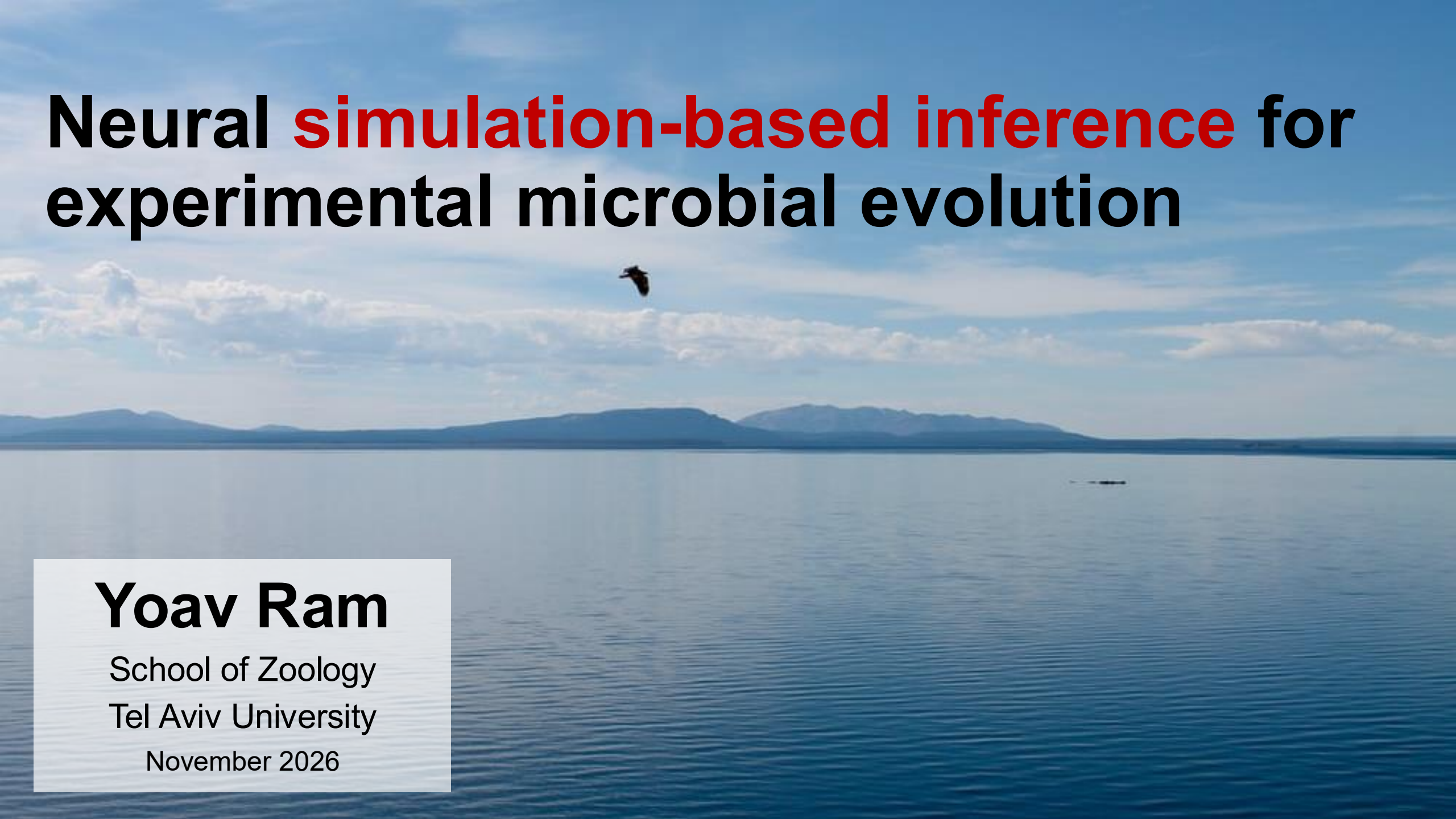


Neural simulation-based inference for experimental microbial evolution

Yoav Ram

School of Zoology

Tel Aviv University

November 2026

Funding















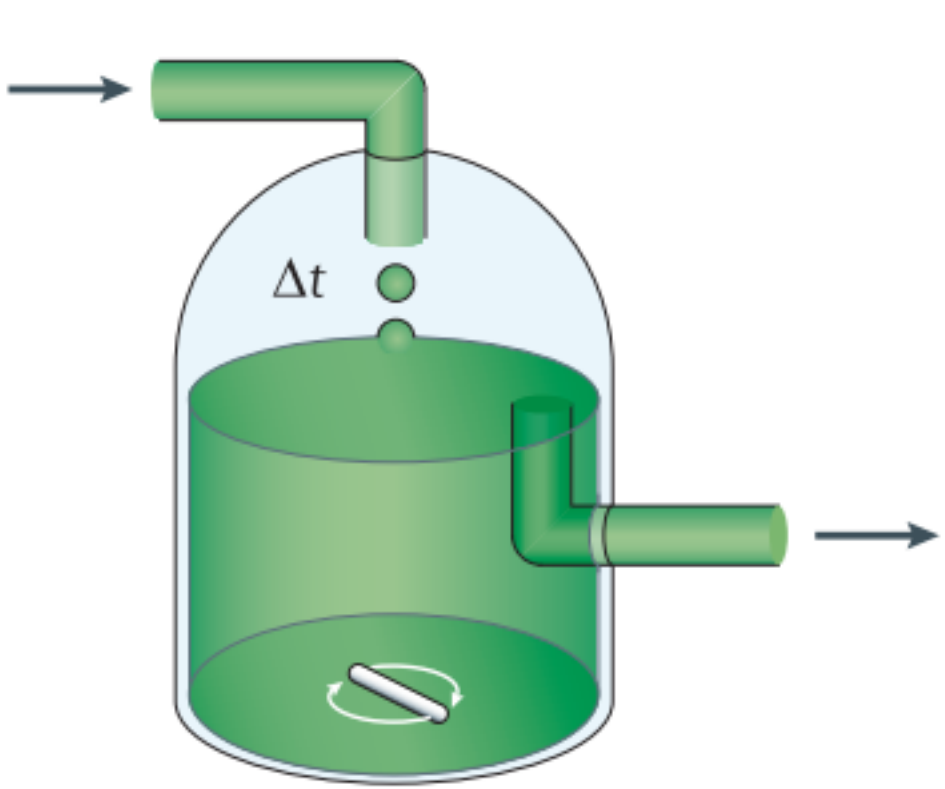




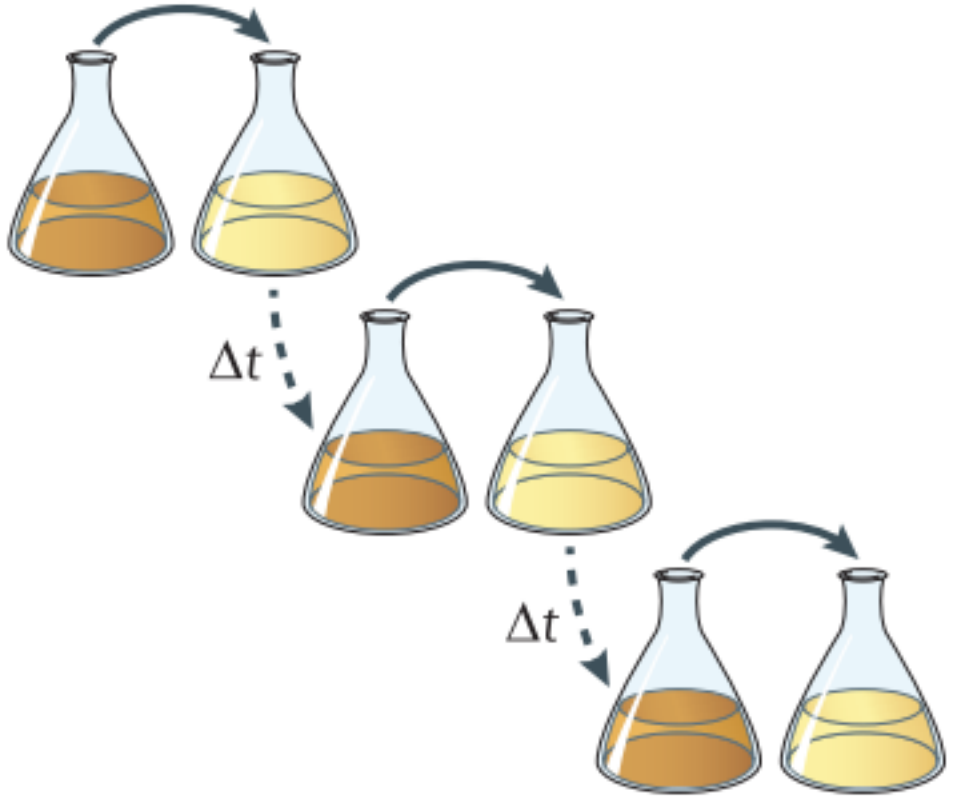
TAD - The Center for AI & Data Science
Tel Aviv University

