Approximate Bayesian Computation with Deep Learning and Conformal prediction

Meïli Baragatti, Céline Casenave, Bertrand Cloez, David Métivier,
Isabelle Sanchez

Applibugs, 27 juin 2025

Presentation outline

Parameter inference using ABC

Parameter inference in the Bayesian framework : $\theta \in \mathbb{R}^d$ parameter of interest, $\pi(.)$ the prior of θ and $x \in \mathbb{D} \subseteq \mathbb{R}^n$ observed vector. The **target posterior** is :

$$\pi(\theta \mid \mathrm{x}) = \frac{f(\mathrm{x} \mid \theta)\pi(\theta)}{\int_\Theta f(\mathrm{x} \mid \theta)\pi(\theta)\mathrm{d}\theta} \propto f(\mathrm{x} \mid \theta)\pi(\theta).$$

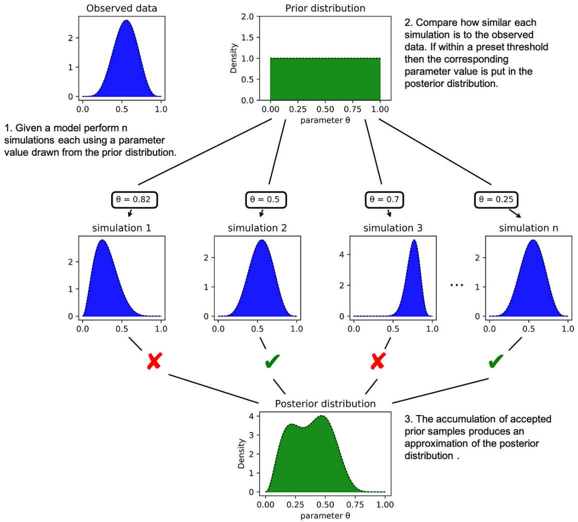
Parameter inference : key role of the likelihood.
Can be difficult to work with :

- no closed form expression.
- prohibitive computational cost.

ABC framework : Computation of the likelihood replaced by the need to be able to simulate from the model under consideration.

$\Longrightarrow$ Approximate the (whole) posterior distribution of interest.

Standard ABC algorithm (Pritchard *et al.* 1999)



source : https ://towardsdatascience.com/the-abcs-of-approximate-bayesian-computation-bfe11b8ca341

Drawbacks :

1. Summary statistics
2. Distance
3. Tolerance level

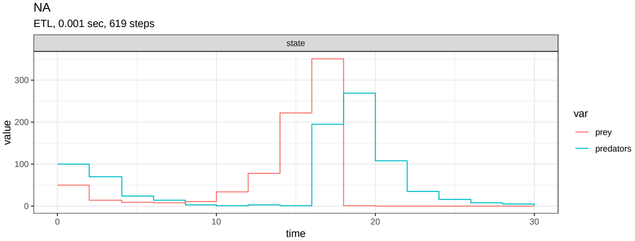
¿ Theoretical guarantees ? Frazier et al. (2018)

Lotka-Volterra

Dynamics of biological systems in which two species interact : prey-predator.
Stochastic Markov jump process version of this model with state $(X_1, X_2) \in \mathbb{Z}^2$.
Three transitions are possible, with hazard rates $\theta_1 X_1$, $\theta_2 X_1 X_2$ and $\theta_3 X_2$:

$$(X_1, X_2) \xrightarrow{\theta_1 X_1} (X_1 + 1, X_2) \qquad \text{(prey growth)}$$
$$(X_1, X_2) \xrightarrow{\theta_2 X_1 X_2} (X_1 - 1, X_2 + 1) \qquad \text{(predation, interaction)}$$
$$(X_1, X_2) \xrightarrow{\theta_3 X_2} (X_1, X_2 - 1) \qquad \text{(predator death)}$$

Parameter of interest
$\theta = (\theta_1, \theta_2, \theta_3)$.
A dataset corresponds to
observations of (X_1, X_2) at times 0, 2, 4, ..., 36, with a non-extinction condition.



Lotka-Volterra

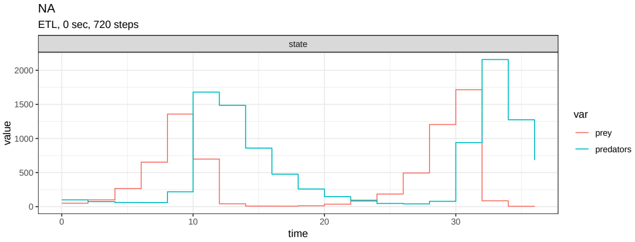
Dynamics of biological systems in which two species interact : prey-predator.
Stochastic Markov jump process version of this model with state $(X_1, X_2) \in \mathbb{Z}^2$.
Three transitions are possible, with hazard rates $\theta_1 X_1$, $\theta_2 X_1 X_2$ and $\theta_3 X_2$:

$$(X_1, X_2) \xrightarrow{\theta_1 X_1} (X_1 + 1, X_2) \qquad \text{(prey growth)}$$

$$(X_1, X_2) \xrightarrow{\theta_2 X_1 X_2} (X_1 - 1, X_2 + 1) \qquad \text{(predation, interaction)}$$

$$(X_1, X_2) \xrightarrow{\theta_3 X_2} (X_1, X_2 - 1) \qquad \text{(predator death)}$$

Parameter of interest
$\theta = (\theta_1, \theta_2, \theta_3)$.
A dataset corresponds to
observations of (X_1, X_2) at times 0, 2, 4, ..., 36, with a non-extinction condition.



Lotka-Volterra

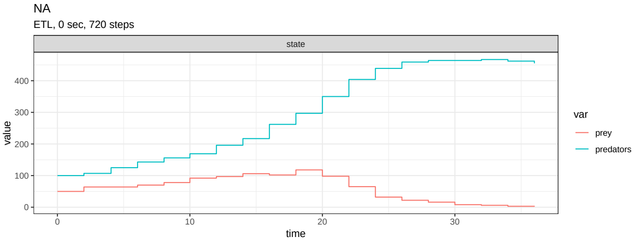
Dynamics of biological systems in which two species interact : prey-predator.
Stochastic Markov jump process version of this model with state $(X_1, X_2) \in \mathbb{Z}^2$.
Three transitions are possible, with hazard rates $\theta_1 X_1$, $\theta_2 X_1 X_2$ and $\theta_3 X_2$:

$$(X_1, X_2) \xrightarrow{\theta_1 X_1} (X_1 + 1, X_2) \qquad \text{(prey growth)}$$

$$(X_1, X_2) \xrightarrow{\theta_2 X_1 X_2} (X_1 - 1, X_2 + 1) \qquad \text{(predation, interaction)}$$

$$(X_1, X_2) \xrightarrow{\theta_3 X_2} (X_1, X_2 - 1) \qquad \text{(predator death)}$$

Parameter of interest
$\theta = (\theta_1, \theta_2, \theta_3)$.
A dataset corresponds to
observations of (X_1, X_2) at times 0, 2, 4, ..., 36, with a non-extinction condition.



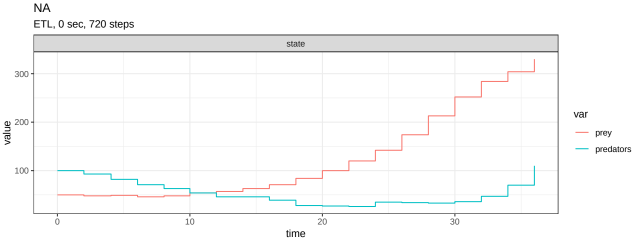
Lotka-Volterra

Dynamics of biological systems in which two species interact : prey-predator.

Stochastic Markov jump process version of this model with state $(X_1, X_2) \in \mathbb{Z}^2$.
Three transitions are possible, with hazard rates $\theta_1 X_1$, $\theta_2 X_1 X_2$ and $\theta_3 X_2$:

$$(X_1, X_2) \xrightarrow{\theta_1 X_1} (X_1 + 1, X_2) \qquad \text{(prey growth)}$$
$$(X_1, X_2) \xrightarrow{\theta_2 X_1 X_2} (X_1 - 1, X_2 + 1) \qquad \text{(predation, interaction)}$$
$$(X_1, X_2) \xrightarrow{\theta_3 X_2} (X_1, X_2 - 1) \qquad \text{(predator death)}$$

Parameter of interest
$\theta = (\theta_1, \theta_2, \theta_3)$.
A dataset corresponds to
observations of (X_1, X_2) at times 0, 2, 4, ..., 36, with a non-extinction condition.



