemble-based method does not fit the observed states well; it overestimates the noise and struggles to identify b_4 (see Figures S4 and S5).

Since the best negative log-likelihood of the ensemble members (-3.83) is larger than the negative log-likelihood of the ground truth parameters (-4.17) this indicates that the optimiser tends to get stuck in suboptimal local minima. Note, that the initial parameter values are sampled in a range that contains the ground truth values.

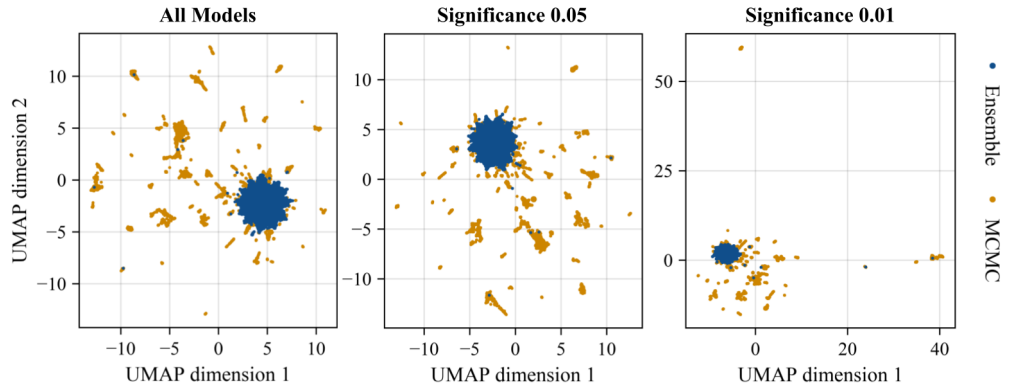

Figure S6: UMAP for the set of parameters obtained using the MCMC and ensemble-based method on the SEIR Pulse problem (Gaussian noise, 0.01). The ensemble members were selected based on a threshold using a significance level of 0.01, 0.05 and using no threshold. The parameter vectors contained mechanistic and neural network parameters.

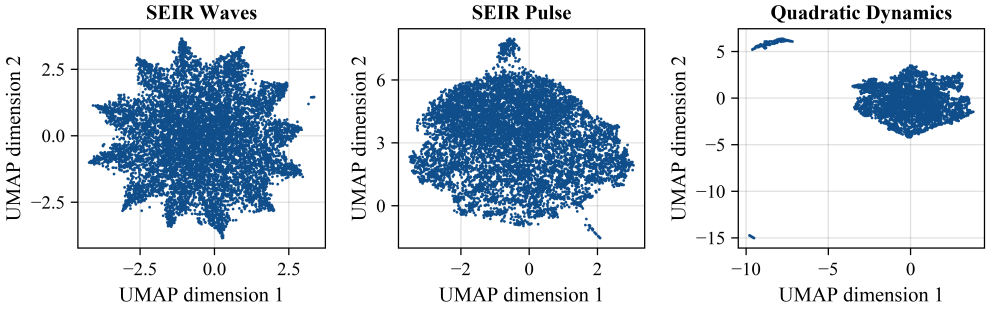

Figure S7: UMAP for the set of parameters obtained using the ensemble-based method for the SEIR Waves, SEIR Pulse and Quadratic Dynamics problems with a Gaussian noise of 0.01. The parameter vectors contained mechanistic and neural network parameters.

H. Extended UMAP analysis

To investigate the clustering behaviour of the ensemble-based method visible in Figure 4, we extended the analysis. Changing the significance level from 0.05 to 0.01 to obtain a threshold, as well as not using a threshold value, does not change the general clustering behaviour observed; see Figure S6. Furthermore, we investigated the ensembles obtained on the SEIR Waves, SEIR Pulse and Quadratic Dynamics problems with a Gaussian noise of 0.01. Again, we find that the clustering seems to be a general behaviour of the ensemble based method; see Figure S7. Both analysis indicate that further research for the ensemble-based method is necessary to avoid clustering, e.g. by using advanced optimizers that overcome local minima or deepening the work on the definition of a suitable threshold.

