



# pyABC: scalable likelihood-free inference

Yannik Schälte<sup>1,2</sup>, Elba Raimúndez-Álvarez<sup>1,2</sup>, Emmanuel Klinger<sup>1,2,3</sup>, Jan Hasenauer<sup>1,2</sup>

<sup>1</sup>Helmholtz Center Munich, Institute of Computational Biology and <sup>2</sup>Technical University Munich, Department of Mathematics and <sup>3</sup>Max Planck Institute for Brain Research Frankfurt, Department of Connectomics

## Why should I care?

### Goal

estimate model parameters by analyzing the posterior distribution of parameters  $\theta$  given observed data  $y_{obs}$ :

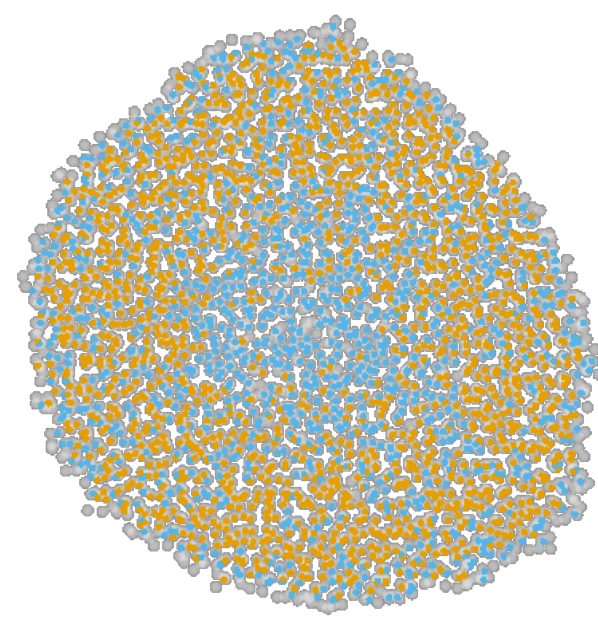
$$\text{posterior } \pi(\theta|y_{obs}) = \frac{\text{likelihood } \pi(y_{obs}|\theta) \pi(\theta) \text{ prior}}{\text{evidence } \pi(y_{obs})}$$

### Problem

evaluating the **likelihood** infeasible, but **possible to simulate data** from the model

### Where in Systems Biology?

- commonly the case for **complex stochastic models**, where evaluating the likelihood means integrating over all possible random paths
- low copy numbers of e.g. molecules or cells
- a particular application is in the emerging field of **multi-scale modeling**