Lake ecosystem : Phytoplankton + Phosphorus dynamics

Observation $X = (C_{\mathrm{P}}(t), C_{\mathrm{IP}}(t)) \in \mathbb{R}^{730\times 2}$.

$$\frac{dC_{\mathbf{P}}}{dt} = \mu(C_{\mathbf{IP}}, T)C_{\mathbf{P}} - d_{\mathbf{P}}C_{\mathbf{P}} \qquad (1)$$

$$\frac{dC_{\mathbf{IP}}}{dt} = V_{\mathbf{min}}(C_{\mathbf{OP}}, C_{\mathbf{IP}}, T) - a_{\mathbf{pc}}\mu(C_{\mathbf{IP}}, T)C_{\mathbf{P}} + (1 - f_{\mathbf{OP}})a_{\mathbf{pc}}d_{\mathbf{P}}C_{\mathbf{P}} \qquad (2)$$

Organic phosphorus concentration :

$$C_{\mathbf{OP}} = \varepsilon(K_{\mathbf{0}} - a_{\mathbf{pc}}C_{\mathbf{P}} - C_{\mathbf{IP}}) \qquad (3)$$

Temperature of water :

$$T(t) = \left(\cos\left(\frac{\pi}{f}t + \pi\right) + 1\right)\frac{T_{\mathbf{max}} - T_{\mathbf{min}}}{2} + T_{\mathbf{min}} \qquad (4)$$

Growth rate of phytoplanton :

$$\mu(C_{\mathbf{IP}}, T) = \mu_{\mathbf{max}}\frac{C_{\mathbf{IP}}}{K_{\mathbf{IP}} + C_{\mathbf{IP}}}\theta_{gr}^{T-20} \qquad (5)$$

Mineralisation speed of the phosphorus :

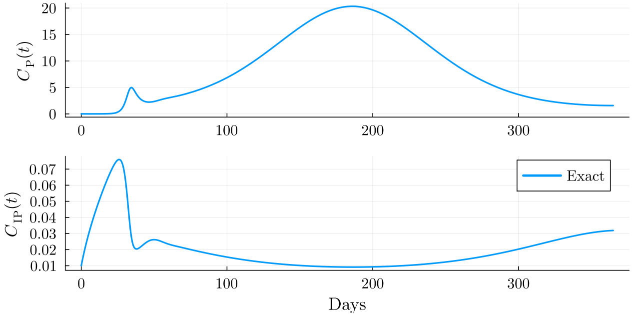
$$V_{\mathbf{min}}(C_{\mathbf{OP}}, C_{\mathbf{IP}}, T) = V_{\mathbf{max}}\frac{C_{\mathbf{OP}}}{K_{\mathbf{OP}}\left(1 + \frac{C_{\mathbf{IP}}}{K_{\mathbf{I}}}\right) + C_{\mathbf{OP}}}\theta_{\mathbf{min}}^{T-20} \qquad (6)$$

Lake ecosystem : Phytoplankton + Phosphorus dynamics

15 parameters, among which 6 have a fixed value
$\longrightarrow \theta$ of dim 9 : $\theta = (\mu_{\text{max}}, K_{\text{IP}}, d_{\text{P}}, f_{\text{OP}}, V_{\text{max}}, K_{\text{OP}}, K_{\text{I}}, \theta_{\text{min}}, \theta_{\text{gr}})$.

A dataset corresponds to obs of $(C_{\text{P}}, C_{\text{IP}})$ at times 0, 0.5, 1, ..., 364.5 days (time series of lengths 730).

1. Summary Stats ? ?
2. Distance : $\mathbb{L}^2$/NMAE ? ?
3. Tolerance level ? ?



Models with high-dimensional parameters → curse of dimensionality.

Towards more efficient approaches

Computational efficiency :

- MCMC algorithms Marjoram et al. (2003), Sisson et al. (2007), Picchini (2014)
- Sequential Monte carlo samplers : Beaumont et al. (2009), Del Moral et al. (2012), Picchini and Tamborrino (2024)

Degree of approx. still depends on the choices of tolerance threshold, the distance, and the summary statistics.

Get rid of tolerance levels and summary statistics, using ML :

- Random Forests, selection of relevant summary statistics Raynal et al. (2018)
- Neural Networks, construction of relevant summary statistics Jiang et al. (2017), Akesson et al. (2022)

Our contribution

A new ABC implementation, ABC Deep Learning & Conformal :

- \+ No summary statistics.
- \+ No distance.
- \+ No tolerance level.
- \+ For multidimensional parameters.
- – Does not give an approx of the whole posterior distribution, but point estimates.
- \+ Gives associated confidence sets (with a proper frequentist coverage).

Based on neural networks with Monte Carlo dropout + conformal prediction

It requires :

1. A sampling function.
2. A neural network whose architecture is tailored to our problem of interest.
3. A confidence level δ.

Presentation outline

Principle

Focus on functionals of the form $\mathbb{E}_\pi[T(\theta) \mid x]$:

moments of the posterior distribution $\pi(\theta \mid x)$, posterior mean $T(\theta) = \theta$, posterior variance $T(\theta) = \theta^2$, posterior quantiles $T(\theta) = 1_{\theta \leq q}$...

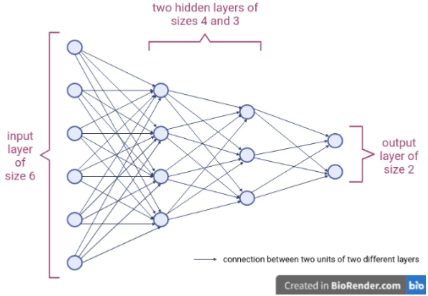
Neural Network (NN) :

- Very powerful for complex problems : multidimensional inputs or outputs, local dependence structure, etc.
- Raw datasets x as inputs.
- $T(\theta)$, as output, can be multidimensional.

Valid confidence sets :

1. Dropout layers in NN : estimate $\mathbb{E}_\pi[T(\theta) \mid x]$ with an associated uncertainty.
2. Conformal prediction.

Neural Network (NN)



Architecture :

- number of layers.
- number of units per layer.
- Activation functions.
- Type of layers (dense, recurrent, convolutional,...)

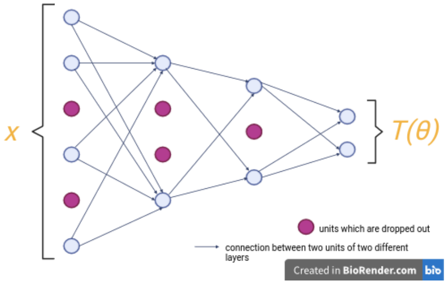
Parameters :

- Weight matrices.
- Biases.

The relationship between the input and the output can be non-linear and quite complex.

NN with Dropout

Dropout layer : **at every training step, some neurons are randomly dropped.**



Dropout introduces randomness in the NN, and so in the output.

Used to associate uncertainties with predictions from the network. Gal (2016), Gal and Ghahramani (2016)

Repeating the same prediction task several times (Monte carlo Dropout)

$\Longrightarrow$ different outputs $\Longrightarrow$ a prediction with an associated uncertainty.

NN with dropout : Monte Carlo approach

k^{th} iteration : the NN outputs a prediction $f^{\widehat{\omega_k}}(x)$ + an associated model precision $\tau_{\widehat{\omega_k}}$.

NN with dropout : Monte Carlo approach

k^{th} iteration : the NN outputs a prediction $f^{\widehat{\omega_k}}(x)$ + an associated model precision $\tau_{\widehat{\omega_k}}$.

Approximation of $\mathbb{E}_\pi[\theta \mid x]$:

$$\widehat{\theta}(x) = \frac{1}{K}\sum_{k=1}^{K} f^{\widehat{\omega_k}}(x) \tag{7}$$

Empirical mean of the predictions of the K forward stochastic passes through the network with Dropout.

Associated uncertainty :

$$\widehat{\mathbb{V}}[\theta \mid x] = \underbrace{\tau^{-1}I}_{\widehat{\mathbb{V}}_a[\theta|x]} + \underbrace{\frac{1}{K}\sum_{k=1}^{K} f^{\widehat{\omega_k}}(x)^T f^{\widehat{\omega_k}}(x) - \widehat{\theta}(x)^T \widehat{\theta}(x)}_{\widehat{\mathbb{V}}_e[\theta|x]}, \tag{8}$$

Inverse model precision plus the sample variance of the predicions of the K stochastic passes through the network with Dropout.

τ^{-1} estimated by the empirical mean of the $\tau_{\widehat{\omega_k}}^{-1}$.

Gal [2016], Gal et al. [2017].

Conformal prediction : principle

1. Computation of the conformal quantile :

Calibration set $\big((\theta_j, x_j), j = 1, \ldots, N_{\text{cal}}\big)$, i.i.d. with test data + a level $\delta \in [0, 1]$.
Each x_j in the calibration set is passed through the trained NN with Dropout, K times :

- Averaging the K predictions to obtain an approx. of $\mathbb{E}_\pi[\theta_j \mid x_j]$: $\widehat{\theta}(x_j)$
- Associated heuristic uncertainties are obtained : $\widehat{\mathbb{V}}(x_j)$
- A score is computed for x_j : $s(\theta, x_j) = \sqrt{(\theta_j - \widehat{\theta}(x_j))^t \widehat{\mathbb{V}}(x_j)^{-1} (\theta_j - \widehat{\theta}(x_j))}$.

Conformal quantile $\hat{q}$: the $\lceil (N_{\text{cal}} + 1)(1 - \delta) \rceil / N_{\text{cal}}$ quantile of the N_{cal} scores.

Conformal prediction : principle

1. Computation of the conformal quantile :

Calibration set $\left((\theta_j, x_j), j = 1, \ldots, N_{cal}\right)$, i.i.d. with test data + a level $\delta \in [0, 1]$.

