ABC: a review and some recent developments

Dennis Prangle

November 2019

Newcastle University, UK

Traditional modelling and inference

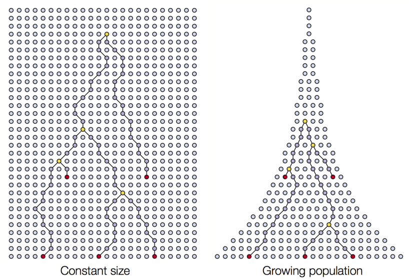
- Observe some data y_{obs}
- Define **model** with density $p(y|\theta)$ and parameters θ
- **Likelihood function** is $L(\theta) = p(y_{\text{obs}}|\theta)$
- Use $L(\theta)$ to learn parameters
  e.g. maximum likelihood or Bayesian inference by MCMC
- Inference output + model allows:
  - **Prediction** of future behaviour
  - **Explanation** of phenomena
  - **Control** by understanding effect of interventions
  - ...

Generative models

- Often hard to directly define probability models for complex systems!
- Easier to describe how system evolves
- Can then simulate data given parameters easily
- i.e. **generative** model
- Likelihood calculations for generative models often very hard!
- Goal: **likelihood-free inference** methods
- Inference without making use of likelihood function
- One such method is ABC: *approximate Bayesian computation*

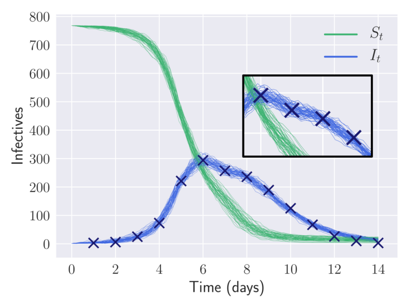
Applications - population genetics

- Population genetics was first ABC application (late 90s)
- Data are genetic sequences from a population
- Parameters include mutation rates etc
- And also demographic history of population (migration, growth etc)
- Easy to simulate data from parameters
- Likelihood not available



Applications - infectious disease epidemiology

- Can model infectious disease by population counts of susceptible, infectious and recovered
- Parameters control rates of change between populations
- Model can be ODE, SDE, jump process etc
- All easy to simulate from (at least approximately)



Applications - infectious disease epidemiology

- Can add extra structure e.g. exposed stage, immigration
- Or move from population level model to individual based model
- Many other applications for ODEs, SDEs, jump processes, IBMs!

More applications

THE ASTROPHYSICAL JOURNAL, 764:116 (15pp), 2013 February 20 doi:10.1088/0004-637X/764/2/116

LIKELIHOOD-FREE COSMOLOGICAL INFERENCE WITH TYPE Ia SUPERNOVAE: APPROXIMATE BAYESIAN COMPUTATION FOR A COMPLETE TREATMENT OF UNCERTAINTY

ANJA WEYANT[1], CHAD SCHAFER[2], AND W. MICHAEL WOOD-VASEY[1]
[1] Pittsburgh Particle Physics, Astrophysics, and Cosmology Center (PITT PACC), Physics and Astronomy Department, University of Pittsburgh, Pittsburgh, PA 15260, USA; anw19@pitt.edu
[2] Department of Statistics, Carnegie Mellon University, Pittsburgh, PA 15213, USA
Received 2012 June 12; accepted 2012 December 10; published 2013 January 30

ABSTRACT

Cosmological inference becomes increasingly difficult when complex data-generating processes cannot be modeled by simple probability distributions. With the ever-increasing size of data sets in cosmology, there is an increasing

More applications

THE ASTROPHYSICAL JOURNAL, 764:116 (15pp), 2013 February 20 doi:10.1088/0004-637X/764/2/116

LIKELIHOOD-FREE COSMOLOGICAL INFERENCE WITH TYPE Ia SUPERNOVAE: APPROXIMATE BAYESIAN COMPUTATION FOR A COMPLETE TREATMENT OF UNCERTAINTY