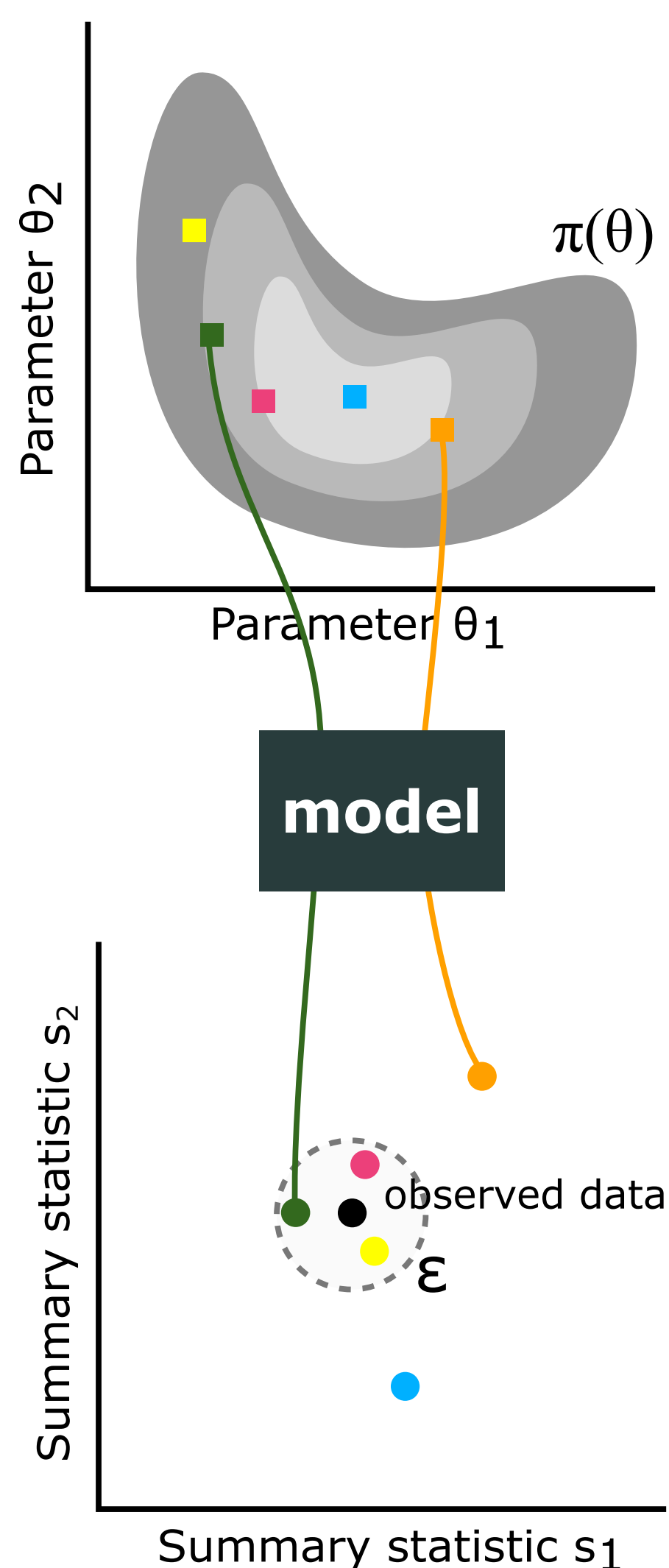


multi-scale agent-based model of tumor growth<sup>1</sup>

## Solution: Approximate Bayesian Computation

### Idea

circumvent likelihood evaluation by measuring distance between simulated and observed data



### Method

with a distance metric  $d$ , a threshold  $\epsilon$ , and summary statistics  $s$ :

### Algorithm (ABC-Rejection)

- until  $N$  acceptances:
- sample parameters  $\theta^* \sim \pi(\theta)$
  - simulate data  $y^* \sim \pi(y|\theta^*)$
  - accept if  $d(s(y^*), s(y_{obs})) \leq \epsilon$

### What you get

$$\pi_{ABC}(\theta|s_{obs}) \propto \int I(d(s, s_{obs}) \leq \epsilon) \pi(s|\theta) \pi(\theta) ds$$

$$\approx \frac{1}{N} \sum_{i=1}^N \delta_{\theta^{(i)}}(\theta)$$

### Theorem

Under mild assumptions it holds that

- $\frac{1}{N} \sum_{i=1}^N \delta_{\theta^{(i)}}(\theta) \xrightarrow{w} \pi_{ABC}(\theta|s_{obs})$  for  $N \rightarrow \infty$
- $\pi_{ABC}(\theta|s_{obs}) \xrightarrow{w} \pi(\theta|s_{obs})$  for  $\epsilon \rightarrow 0$

## Make it efficient: ABC-SMC<sup>2</sup>

### Problems

- smaller  $\epsilon$  reduces approximation error, but also decreases acceptance rates
- also, sampling from the prior is inefficient

### Solution

- combine with a Sequential-Monte-Carlo scheme
- iteratively decrease  $\epsilon$  while sampling from an increasingly better approximation of the posterior

## Software: pyABC<sup>3</sup>

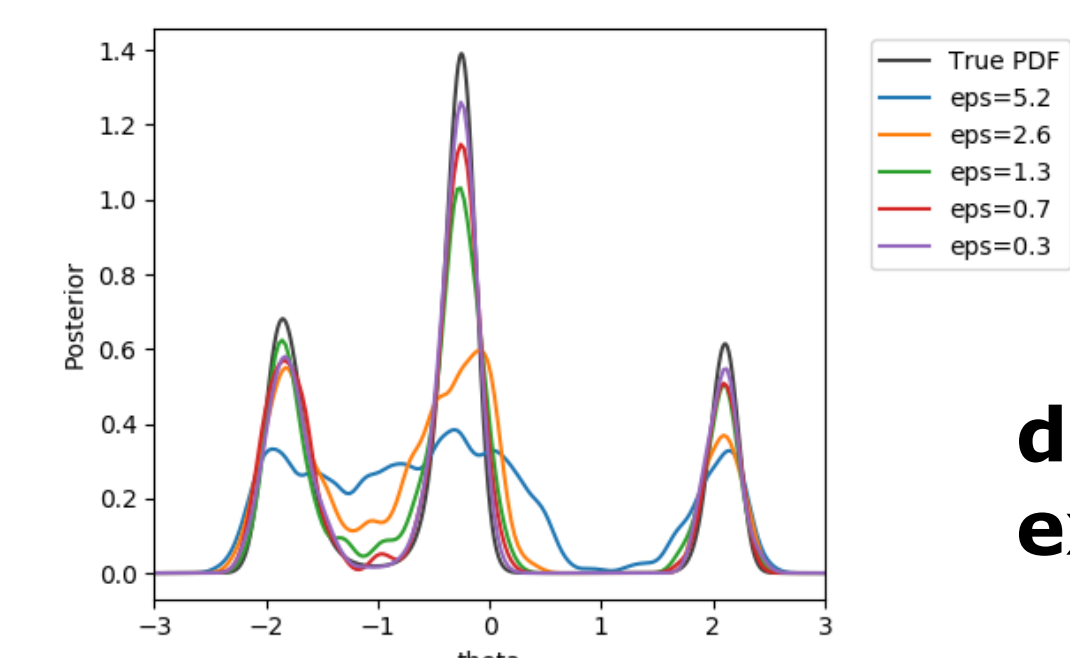
### easy to get started

```
# pass your model
abc = pyabc.ABCSMC(model, prior)

# pass your data
abc.new(db_file, data)

# run it
abc.run(min_eps, max_iter)
```