Each x_j in the calibration set is passed through the trained NN with Dropout, K times :

- Averaging the K predictions to obtain an approx. of $\mathbb{E}_\pi[\theta_j \mid x_j]$: $\widehat{\theta}(x_j)$
- Associated heuristic uncertainties are obtained : $\widehat{\mathbb{V}}(x_j)$
- A score is computed for x_j : $s(\theta, x_j) = \sqrt{(\theta_j - \widehat{\theta}(x_j))^t \widehat{\mathbb{V}}(x_j)^{-1}(\theta_j - \widehat{\theta}(x_j))}$.

Conformal quantile $\hat{q}$: the $\lceil (N_{cal} + 1)(1 - \delta) \rceil / N_{cal}$ quantile of the N_{cal} scores.

2. A confidence set for θ given any new x :

x is passed through the trained NN with Dropout, K times : $\widehat{\theta}(x)$ and $\widehat{\mathbb{V}}(x)$.

$$\mathcal{C}(x) = \{\theta \mid s(\theta, x) \leq \widehat{q}\}, \tag{9}$$

Ellipsoid of center $\widehat{\theta}(x)$ and cov. matrix $\frac{\widehat{\mathbb{V}}(x)^{-1}}{\widehat{q}^2}$; in dim 1 : $\left[\widehat{\theta}(x) - \widehat{q}\sqrt{\widehat{\mathbb{V}}(x)}; \widehat{\theta}(x) + \widehat{q}\sqrt{\widehat{\mathbb{V}}(x)}\right]$.

Conformal prediction : property

Marginal coverage property

for $\delta \in [0,1]$ a user-chosen level :

$$1 - \delta \leq \mathbb{P}[\theta \in \mathcal{C}(\mathrm{x})] \leq 1 - \delta + \frac{1}{N_{cal} + 1}. \quad (10)$$

Conformal prediction : property

Marginal coverage property

for $\delta \in [0,1]$ a user-chosen level :

$$1 - \delta \leq \mathbb{P}[\theta \in \mathcal{C}(\mathrm{x})] \leq 1 - \delta + \frac{1}{N_{cal} + 1}. \tag{10}$$

$\hat{\theta}(\mathrm{x})$ + heuristic uncertainty $\widehat{\mathbb{V}}[\theta \mid \mathrm{x}] \xrightarrow[\text{procedure}]{\text{conformal}}$ rigourous confidence set !

- Provides a valid frequentist coverage in an ABC framework.
- Non-asymptotic guarantees, even without distributional assumptions or model assumptions.
- User-friendly, quick and easy to implement.

ABCD-conformal implementation

1. Generate a **reference table** (training dataset), **(a validation) and a calibration dataset**.

2. **Train the NN with concrete dropout** on the reference table. A validation dataset can be used to compare different architectures of NNs.

3. **Monte Carlo dropout prediction for each x_j in the calibration set to obtain the conformal quantile**.

4. **For a given new data sample x, approx. of $\mathbb{E}_\pi[\theta \mid x]$ + confidence set** :
   x is passed through the trained NN with Dropout, K times :
   - Averaging the K predictions to obtain an approx. of $\mathbb{E}_\pi[\theta \mid x]$.
   - Associated heuristic uncertainties are obtained.

   approx. of $\mathbb{E}_\pi[\theta \mid x]$ + an associated uncertainty + the conformal quantile $\Longrightarrow$ confidence set.

Presentation outline

Compared methods

Example	θ dimension	ABC methods					Language
		stand.	SMC	CNN	RF	D-conf	
MA2	2	✓	✓	✓	✗	✓	Julia, R
Gaussian field	1	✓	✗	✓	✓	✓	R
Lotka-Volterra	3	✓	✓	✓	✗	✓	Julia, R
Lake model	9	✓	✓	✓	✗	✓	Julia

Same datasets for all methods, except for ABC-SMC.

- Standard ABC, ABC-SMC, ABC-CNN : approx. of the whole posterior distribution $\pi(\theta \mid x)$.
- ABC-RF, ABCD-conf : approx. of $\mathbb{E}_\pi[\theta \mid x]$ to estimate true θ + confidence sets.

Comparison criteria between methods

For each test sample x generated from $\theta \xrightarrow[\text{method}]{\text{ABC}} \widehat{\theta}$ + credible or confidence set.

Normalized Mean Absolute Error (for each marginal of θ) :

$$\text{NMAE} = \frac{\sum_{i=1}^{N_{\textbf{test}}} \mid \theta_i - \widehat{\theta}_i \mid}{\sum_{i=1}^{N_{\textbf{test}}} \theta_i},$$

Standard deviations of the absolute errors (for each marginal of θ) :

$$sd(|\theta_i - \widehat{\theta}|) = \sqrt{\frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\textbf{test}}} (\theta - \widehat{\theta}_i)^2}.$$

Credible and confidence sets/intervals : comparisons of coverages, mean and median lengths for intervals, volumes for multidimensional ellipsoids.

Lotka-Volterra (1/3)

	Standard ABC	ABC-SMC	ABC-CNN	ABCD-Conf. overall	ABCD-conf. constant
θ_3 parameter					
$NMAE(\theta_3)$	0.195	0.173	0.091	**0.081**	
$sd(\lvert\theta_3 - \widehat{\theta}_3\rvert)$	0.438	0.461	**0.218**	0.222	
mean length $CI(\theta_3)$	0.697	0.365	**0.278**	0.318	
median length $CI(\theta_3)$	0.268	0.175	0.177	**0.113**	
coverage $CI(\theta_3)$	**97.2%**	92.4%	95.7%	94.6%	
θ parameter (3 dimensional)					
mean volume $CE(\theta)$	0.0197	0.00269	**0.00207**	0.0336	0.0230
median volume $CE(\theta)$	0.00028	**0.00008**	0.0004	0.00015	0.0230
coverage $CE(\theta)$	95.9%	91.5%	88.8%	94.6%	**96.7%**

- NMAE and sd : ABCD-Conformal & ABC-CNN outperform stand. ABC and ABC-SMC.
- Mean and median lengths of CI : sometimes ABC-CNN is better, other times it is ABCD-conformal.
- coverages of CI or CE : ABC-CNN & ABC-SMC are discarded.
- Impact of the heuristic uncertainty measure used for the conformal procedure.
- Big differences between mean and median lengths and volumes.

Marginally and globally, ABCD-conformal gives the most acceptable results.

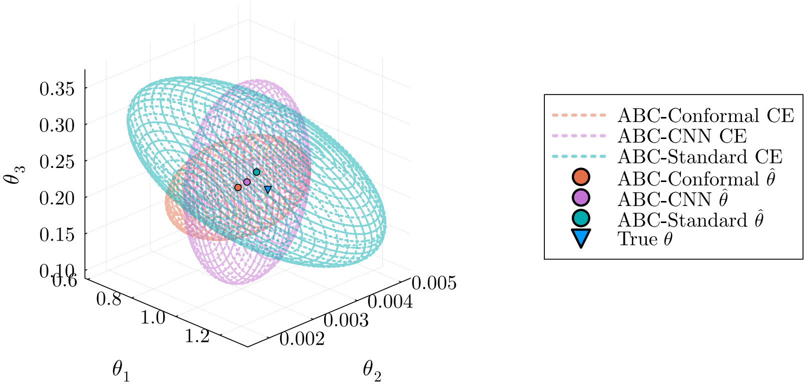
Lotka-Volterra (2/3)

	Stand. ABC	ABC-SMC	ABC-CNN	ABCD-Conf.
$\theta_3 < 1$, 782 test samples				
mean length $CI(\theta_3)$	0.255	0.164	0.166	**0.120**
median length $CI(\theta_3)$	0.192	0.120	0.148	**0.0967**
coverage $CI(\theta_3)$	98.1%	94.4%	**97.1%**	94.8%
$1 \leq \theta_3 \leq 3$, 178 test samples				
mean length $CI(\theta_3)$	1.922	0.889	0.580	**0.560**
median length $CI(\theta_3)$	1.41	0.745	0.517	**0.399**
coverage $CI(\theta_3)$	97.8%	91.0%	94.4%	95.5%
$3 < \theta_3$, 40 samples				
mean length $CI(\theta_3)$	3.89	1.97	**1.13**	3.11
median length $CI(\theta_3)$	4.19	1.98	**1.13**	2.41
coverage $CI(\theta_3)$	77.5%	60.0%	75.0%	**87.5%**

- Lengths of CI and coverage depends on the regions for ALL methods.
- ABCD-Conformal is the least bad on harder regions (coverage) and very good in easier regions. (coverage + lengths).

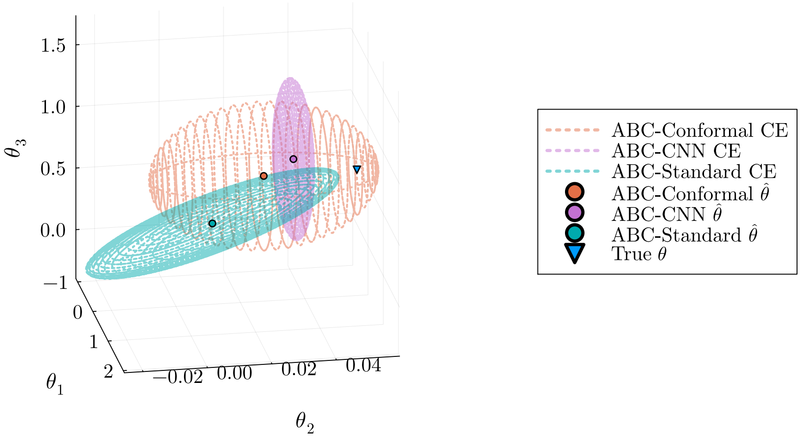
ABCD-Conformal is often better than the other methods, with better predictions and smaller confidences sets, associated to good coverages.