ANJA WEYANT[1], CHAD SCHAFER[2], AND W. MICHAEL WOOD-VASEY[1]
[1] Pittsburgh Particle Physics, Astrophysics, and Cosmology Center (PITT PACC), Physics and Astronomy Department, University of Pittsburgh, Pittsburgh, PA 15260, USA; anw19@pitt.edu
[2] Department of Statistics, Carnegie Mellon University, Pittsburgh, PA 15213, USA
Received 2012 June 12; accepted 2012 December 10; published 2013 January 30

ABSTRACT

Cosmological inference becomes increasingly difficult when complex data-generating processes cannot be modeled by simple probability distributions. With the ever-increasing size of data sets in cosmology, there is an increasing

Distance-learning For Approximate Bayesian Computation To Model a Volcanic Eruption

Lorenzo Pacchiardi[1], Pierre Kunzli[2]*, Marcel Schoengens[3], Bastien Chopard[2], Ritabrata Dutta[4]†

More applications

Approximate Bayesian computation for estimating number concentrations of monodisperse nanoparticles in suspension by optical microscopy

Magnus Röding*
SP Food and Bioscience, Soft Materials Science, Göteborg, Sweden and School of Energy and Resources, UCL Australia, University College London, Adelaide, Australia

Elisa Zagato, Katrien Remaut, and Kevin Braeckmans
Laboratory of General Biochemistry and Physical Pharmacy, Ghent University, Ghent, Belgium and Center for Nano- and Biophotonics, Ghent University, Ghent, Belgium

(Dated: June 3, 2016)

We present an approximate Bayesian computation (ABC) scheme for estimating number concentrations of monodisperse diffusing nanoparticles in suspension by optical particle tracking mi-

More applications

Approximate Bayesian computation for estimating number concentrations of monodisperse nanoparticles in suspension by optical microscopy

Magnus Röding*
SP Food and Bioscience, Soft Materials Science, Göteborg, Sweden and School of Energy and Resources, UCL Australia, University College London, Adelaide, Australia

Elisa Zagato, Katrien Remaut, and Kevin Braeckmans
Laboratory of General Biochemistry and Physical Pharmacy, Ghent University, Ghent, Belgium and Center for Nano- and Biophotonics, Ghent University, Ghent, Belgium
(Dated: June 3, 2016)

We present an approximate Bayesian computation (ABC) scheme for estimating number concentrations of monodisperse diffusing nanoparticles in suspension by optical particle tracking mi-

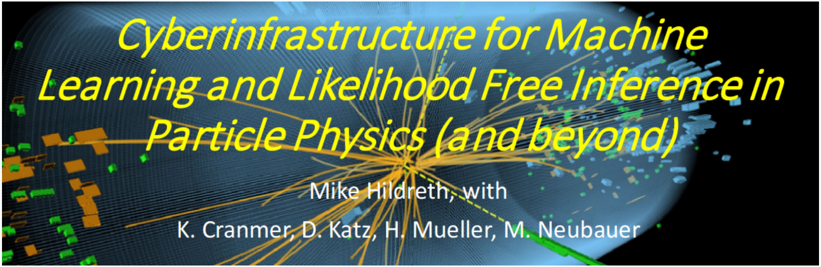
Bayesian semi-individual based model with approximate Bayesian computation for parameters calibration: Modelling Crown-of-Thorns populations on the Great Barrier Reef

C.C.-M. Chen [a, b], C.C. Drovandi [b, c], J.M. Keith [d], K. Anthony [a], M.J. Caley [b, c], K.L. Mengersen [b, c]

⊞ Show more

https://doi.org/10.1016/j.ecolmodel.2017.09.006

More applications



Queueing example

- Example used throughout talk
- A single queue
- Gaps between arrivals at back are $Exp(\theta_1)$
- Service times on reaching front are $U(\theta_2, \theta_3)$
- Observations are times between queue departures
- Easy to simulate data
- But probability calculations for observations challenging!
- Exact MCMC is possible (Shestopaloff and Neal 2014)
- More complex queueing network models useful in applications
    - Passenger flow in airport (Ebert et al 2018)
    - Computing jobs in data centre (Sutton and Jordan 2011)

Overview of talk

- Standard ABC approaches (approximate Bayesian computation)
- DE methods (density estimation)

Overview of talk

- Standard ABC approaches (approximate Bayesian computation)
- DE methods (density estimation) – including my recent work!

