Lecture 13: Topics in Approximate Bayesian Computation

Examinable assignment 4 - part 1

- Use the Urn model starting with 2 red balls of weight 1, and one black (mutation) ball of weight w.
- Draw balls until you have 10 non-black balls.

- If all non-black balls are the same color at the end, what is the posterior distribution of the weight of the black ball?
- If we observe exactly 2 non-black colors at the end, what is the posterior distribution of the weight of the black ball?
-

Use a Uniform[0,20] prior for the weight of the black ball[2]

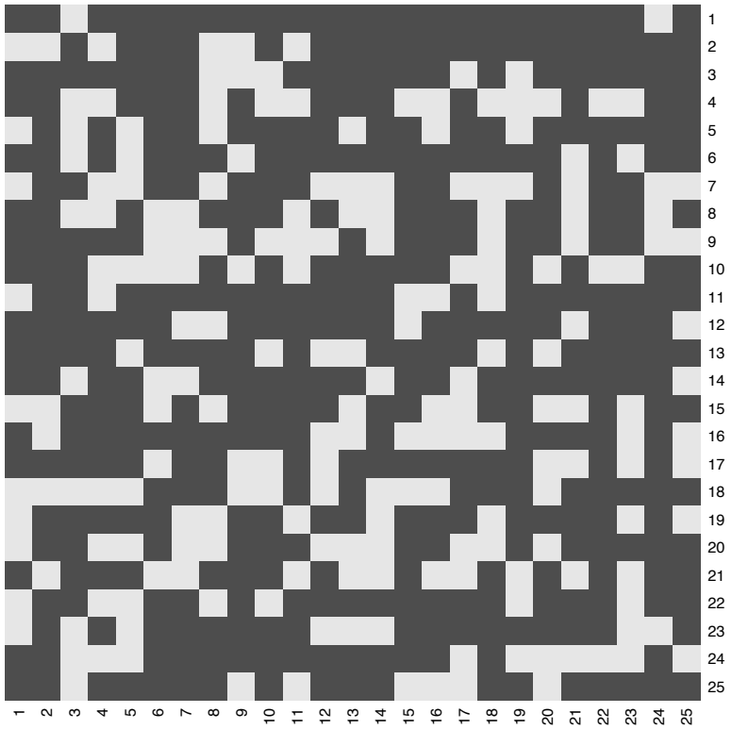
Examinable assignment - part 2

- Repeat the Urn model assignment to find the posterior **mass** of the black ball given that we observe just one (non-black0 color in an Urn containing 10 (non-black) balls.
- This time, sample θ' (black ball **mass**) from ξ, an expo(λ) distribution, rather than a Uniform distribution. (So this is Importance sampling.)
- If you accept θ' then add a point with (**Importance Sampling**) **weight** $\pi(\theta')/\xi(\theta')$ to the posterior distribution of θ.
- If we restrict θ' to be in the interval [0,20] (for simplicity's sake), then this means that we have:
  - $\pi(\theta') = 1/20$
  - $\xi(\theta') = \lambda e^{-\lambda\theta'} / (1-e^{-20\lambda})$
- Compare efficiency to a rejection method that samples directly from a Uniform[0,20] distribution, by comparing the number of iterations you need to run in order to collect 10000 accepted θ's.

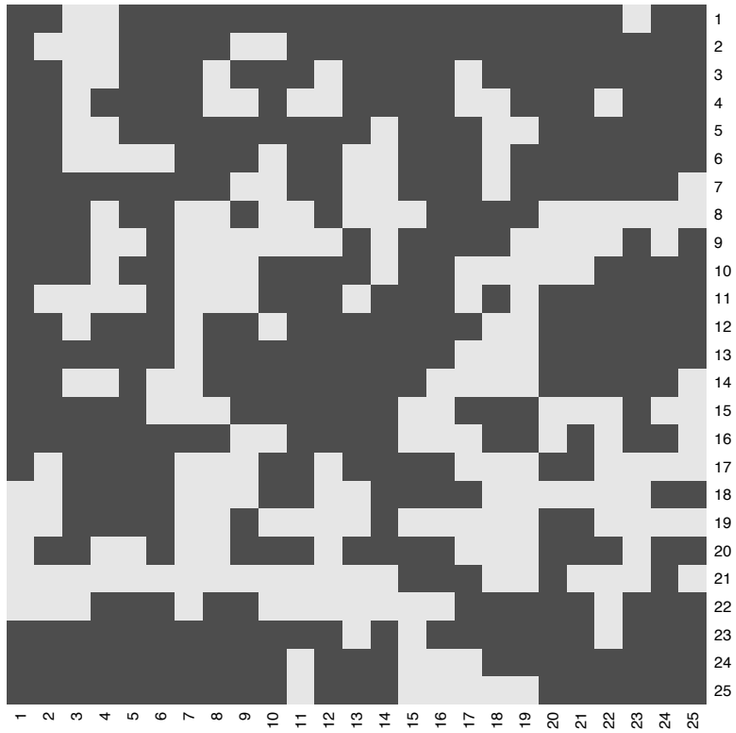
Examinable Assignment - Part 3

- Many tissues contain 2 (or more) cell-types.
- Investigator wants a way of testing whether each of 2 cell types in a given tissue is homogeneously distributed (in space). So:
  - H_0: cell types are homogeneously (randomly) arranged
  - H_1: cell types are not homogeneously arranged (i.e. they are clustered in some fashion).
- Your job, is to come up with such a test, using Monte Carlo methods.
  1. Formulate your test.
  2. Apply it to each of Grid1, Grid2, Grid3 and determine a p-value for rejecting the H_0 in favor of H_1.
  3. Write-up your test (i.e. Methods) and your Results.

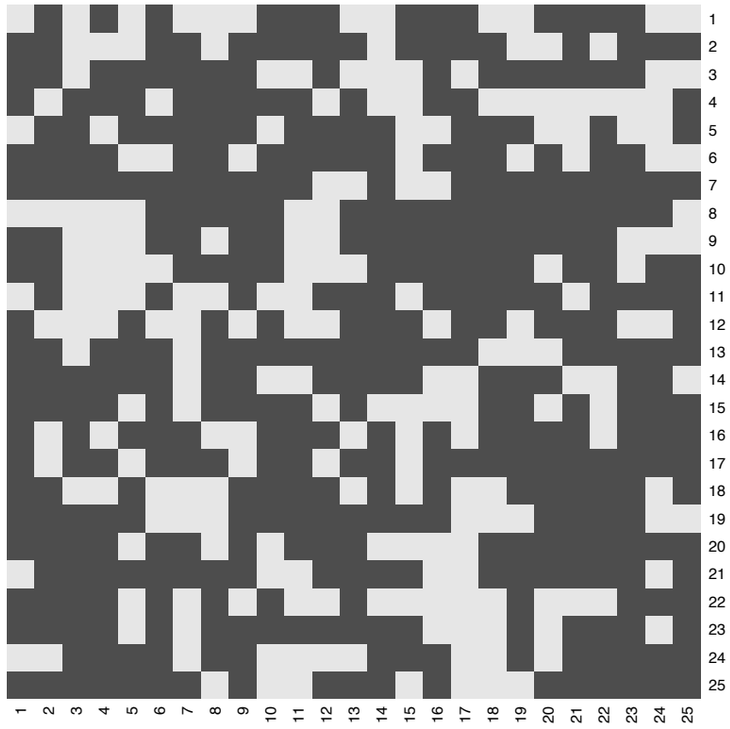
Examinable Assignment - Part 2



“Grid1.txt”



“Grid2.txt”



“Grid3.txt”

‘Final’ Presentation

- Everybody must present something.
- 5-10 minutes per person.
- Can do it in groups of 1-3.
- Will be held **via Zoom** on Friday May 6th, 10-12pm.
- If you have a request for when to speak please let me know - presentation order will be random subject to those requests.
- Please also upload your presentation to the repository called “Final_Presentation” on Github. (Preferably as a pdf)

- Possible topics:
  - **A topic from your own research (at least tangentially statistical computing)**
  - **A Stats computing topic that we didn’t cover in this course**
  - Sudoku exercise (slides 9 onwards)
  - Tempered MCMC for Gamma rvs. (next slide)
  - Use simulation to tackle one of the “Riddler” problems on slides 7 or 8.
  - Report on a question from your own research that uses statistical computation.
  - Traveling salesperson or 4-color problem
  - Grasshopper problem (slides 18-22)
  - ???

Final project option - Gamma rvs.

- Use Tempered MCMC to produce samples from a Gamma random variable
- Do both Gamma(2.3,2.7) and Gamma(0.1,0.01).
- You must use a perturbation kernel (not an independence sampler). So something like q' = q + Normal(0,σ^2).
- Use the Gelman plots to check convergence.
- Use a qqplot to check that it has converged to a gamma distribution.

Sample Riddler Problem #1

Consider a game of chance called Left, Right, Center. Everyone sits in a circle and begins with some $1 bills. Taking turns, each person rolls three dice. For each die, if you roll a 1 or 2 you give a dollar to the person on your left, if you roll a 3 or 4 you give a dollar to the person on your right, and if you roll a 5 or 6 you put a dollar in the middle. The moment only a single person has any money left, the game ends and that person gets all the money in the center.

How long is the game expected to last for six players each starting with three $1 bills? For X players each starting with Y $1 bills?

Sample "Riddler" Problem #2

You are throwing darts at a dartboard that has a radius of 1 foot. Due to a gift of miraculous marksmanship, your darts always fall on the board and never outside. (Darts *can* land on the very edge of the circular board, and if they do they're considered as landing inside the scoring area.) Furthermore, your chances of hitting any area on the board are exactly proportional to the area of the patch — *your darts land according to a uniform probability distribution*.

You keep throwing darts until your *n*th dart hits a location that is less than 1 foot from some other dart. You are then "out," and n-1 is your final score. Here are three questions of increasing difficulty about this game:

1. What is the maximum possible score?
2. What is the probability of getting a score greater than 1 (i.e., that the second dart falls more than 1 foot away from the first)?
3. What is the expected value of your score?

Is there any theory you can use here?