I. Additional figures

(a) Additional figures for UDE ensembles

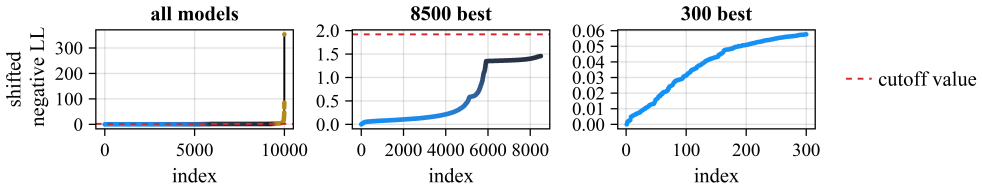

Figure S8: Waterfall plot for the SEIR Pulse problem with Gaussian noise (0.01) for all models (left), the best 8500 models (middle) and the best 300 models (right) according to the negative log-likelihood values (negative LL). The y-values are shifted by the minimal value obtained. The cut-off value clearly discards failed model trainings. Unlike many mechanistic systems, UDEs tend to not converge to a global minimum when using multistart optimization.

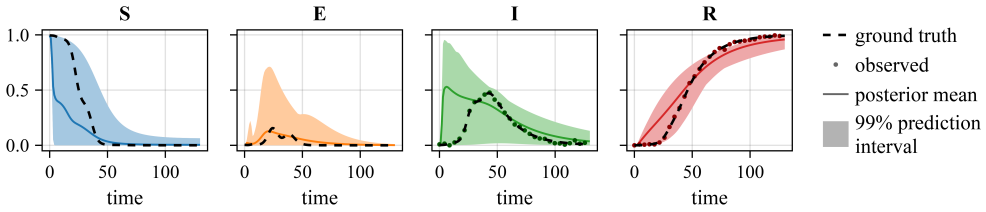

Figure S9: Visualization of the ensemble-based parameter uncertainty estimation assuming a constant β instead of the proposed neural network for the SEIR Waves scenario with Gaussian noise (0.01). Four observations of the state R lie not within the prediction bands.

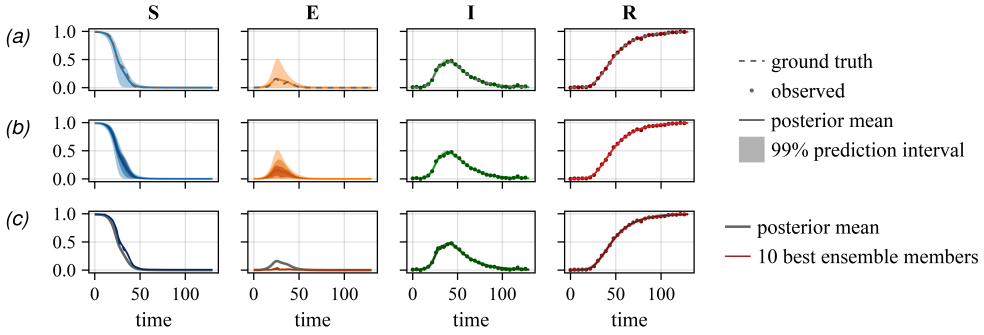

Figure S10: Visualization of the ensemble-based epistemic predictive uncertainty estimation of the SEIR Waves scenario with Gaussian noise (0.01). *(a)* Area covered by all ensemble members per state in comparison to the data generating dynamics (ground truth). *(b)* Area coverages for 99%, 80% or 50% of the best ensemble members. *(c)* Visualization of the trajectories of the 10 best ensemble members (according to the negative log-likelihood). While the 10 best ensemble members approximately follow the same trajectories, they do not coincide with the posterior mean for the states S and E.

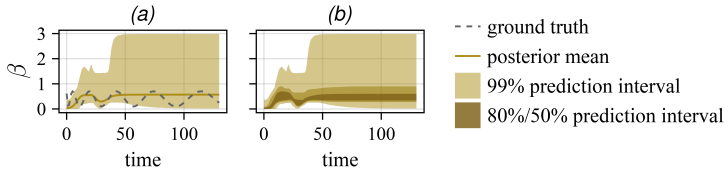

Figure S11: Visualization of the ensemble-based uncertainty of the neural network prediction of β for the SEIR Waves scenario with Gaussian noise (0.01).