Approximate Bayesian Computation

ABC idea

- General idea:
  - Simulate data y from various parameter values θ
  - Consider closest matches of y to y_{obs}
  - Use corresponding parameters for inference
- Can be implemented in many different ways
- Many approaches suggested in various fields over last 50 years
- ABC puts this idea into a **Bayesian** framework

Bayesian inference

- Specify prior density $\pi(\theta)$
  - Beliefs about parameters before data observed
- Ideal inference goal is **posterior density** $p(\theta|y_{\text{obs}})$
  - Beliefs updated to take data into account
- Posterior depends on prior and likelihood through Bayes theorem:

$$p(\theta|y_{\text{obs}}) \propto \pi(\theta) L(\theta)$$

$$\text{i.e. posterior} \propto \text{prior} \times \text{likelihood}$$

- Posterior is parameters conditioned on observing y_{obs}

Basic ABC

Input:

- Observed data y_{obs}
- Threshold $\epsilon \geq 0$
- Distance function $d(y, y')$

Loop over $i = 1, 2, \ldots, N$:

1. Sample θ_i from prior $\pi(\theta)$
2. Simulate y from model $p(y|\theta_i)$
3. If $d(y, y_{\text{obs}}) \leq \epsilon$ accept θ_i

Output: accepted θ_i values

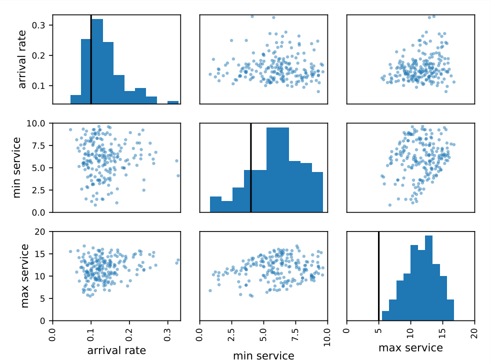
Popular variant – do N simulations then accept M with smallest distance

ABC target distribution

- ABC output is sample from **approximate** posterior
- Ideal Bayesian inference: sample from prior + model
  and condition on exact match to y_{obs}
- ABC: sample from prior + model
  and condition on approximate match to y_{obs}
- Reducing ϵ gives closer matches
  - more accurate output
  - but acceptances rarer
- ϵ controls trade-off between accuracy and cost

Example - queueing model

- Data: 20 inter-departure times
- Uniform priors for:
  - Service rate θ_1 on $[0, 1/3]$
  - Min service time θ_2 on $[0, 10]$
  - Service time range $\theta_3 - \theta_2$ on $[0, 10]$
- 1 million ABC simulations, best 200 accepted



ABC distances

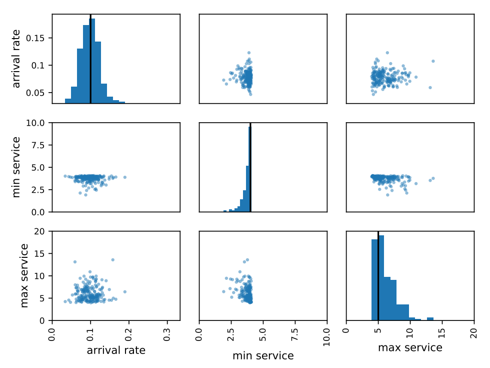
- Choice of distance function crucial
- Simplest setting: y is fixed-length vector
- Could take d as Euclidean distance
- Usually sensible to use weighted version
- i.e. normalise y components so on similar scales
- Many more sophisticated distances proposed:
    - Application specific
    - Based on theory (Wasserstein, MMD)
    - Using machine learning (kernel methods, deep distance learning)