Evolution in the lab

chemostat



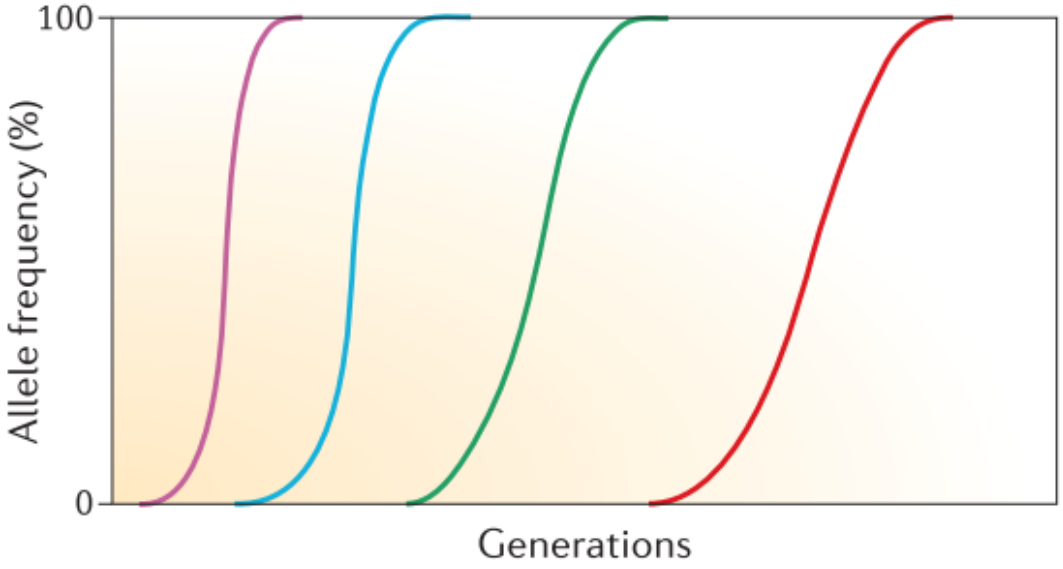
serial transfer

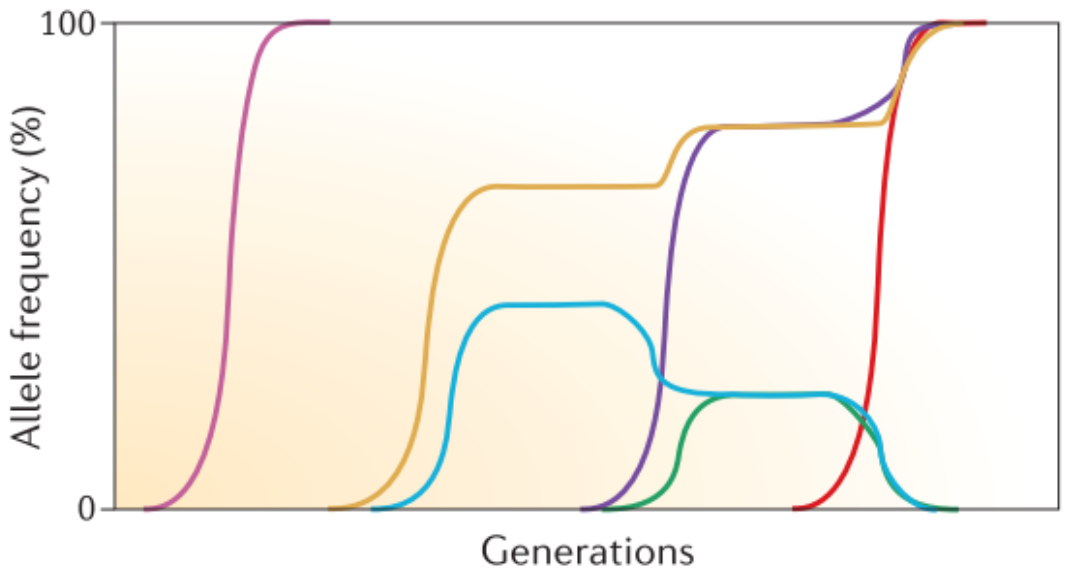


Barrick & Lenski *Nat Rev Genet* 2013

Evolutionary model

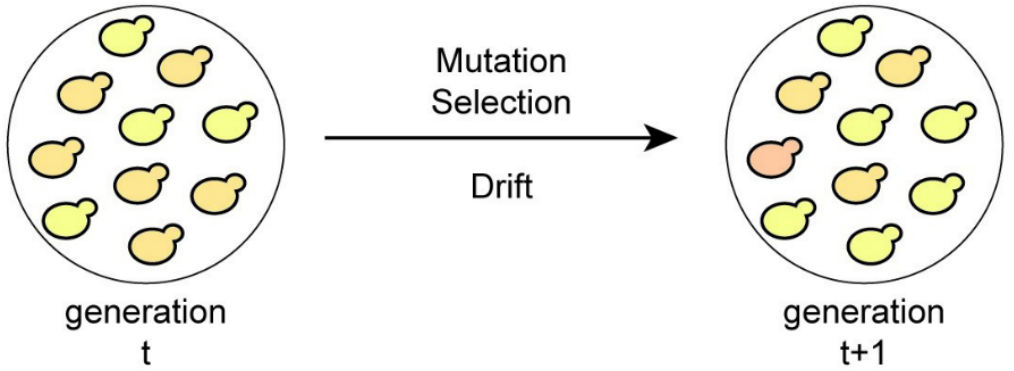
Model for change in frequency of genotypes / alleles over time





Wright-Fisher model

- Effects of
  - Mutation - δ_i
  - Selection – s_i
  - Population size - N
- **Forward:** computationally efficient
- **Backward:** …



Mutation (new variants)

$$x^{\dagger}_A = (1 - \delta_B - \delta_C)x_A,$$

$$x^{\dagger}_B = x_A\delta_B + x_B,$$

$$x^{\dagger}_C = x_A\delta_C + x_C$$

Selection (survival of the fittest)

$$x^*_i = \frac{w_k x_i}{\bar{w}}, \quad \bar{w} = \sum_{i\in\{A,B,C\}} w_i x_i,$$

$$w_i = 1 + s_i, i = B, C$$

Genetic drift (stochastic effects)

$$n_i = Multinomial(N, (x^*_A, x^*_B, x^*_C)), \quad x'_i = \frac{n_i}{N},$$



Inference

Bayesian inference

$$P(\theta|x) = P(x|\theta) \cdot P(\theta) / P(x)$$

$$\text{posterior} = \text{likelihood} \cdot \text{prior} / \text{evidence}$$

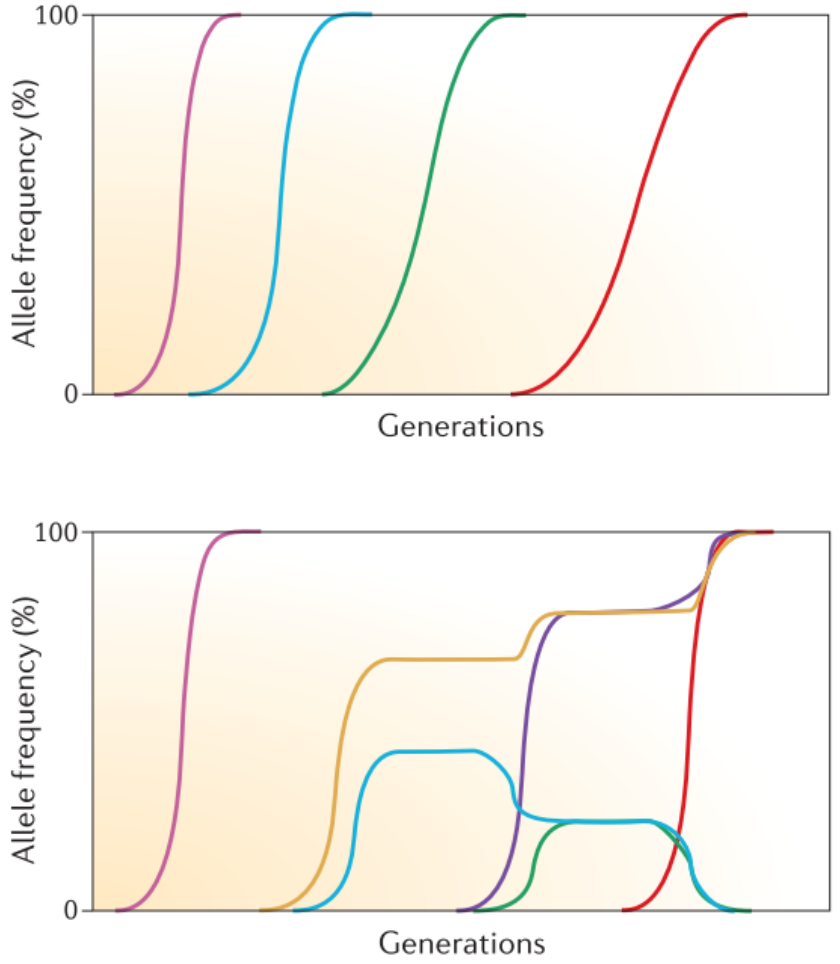
Inference

Infer **posterior distribution** $P(\theta|x)$

where x is the **data**

and θ are the model **parameters**:

- Selection coefficient s
- Mutation rate u
- Population size N

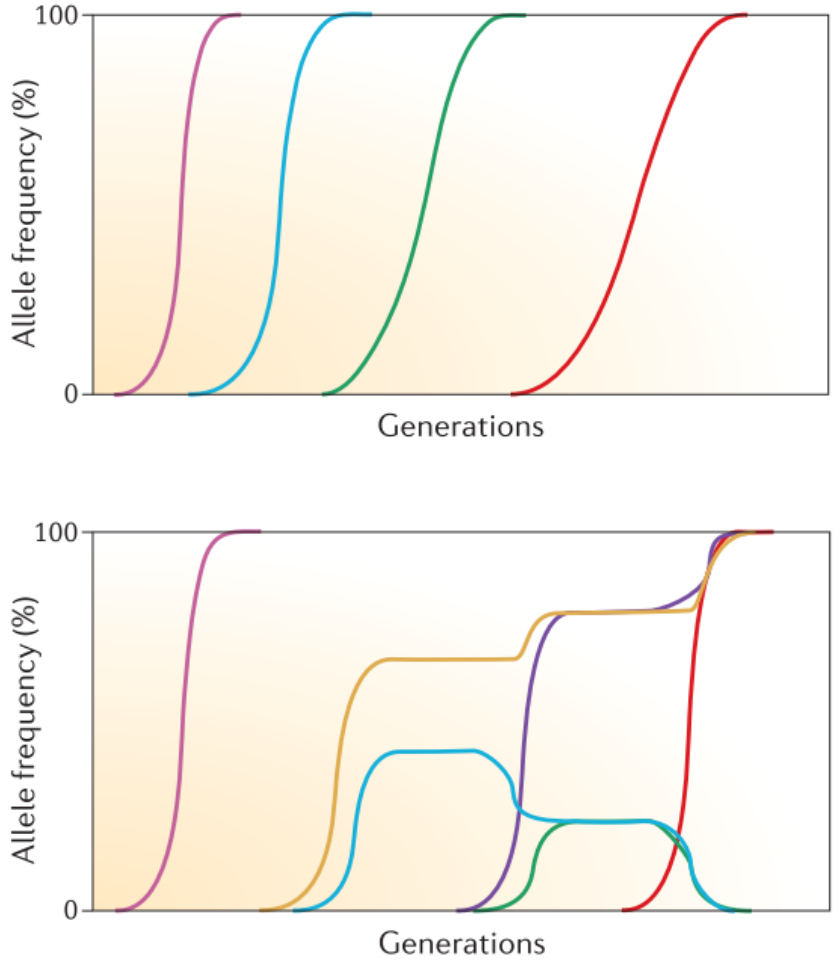


Inference

Infer **posterior distribution** $P(\theta|x)$
where x is the **data**
and θ are the model **parameters**:

- Selection coefficient s
- Mutation rate u
- Population size N

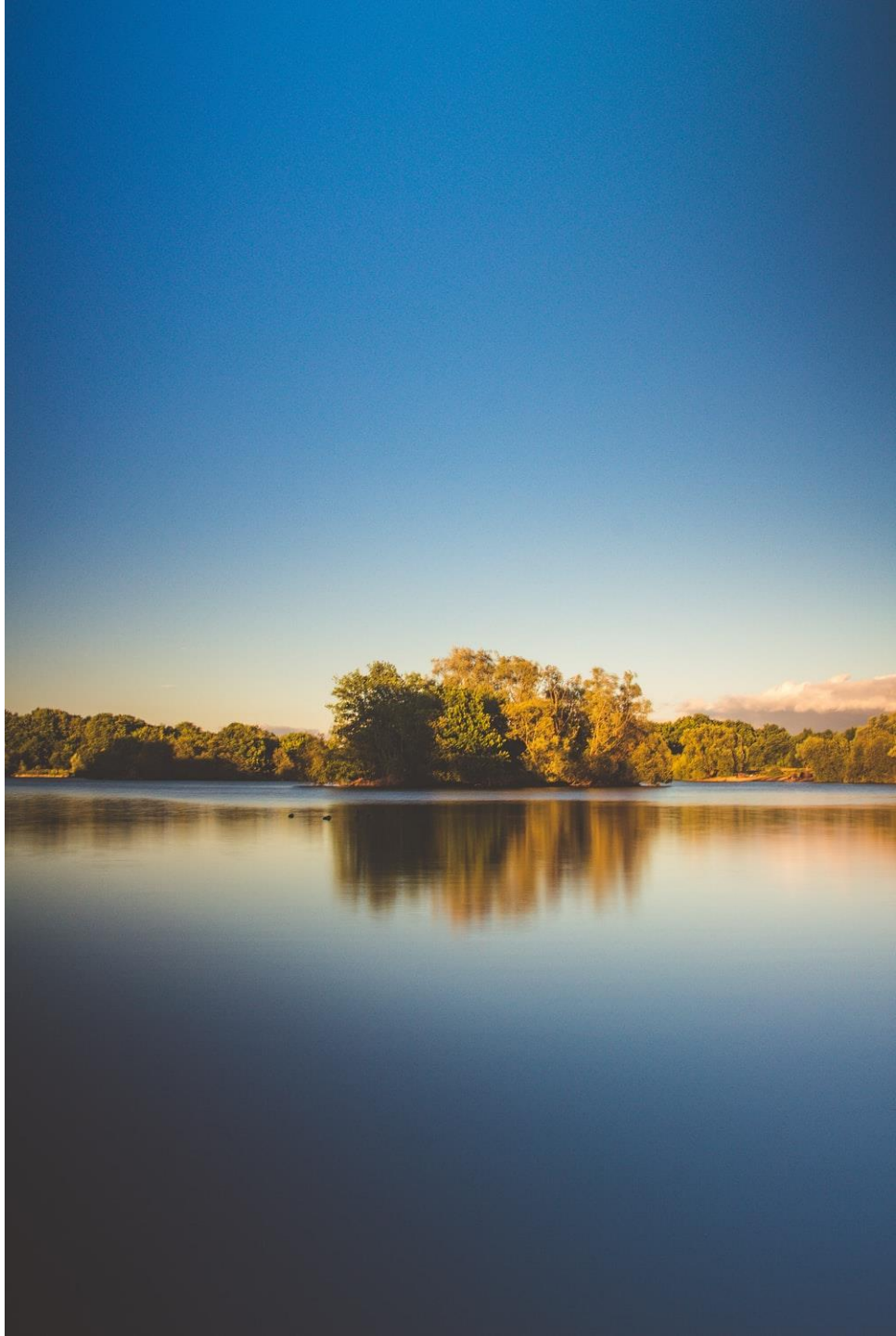
Complex dynamics
Intractable likelihood function…



Simulation-based inference (SBI)

Bayesian inference…
without likelihoods…
using stochastic simulations

Also called:
Likelihood-free inference (LFI)



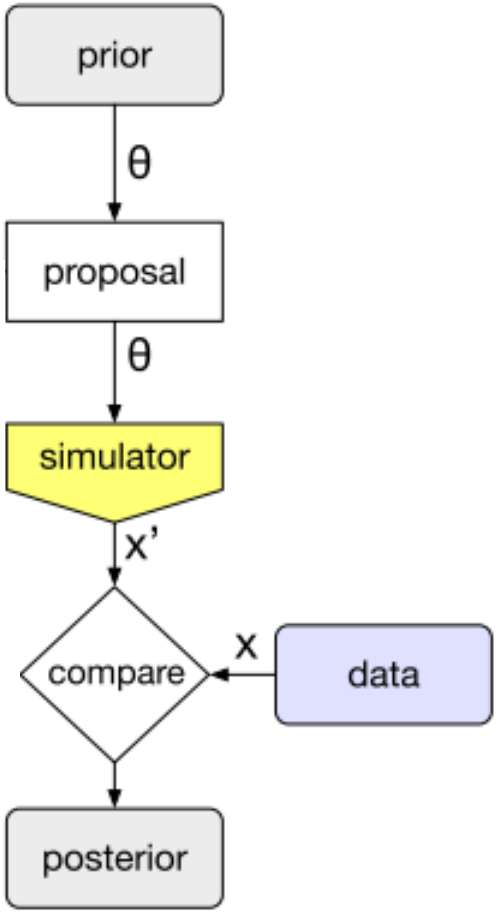
Approximate Bayesian computation (ABC)

Rejection-ABC

- Draw parameter proposal $\boldsymbol{\theta}$ from prior $\boldsymbol{P(\theta)}$
- Simulate synthetic data $\boldsymbol{x'}$
- Compare to empirical data $\boldsymbol{x}$
- Accept $\boldsymbol{\theta}$ if $\boldsymbol{D(x,x')} < \varepsilon$

Result

Accepted $\boldsymbol{\theta}$s are (approximately) samples from the posterior distribution $\boldsymbol{P(\theta|x)}$



Tavaré et al 1997; Fu, Li 1997; Pritchard et al 1999

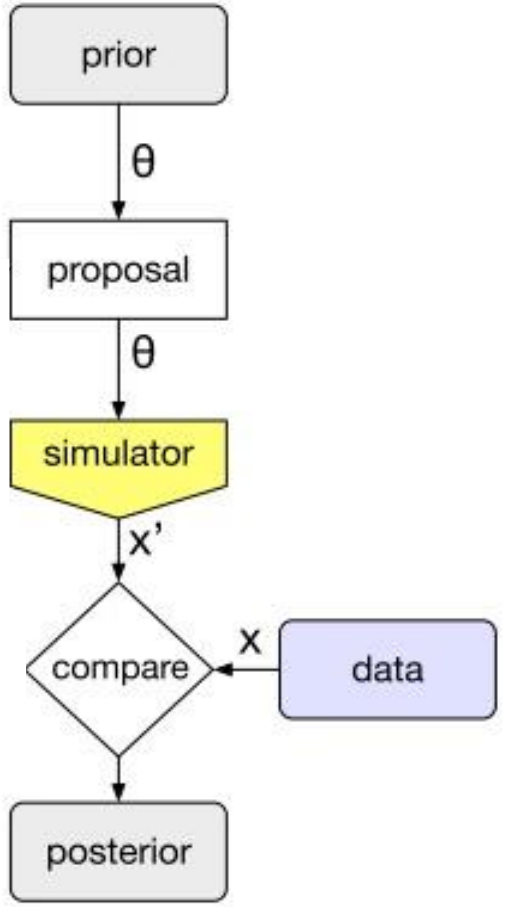
Approximate Bayesian computation (ABC)