Sudoku

	7		1		5			
5						3		
		6	7	9		1	2	5
1	5	2	8		6		9	
				3			5	
3				5			8	
8	6			7		5	1	
				6				
		9			2		4	

Sudoku

9	7	3	1	2	5	4	6	8
5	2	1	6	8	4	3	7	9
4	8	6	7	9	3	1	2	5
1	5	2	8	4	6	7	9	3
6	4	8	9	3	7	2	5	1
3	9	7	2	5	1	6	8	4
8	6	4	3	7	9	5	1	2
2	1	5	4	6	8	9	3	7
7	3	9	5	1	2	8	4	6

Sudoku rules

9	7	3	1	2	5	4	6	8
5	2	1	6	8	4	3	7	9
4	8	6	7	9	3	1	2	5
1	5	2	8	4	6	7	9	3
6	4	8	9	3	7	2	5	1
3	9	7	2	5	1	6	8	4
8	6	4	3	7	9	5	1	2
2	1	5	4	6	8	9	3	7
7	3	9	5	1	2	8	4	6

- Each 3x3 square must contain all the numbers 1,2,3,4,5,6,7,8,9.

Sudoku rules

9	7	3	1	2	5	4	6	8
5	2	1	6	8	4	3	7	9
4	8	6	7	9	3	1	2	5
1	5	2	8	4	6	7	9	3
6	4	8	9	3	7	2	5	1
3	9	7	2	5	1	6	8	4
8	6	4	3	7	9	5	1	2
2	1	5	4	6	8	9	3	7
7	3	9	5	1	2	8	4	6

- Each 3x3 square must contain all the numbers 1,2,3,4,5,6,7,8,9.

Sudoku rules

9	7	3	1	2	5	4	6	8
5	2	1	6	8	4	3	7	9
4	8	6	7	9	3	1	2	5
1	5	2	8	4	6	7	9	3
6	4	8	9	3	7	2	5	1
3	9	7	2	5	1	6	8	4
8	6	4	3	7	9	5	1	2
2	1	5	4	6	8	9	3	7
7	3	9	5	1	2	8	4	6

- Each row must contain all the numbers 1,2,3,4,5,6,7,8,9.

Sudoku rules

9	**7**	3	**1**	2	**5**	4	6	8
5	2	1	6	8	4	**3**	7	9
4	8	**6**	**7**	**9**	3	**1**	**2**	**5**
1	**5**	**2**	**8**	4	**6**	7	**9**	3
6	4	8	9	**3**	7	2	**5**	1
3	9	7	2	**5**	1	6	**8**	4
8	**6**	4	3	**7**	9	**5**	**1**	2
2	1	5	4	**6**	8	9	3	7
7	3	**9**	5	1	**2**	8	**4**	6

- Each row must contain all the numbers 1,2,3,4,5,6,7,8,9.

Sudoku rules

9	7	3	1	2	5	4	6	8
5	2	1	6	8	4	3	7	9
4	8	6	7	9	3	1	2	5
1	5	2	8	4	6	7	9	3
6	4	8	9	3	7	2	5	1
3	9	7	2	5	1	6	8	4
8	6	4	3	7	9	5	1	2
2	1	5	4	6	8	9	3	7
7	3	9	5	1	2	8	4	6

- Each column must contain all the numbers 1,2,3,4,5,6,7,8,9.

Sudoku rules

9	7	3	1	2	5	4	6	8
5	2	1	6	8	4	3	7	9
4	8	6	7	9	3	1	2	5
1	5	2	8	4	6	7	9	3
6	4	8	9	3	7	2	5	1
3	9	7	2	5	1	6	8	4
8	6	4	3	7	9	5	1	2
2	1	5	4	6	8	9	3	7
7	3	9	5	1	2	8	4	6

See
"TestSudoku.txt"
"SudokuPlot.R"
on Github

- Each column must contain all the numbers 1,2,3,4,5,6,7,8,9.

Sudoku

- Attack this problem as an optimization problem?
- Test data is uploaded to Github repo “Week5—Sudoku” (“TestSudoku.txt”), along with a function to draw your sudoku to the screen (“SudokuPlot.R”)
- Need to define an ‘energy’ function. (How good a given solution is.) What might you use for this?
- Need to define a rule q(x -> x’), where x,x’ are potential solutions. This is how you will change your solution from iteration to iteration of the optimization algorithm. This is the bit that will probably determine how well your algorithm works.
- Need to choose a configuration from which to start the algorithm.
- Virtually complete “pseudo”-code is on Github “SudokuPseudoCode.R”. You just need to write a function to change the configuration of numbers in the grid.
- Test it on the Sudoku in “TestSudoku.txt” (a relatively easy sudoku)
- Can you solve the ‘moderate’ sudoku I uploaded? (a harder sudoku)

Grasshopper problem

- Informal statement: You are given a bag of grass seed from which you can grow a lawn of any shape (not necessarily connected) with unit area on a planar surface. A grasshopper lands at a random point on your lawn, then jumps a given distance d in a random direction. What lawn shape should you choose to maximise the probability that the grasshopper remains on your lawn after jumping?

- Warning: this is a very hard problem! So have fun with it, but don't expect to solve it (and neither will I expect you to do so). Full credit is available for "noble efforts"

Grasshopper problem

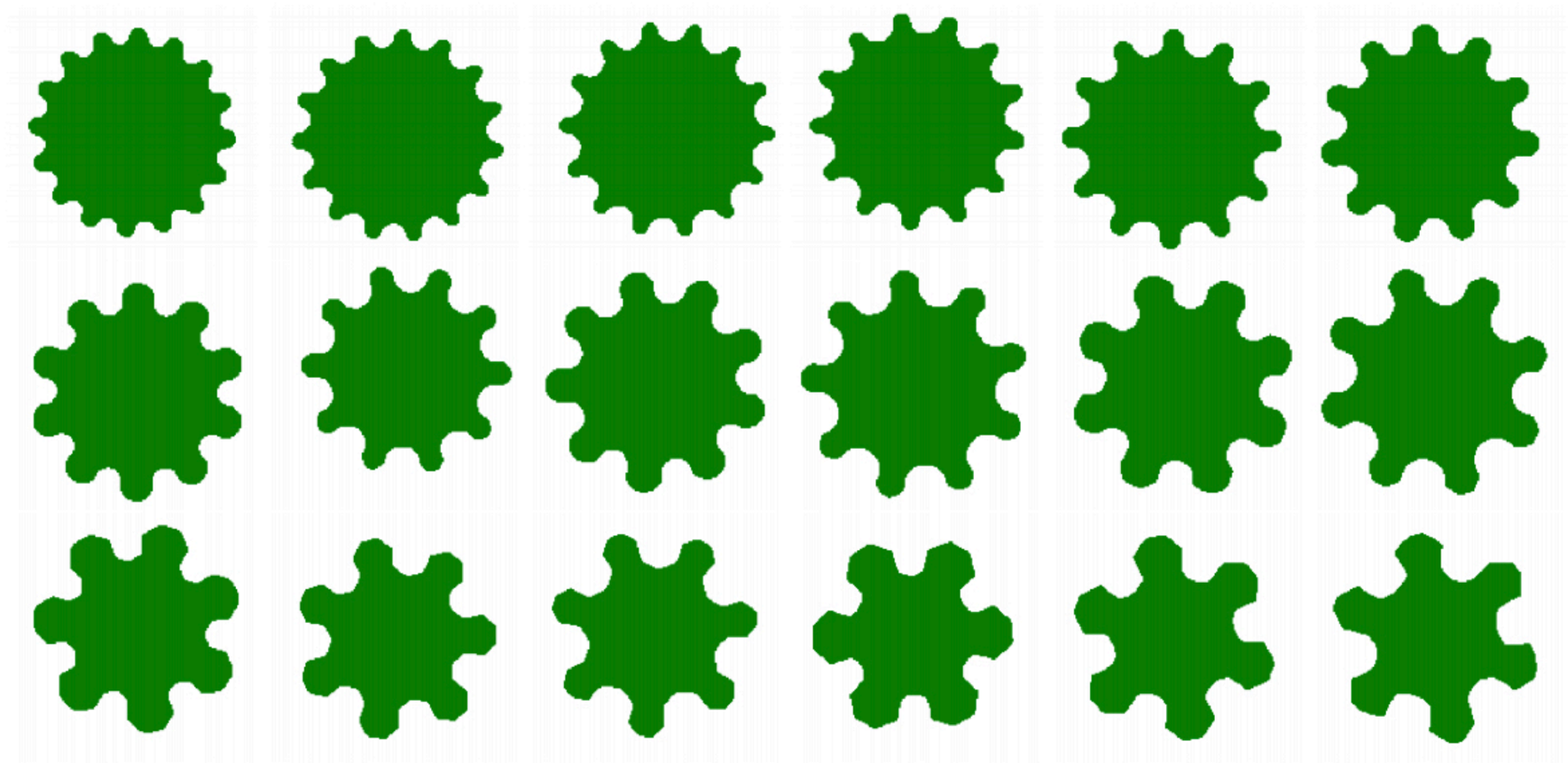


FIG. 3. Best configurations found for d ranging from 0.22 (top left) to 0.56 (bottom right), where d increases by 0.02 in each subsequent panel going first from left to right and then from top to bottom.