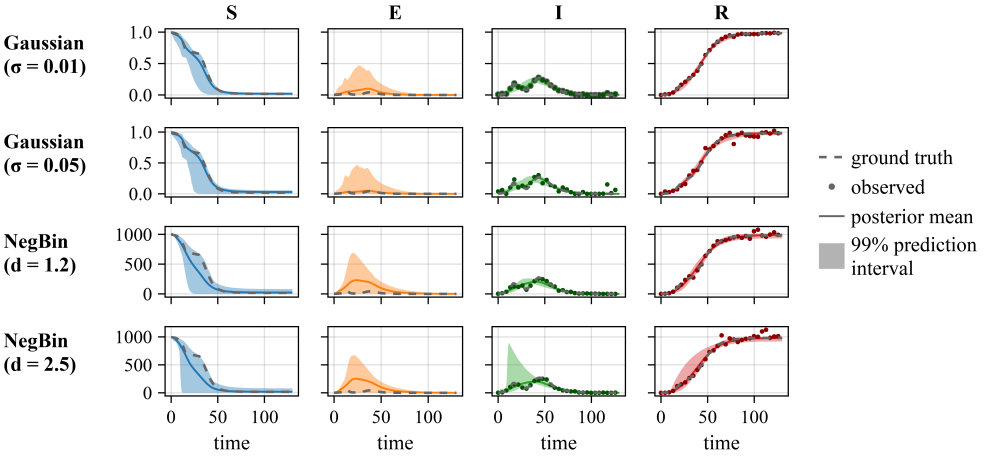

Figure S12: Comparison of different noise model scenarios for the SEIR Pulse problem when using the ensemble-based UQ method.

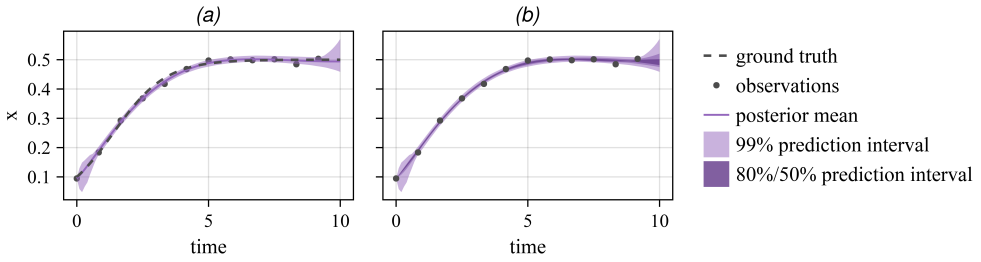

Figure S13: Visualisation of the ensemble-based UQ method for the Quadratic Dynamics problem (Gaussian noise, $\sigma = 0.01$)

(b) Additional figures for UQ based on MCMC

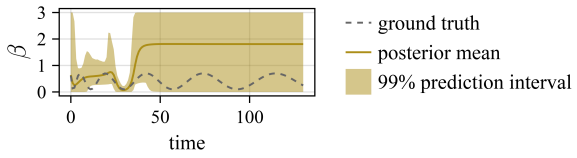

Figure S14: Visualization of the MCMC-based uncertainty of the neural network prediction of β for the SEIR Waves scenario with Gaussian noise ($\sigma = 0.01$).

(c) Additional figures for UQ based on Variational Inference

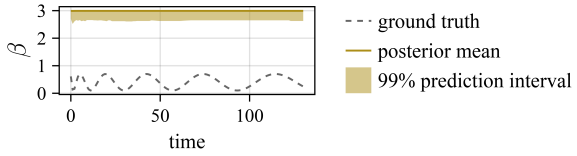

Figure S15: Visualization of UQ of the neural network prediction of β for the SEIR Waves scenario with Gaussian noise ($\sigma = 0.01$) using Variational Inference.