Requires

- Prior distribution – always an issue
- Simulator
- Distance function ***D(x,x')***
- Acceptance threshold ***ε***

<u>Problems</u>

- Inefficient: Low acceptance rate
- Choosing ***D(x,x')*** and ***ε*** can be hard
  - Especially when ***x*** is high-dimensional / multi-modal



Efficient sampling algorithms

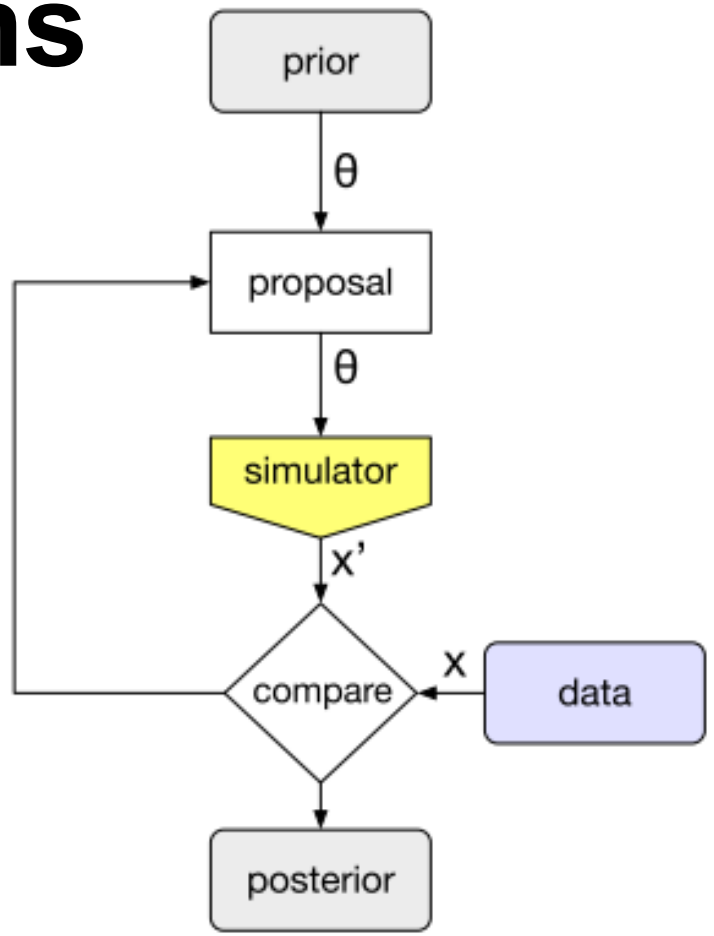
Markov Chain Monte Carlo ABC (MCMC-ABC)

Marjoram et al. 2003, Tanaka et al. 2006

Sequential Monte Carlo ABC (SMC-ABC)

Sisson et al. 2007; Toni et al. 2009; Python package: pyABC

Higher acceptance rate



<u>Problems</u>

- Still needs ***D(x,x')*** and ***ε***
- Choosing proposal distribution can be hard
- Re-run for every new dataset

Regression approaches to ABC

- Compare summary statistic $s(x)$ instead of data x
- Simulate (θ_i, x_i') pairs from prior and simulator
- Solve regression problem $\theta_i = \alpha + (s(x_i') - s(x))\,\beta$
- Then $\theta_i - (s(x_i') - s(x))\,\beta$ are approximately samples from $P(\theta|x)$

Beaumont, Zhang, Balding 2002

Blum, François 2010

Problems

- Choosing a summary statistic $s(x)$
- What kind of regression?
- Assumptions on the posterior shape

Neural Networks!



Neural simulation-based inference

Train a neural network to estimate the

- ***NPE*: posterior density function $P(\theta|x)$**

Papamakarios, Murray *NeurIPS* 2016

- *NLE*: likelihood function $P(x|\theta)$

Papamakarios, Sterratt, Murray *AISTATS* 2019

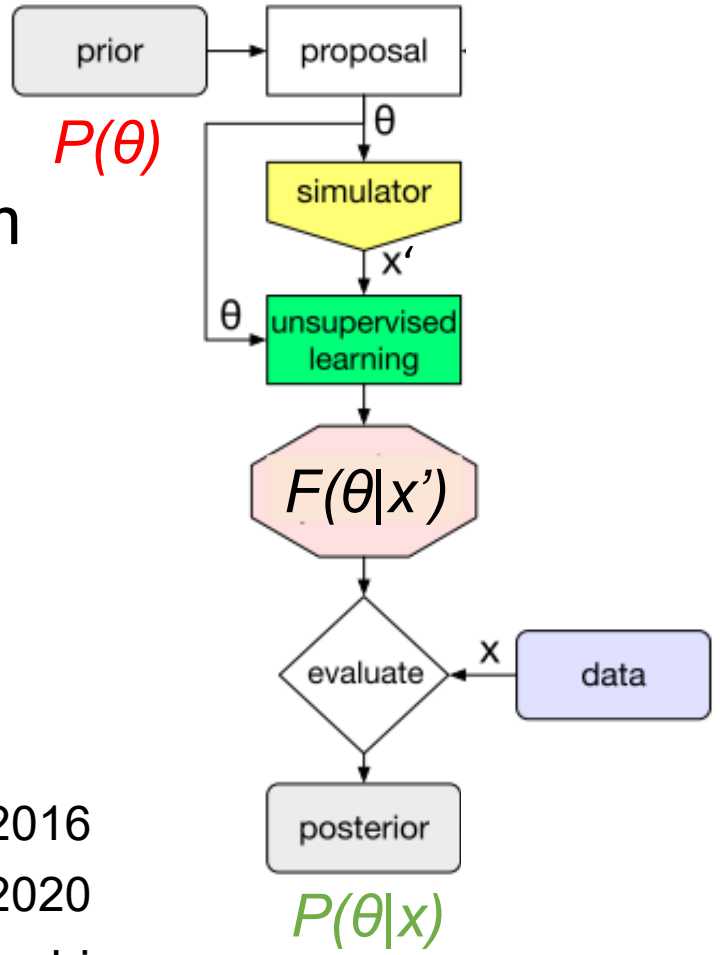
- *NRE*: likelihood-to-evidence ratio $P(x|\theta)/P(x)$

Hermans, Begy, Louppe *ICML* 2020

Review: Cranmer, Brehmer, Louppe *PNAS* 2020

Neural posterior estimation (NPE)

- Simulate (θ_i, x_i') pairs from prior and simulator
- Train neural density estimator $F(\theta|x')$ to approximate the posterior distribution $P(\theta|x')$ from the simulated pairs (θ_i, x_i')
- Evaluate F on empirical data x to get posterior distribution $P(\theta|x)$ conditioned on data x



Papamakarios & Murray *NeurIPS* 2016

Review: Cranmer et al. *PNAS* 2020

Python, PyTorch, sbi

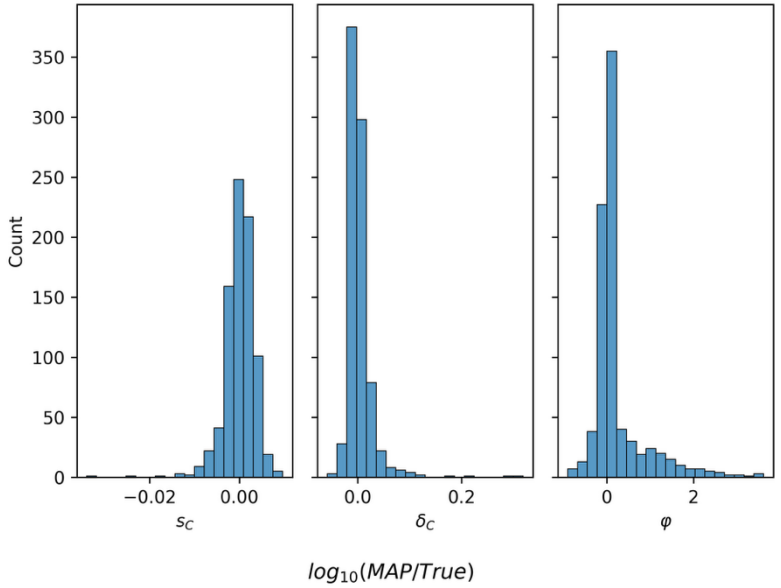
Neural density estimation

Neural network *F* architecture:

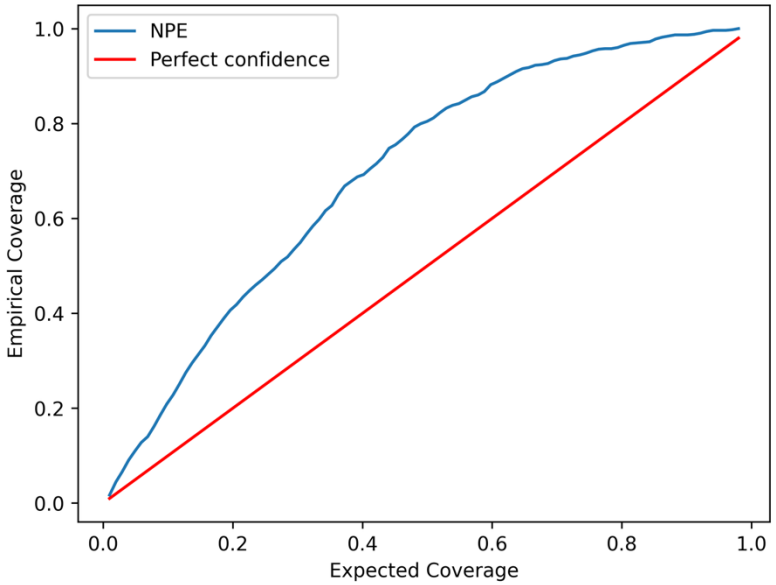
- MDN: Mixture-density network (MoG output) Bishop 1994
- **MADE: Masked autoencoder for distribution estimation** Germain et al. 2015
- MAF: Masked autoregressive flow Papamakarios et al. 2017
- NSF: Neural spline flows Durkan et al. 2019
- FMPE: Flow matching posterior estimation Dax et al. 2023
- NPSE: Diffusion posterior estimation Geffner et al. 2023, Sharrock et al. 2024

Evaluation

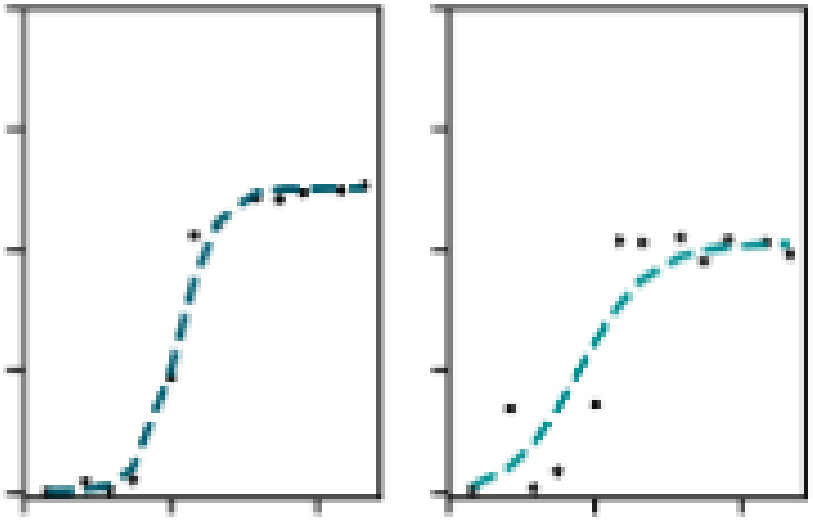
$\log_{10}\frac{MAP}{True}$ histograms

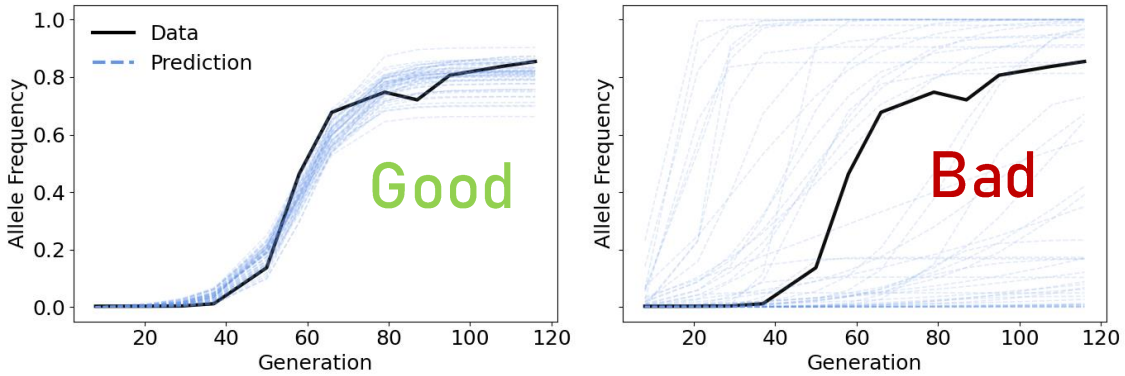


Coverage plot



Posterior Predictive checks





Chuong et al., 2025

Application to lab evolution: yeast

Avecilla et al., *PLOS Biology* 2022
Chuong, et al., *eLife* 2025
De, et al., *Molecular Biology and Evolution*, 2026
Sakrikar, et al., *bioRxiv*, 2026


David Gresham


Grace Avecilla


Julie Chuong


Titir De


Saaz Sakrikar

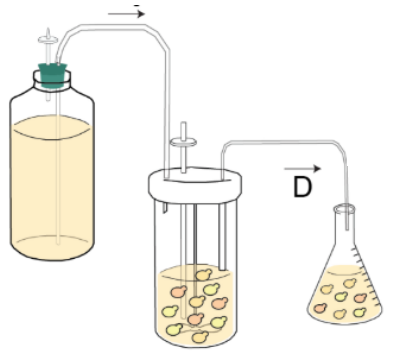

Nadav Ben Nun

Application to lab evolution

11 yeast populations

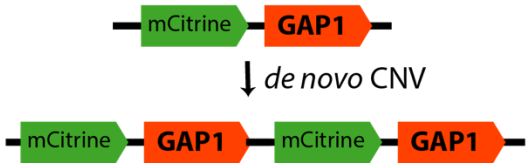
Evolution in glutamine-limited chemostats

267 generations (65 days)

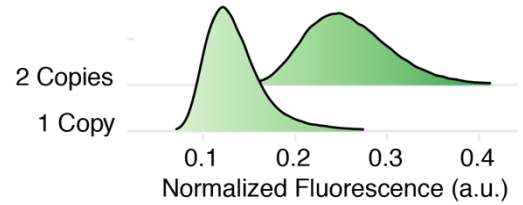


GAP1 gene with fluorescent marker

(*GAP1* = general amino-acid permease gene)



Monitor copy number using flow cytometry



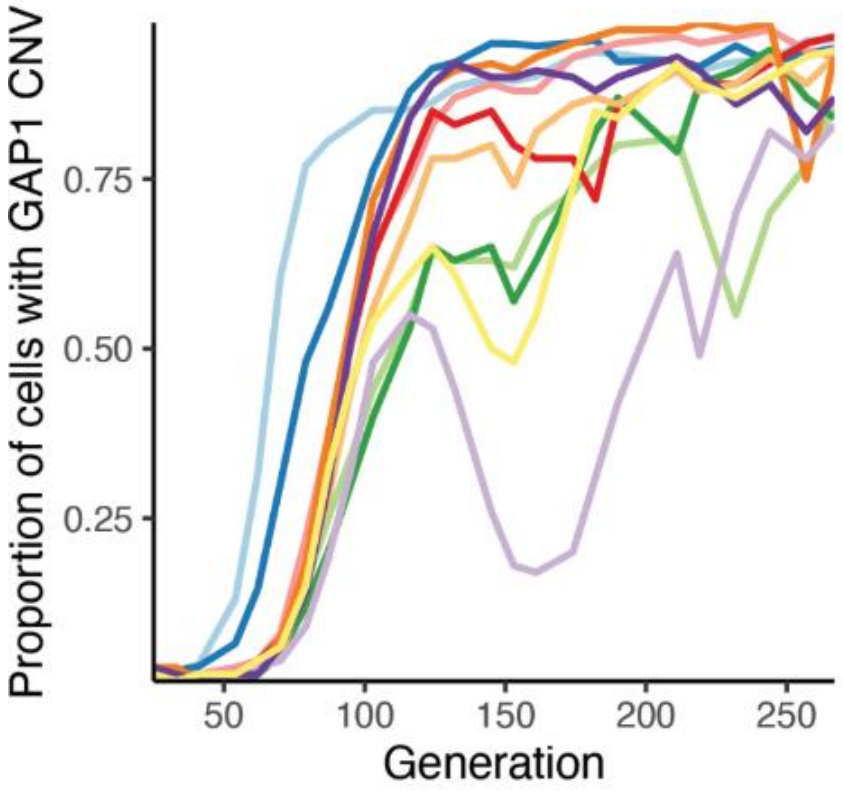
Lauer, Avecilla, Spealman, Sethia, Brandt, Levy, Gresham *PLoS biology* 2018

Data

11 populations

25 time points

Proportion of cells with >1 *GAP1*

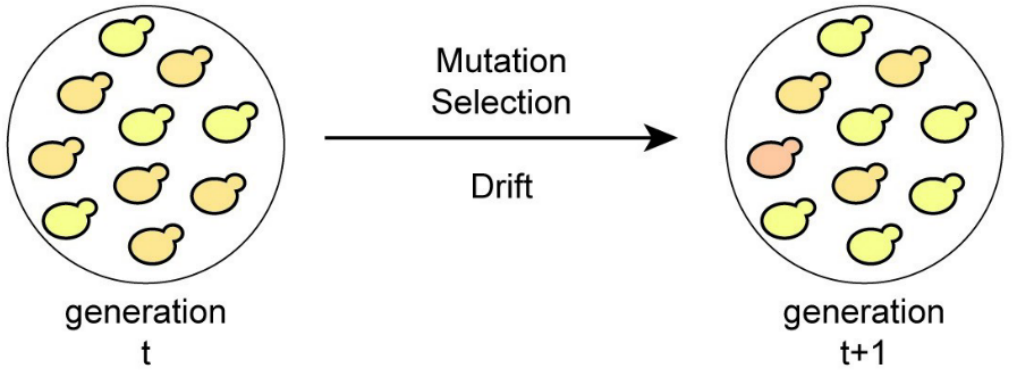


Lauer, Avecilla, Spealman, Sethia, Brandt, Levy, Gresham *PLoS biology* 2018

Wright-Fisher model

- Effects of
  - Mutation - δ_i
  - Selection – s_i

- Non-overlapping generations
- **Computationally efficient**



Mutation & CNV

$$x^{\dagger}_{A} = (1 - \delta_B - \delta_C)x_A,$$

$$x^{\dagger}_{B} = x_A\delta_B + x_B,$$

$$x^{\dagger}_{C} = x_A\delta_C + x_C$$

Selection

$$x^{*}_{i} = \frac{w_k x_i}{\bar{w}}, \quad \bar{w} = \sum_{i\in\{A,B,C\}} w_i x_i,$$

$$w_i = 1 + s_i, i = B, C$$

Genetic drift

$$n_i = Multinomial(N, (x^{*}_{A}, x^{*}_{B}, x^{*}_{C})), \quad x'_i = \frac{n_i}{N},$$