- https://arxiv.org/pdf/1705.07621.pdf

Grasshopper problem

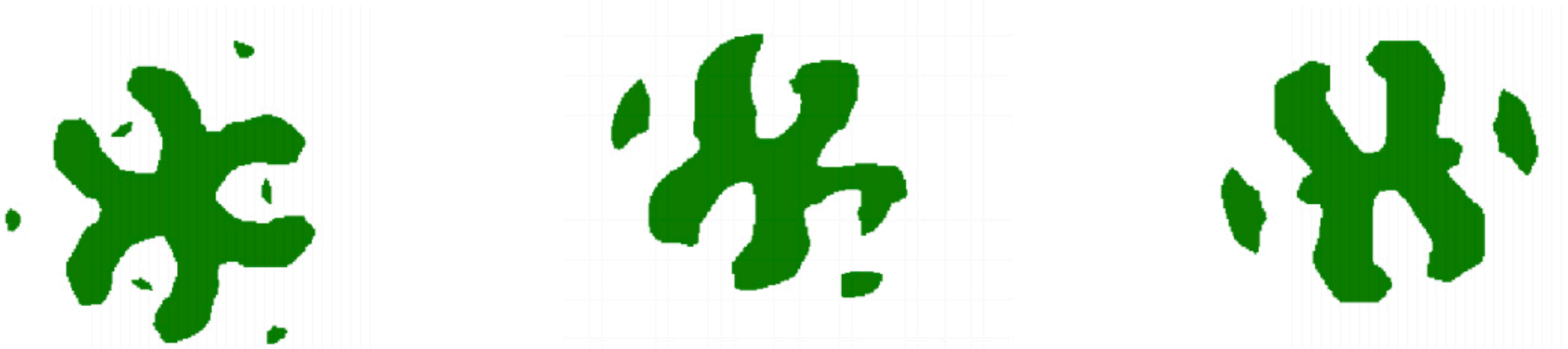


FIG. 9. Three types of shape that were found to be optimal for $0.58 \leq d \leq 0.64$. All shapes are disconnected. Left panel: Best configuration found for $d = 0.58$ (the same type of shape was also the best found for $d = 0.59$). Middle panel: Best configuration found for $d = 0.60$. Its shape appears to lack any symmetry. The same type of shape was the best found for $d = 0.61$ (in most discretization setups) and found to be nearly optimal for a range of d between 0.58 and 0.61. Right panel: Best configuration found for $d = 0.62$. The same type of shape was the best found for $0.62 \leq d \leq 0.64$ and found to be nearly optimal for a range of d between 0.59 and 0.66.

- https://arxiv.org/pdf/1705.07621.pdf

Grasshopper problem

C. Three-bladed fan regime

For $0.65 \leq d \leq 0.87$ the best configurations found are shaped like a three-bladed "fan" with additional patches between the blades, as shown in Fig. 10. These configurations have three-fold rotational and mirror symmetries, corresponding to the dihedral group D_3. Starting from approximately $d = 0.78$ a hole starts to develop in the centre of the fan. The hole becomes larger with increasing d. The three-bladed fan was also occasionally generated as an (apparently) nearly optimal configuration at larger values of d.

FIG. 10. Best configurations found for $d = 0.65$, 0.72, 0.78, 0.82, 0.87 (left to right).

- https://arxiv.org/pdf/1705.07621.pdf

Grasshopper problem

FIG. 11. The best configuration found for $d = 0.9$ consists of four stripes (left). The same type of shape was the best found for $d \geq 0.89$ and up to the largest d considered in this study. The five striped configuration (right) has lower $P_{\{s\}}(d)$.

- https://arxiv.org/pdf/1705.07621.pdf

ABC MCMC

MCMC: Metropolis-Hastings

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Calculate

$$h = \min\left\{1, \frac{P(D \mid \theta')\pi(\theta')q(\theta' \to \theta)}{P(D \mid \theta)\pi(\theta)q(\theta \to \theta')}\right\}$$

3. Move to θ' with prob. h, else remain at θ.
4. Return to 1.

Result: correlated samples from f(θ |D)

(Metropolis et al. 1953, Hastings 1970)

MCMC without likelihoods

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Generate data D′ using θ'.
3. If D'=D go to 4. Else, stay at θ and go to 1.
4. Calculate

$$h = \min\left\{1,\ \frac{\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right\}$$

5. Move to θ' with prob. h, else remain at θ.
6. Return to 1.

Result: correlated samples from f(θ |D)

[Marjoram, Molitor, Plagnol and **Tavaré** 2003]

MCMC without likelihoods - 1

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Generate data D′ using θ'.
3. If D'=D go to 4. Else, stay at θ and go to 1.
4. Calculate

$$h = \min\left\{1,\ \frac{\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right\}$$

5. Move to θ' with prob. h, else remain at θ.
6. Return to 1.

Result: correlated samples from f(θ |D)

MCMC without likelihoods -1

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Generate data D′ using θ'.
3. If D'~D go to 4. Else, stay at θ and go to 1.
4. Calculate

$$h = \min\left\{1,\ \frac{\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right\}$$

5. Move to θ' with prob. h, else remain at θ.
6. Return to 1.

Result: correlated samples from φ (θ |D) ≠ f(θ |D)

MCMC without likelihoods - 1 (restated)

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Generate data D′ using θ'.
3. Calculate

$$h = \min\left\{1, \frac{I_{\theta'}(D = D')\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right\}$$

where $I_\theta()$ is the Indicator function (when the data is generated using θ).

4. Move to θ' with prob. h, else remain at θ.
5. Return to 1.

Result: correlated samples from f(θ |D)

MCMC without likelihoods - 1 (restated)

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Generate data D′ using θ'.
3. Calculate

$$h = \min\left\{1, \frac{I_{\theta'}(D = D')\pi(\theta')q(\theta' \to \theta)}{I_{\theta}(D = D')\pi(\theta)q(\theta \to \theta')}\right\}$$

where $I_\theta()$ is the Indicator function (when the data is generated using θ).

4. Move to θ' with prob. h, else remain at θ.
5. Return to 1.

Result: correlated samples from f(θ |D)

Note: **MUST** re-use the value of $I_\theta()$ that was used when accepting θ

MCMC without likelihoods - 1 (restated)

In general, any unbiased estimator of the likelihood, $\mathcal{E}()$ say, can be used.

1. If at θ, propose move to θ' according to transition kernel q(θ -> θ').
2. Generate data D′ using θ'.
3. Calculate

$$h = \min\left\{1, \frac{\mathcal{E}(D \mid \theta')\pi(\theta')q(\theta' \to \theta)}{\mathcal{E}(D \mid \theta)\pi(\theta)q(\theta \to \theta')}\right\}$$

where $I_\theta()$ is the Indicator function (when the data is generated using θ).

4. Move to θ' with prob. h, else remain at θ.
5. Return to 1.

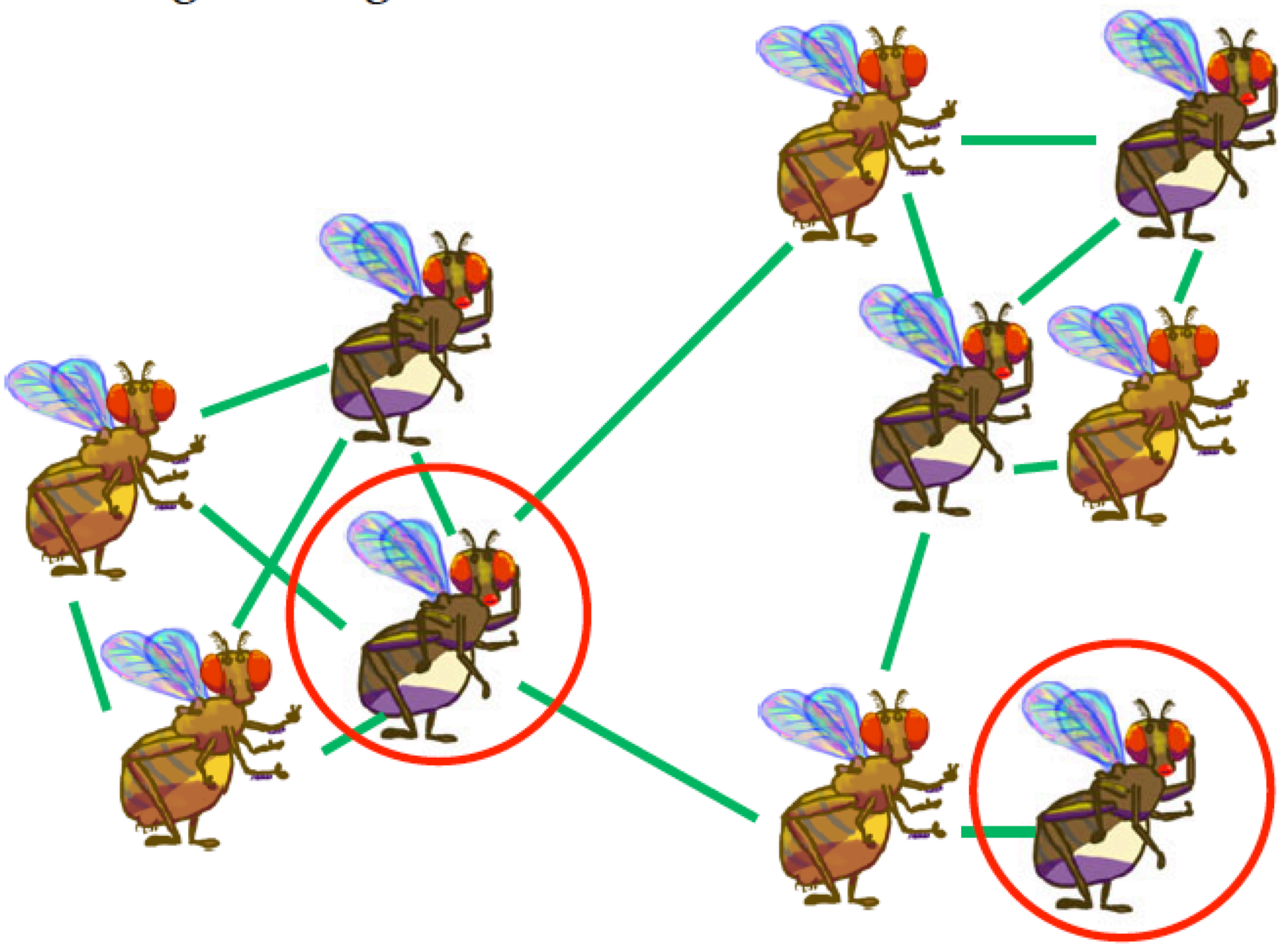
Result: correlated samples from f(θ |D)