### visualizations



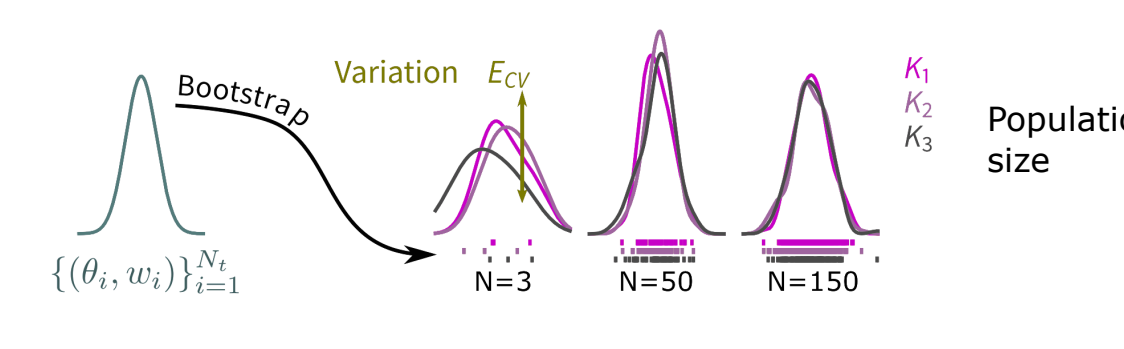
user-friendly

docs and examples



### state-of-the-art algorithms

- adaptive population sizes



- transition kernels



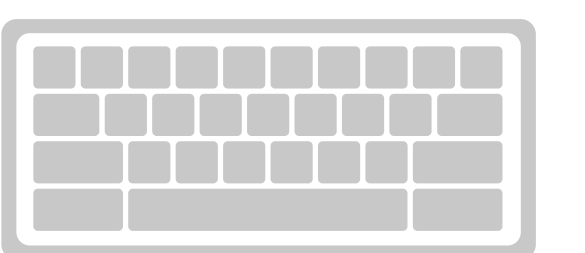
flexible

- adaptive distance functions<sup>4</sup>

$$d(x, y) = \left( \sum_i |\omega_i^{(k)}(x_i - y_i)|^p \right)^{1/p}$$

- epsilon thresholds

customize



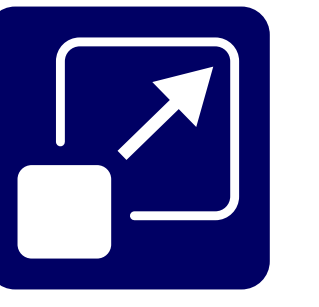
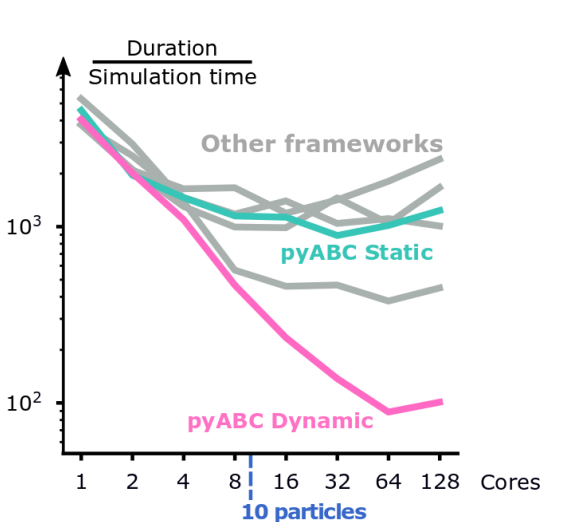
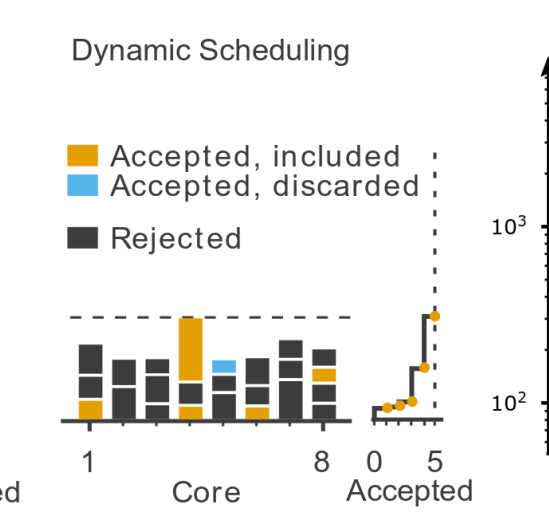
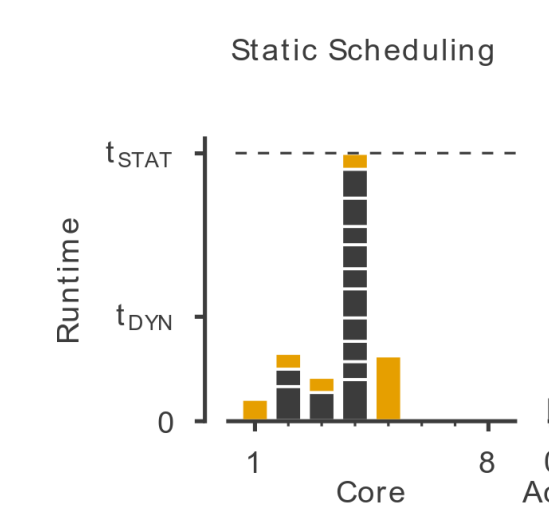
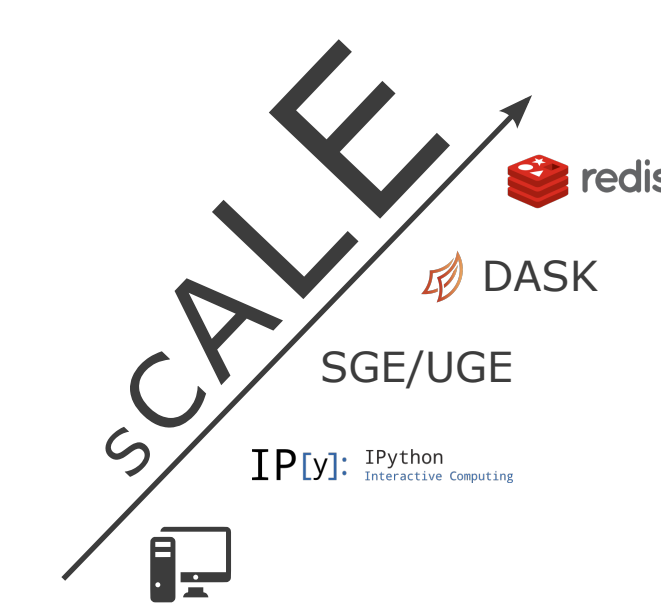
- measurement noise assessment<sup>5</sup>

$$y \sim \pi(y|\theta) + \delta_\epsilon$$

### run where you like

### make your cores count

### finish first



scalable

## Next steps

- weight summary statistics with correlations and model error ...
- efficiently handle measurement noise in ABC-SMC ...
- (semi-)automatically select informative summary statistics for different data types ...
- various application projects underway ...

## Conclusion

- likelihood-free parameter estimation is challenging
- ABC enables reliable statistical inference
- broadly applicable
- pyABC offers a scalable framework

## References and Contact

- [1] Jagiella et al.; Parallelization and high-performance computing enables automated statistical inference of multi-scale models; Cell Systems; 2017
- [2] Toni et al.; Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems; Journal Royal Statistical Society; 2008
- [3] Klinger et al.; pyABC: distributed, likelihood-free inference; Bioinformatics; 2018
- [4] Prangle; Adapting the ABC distance function; Bayesian Analysis; 2015
- [5] Wilkinson; Approximate Bayesian computation gives exact results under the assumption of model error; Statistical applications in genetics and molecular biology; 2013



github.com/icb-dcm/pyabc

yannik.schaelte@helmholtz-muenchen.de