Inferred parameters

- CNV adaptive fitness effect s_c
- CNV formation rate per generation δ_c

Avecilla, Chuong, Li, Sherlock, Gresham, Ram *PLOS Biology* 2022

Results: Validation

Synthetic data

Simulate from one model

Infer with the other model

NPE works well and better than SMC-ABC

Avecilla, Chuong, Li, Sherlock, Gresham, Ram *PLOS Biology* 2022

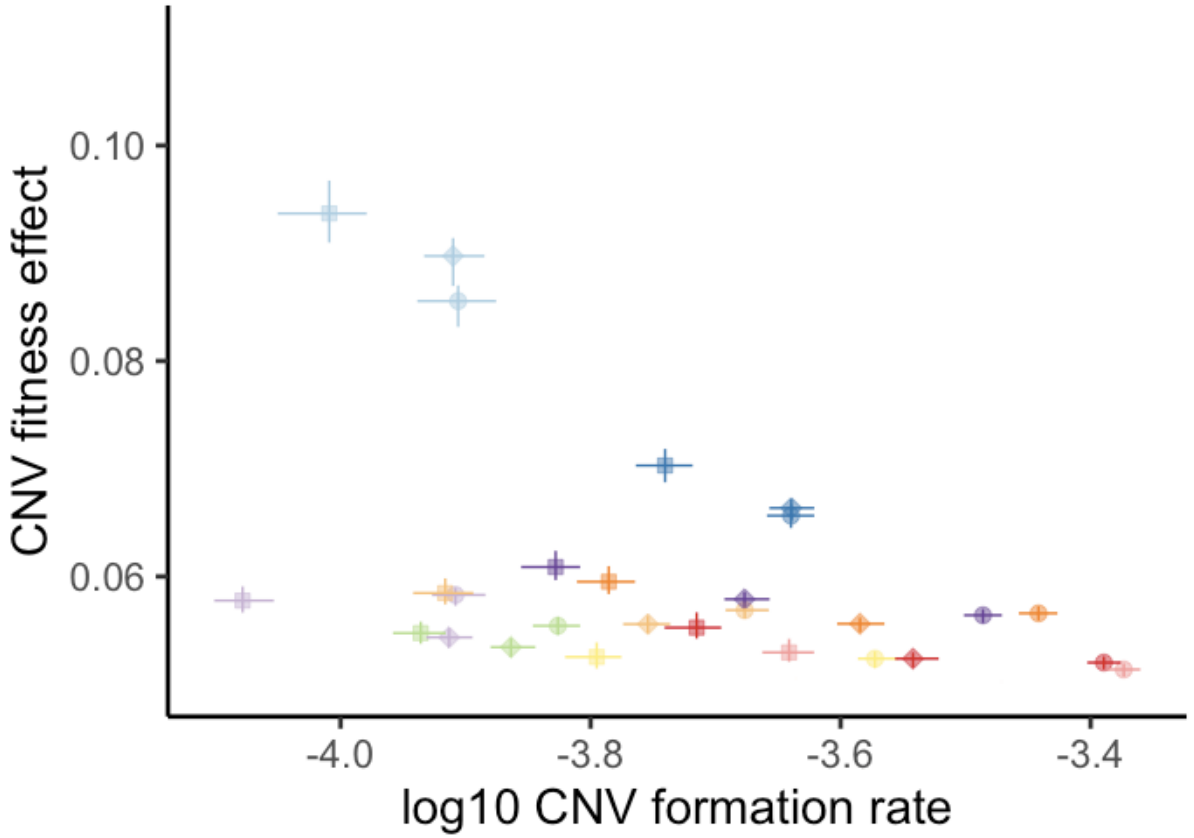
Results

11 populations (color)