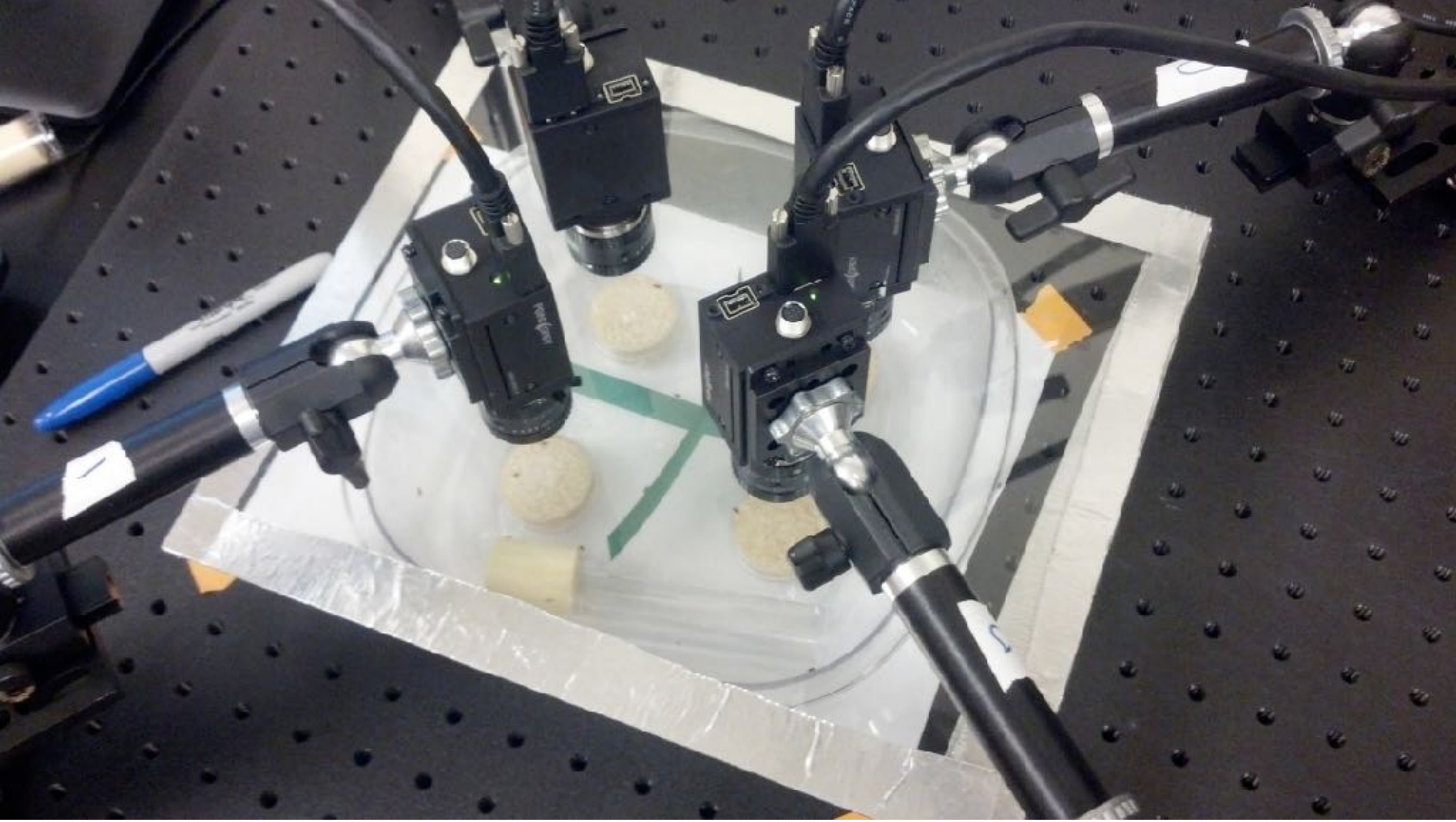
Note: **MUST** re-use the value of $\mathcal{E}$(D | θ) that was used when accepting θ.

[Beaumont et al 2003]

Example Applications

Measuring social genetics





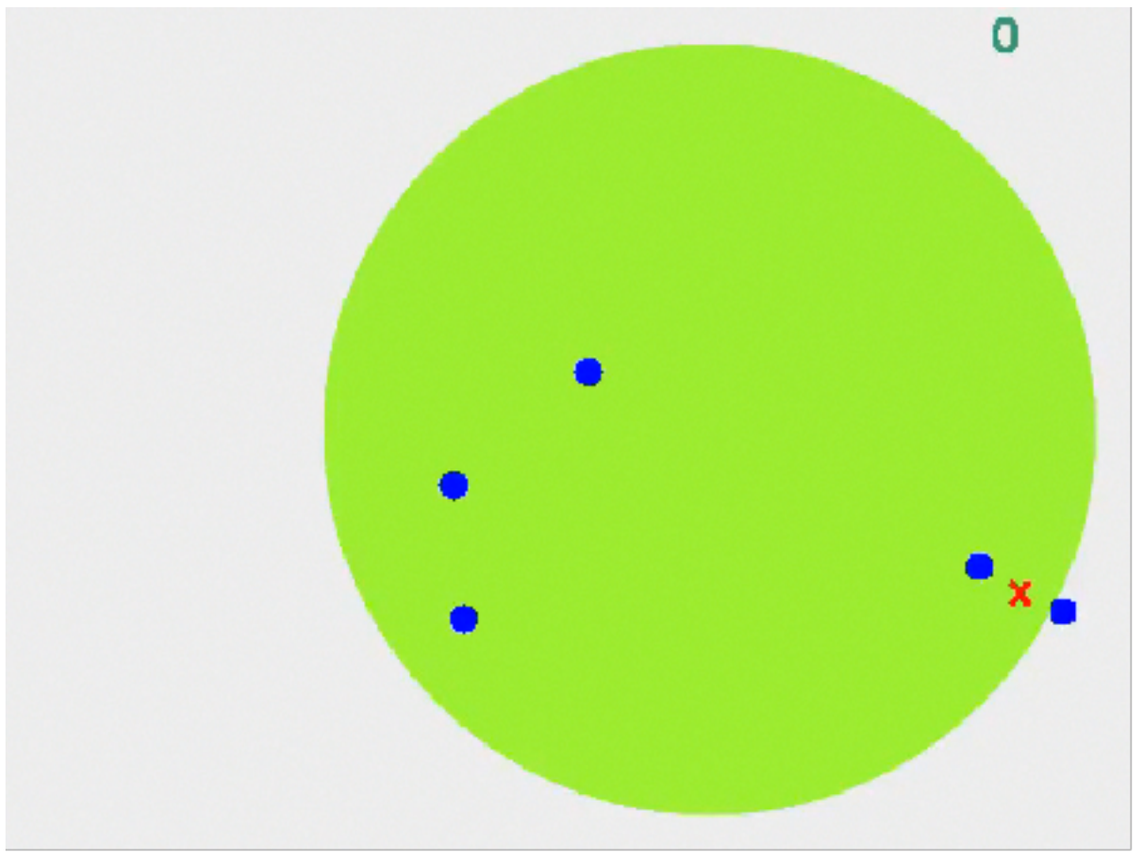
Data collection:

Past: 1 very bored student,

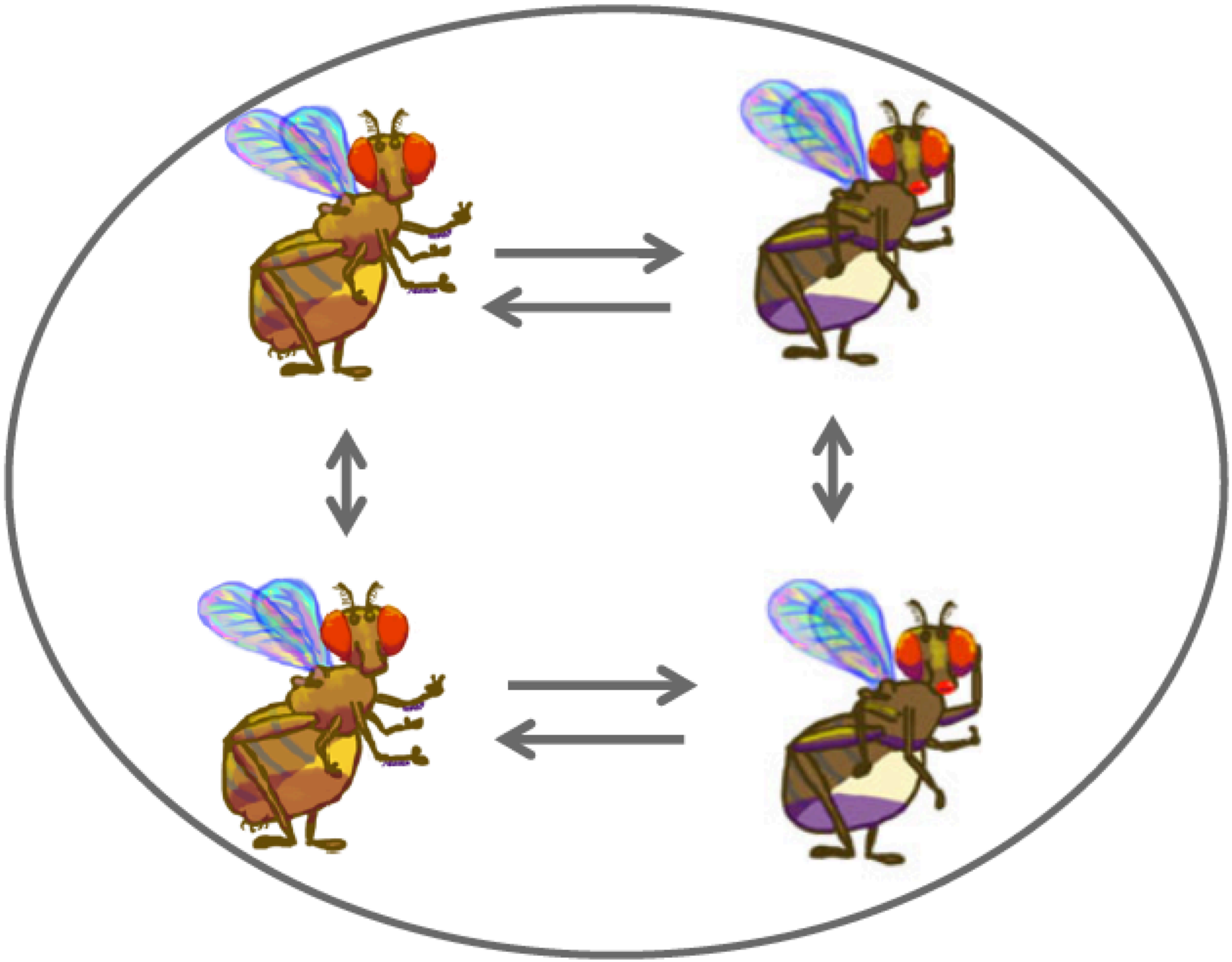
Present: 1 camera plus some smart software