Summary statistics

- ABC typically poor with high dimensional data
- So lots of data dimensions must match
- V unlikely to get good matches in all!
- So ABC usually reduces data to low dim summary statistics
- Then computes distance e.g. Euclidean distance
- Theory: rate of convergence as $\epsilon \to 0$ worsens quickly with high $\text{dim}(y)$
    - See e.g. "The rate of convergence for approximate Bayesian computation" (Barber et al 2015)
- **Dimension reduction** using summary statistics helps

Example - queueing model

- Same data (dimension 20) and simulations as before
- 5 summary statistics: min, lower quartile, median, upper quartile, max (all normalised)



Other ABC algorithms

- Always sampling θ from prior is inefficient
- Posterior ususally more concentrated than prior
- Lots of prior θs samples will be poor
- Simulating data from these **wastes time**!
- More efficient to propose promising θs based on previous results
- Can use MCMC (Markov chain Monte Carlo)
- Or **SMC** (sequential Monte Carlo)

Other ABC algorithms

- MCMC
  - Propose next θ near to previous accepted one
- SMC
  - Propose a group of θs close to previous accepted sample
- In ABC, SMC is more popular
- Can reduce ϵ adaptively during algorithm
- Several versions of ABC-SMC algorithm
- ϵ tuning harder for MCMC (but see Vihola and Franks 2019)

More ABC contributions

- Lots more in the literature!
- Diagnostics
- Asymptotic theory
- ABC for model choice
- Choosing summary statistics
- Interpreting approximate results
- Lots of variants on ABC algorithms

ABC software

- R
  - abc
  - easyABC
- python
  - ABCPy
  - PyAbc
  - ELFI
- julia
  - GpABC
- (not an exhaustive list!)
- Also, basic ABC v easy to code yourself

ABC references

- Handbook of ABC
- Review papers
  - "Approximate Bayesian Computation for infectious disease modelling" - Minter and Retkute (2019)
  - "Fundamentals and Recent Developments in Approximate Bayesian Computation" - Lintusaari et al (2017)
  - Wikipedia article (commissioned by PLOS comp bio - 2013)
  - "Approximate Bayesian computational methods" - Marin et al (2012)
  - And many more

Pros and cons

- Pros
  - Basic idea very simple to understand
  - Easy to implement
  - Widely applicable
  - Easy to get rough parameter estimates
- Cons
  - Difficult tuning choices – distance / summary statistics
  - Approximate results – hard to quantify error
  - Doesn't scale up well to expensive simulators
  - Doesn't scale up easily to high dimensional data

Density estimation approaches

Density estimation for likelihood-free inference

- Lots of powerful density estimation methods developed in recent decade(s)
    - Mixture models
    - Copulas
    - Gaussian processes
    - Invertible neural networks (aka normalising flows)
- Can be applied to likelihood-free inference
- Very broad idea:
    1. Simulate parameters + data
    2. Estimate some relevant density
    3. Use to approximate posterior
- n.b. ABC uses nearest neighbours density estimation!

Conditional density estimation

- General idea:
  - Simulate many (θ, y) pairs from prior and model
  - Estimate joint density $\hat{\pi}(\theta, y)$
  - Condition on y_{obs} and output $\hat{\pi}(\theta | y_{\text{obs}})$
- Variation:
  - Directly estimate $\hat{\pi}(\theta | y)$
  - Output $\hat{\pi}(\theta | y_{\text{obs}})$

Examples

- Mixture of Gaussians (Bonassi, You and West 2011)
- Random forest (Pudlo et al 2016)
- Extreme gradient boosted tree (Lamperti et al 2017)
- Mixture density neural networks (Papamakarios and Murray 2016, Lueckmann et al 2017)

Likelihood estimation

- General idea:
  - Simulate many (θ, y) pairs from prior and model
  - Estimate data density $\hat{p}(y|\theta)$
  - Likelihood estimate is $\hat{L}(\theta) = \hat{p}(y_{\text{obs}}|\theta)$
- Examples
  - Kernel density estimation (Diggle and Gratton 1984)
  - Gaussian processes (Wilkinson 2014, Gutmann and Corander 2016)
  - Normalising flows (Papamakarios, Sterrat and Murray 2018)