Lotka-Volterra (3/3)



For a given sample, estimated parameter θ with associated 3D confidence ellipsoids obtained using standard ABC, ABC-CNN and ABCD-conformal, and true value of θ.

Lotka-Volterra (3/3)



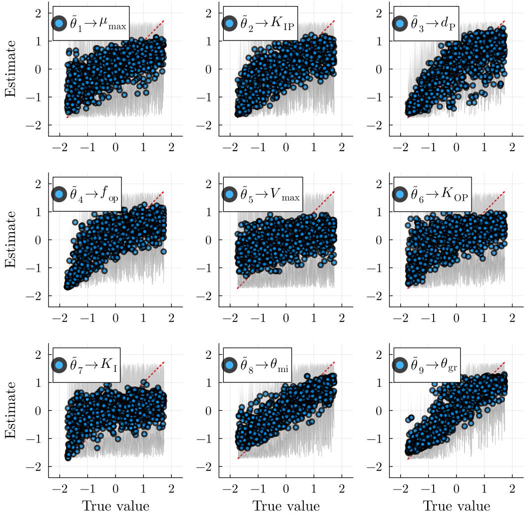
For a given sample, estimated parameter θ with associated 3D confidence ellipsoids obtained using standard ABC, ABC-CNN and ABCD-conformal, and true value of θ.

Lake ecosystem dynamics (1/3)

	Stand. ABC	ABC-SMC	ABC-CNN	ABCD-Conf.
θ_1 parameter				
$NMAE(\theta_1)$	0.6988	0.6298	0.3438	**0.2318**
$sd(\lvert\theta_1 - \widehat{\theta_1}\rvert)$	0.4150	0.4523	0.2096	**0.1782**
mean length $CI(\theta_1)$	2.5674	2.0100	1.6073	**1.1071**
median length $CI(\theta_1)$	2.6670	2.3142	1.6073	**1.0822**
coverage $CI(\theta_1)$	93.9%	91.6%	95.4%	**96.0%**
θ_6 parameter				
$NMAE(\theta_6)$	0.7596	0.6933	0.7396	**0.6555**
$sd(\lvert\theta_6 - \widehat{\theta_6}\rvert)$	0.4392	0.4596	0.4870	**0.4180**
mean length $CI(\theta_6)$	2.5613	2.1558	**1.9344**	2.5650
median length $CI(\theta_6)$	2.6478	2.3235	**1.9508**	2.9140
coverage $CI(\theta_6)$	90.0%	86.9%	78.6%	**95.4%**
θ parameter, 9-dimensional				
mean volume $CE(\theta)$	2235	**57.12**	823.3	110.5
median volume $CE(\theta)$	11683	**14.70**	765.0	102.8
coverage $CE(\theta)$	90.3%	73.4%	87.9%	**95.4%**

- NMAE and sd : ABCD-conf often outperforms standard ABC, ABC-SMC and ABC-CNN.
- Mean and median lengths of CI : often smaller for ABCD-conf. If not, other methods have too small coverages.
- ABCD-conf always has a good coverage on the contrary to other methods.
- Confidence ellipsoids : ABCD-conformal is the best considering volume & coverage.

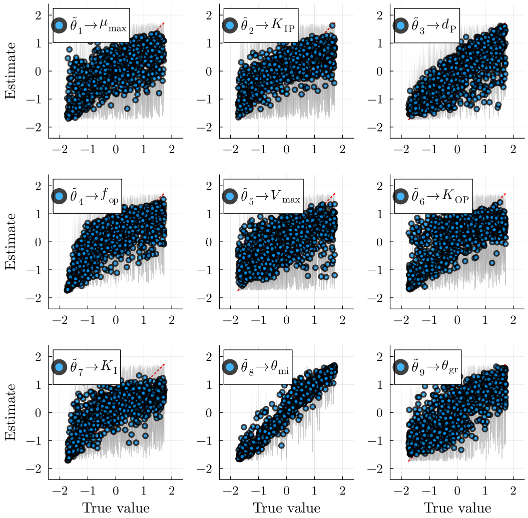
Lake ecosystem dynamics (2/3)



Standard ABC : estimated against true value for each component.

Not very good.

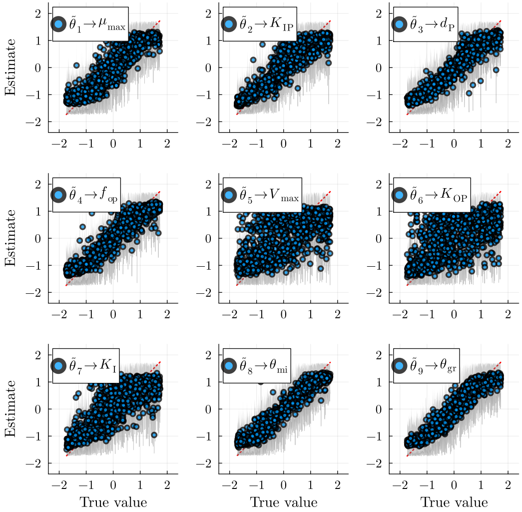
Lake ecosystem dynamics (2/3)



ABC-SMC : estimated against true value for each component.

Not very good but slighty better for θ_8.

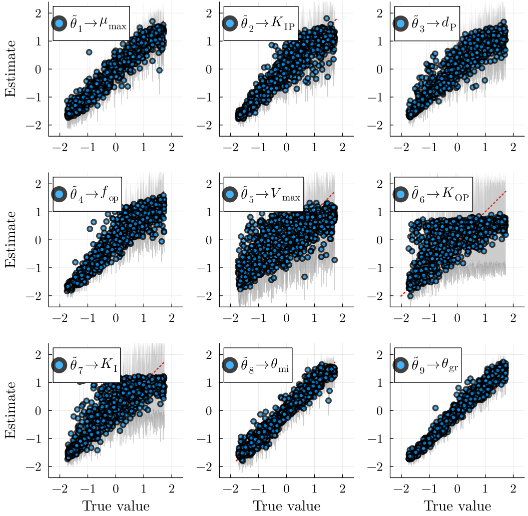
Lake ecosystem dynamics (2/3)



ABC-CNN : estimated against true value for each component.

Improvement thanks to NN
$\Longrightarrow$ pitfall of using summary statistics.

Lake ecosystem dynamics (2/3)



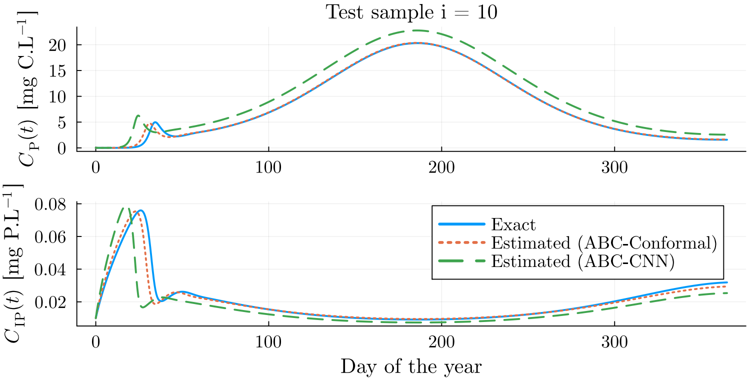
ABCD-conformal : estimated against true value for each component.

Improvement
$\Longrightarrow$ pitfall of using distance and threshold.

Some components are not identifiable (e.g. $\theta_5, \theta_6, \theta_7$).

Lake ecosystem dynamics (3/3)

It can be more a calibration problem than a parameter estimation problem.
$\longrightarrow$ finding a parameter set for which the simulated outputs are close to the observed data.



Reconstructions of times series of P and IP, using the estimated θ parameters, for the 10^th test sample.

Presentation outline

Strengths of ABCD-conformal

A new ABC implementation that combines Neural Networks with Monte Carlo dropout and a conformal procedure :

- No summary statistic.
- No distance nor tolerance level.
- Deals with multidimensional parameters of interest.
- Confidence sets with guaranteed non asymptotic coverage.

Strengths of ABCD-conformal

A new ABC implementation that combines Neural Networks with Monte Carlo dropout and a conformal procedure :

- No summary statistic.
- No distance nor tolerance level.
- Deals with multidimensional parameters of interest.
- Confidence sets with guaranteed non asymptotic coverage.

In practice :

- Computationally efficient, good results.
- Among the best if not the best when compared to other methods.
- Suffer less of the curse of dimensionality than other methods.
- An alternative to other methods when there is no obvious summary statistic.
- Amortized.

Drawbacks and perspectives

Drawbacks

Multimodal posteriors ABCD-conf unable to detect multimodality.