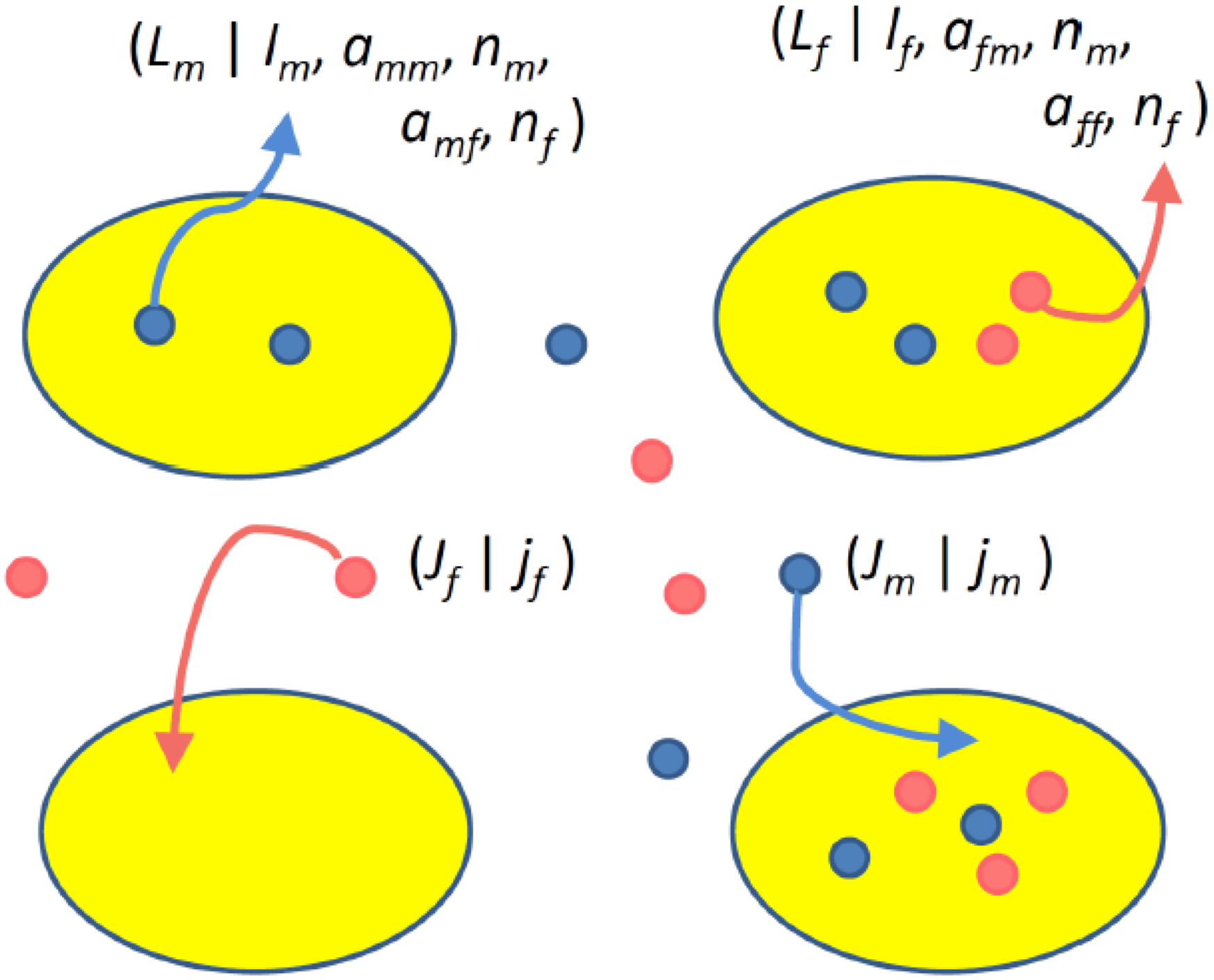

0
(414,259)
(318,340)
(692,398)
(325,435)


0
(414,250)
(318,340)
(692,398)
(325,435)


0

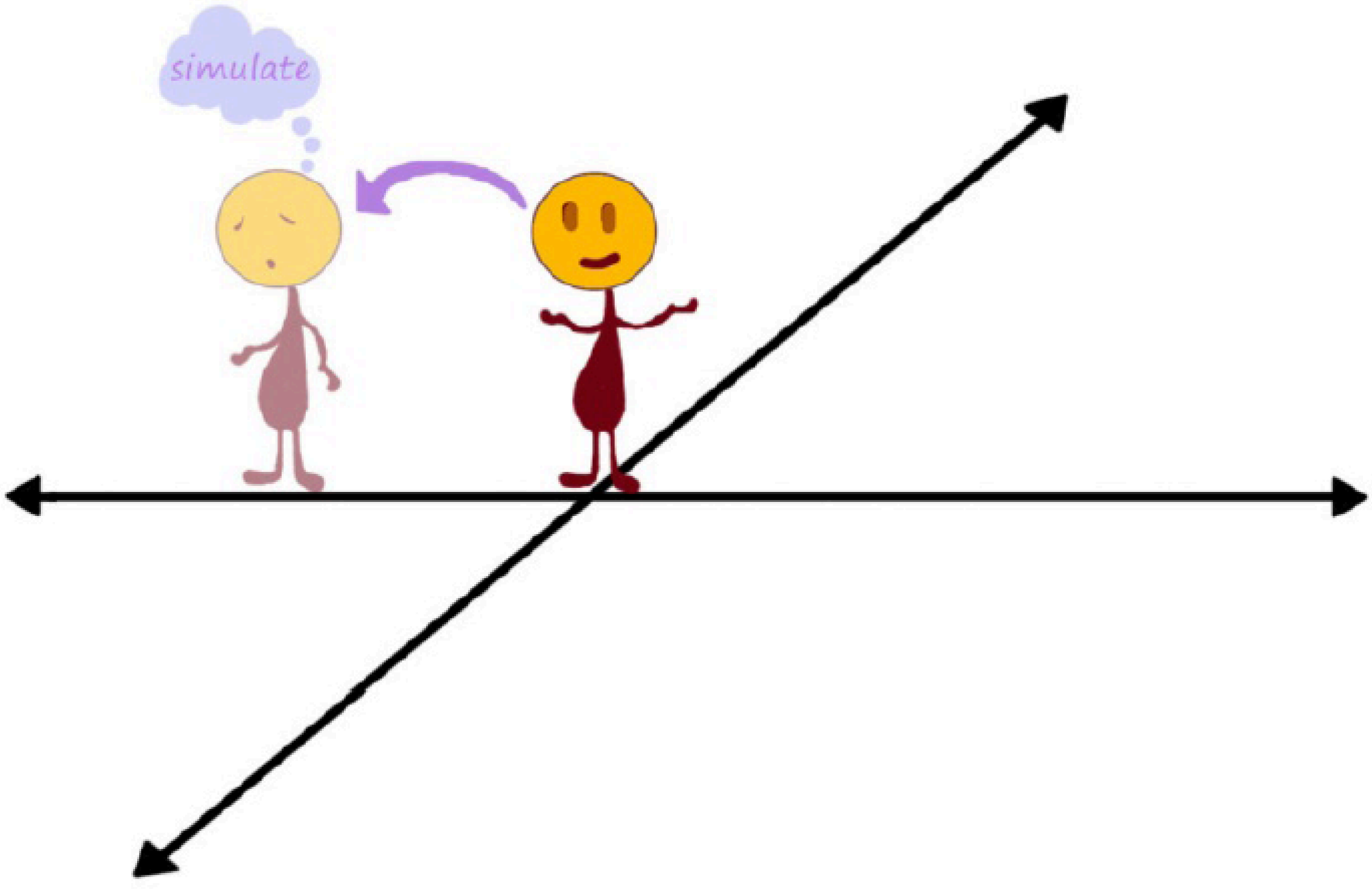

leave!
stay.
leave!
stay.
join!
join!
stay.