Other related methods

- ABC regression correction (Beaumont et al 2002)
  - Run ABC
  - Fit regression to output (predict θ given y)
  - Use to correct ABC sample
- Likelihood ratio estimation (Cranmer, Pavez and Louppe 2015)
  - Fit a neural network to estimate likelihood ratio given θ_1, θ_2
  - (Equivalent to a classification problem)
  - Use in inference e.g. MCMC

Review papers

- "Learning in implicit generative models" (Mohamed and Lakshminarayanan, 2016)
- "A review of approximate Bayesian computation methods via density estimation: Inference for simulator models" (Grazian and Fan 2019)
- "The frontier of simulation-based inference" (Cranmer Brehmer and Louppe 2019)

Pros and cons

- Pros
  - Scales up to bigger problems better than ABC
  - Most methods don't need ϵ (less tuning!)
  - Reduces need for summary statistics
- Cons
  - Less transparent than ABC
  - Unclear what best method is!
  - Often still needs some dimension reduction by summary statistics
  - Still hard to quantify approx error
  - Little general purpose software yet

Distilled importance sampling

Motivation

- **ABC** relies on simulator producing close matches to data
- Rare even under best parameter values!
- **Density estimation** methods must estimate densities for wide range of ys
- Hard modelling task!
- We learn to control simulator to always output data close to y_{obs}

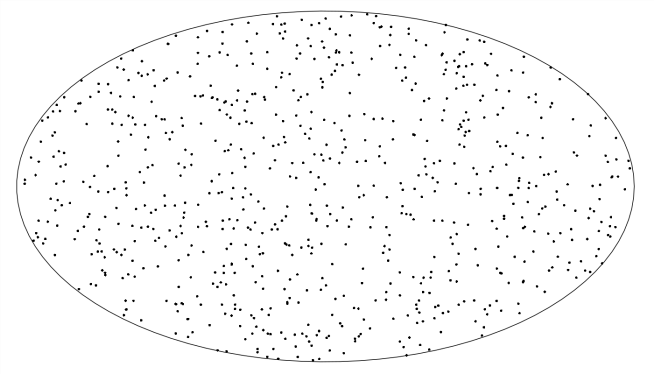
Framework

- Parameters are θ
- Let u be all random draws used in simulator
- Then simulator is function $y(\theta, u)$
- We try to learn density $p(\theta, u | y_{\text{obs}})$
- (Tricky as often a near-singular density)

Main idea

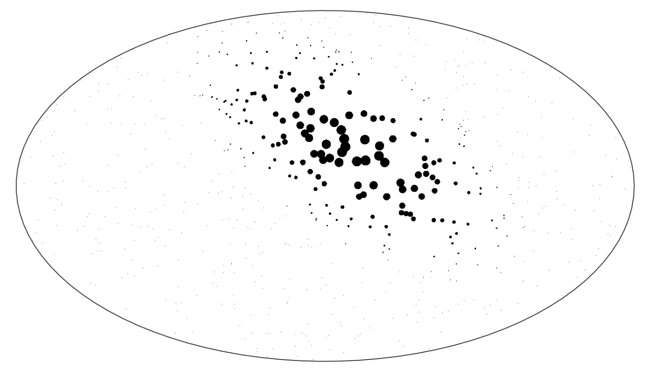
- **Iteration 1**
- Sample θ, u from prior distributions
- Weight sample depending closeness of $y(\theta, u)$ to y_{obs}
  Give moderate weights even to poor matches
- Use weighted sample to train density estimate
- **Iteration 2**
- Sample θ, u from density estimate
- Weight sample depending closeness of $y(\theta, u)$ to y_{obs}
  Require slightly close matches to get moderate weights
- Use weighted sample to train density estimate
- ...