(d) Additional figures for the method comparison

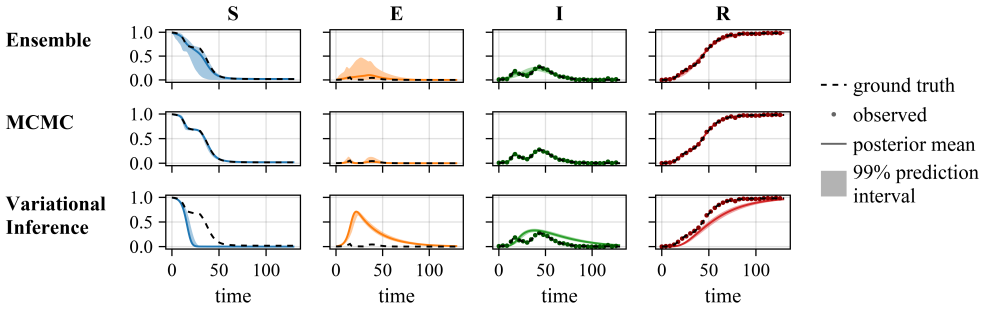

Figure S16: Comparison of the UQ methods on the SEIR Pulse problem (Gaussian noise, $\sigma = 0.03$).

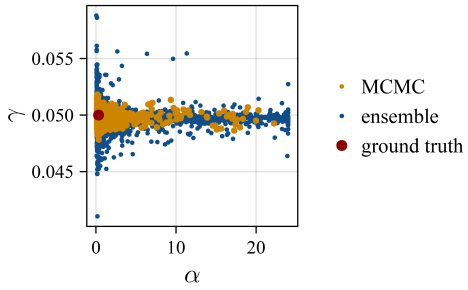

Figure S17: Comparison of the mechanistic parameters of samples of the posterior distributions for the SEIR Waves problem (Gaussian noise, $\sigma = 0.01$).

References

1. Jacquez JA, Greif P. 1985 Numerical parameter identifiability and estimability: Integrating identifiability, estimability, and optimal sampling design. *Mathematical Biosciences* **77**, 201–227.
2. Kreutz C, Raue A, Kaschek D, Timmer J. 2013 Profile likelihood in systems biology. *FEBS* **280**, 2564–2571. (10.1111/febs.12276)
3. Villaverde AF, Raimúndez E, Hasenauer J, Banga JR. 2023 Assessment of Prediction Uncertainty Quantification Methods in Systems Biology. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **20**, 1725–1736. (10.1109/TCBB.2022.3213914)
4. Bezanson J, Edelman A, Karpinski S, Shah VB. 2017 Julia: A fresh approach to numerical computing. *SIAM Review* **59**, 65–98. (10.1137/141000671)
5. Rackauckas C, Ma Y, Martensen J, Warner C, Zubov K, Supekar R, Skinner D, Ramadhan A, Edelman A. 2021 Universal Differential Equations for Scientific Machine Learning. *arxiv*.
6. Ge H, Xu K, Ghahramani Z. 2018 Turing: a language for flexible probabilistic inference. In *AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain* pp. 1682–1690.
7. Surjanovic N, Biron-Lattes M, Tiede P, Syed S, Campbell T, Bouchard-Côté A. 2023 Pigeons.jl: Distributed sampling from intractable distributions. *arXiv:2308.09769*.
8. Hass H, Loos C, Raimúndez-Álvarez E, Timmer J, Hasenauer J, Kreutz C. 2019 Benchmark problems for dynamic modeling of intracellular processes. *Bioinformatics* **35**, 3073–3082.
9. Glorot X, Bengio Y. 2010 Understanding the difficulty of training deep feedforward neural networks. In *PMLR* vol. 9 pp. 249–256 Chia Laguna Resort, Sardinia, Italy.
10. Schälte Y, Fröhlich F, Jost PJ, Vanhoefer J, Pathirana D, Stapor P, Lakrisenko P, Wang D, Raimúndez E, Merkt S, Schmiester L, Städter P, Grein S, Dudkin E, Doresic D, Weindl D, Hasenauer J. 2023 pyPESTO: a modular and scalable tool for parameter estimation for dynamic models. *Bioinformatics* **39**. (10.1093/bioinformatics/btad711)