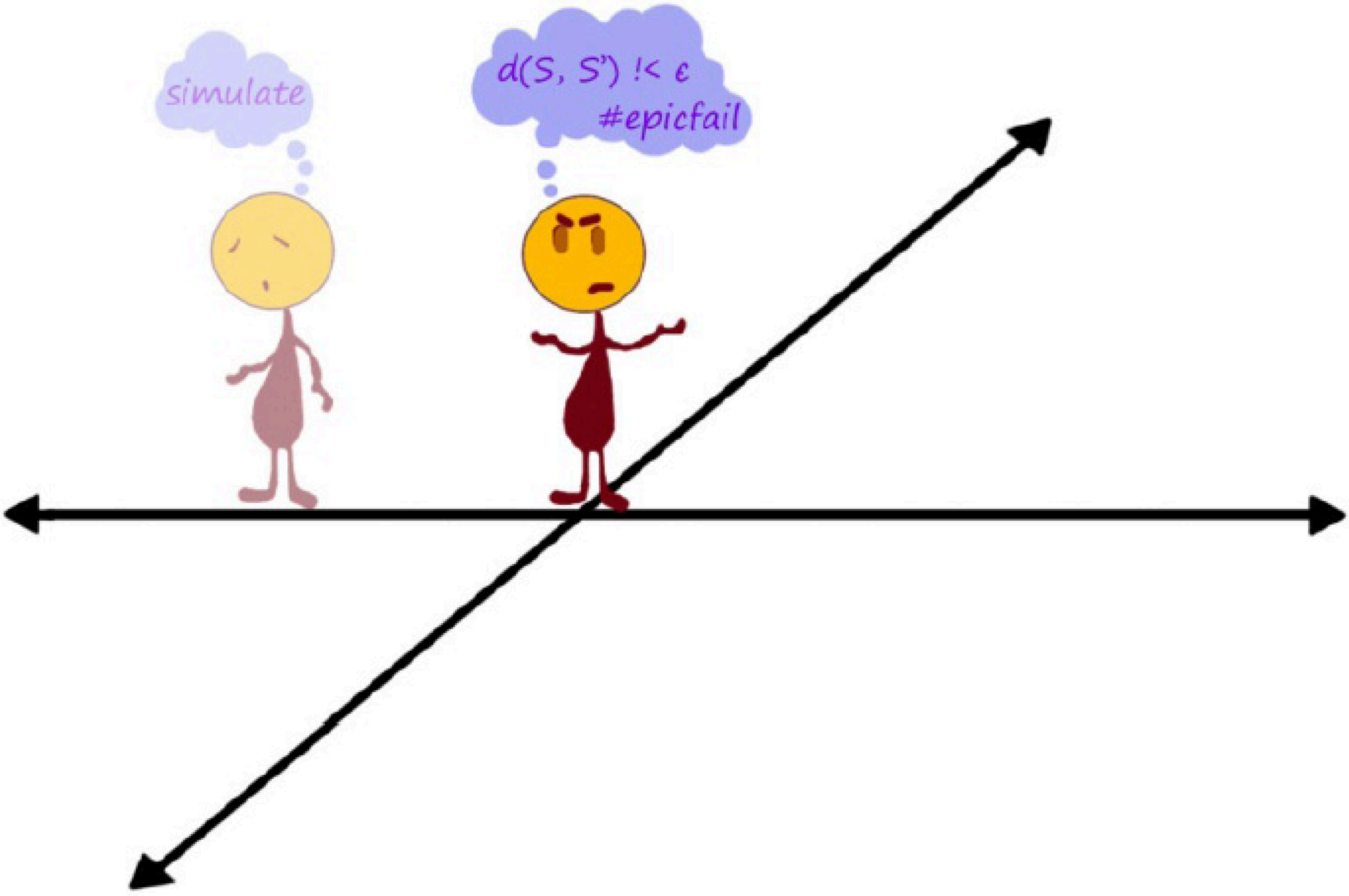


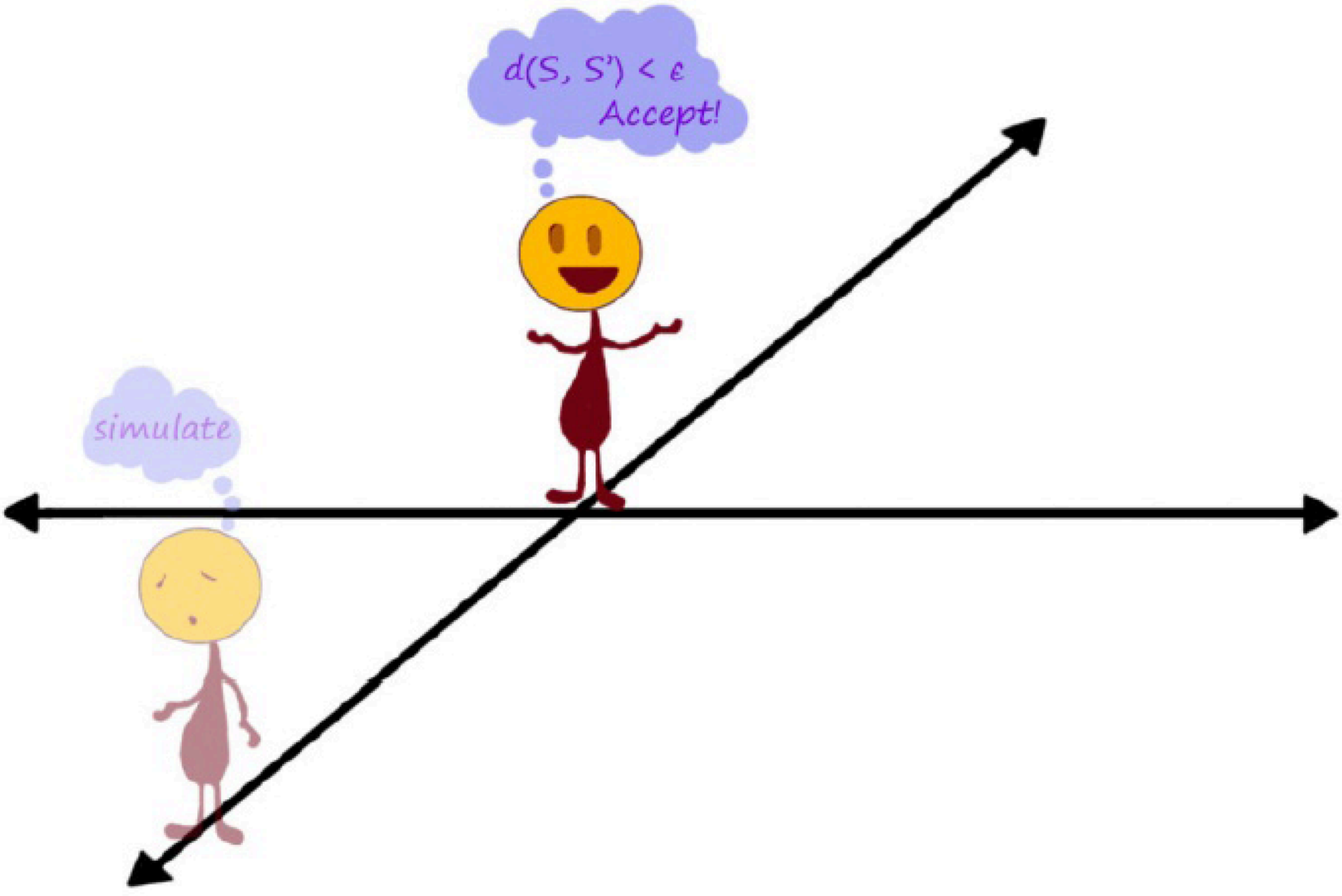

$(L_m \mid l_m, a_{mm}, n_m, a_{mf}, n_f)$
$(L_f \mid l_f, a_{fm}, n_m, a_{ff}, n_f)$
$(J_f \mid j_f)$
$(J_m \mid j_m)$