Coverage valid marginally and not conditionally

- Performances can vary depending on the region of the parameter.
- Like all the methods in practice !
- ABCD-conf able to adapt confidence sets to easy/difficult values if we have a good heuristic uncertainty.

Perspectives

Heuristic uncertainty Other uncertainty methods than using MC dropout could be used.

Conformal procedure we focused on conformalizing a scalar uncertainty estimate in a split conformal procedure. Other conformal procedures can be used.

Approximate the full posterior

Supplementary material and softwares

- GitLab repository with source codes and datasets : https://forge.inrae.fr/mistea/codes_articles/abcdconformal.
- Quarto and Rmarkdown notebooks directly available : https://mistea.pages-forge.inrae.fr/codes_articles/abcdconformal/.
- Julia package ABCMethods.jl accessible in https://github.com/dmetivie/ABCMethods.jl.

Paper on arXiv : https://arxiv.org/abs/2406.04874, and recently accepted in the Journal of Computational and Graphical Statistics.

References

- Akesson et al. *Convolutional Neural Networks as Summary Statistics for Approximate Bayesian Computation.* IEEE/ACM Trans Comput Biol Bioinform (2022).
- Del Moral et al. *An adaptive sequential monte carlo method for approximate bayesian computation.* in Conformal and Probabilistic Prediction with Applications, PMLR (2022).
- Messoudi et al. *Ellipsoidal conformal inference for multi-target regression.* Statistics and Computing (2012).
- Raynal et al. *ABC random forests for Bayesian parameter inference.* Bioinformatics (2018).
- Gal and Ghahramani. *Dropout as a bayesian approximation : Representing model uncertainty in deep learning.* International conference on machine learning (2016).
- Gal et al. *Concrete dropout.* 31st Conference on Neural Information Processing System (2017).
- Frazier et al. *Asymptotic properties of approximate Bayesian computation.* Biometrika (2018).
- Rousseau and Szabo. *Asymptotic frequentist coverage properties of Bayesian credible sets for sieve priors in general settings.* Annals of Statistics (2016).
- Angelopoulos and Bates. *A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification.* arXiv (2022).

Appendix

Presentation outline

Quantification of uncertainties

Posterior credible sets :

- Highest posterior density regions.
- Use of posterior quantiles.

For simple parametric models : posterior credible sets $\overset{\text{asymp.}}{=}$ confidence sets.

Quantification of uncertainties

Posterior credible sets :

- Highest posterior density regions.
- Use of posterior quantiles.

For simple parametric models : posterior credible sets $\overset{\text{asymp.}}{=}$ confidence sets.

How much we can trust Bayesian credible sets as a measure of confidence in the statistical procedure from a frequentist perspective ? Rousseau and Szabo (2016)

Quantification of uncertainties

Posterior credible sets :

- Highest posterior density regions.
- Use of posterior quantiles.

For simple parametric models : posterior credible sets $\overset{\text{asymp.}}{=}$ confidence sets.

How much we can trust Bayesian credible sets as a measure of confidence in the statistical procedure from a frequentist perspective ? Rousseau and Szabo (2016)

If the posterior coverage probability of a Bayesian credible set does not match with the corresponding frequentist coverage probability, what does the Bayesian credible set mean objectively or empirically ?

Is it relevant for applied statisticians ?

Quantification of uncertainties

Posterior credible sets :

- Highest posterior density regions.
- Use of posterior quantiles.

For simple parametric models : posterior credible sets $\stackrel{\text{asymp.}}{=}$ confidence sets.

How much we can trust Bayesian credible sets as a measure of confidence in the statistical procedure from a frequentist perspective ? Rousseau and Szabo (2016)

If the posterior coverage probability of a Bayesian credible set does not match with the corresponding frequentist coverage probability, what does the Bayesian credible set mean objectively or empirically ?

Is it relevant for applied statisticians ?

Probability matching criterion :

- For nonparametric and high-dimensional models ? Rousseau and Szabo (2016), Datta et al. (2000), Hoff (2021), Wasserman (2011)
- In an ABC framework ? Frazier et al. (2018)

Presentation outline

Marginal vs conditional coverage

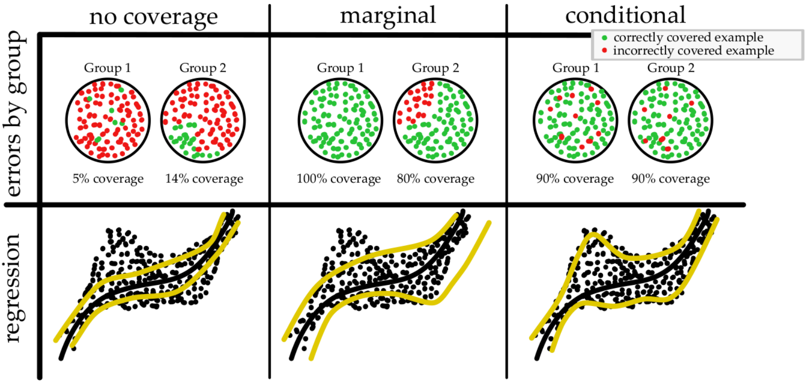


Figure 10: Prediction sets with various notions of coverage: *no coverage, marginal coverage, or conditional coverage (at a level of 90%). In the marginal case, all the errors happen in the same groups and regions in X-space. Conditional coverage disallows this behavior, and errors are evenly distributed.*

source : Angelopoulos and Bates (2022).

Conformal prediction : performances

A "good" conformal prediction procedure :

- Small sets on easy inputs.
- Large sets on hard inputs in a way that faithfully reflects the model's uncertainty.

Performance depends only on the quality of the uncertainty measure used :
it should reflect the magnitude of model error, smaller for easy input, and larger for hard ones.

A conditional coverage would be preferable :
Marginal property guarantees average coverage over the whole domain, and not the coverage for any specific x :

$$\mathbb{P}[\theta \in \mathcal{C}(\mathrm{x}) \mid \mathrm{x}] \geq 1 - \delta. \quad (11)$$

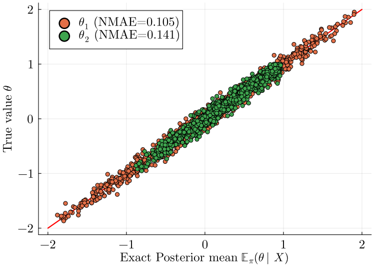
Presentation outline

True value vs posterior mean of θ

ABCD-conformal aims to provide $\widehat{\theta}(x)$ an estimate of the posterior mean $\mathbb{E}_\pi[\theta \mid x]$, but we compare $\widehat{\theta}(x)$ to the true value of θ used to generate x.
These two quantities $\mathbb{E}_\pi[\theta \mid x]$ and θ can be quite bifferent, but :

- The exact posterior means are often unknown.
- Likelihood based on a large number of observations : results such as the Bernstein-von Mises Theorem, ensure that, often, these two quantities are close.

MA2 example :



ABCD-conformal enables to obtain a confidence set for θ and not $\mathbb{E}_\pi[\theta \mid x]$.

Estimate of the posterior mean

Estimate of the posterior mean $\mathbb{E}_\pi[\theta \mid x]$

- Standard ABC and ABC-CNN : $\widehat{\theta}_i$ = empirical mean of the θ corresponding to the αN_{train} samples kept to approximate the posterior distribution of θ given the i^{th} dataset.
- ABC-RF : directly gives estimates $\widehat{\theta}_i$.
- ABC-SMC : $\widehat{\theta}_i$ = the weighted mean of the last population.
- ABCD-Conformal : $\widehat{\theta}_i$ = empirical mean of K stochastic forward passes through the network with Dropout, given the i^{th} dataset as input.

Confidence/credible sets

Credible and confidence intervals (for each marginal of θ)

- Standard ABC and ABC-CNN : we used the $\delta/2$ and $1-\delta/2$ quantiles of the marginals of the approximated posterior distributions.
- ABC-SMC : idem with weighted quantiles.
- ABC-RF : directly gives CI.
- ABCD-Conformal : conformal prediction using diagonal terms of the uncertainty matrix $\widehat{\mathbb{V}}(x)$.

Credible and confidence sets (multidimensional)

- Standard ABC, ABC-CNN and ABC-SMC : define a 95% credible set using the Highest Posterior Density Region, assuming the posterior to be a Multivariate Normal distribution.
- ABC-RF : not possible.
- ABCD-Conformal : conformal prediction using the uncertainty matrix $\widehat{\mathbb{V}}(x)$.

Presentation outline

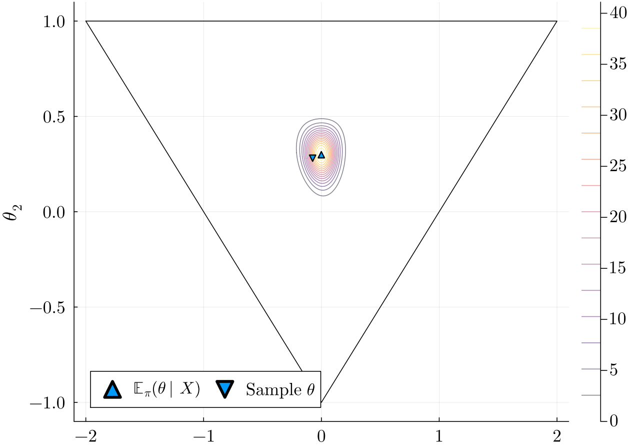
Moving Average 2 toy example (1/3)

Observation $X = (X_1, \ldots, X_{100})$, parameter $\theta = (\theta_1, \theta_2)$

$$X_j = Z_j + \theta_1 Z_{j-1} + \theta_2 Z_{j-2}, \qquad j = 1, \ldots, 100,$$

where $(Z_j)_{-2<j\leq 100}$ is an i.i.d. sequence of standard Gaussians $\mathcal{N}(0,1)$.

