ABCRanger

A fast and scalable random forest library for ABC model choice and parameter estimation

F.-D. Collin [2] A. Estoup [1] J.-M. Marin [2] L. Raynal [2]

[1]CBGP, INRA, CIRAD, IRD, Montpellier SupAgro, Univ. Montpellier

[2]Université de Montpellier, CNRS, IMAG UMR 5149

January 13, 2020

Outline

Approximate Bayesian Computation

It is defined by :

- *Bayesian Inference* context
- *Likelihood-free inference* method

Bayesian Inference

"If I believe in some model and I have some new data, what is the probability of this model, knowing this new data?". Baye's Theorem gives the answer :

$$P(\Theta|Y) = \frac{P(Y|\Theta) * P(\Theta)}{P(Y)}$$

Where :

Y	the *data*, observations, evidence and so on.
Θ	the *model* (hypothesis) we want to run
$P(\Theta)$	the *prior probability*
$P(\Theta\|Y)$	the *posterior probability*
$P(Y\|\Theta)$	the *likelihood*
$P(Y)$	the *marginal* likehood

Likelihood free

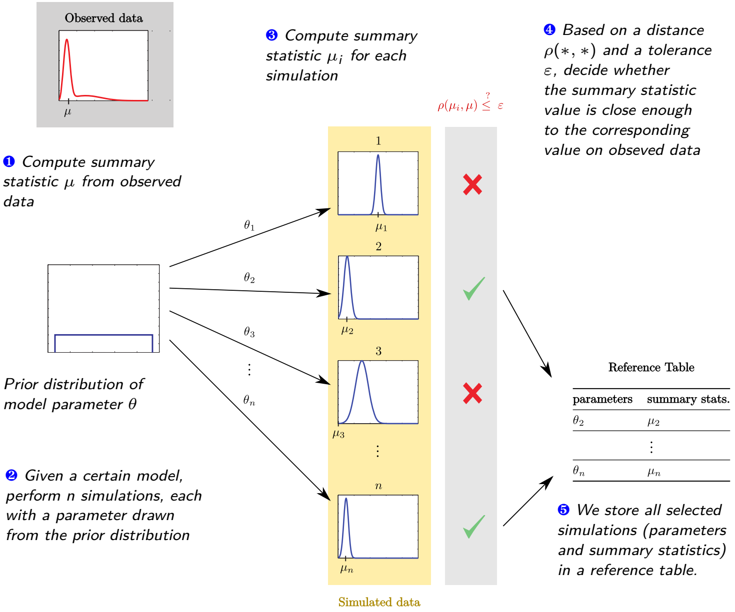
- $P(\Theta)$ is the easy part, $P(Y)$ should be too (and sometimes we can bypass it).
- Computing the *likelihood* $P(Y|\Theta)$ is the name of the game.

Often we can't have a function for the likelihood, or it is intractable, too complex and so on.
➧ Enter the *Likelihood-free* Kingdom and *ABC (Approximate Bayesian Computation)*.

ABC in short

Given an observed data, the basic idea of ABC is to approximate the likelihood of a parametrized model with selected simulations, by comparing the observed data and simulated ones via computed *summary statistics*.
The table of summary statistics for simulated data is called *the reference table*.

ABC schema


Observed data
μ
❶ Compute summary statistic μ from observed data
Prior distribution of model parameter θ
❷ Given a certain model, perform n simulations, each with a parameter drawn from the prior distribution
θ_1
θ_2
θ_3
$\vdots$
θ_n
❸ Compute summary statistic μ_i for each simulation
1
μ_1
2
μ_2
3
μ_3
$\vdots$
n
μ_n
Simulated data
$\rho(\mu_i, \mu) \overset{?}{\leq} \varepsilon$
❹ Based on a distance $\rho(*, *)$ and a tolerance ε, decide whether the summary statistic value is close enough to the corresponding value on obseved data
Reference Table
parameters summary stats.
θ_2 μ_2
$\vdots$
θ_n μ_n
❺ We store all selected simulations (parameters and summary statistics) in a reference table.