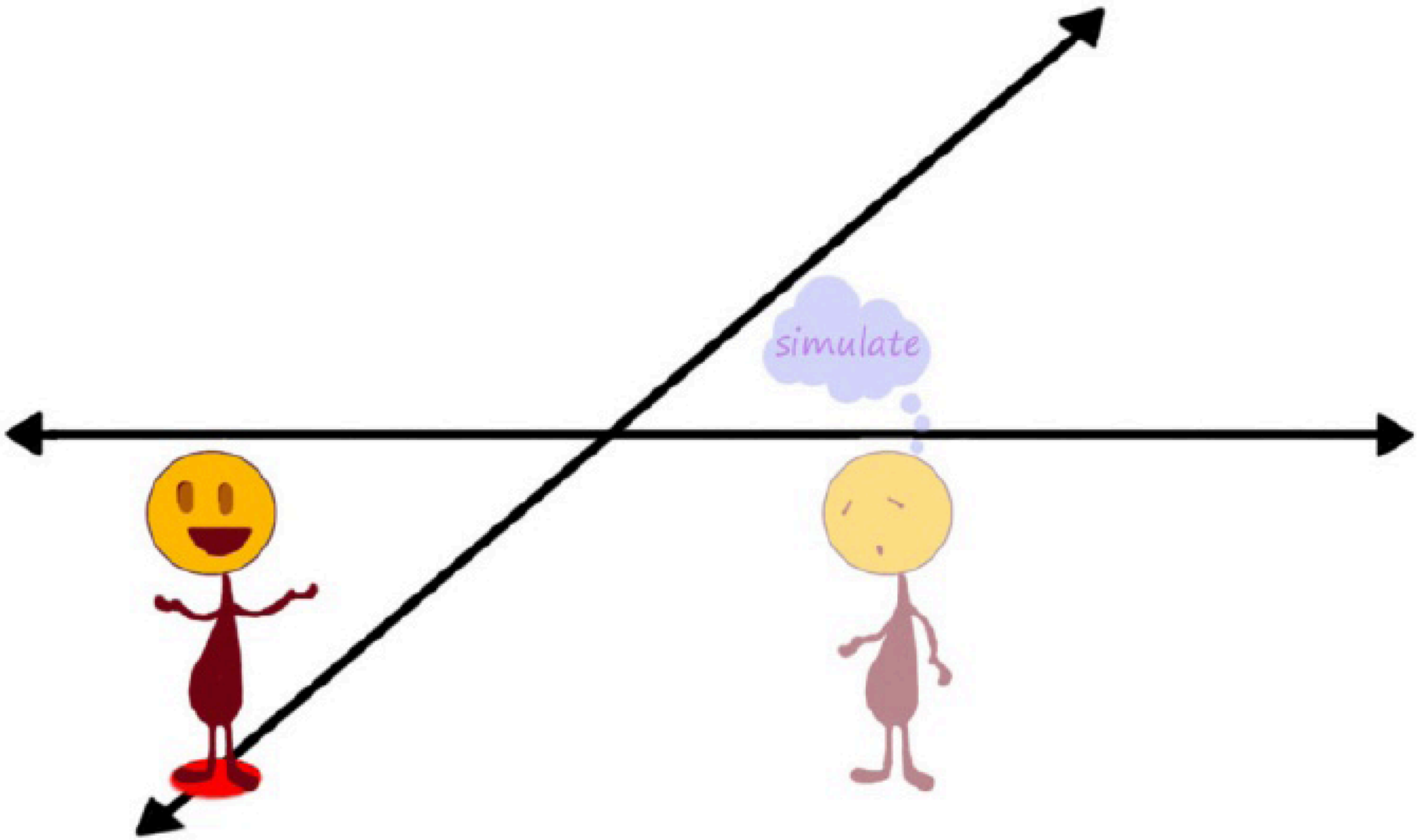
Summary statistics

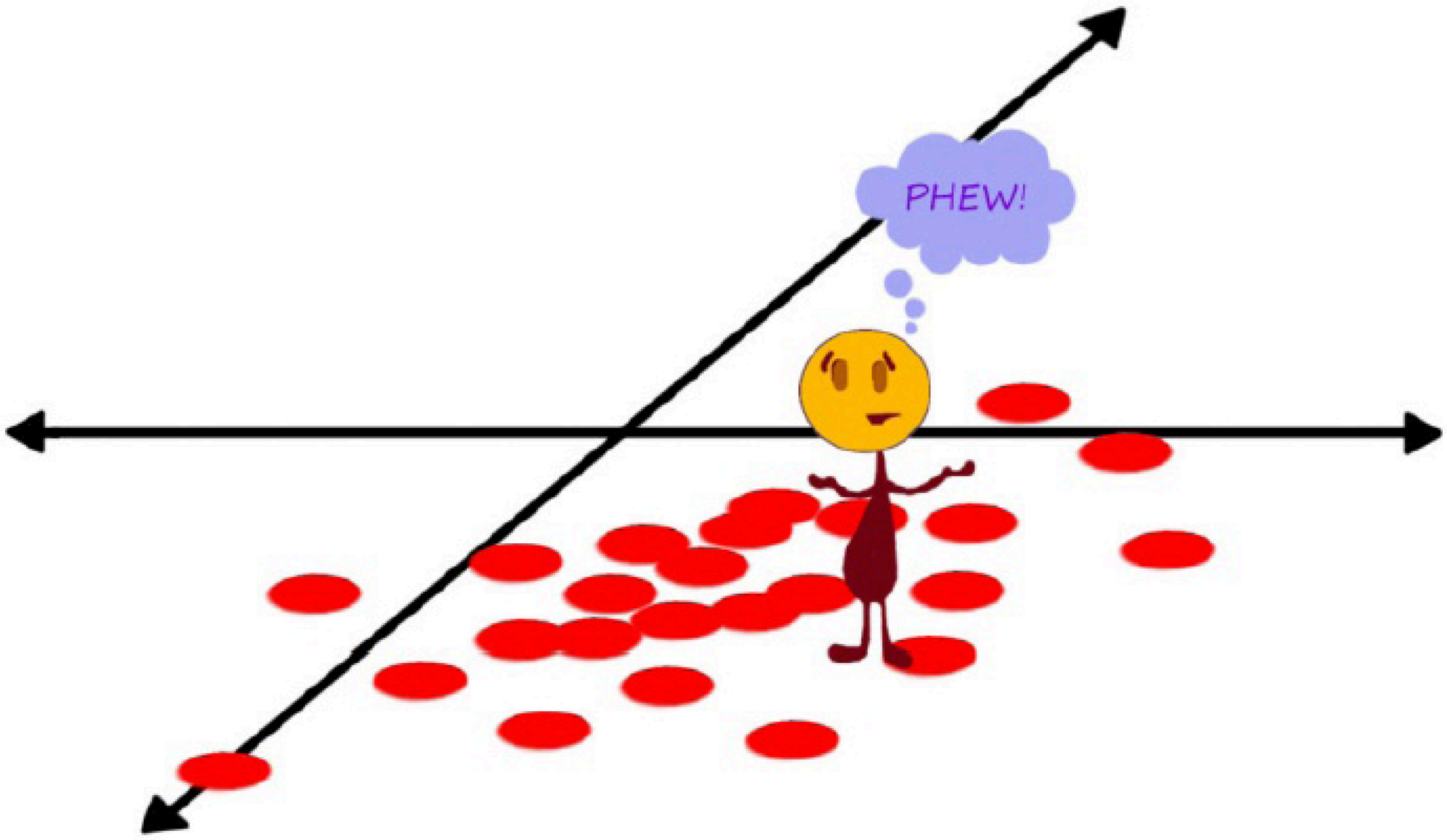
Statistic	Description
mAv/fAv	Average number of males, or females, in groups
mAvSD/fAvSD	Sampling variance in mAv, or fAv
mVar/fVar	Variance in the number of males, or females, among groups
mfCor	Correlation in the number of males and females among groups
mPerM/fPerM /mPerF/fPerF	The average number of (other) males or females each male, or female, experiences in their group.