1. Summary Stats : autocovariance 1 & 2
2. Distance : $\mathbb{L}^2$
3. Tolerance level : 500 samples



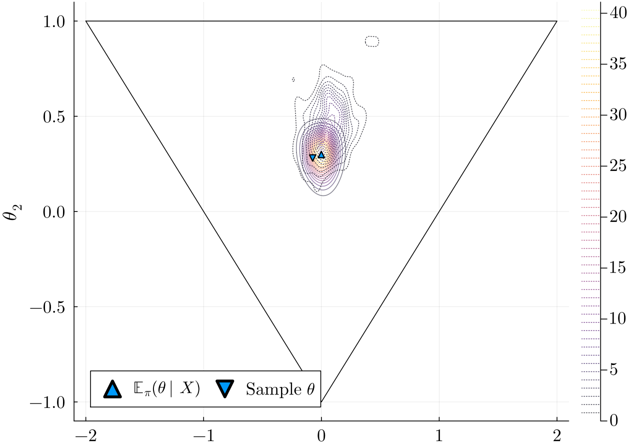
Moving Average 2 toy example (1/3)

Observation $X = (X_1, \ldots, X_{100})$, parameter $\theta = (\theta_1, \theta_2)$

$$X_j = Z_j + \theta_1 Z_{j-1} + \theta_2 Z_{j-2}, \qquad j = 1, \ldots, 100,$$

where $(Z_j)_{-2<j\leq 100}$ is an i.i.d. sequence of standard Gaussians $\mathcal{N}(0,1)$.

1. Summary Stats : autocovariance 1 & 2
2. Distance : $\mathbb{L}^2$
3. Tolerance level : 500 samples



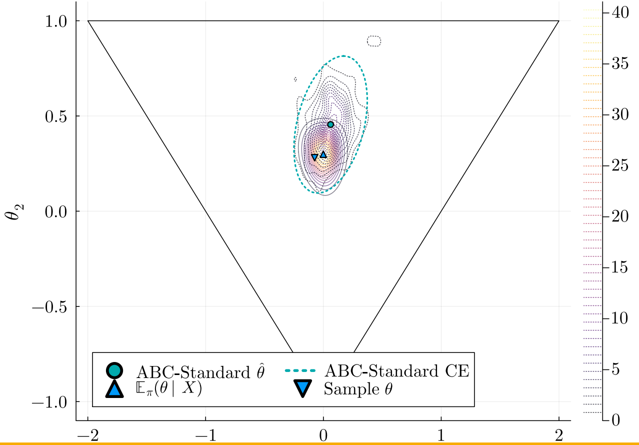
Moving Average 2 toy example (1/3)

Observation $X = (X_1, \ldots, X_{100})$, parameter $\theta = (\theta_1, \theta_2)$

$$X_j = Z_j + \theta_1 Z_{j-1} + \theta_2 Z_{j-2}, \qquad j = 1, \ldots, 100,$$

where $(Z_j)_{-2<j\leq 100}$ is an i.i.d. sequence of standard Gaussians $\mathcal{N}(0,1)$.

1. Summary Stats : autocovariance 1 & 2
2. Distance : $\mathbb{L}^2$
3. Tolerance level : 500 samples



Lotka-Volterra

1. **Summary statistics and distance** : for stand. ABC and ABC-SMC.
   No obvious summary statistic. The distance function between a sample from the test set and a sample from the training set is :

$$d\Big((x_{1,train}, x_{2,train}), (x_1, x_2)\Big) = \sum_{i=1}^{19} \Big(\big(x_{1,train}[i] - x_1[i]\big)^2 + \big(x_{2,train}[i] - x_2[i]\big)^2\Big)$$

2. **NN architecture** : for ABC-CNN and ABCD-conformal.
   Simple, without dropout for ABC-CNN, with for ABCD-conformal.

   Input 2 time series of lengths 19.
   Output θ of dimension three.

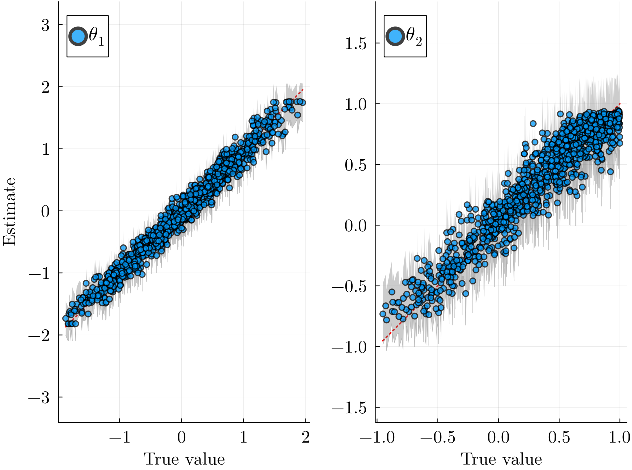
3. **Heuristic uncertainty** : for ABCD-conformal.
   "good" overall uncertainty + "bad" constant uncertainty.

Moving Average 2 toy example (2/3)

Observation $X = (X_1, \dots, X_{100})$, parameter $\theta = (\theta_1, \theta_2)$.

1. A simple NN architecture.
2. Without dropout for ABC-CNN, with for ABCD-conformal.

   Input Temporal series of length 100.
   Output $\theta = (\theta_1, \theta_2)$.



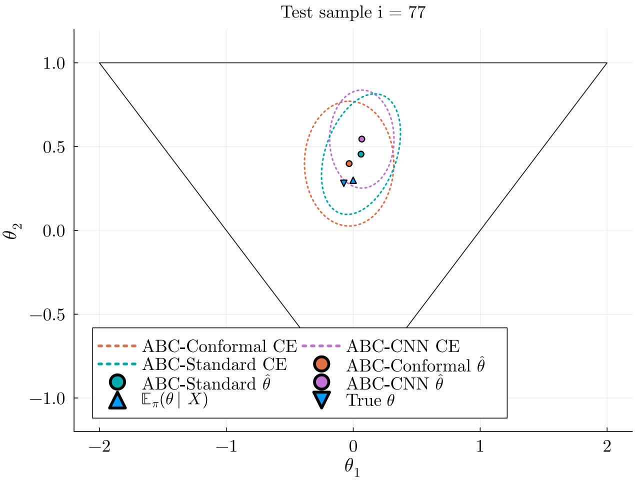
Moving Average 2 toy example (2/3)

Observation $X = (X_1, \ldots, X_{100})$, parameter $\theta = (\theta_1, \theta_2)$.

1. A simple NN architecture.
2. Without dropout for ABC-CNN, with for ABCD-conformal.

   Input Temporal series of length 100.

   Output $\theta = (\theta_1, \theta_2)$.



Lake ecosystem dynamics

1. **Summary statistics and distance :** for stand. ABC and ABC-SMC.
No obvious summary statistic. The distance function between a sample from the test set and a sample from the training set is :

$$\frac{1}{2}\Big(\mathrm{NMAE}(C_{\mathrm{P,test}}, C_{\mathrm{P,train}})^2 + \mathrm{NMAE}(C_{\mathrm{IP,test}}, C_{\mathrm{IP,train}})^2\Big)$$

2. **NN architecture :** for ABC-CNN and ABCD-conformal.
Simple, without dropout for ABC-CNN, with for ABCD-conformal.

   Input two time series of lengths 730.
   Output θ of dimension nine.

3. **Heuristic uncertainty :** for ABCD-conformal.
"good" overall uncertainty.

Moving Average 2 toy example (3/3)

	Standard ABC	ABC-SMC	ABC-CNN	ABC-Conformal
NMAE(θ_1)	0.174	0.176	0.177	**0.166**
$sd(\|\theta_1 - \widehat{\theta_1}\|)$	0.087	0.089	0.091	**0.083**
mean length $CI(\theta_1)$	0.544	0.626	**0.377**	0.560
median length $CI(\theta_1)$	0.535	0.609	**0.375**	0.559
coverage $CI(\theta_1)$	94.0%	**96.5%**	81.2%	95.7%
NMAE(θ_2)	0.273	0.272	0.264	**0.234**
$sd(\|\theta_2 - \widehat{\theta_2}\|)$	0.101	0.101	0.099	**0.089**
mean length $CI(\theta_2)$	0.615	0.667	**0.391**	0.583
median length $CI(\theta_2)$	0.596	0.653	**0.394**	0.585
coverage $CI(\theta_2)$	93.2%	95.8%	78.1%	**94.8%**
mean area $CE(\theta)$ (2D)	0.437	0.507	**0.187**	0.409
median area $CE(\theta)$ (2D)	0.412	0.479	**0.187**	0.411
coverage $CE(\theta)$ (2D)	**95.1%**	96.7%	75.6%	94.5%

- Similar results concerning NMAE and sd of the absolute error.
- Coverage is sharp for standard ABC, ABC-SMC and ABCD-conformal.
- Mean and median lengths of CI : standard ABC and ABCD-conf are the best.
- Confidence ellipses : ABCD-conformal is the best.