Sample from approximate density

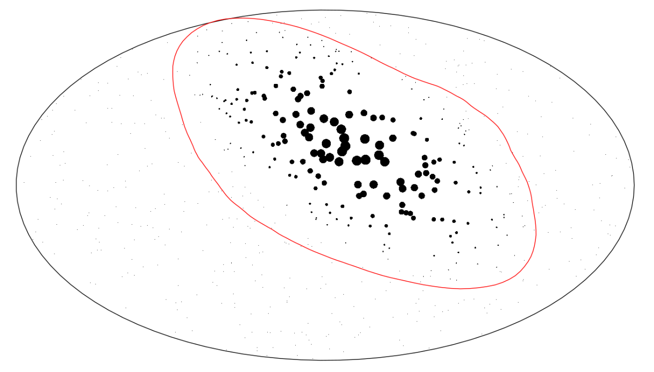


Illustration

Weight based on closeness

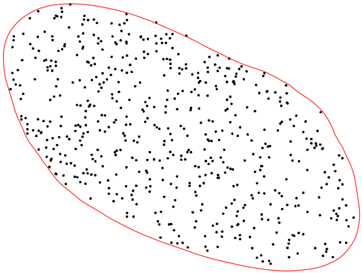


Apply density estimation

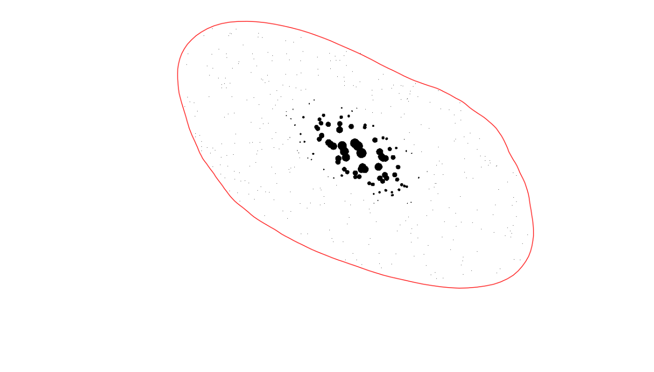


Illustration

Sample from new approximate density



Weight based on closeness



Comments

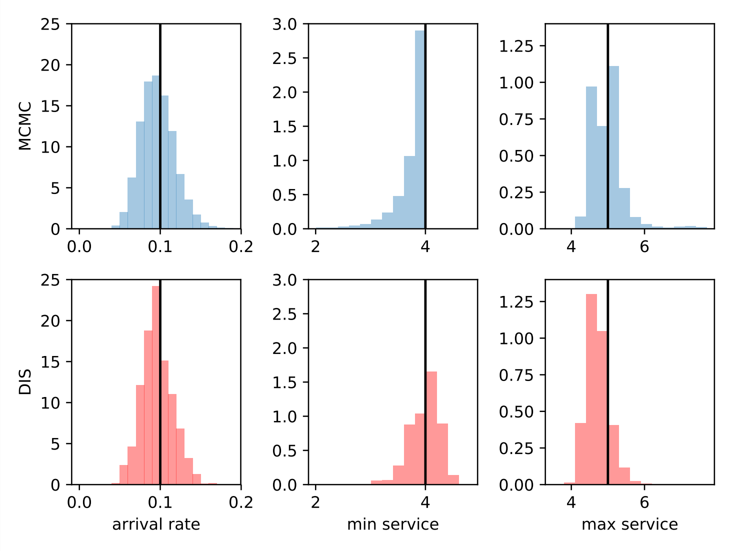
- Each iteration uses **importance sampling**
- Then uses output in density estimation
- "Distilled imporance sampling" (preprint on arxiv)
- Final results still approximate posterior
- Related to:
  - Adaptive importance sampling
  - Sequential importance sampling
  - Cross-entropy method
- We use normalising flows for density estimation
- Algorithm reduces to stochastic gradient optimisation
- Similar to variational inference

Queue example

- Need two u variables for individual in queue
- Seed for arrival time
- Seed for departure time
- Overall: 40 u variables and 3 θ variables

MG1 queue results

- Inference time: 3 hours on desktop PC, final $\epsilon = 0.283$



Conclusion

Summary

- **Generative models** allow data simulation but not easy likelihod calculation
- **Likelihood-free inference** does approximate inference just using simulations
- **ABC** simulates data under many parameter values looking for good matches
- Alternative: use simulations in modern **density estimation** methods
- My recent work improves efficiency by learning to **control simulator**