3 inferences per population

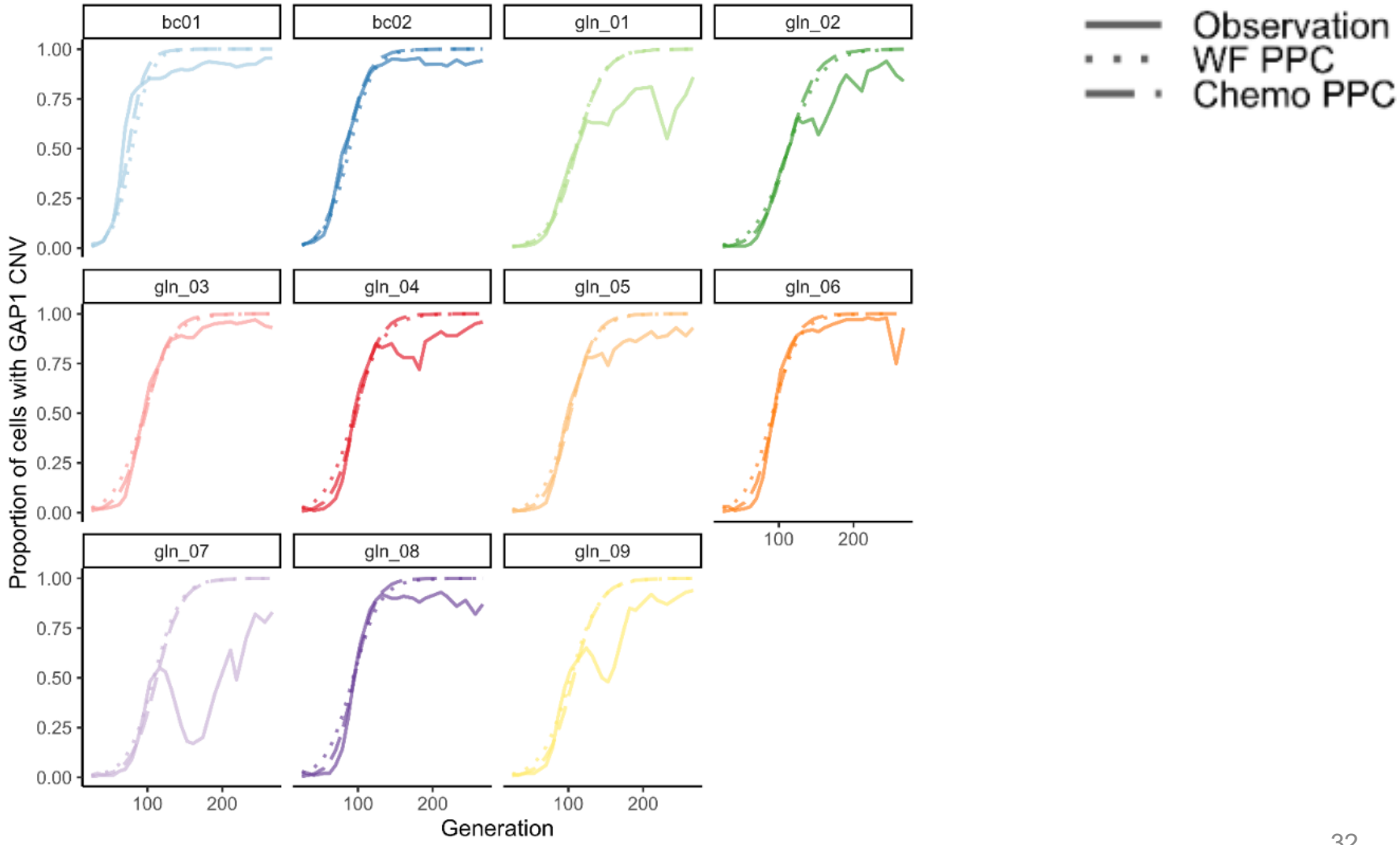
Fitness effect: ~0.06

CNV formation rate: ~0.0002



Avecilla, Chuong, Li, Sherlock, Gresham, Ram *PLOS Biology* 2022

Predictive checks



Avecilla, Chuong, Li, Sherlock, Gresham, Ram *PLOS Biology* 2022

Application to lab evolution: viruses

Caspi et al., *Virus Evolution* 2023

Maoz et al., bioRxiv, 2026



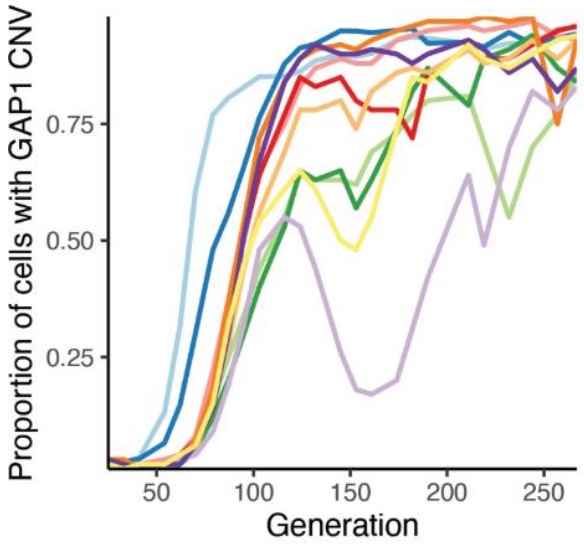





Collective Posterior
Distribution

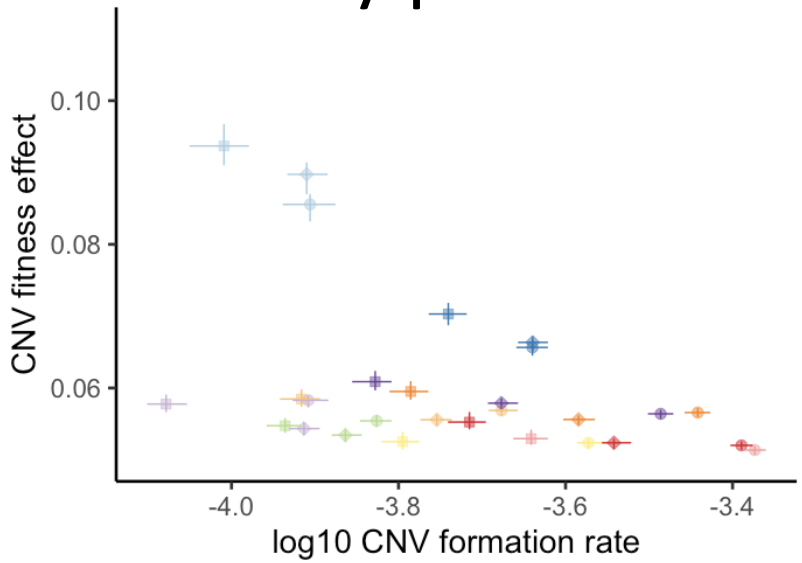
Posterior Inference from Replicates

Many replicates



SBI

Many posteriors

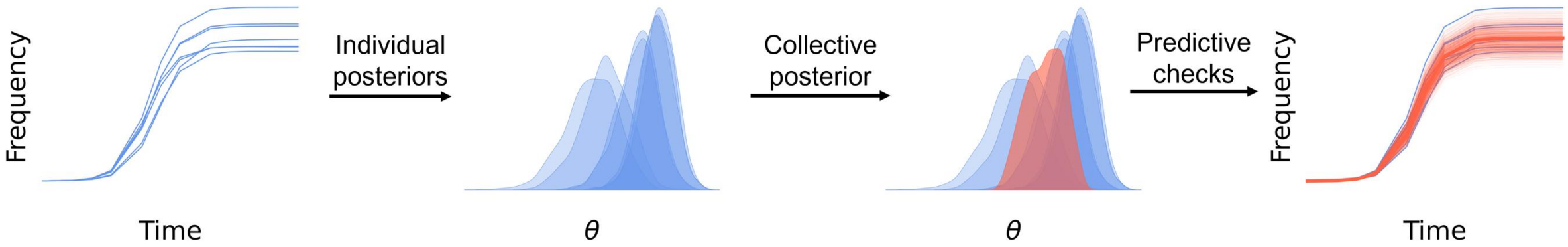


Goal: Aggregate individual posteriors to a collective posterior

Collective posterior by Bayes' law

Ben Nun, et al., *PLOS Computational Biology*, 2026

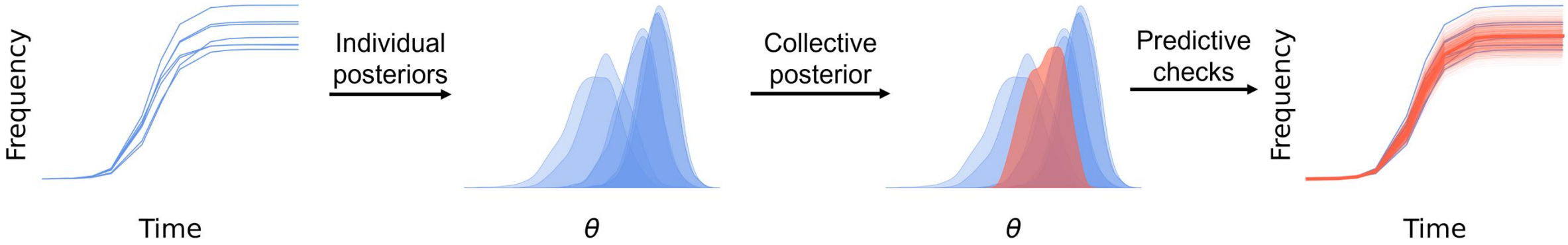
$$P(\theta|x_1,\ldots,x_n) = \frac{P(\theta)^{1-n}\Pi_i\left[P\left(\theta|x_i\right)\right]}{\int P\left(\xi\right)^{1-n}\Pi_i\left[P\left(\xi|x_i\right)\right]d\xi}$$



Aggregating at inference time

- Training time on single replicates is faster
- No constraint on number of replicates
- No added complexity
- Plug and play

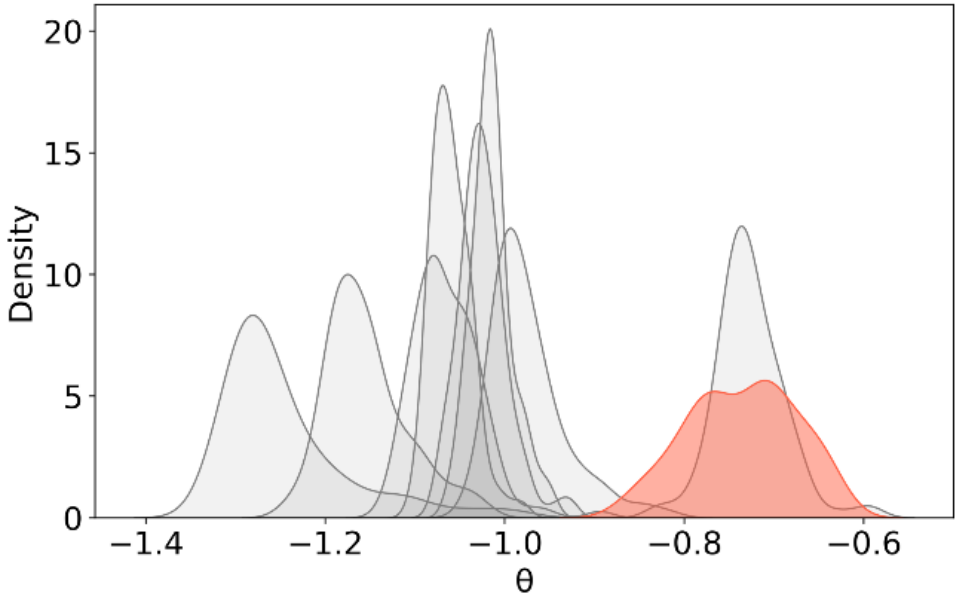
$$P(\theta|x_1, \ldots, x_n) = \frac{P(\theta)^{1-n} \Pi_i [P(\theta|x_i)]}{\int P(\xi)^{1-n} \Pi_i [P(\xi|x_i)] \, d\xi}$$



Naïve solution doesn't work

- Individual posteriors $P(\theta|x_i)$ are available from ABC/NPE/etc.
- Let's use them!

$$P(\theta|x_1, \ldots, x_n) = \frac{P(\theta)^{1-n}\Pi_i[P(\theta|x_i)]}{\int P(\xi)^{1-n}\Pi_i[P(\xi|x_i)]\, d\xi}$$

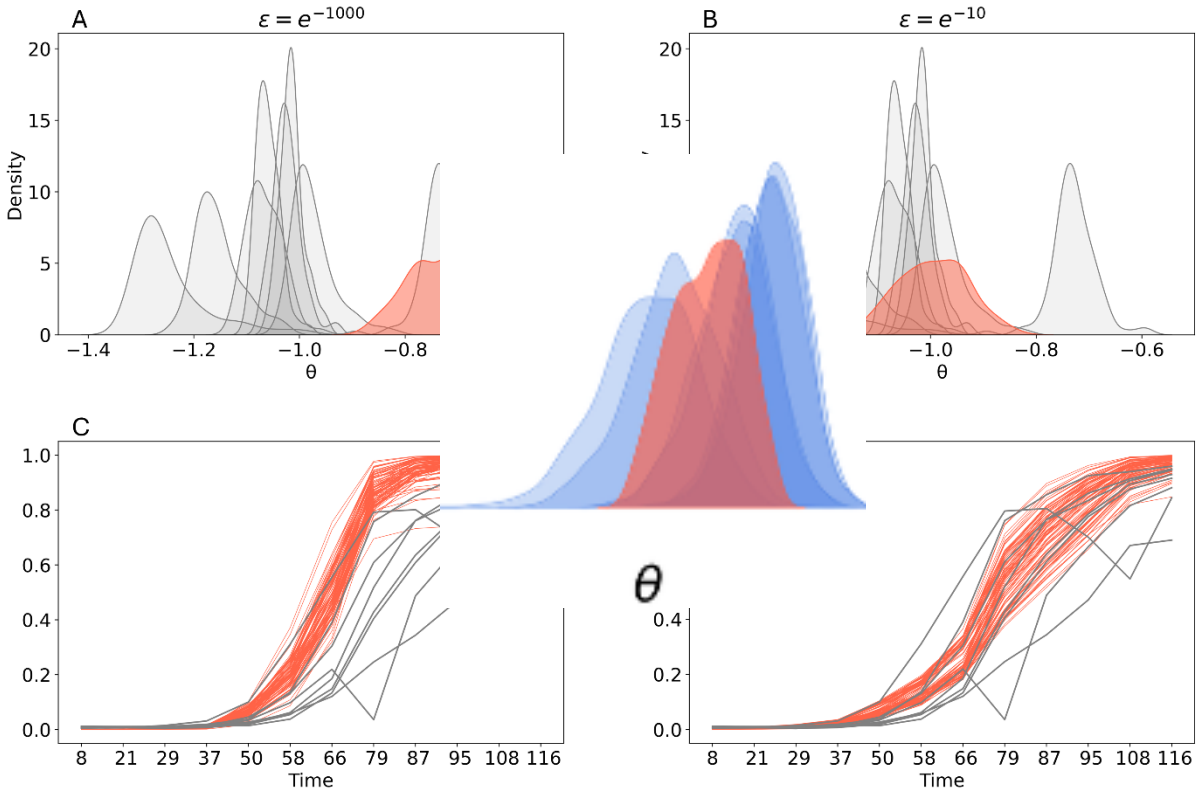


Problem:
- An outlier can take over
- We want to focus on "least disagreement"