AbcRf/AbcRanger, presentation

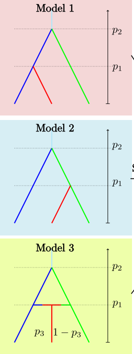
AbcRanger is a software for ABC posterior methodologies. It gets the output from an ABC run and provides :

Model choice: Simulate data for several models and *choose the best model to fit our data*

Parameter estimation: Simulate data for one model and *infer one or several parameters for this model given the observed data*

ABC workflow with AbcRanger

❶ *Compute simulations with several models, and the reference table with model-indexed lines using a simulator (DIYAC, PyABC etc.)*



Simulations

❷ *Apply Model Choice Methodology with AbcRanger*

Reference Table with multiple models

Model	p_1	p_2	p_3	s_1	s_2	s_3	s_4	s_5
2	38	2	0.783	0.559	0.409	0.591	0.393	0.601
2	40	5	0.141	0.294	0.386	0.469	0.515	0.542
1	35	1	0.445	0.252	0.481	0.265	0.532	0.579
3	38	2	0.706	0.250	0.308	0.359	0.372	0.740
2	37	4	0.267	0.287	0.363	0.459	0.434	0.690
				⋮				
1	38	1	0.331	0.507	0.305	0.303	0.525	0.477

Parameters — Summary Statistics

Model Choice : AbcRanger

Scenario 2 Chosen

Reference Table for parameter estimation

p_1	s_1	s_2	s_3	s_4	s_5
38	0.559	0.409	0.591	0.393	0.601
40	0.294	0.386	0.469	0.515	0.542
			⋮		
37	0.287	0.363	0.459	0.434	0.690

❸ *Apply Parameter Estimation Methodology with AbcRanger*

Parameter Estimation : AbcRanger

$p_1 \approx 0.329$

Parameter estimation

Random Forest setup :

- Choose a parameter t of the model
- Train a regression RF on reftable with the t as target
- Evaluate local/posteriors on observed data

➧ Estimator for posterior PDF for the parameter (discretized but obtainable via kde)

Model Choice

Two staged RF setup:

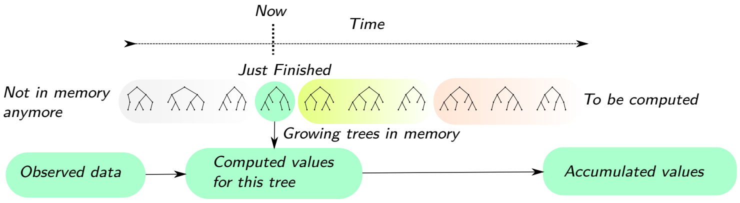
1. Classification :
   - Train a classification RF with the models (classes) as target
   - Eval the RF on observed data to get votes and chosen model
2. Regression :
   - Using the previous RF, get the classified/misclassified on the training set as 0,1 and train a new regression RF with this as target
   - Evalute the obtained RF on the observed data to get the posterior probability of the chosen model

AbcRanger details

- Written in C++, `http://github.com/diyabc/abcranger`, code and binaries (mac/windows/linux)
- Python frontend in the final stage (demos running)
- R frontend WIP
- optimized for large, high dimensional reference table without (too much) memory limits: more than 10^{e5} columns and 10^{e6} rows.

Under the hood, a new RF implementation

Since ABC procedures only use trained Random Forests on a known set of observations, we have altered the random forest training computation by using only a subset of in-memory trees at a time and accumulating the required outcomes (predictions and statistics). Memory footprint is vastly improved and there is no performance cost.



Demo time

- https://github.com/diyabc/abcranger/blob/master/testpy/Parameter%20Estimation%20Demo.ipynb
- https://github.com/diyabc/abcranger/blob/master/testpy/Model%20Choice%20Demo.ipynb

Conclusion

1. Thoughts:
   - nice integration of ML techniques in a model-based approach...
   - ... although the objective there is not better "predictions" or "score" as in ML but easy and accurate posteriors
2. Perspectives:
   - deeper integration in ABC pipeline like the Elfi python package
   - On the RF side, ongoing project *LeafLitter* intends to pursue that line even further: for a growing tree, only encountered leaves are stored. Thus, the memory footprint of the trees becomes negligible, and their growing could finally be parallelized at full scale.

References

[1] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[2] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984.

[3] Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012.

[4] Pierre Pudlo, Jean-Michel Marin, Arnaud Estoup, Jean-Marie Cornuet, Mathieu Gautier, and Christian P Robert. Reliable abc model choice via random forests. *Bioinformatics*, 32(6):859–866, 2015.

[5] Louis Raynal, Jean-Michel Marin, Pierre Pudlo, Mathieu Ribatet, Christian P Robert, and Arnaud Estoup. ABC random forests for Bayesian parameter inference. *Bioinformatics*, 35(10):1720–1728, 10 2018.

[6] Marvin N Wright and Andreas Ziegler. Ranger: a fast implementation of random forests for high dimensional data in c++ and r. *arXiv preprint arXiv:1508.04409*, 2015.