Remark for standard ABC : Good summary statistics, conditions for good asymptotic properties.

Presentation outline

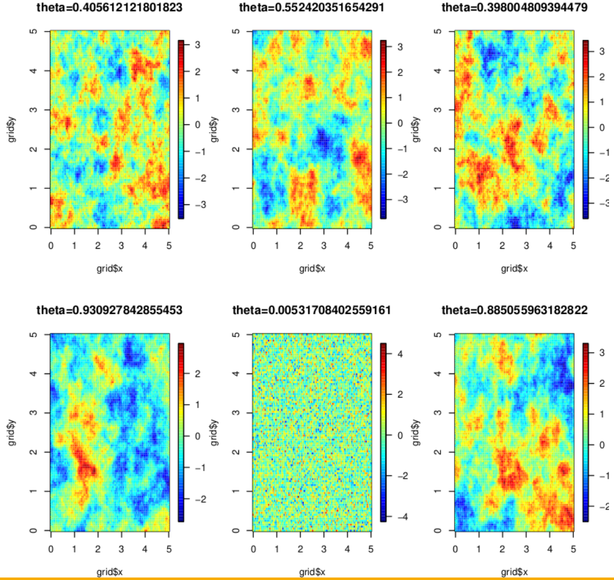
2D Gaussian fields (1/3)

10^4 stationary isotropic Gaussian random fields generated on a 100×100 grid :

- 7000 for training.
- 1000 for calibration.
- 1900 for validation.
- 100 for a test set.

Exponential covariance function :

$$C(z_i, z_j) = \theta e^{-||z_i - z_j||^2}$$



2D Gaussian fields (2/3)

1. **Summary statistics :** for stand. ABC and ABC-SMC.
   The Moran's I statistics from lag 1 to 5, and 15 values of the semi-variogram.

2. **Distance :** for stand. ABC and ABC-SMC.
   Distance to compare two GF is the sum of the quadratic distance between their Moran's correlograms and of the quadratic distance between their semi-variograms.

3. **NN architecture :** for ABC-CNN and ABCD-conformal. Simple, without dropout for ABC-CNN, with for ABCD-conformal.

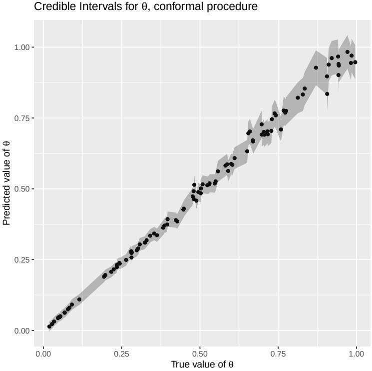
   Input 2D gaussian random fields (100×100 grids).
   Output θ of dimension one.

4. **Heuristic uncertainties :** for ABCD-conformal.
   Overall uncertainty + epistemic uncertainty.

2D Gaussian fields (3/3)

	Standard ABC	ABC-CNN	ABC-RF	ABCD-Conf overall	ABCD-Conf epistemic
$NMAE(\theta)$	0.0436	0.0223	**0.0151**	0.0300	0.0300
$sd(\lvert\theta - \widehat{\theta}\rvert)$	0.0229	0.0106	**0.0076**	0.0141	0.0141
mean length $CI(\theta)$	0.1313	**0.0307**	0.0405	0.0684	0.0711
median length $CI(\theta)$	0.1341	**0.0298**	0.0391	0.0637	0.0672
coverage $CI(\theta)$	**100%**	88%	98%	93%	94%

- ABC-RF outperforms the others methods.
- Standard ABC : largest NMAE and $sd(\lvert\theta - \widehat{\theta}\rvert)$.
- ABC-CNN : small NMAE and $sd(\lvert\theta - \widehat{\theta}\rvert)$, counterbalanced by a too small coverage.
- ABCD-conformal : satisfactory, with coverages of exactly 95%.
- Epistemic and overall uncertainties : similar results.



Presentation outline

NN with Dropout : formalisation (1/2)

- D the training set, containing inputs and outputs $\{(x_j, \theta_j), j = 1, \ldots, N\}$.
- θ the parameter of interest we want to predict, for a new data sample x.
- ω the vector of parameters of the network (weights and bias).

Training step : using D, the goal is to find parameters ω that are likely to have generated the outputs $(\theta_j, j = 1, \ldots, N)$, given the inputs $(x_j, j = 1, \ldots, N)$.

NN with Dropout : formalisation (1/2)

- D the training set, containing inputs and outputs $\{(x_j, \theta_j), j = 1, \ldots, N\}$.
- θ the parameter of interest we want to predict, for a new data sample x.
- ω the vector of parameters of the network (weights and bias).

Training step : using D, the goal is to find parameters ω that are likely to have generated the outputs $(\theta_j, j = 1, \ldots, N)$, given the inputs $(x_j, j = 1, \ldots, N)$.

Prediction : predict θ associated to a new x.
The posterior distribution of interest is then :

$$\pi(\theta \mid x, D) = \int \pi(\theta \mid x, \omega)\pi(\omega \mid D) d\omega. \quad (12)$$

NN with Dropout : formalisation (1/2)

- D the training set, containing inputs and outputs $\{(\mathrm{x}_j, \theta_j), j = 1, \ldots, N\}$.
- θ the parameter of interest we want to predict, for a new data sample x.
- ω the vector of parameters of the network (weights and bias).

Training step : using D, the goal is to find parameters ω that are likely to have generated the outputs $(\theta_j, j = 1, \ldots, N)$, given the inputs $(\mathrm{x}_j, j = 1, \ldots, N)$.

Prediction : predict θ associated to a new x.
The posterior distribution of interest is then :

$$\pi(\theta \mid \mathrm{x}, D) = \int \pi(\theta \mid \mathrm{x}, \omega)\pi(\omega \mid D) d\omega. \tag{12}$$

Approximation : density $\pi(\omega \mid D)$ approximated using a variational approach by $q(\omega)$.
The approximate posterior distribution of interest is then given by

$$q(\theta \mid \mathrm{x}) = \int \pi(\theta \mid \mathrm{x}, \omega) q(\omega) d\omega. \tag{13}$$

NN with Dropout : formalisation (2/2)

$$q(\theta \mid x) = \int \pi(\theta \mid x, \omega) q(\omega) d\omega. \tag{14}$$

The first two moments of $q(\theta \mid x)$ can be estimated empirically following Monte Carlo integration with K samples, using dropout.

- $\widehat{\omega_k}$ associated to the k^{th} network with dropped units.
- $f^\omega(x)$ the model's stochastic output for input x and parameters ω
- Assumption : $\theta \mid x, \omega \sim \mathcal{N}(f^\omega(x), \tau^{-1} I)$ and $\widehat{\omega_k} \sim q(\omega),\ t = 1, \ldots, K$.

NN with Dropout : formalisation (2/2)

$$q(\theta \mid x) = \int \pi(\theta \mid x, \omega) q(\omega) d\omega. \tag{14}$$

The first two moments of $q(\theta \mid x)$ can be estimated empirically following Monte Carlo integration with K samples, using dropout.

- $\widehat{\omega_k}$ associated to the k^{th} network with dropped units.
- $f^{\omega}(x)$ the model's stochastic output for input x and parameters ω
- Assumption : $\theta \mid x, \omega \sim \mathcal{N}(f^{\omega}(x), \tau^{-1} I)$ and $\widehat{\omega_k} \sim q(\omega)$, $t = 1, \ldots, K$.

Estimator for $\mathbb{E}_{q(\theta|x)}[\theta \mid x]$:

$$\widehat{\theta}(x) = \frac{1}{K} \sum_{k=1}^{K} f^{\widehat{\omega_k}}(x) \tag{15}$$

Estimator for $\mathbb{E}_{q(\theta|x)}[\theta^T \theta]$:

$$\widehat{\mathbb{E}}[\theta^T \theta \mid x] = \tau^{-1} I + \frac{1}{K} \sum_{k=1}^{K} f^{\widehat{\omega_k}}(x)^T f^{\widehat{\omega_k}}(x) \tag{16}$$

NN with Dropout : formalisation (2/2)

$$q(\theta \mid x) = \int \pi(\theta \mid x, \omega) q(\omega) d\omega. \tag{14}$$

The first two moments of $q(\theta \mid x)$ can be estimated empirically following Monte Carlo integration with K samples, using dropout.

- $\widehat{\omega_k}$ associated to the k^{th} network with dropped units.
- $f^{\omega}(x)$ the model's stochastic output for input x and parameters ω
- Assumption : $\theta \mid x, \omega \sim \mathcal{N}(f^{\omega}(x), \tau^{-1} I)$ and $\widehat{\omega_k} \sim q(\omega)$, $t = 1, \ldots, K$.