Clamping the posteriors

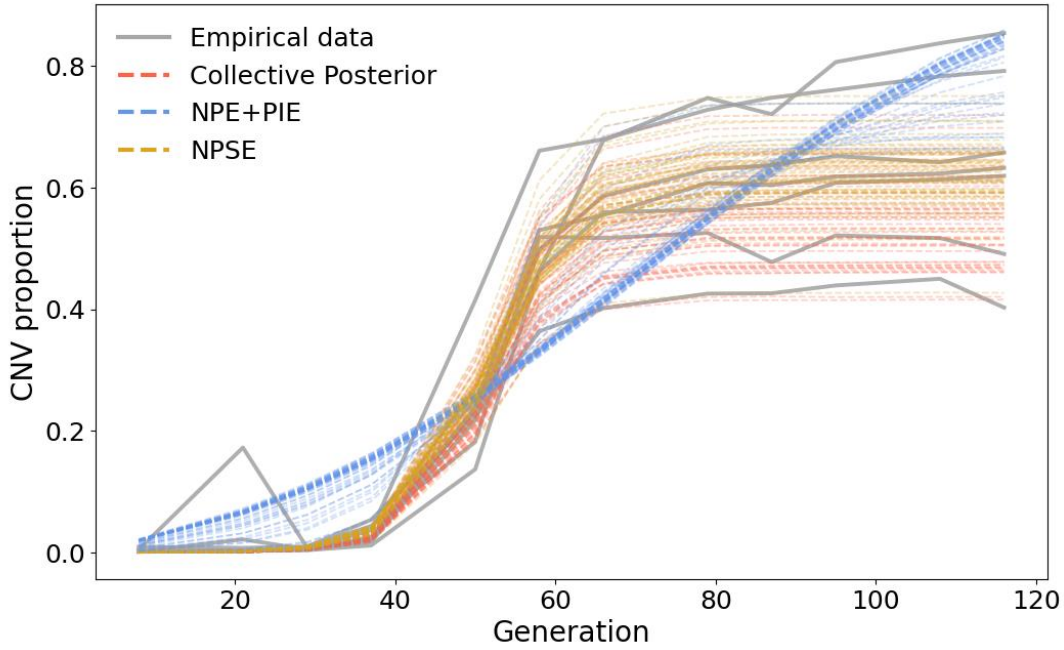
- Set a minimal density value for "acceptance":
$$P_\varepsilon(\theta|x_i) = max[\varepsilon, P(\theta|x_i)]$$
- Choice of ε balances robustness to outliers with accurate parameter estimation
- $P(\theta|x_i) > \varepsilon$ suggests that the individual posterior "agrees" with this θ.
- $P(\theta|x_i) \leq \varepsilon$ doesn't spoil for the rest of the group!

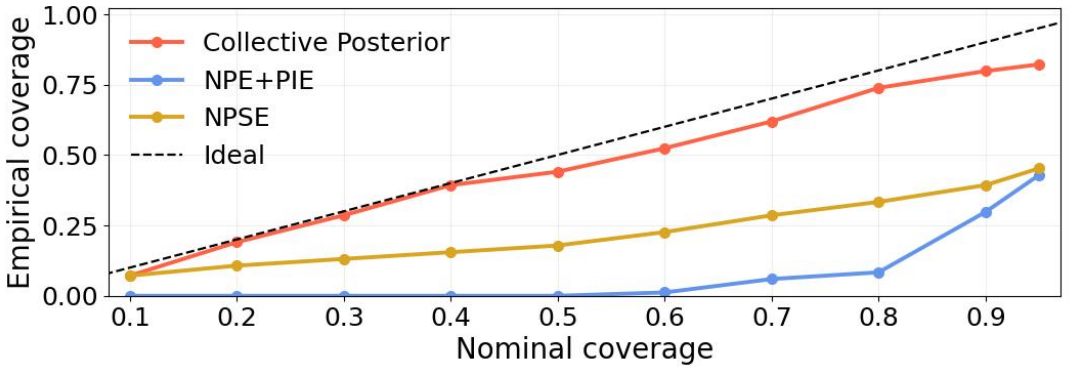
$$P(\theta|x_1, \dots, x_n) = \frac{P(\theta)^{1-n}\Pi_i[P(\theta|x_i)]}{\int P(\xi)^{1-n}\Pi_i[P(\xi|x_i)]\, d\xi}$$



Results

- Yeast evolution (Chuong et al., 2025)
- 3-parameter Wright-Fisher model
  - Selection coefficient s
  - CNV formation rate δ
  - Initial undetectable frequency φ

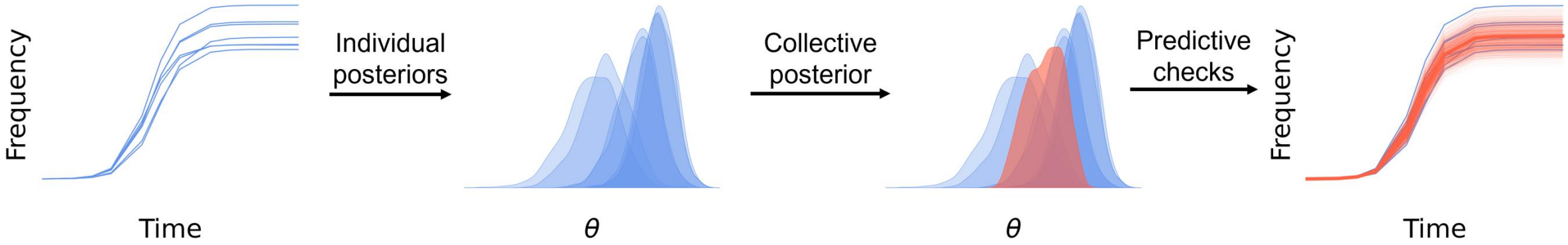




Aggregating at inference time is key

- Training time on single replicates is faster
- No constraint on number of replicates
- No added complexity
- Plug and play with any density estimator

$$P(\theta|x_1, \ldots, x_n) = \frac{P(\theta)^{1-n} \Pi_i [P(\theta|x_i)]}{\int P(\xi)^{1-n} \Pi_i [P(\xi|x_i)] \, d\xi}$$



Ben Nun, et al., *PLOS Computational Biology*, 2026

Summary



Advantages of NPE

- **Black-box simulator**: no likelihood, no gradients, no problem
- **No tuning**: No distance D or threshold ε
- **Amortization**: Learn $F(\theta,x')$ once, infer $P(\theta|x_i)$ many times
- **Efficient**: No wasted simulations; fast inference and sampling
- **Cheap validation**: simulation-based calibration, coverage
- **Automatic summary**: neural encoder
- **Flexible posterior**: flow, diffusion

- **Collective posterior:** inference from replicates

Ben Nun et al., 2026

Ram Lab



Tami Yosef



Yoav Livne



Nadav Ben Nun



Lior Zlotorinsky



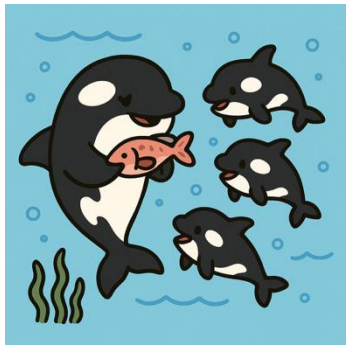
Ari Stolba



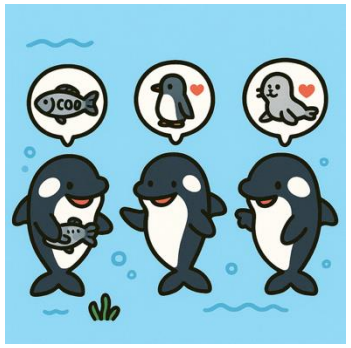
Maayan Yesharim



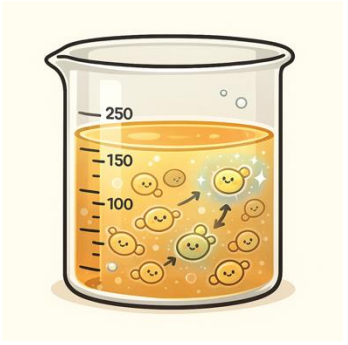
Yotam Caspy



Evolution of menopause



Killer whale diversification



AI for experimental evolution



AI for animal communication



AI for animal behavior



AI for animal identification



Inference of cultural evolution