Estimator for $\mathbb{E}_{q(\theta|x)}[\theta \mid x]$: approximation of $\mathbb{E}_{\pi}[\theta \mid x]$

$$\widehat{\theta}(x) = \frac{1}{K} \sum_{k=1}^{K} f^{\widehat{\omega_k}}(x) \tag{15}$$

Estimator for $\mathbb{E}_{q(\theta|x)}[\theta^T \theta]$: approximation of $\mathbb{E}_{\pi}[\theta^T \theta \mid x]$

$$\widehat{\mathbb{E}}[\theta^T \theta \mid x] = \tau^{-1} I + \frac{1}{K} \sum_{k=1}^{K} f^{\widehat{\omega_k}}(x)^T f^{\widehat{\omega_k}}(x) \tag{16}$$

NN with Dropout : uncertainties

Estimator for an associated variance :

$$\widehat{\mathbb{V}}[\theta \mid \mathrm{x}] = \underbrace{\tau^{-1}\mathrm{I}}_{\widehat{\mathbb{V}}_a[\theta|\mathrm{x}]} + \underbrace{\frac{1}{K}\sum_{k=1}^{K} \mathrm{f}^{\widehat{\omega_k}}(\mathrm{x})^T \mathrm{f}^{\widehat{\omega_k}}(\mathrm{x}) - \widehat{\theta}(\mathrm{x})^T \widehat{\theta}(\mathrm{x}),}_{\widehat{\mathbb{V}}_e[\theta|\mathrm{x}]} \quad (17)$$

In practice : for each of the K Monte Carlo iterations, the weigths $\widehat{\omega_k}$ are different, and the CNN with input x gives as outputs $\mathrm{f}^{\widehat{\omega_k}}(\mathrm{x})$ and $\tau^{-1}_{\widehat{\omega_k}}$.

- τ^{-1} is estimated by the mean of the $\tau^{-1}_{\widehat{\omega_k}}$.
- $\widetilde{\mathbb{V}_e}$ estimated by the sample variance of the $\mathrm{f}^{\widehat{\omega_k}}(\mathrm{x})$.
- Use of Concrete Dropout Gal et al. (2017)

Aleatoric and epistemic uncertainties : their interpretations studied a lot by Gal (2016), Gal and Ghahramani (2016), Gal et al. (2017).

Epistemic, aleatoric, predictive uncertainties (Gal 2016) (1/2)

Aleatoric uncertainty :

- Captures noise inherent in the environment (ex. measurement error).
- Cannot be explained even if more data.

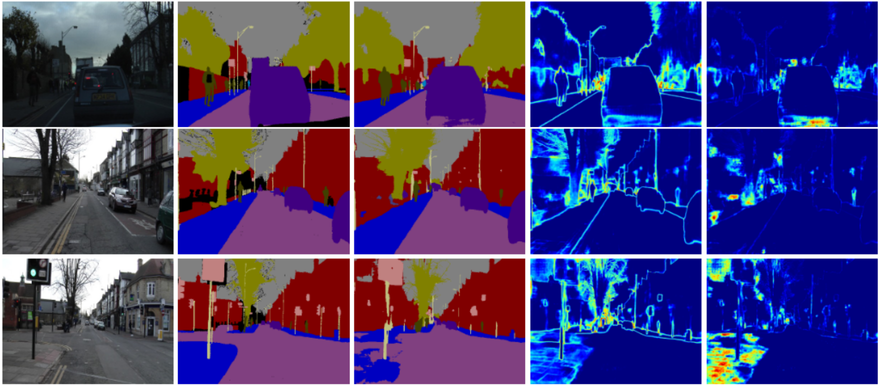
Epistemic uncertainty :

- Captures our ignorance about the model : parameters or structure.
- Can be reduced with more data.

Predictive uncertainty :

- Combines aleatoric and epistemic uncertainties.
- It is the model's confidence in its prediction taking into account noise it can explain away and noise it can not.

Epistemic, aleatoric, predictive uncertainties (Gal 2016) (2/2)



(a) Input Image (b) Ground Truth (c) Semantic Segmentation (d) Aleatoric Uncertainty (e) Epistemic Uncertainty

source : Kendall A. and Gal Y. (2017)

NN with dropout $\iff$ Bayesian Neural Network

optimal parameters found through the optimisation of a dropout neural network
=
optimal variational parameters in a probabilistic Bayesian neural network with the same structure.

A network trained with dropout $\iff$ a Bayesian Neural Network
It possesses all the properties of such a Bayesian Neural Network.

For this equivalence to be true, only one condition should be verified in the variational inference approach, which is about the Kullback-Leibler divergence between the prior distribution of the parameters and an approximating distribution for these parameters.

Gal (2016), Gal and Ghahramani (2016).

Presentation outline

ABCD-conformal algorithm (1/2)

Input A Bayesian parametric model $\{f(\cdot \mid \theta), \pi\}$, a data sample x, integers N_{train}, N_{cal} and K representing sizes of training and calibration sets, and the number of stochastic passes for the dropout. δ between 0 and 1 to obtain $(1-\delta)$ confidence sets.
Output Approximation of the posterior expected value $\mathbb{E}_\pi[\theta \mid \mathrm{x}]$ and a confidence interval for θ.

a) Generation of a reference table (training) and a calibration dataset :
For the reference table for instance :
for $j \leftarrow 1$ **to** N_{train} **do**
  Draw $\theta_j \sim \pi$
  Draw synthetic sample $\mathrm{x}_j = (x_{1,j}, \ldots, x_{d,j})^\top$ from the model $f(\cdot \mid \theta_j)$
end
b) Train a NN with concrete dropout on the reference table :
the training pairs are the $\left\{(\mathrm{x}_j, \theta_j), j = 1, \ldots, N_{\mathrm{train}}\right\}$ and the loss is the heteroscedastic loss.
A validation set can be generated and used to choose the architecture of the NN.

ABCD-conformal algorithm (2/2)

c) Monte Carlo Dropout prediction on the calibration set :

for $j \leftarrow 1$ **to** N_{cal} **do**
  for $k \leftarrow 1$ **to** K **do**
    x_j is given as input to the trained network with Dropout to obtain outputs $\mathrm{f}^{\widehat{\omega_k}}(\mathrm{x}_j)$ and $\tau^{-1}_{\widehat{\omega_k}}$
  end
  Obtain $\widehat{\theta}_j$ and τ^{-1} by averaging the $\mathrm{f}^{\widehat{\omega_k}}(\mathrm{x}_j)$ outputs and the $\tau^{-1}_{\widehat{\omega_k}}$ and an associated uncertainty $\widehat{\mathbb{V}}(\mathrm{x}_j)$ that can be $\widehat{\mathbb{V}}[\theta_j \mid \mathrm{x}_j]$ (see eq. (17)).
end

d) Computation of the conformal quantile on the calibration set :

for $j \leftarrow 1$ **to** N_{cal} **do**
  Compute the calibration score $s_j = \sqrt{(\theta_j - \widehat{\theta}_j)^t \widehat{\mathbb{V}}(\mathrm{x_j})^{-1}(\theta_j - \widehat{\theta}_j)}$.
end

Conformal quantile $\widehat{q}$: the $\frac{\lceil (N_{\mathbf{cal}}+1)(1-\delta) \rceil}{N_{\mathbf{cal}}}$ quantile of the calibration scores $s_1, \ldots, s_{N_{\mathbf{cal}}}$.

e) For the new data sample x**, approx. of** $\mathbb{E}_\pi[\theta \mid \mathrm{x}]$ **and confidence set for** θ :

for $k \leftarrow 1$ **to** K **do**
  x is given as input to the trained network, to obtain an output $\mathrm{f}^{\widehat{\omega_k}}(\mathrm{x})$
end

Obtain $\widehat{\theta}(\mathrm{x})$ an approx. of $\mathbb{E}_\pi[\theta \mid \mathrm{x}]$ by averaging these outputs + an associated uncertainty $\widehat{\mathbb{V}}(\mathrm{x})$
The confidence set for θ is an ellipsoid which center is $\widehat{\theta}(\mathrm{x})$ and covariance matrix is $\widehat{\mathbb{V}}(\mathrm{x})^{-1}/\widehat{q}^2$.

Presentation outline

Comparison of the five methods

Point of comparison	Standard ABC	ABC-SMC	ABC-CNN	ABC-RF	ABCD-Conf
Approx. of the whole posterior or of transforms of interest	whole posterior	whole posterior	whole posterior	transforms of interest	transforms of interest
No need of relevant summary stat	✗	✗	✓	✗	✓
No need of a distance	✗	✗	✓	✓	✓
No need of a tolerance threshold	✗	✗	✗	✓	✓
Deal with multidimensional parameters	✓	✓	✓	✗	✓
No need to choose network or random forest architecture	✓	✓	✗	✗	✗
Datasets needed	1 : training	Simulations made sequentially	2 : training, validation	1 : training	3 : training, validation, calibration
Justification of the method, guarantees	asymptotic under conditions	asymptotic under conditions	conditions too difficult to check	conditions too difficult to check	non asymptotic, no condition
Adapted for different types of data and high-dim data	difficult ✗	difficult ✗	✓	difficult ✗	✓
Computing time	difficult to compare, it depends on examples.				