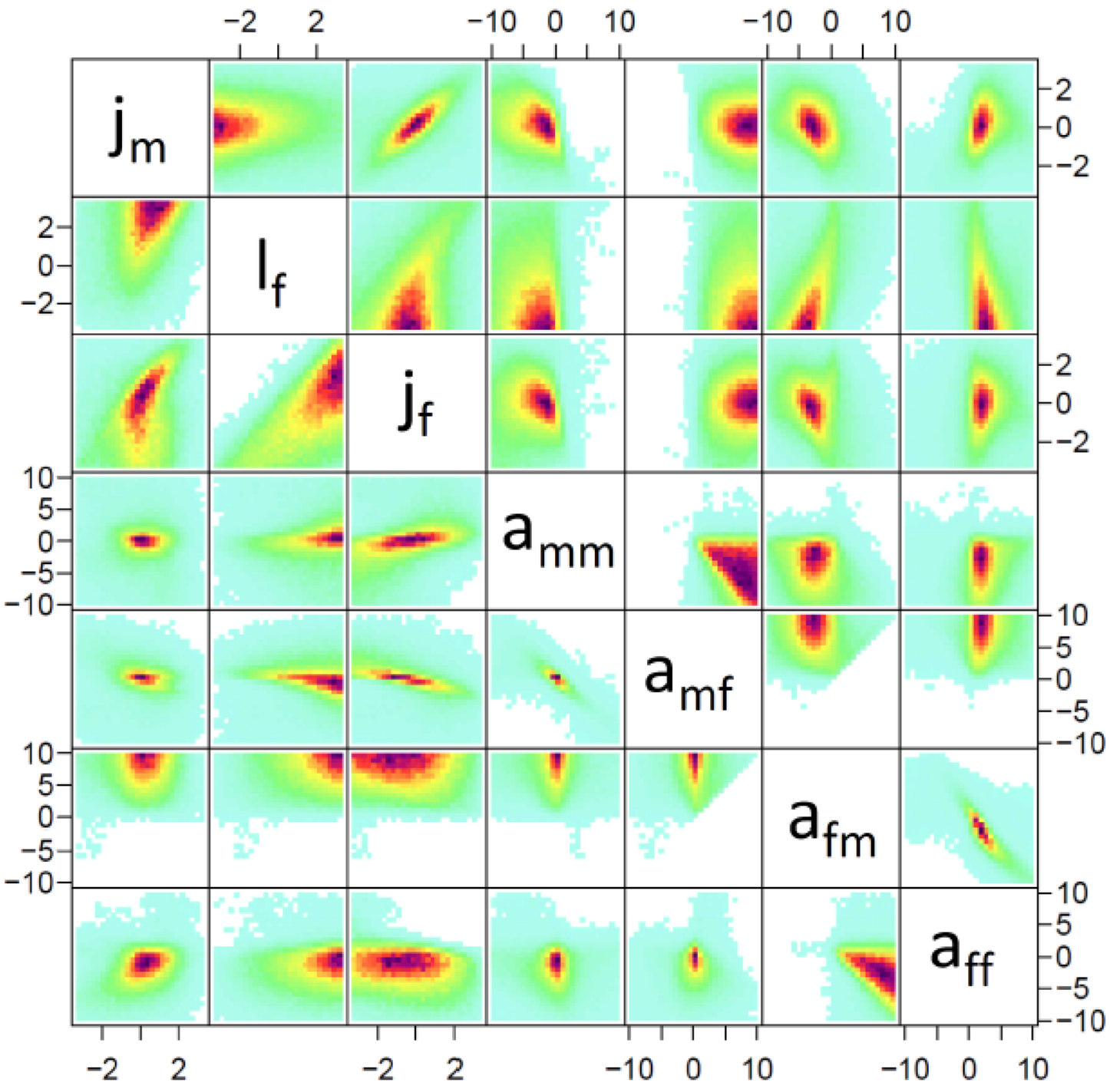

simulate


simulate
d(S, S') !< ε
#epicfail

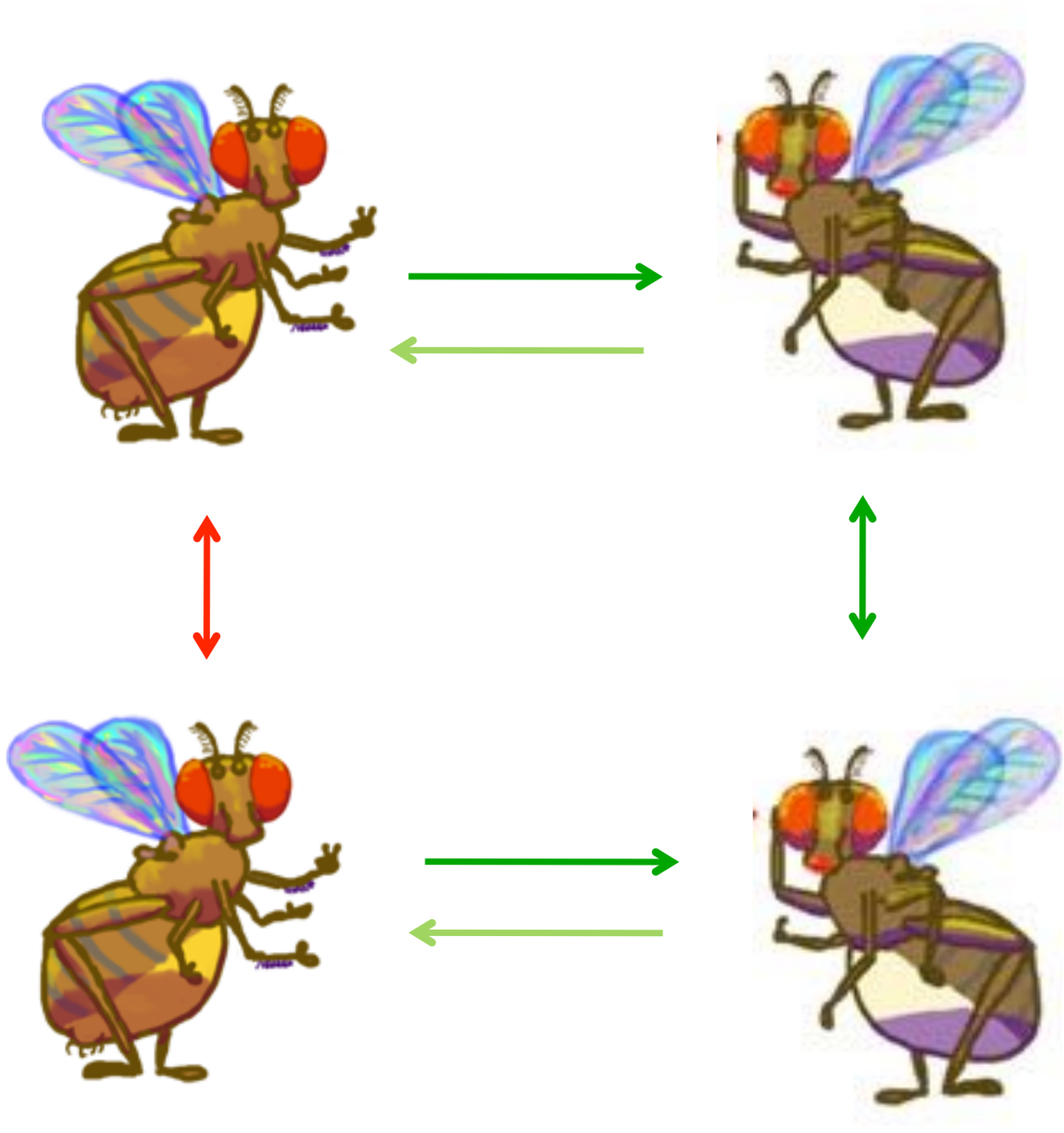

d(S, S') < ε
Accept!
simulate


simulate


PHEW!

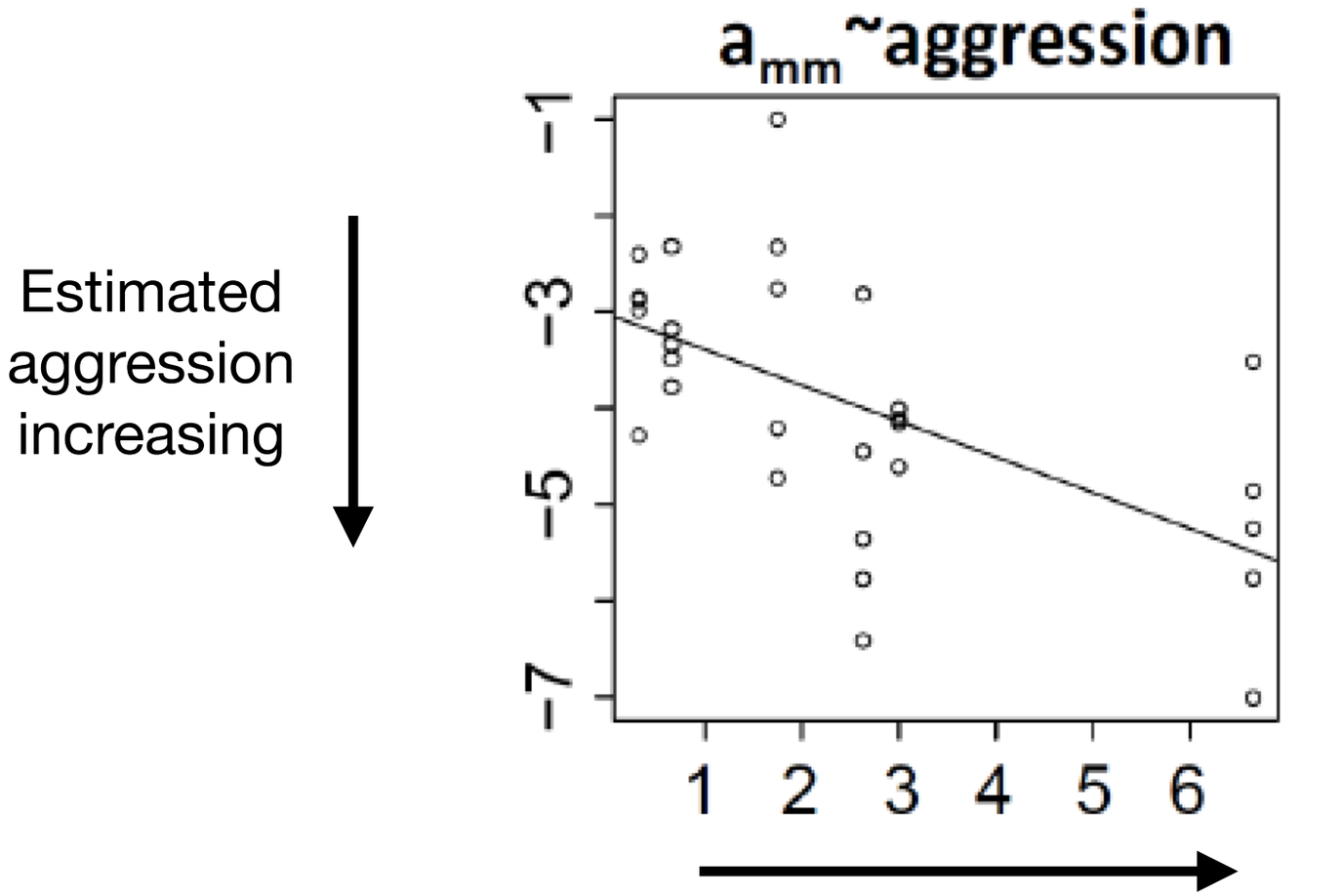

−2
2
−10
0
10
−10
0
10
j_m
l_f
j_f
a_{mm}
a_{mf}
a_{fm}
a_{ff}
2
0
−2
10
5
0
−5
−10

Results - Overview

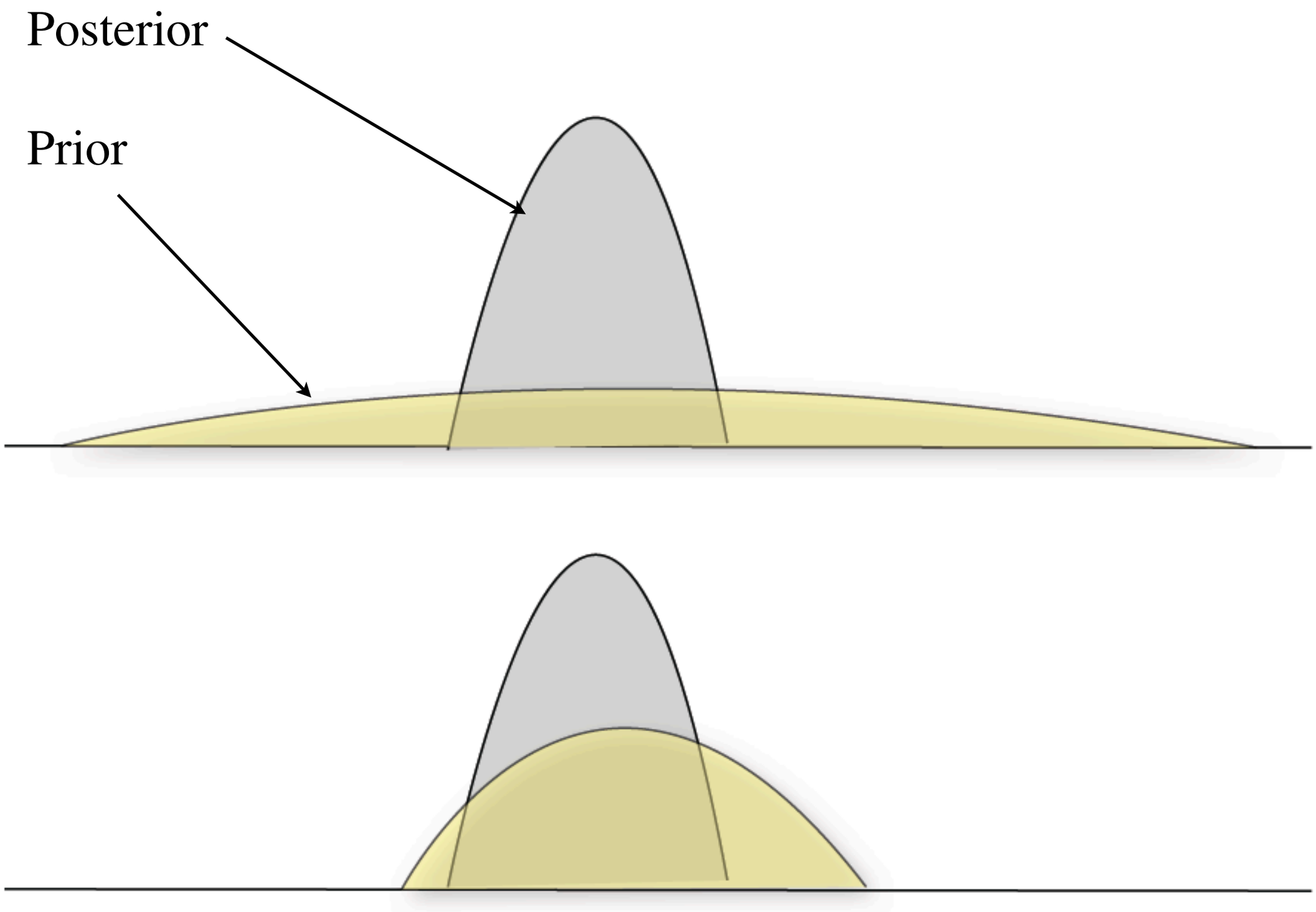


Brad R. Foley, Julia B. Saltz, Sergey V. Nuzhdin, and Paul Marjoram, "A Bayesian Approach to Social Structure Uncovers Cryptic Regulation of Group Dynamics in *Drosophila melanogaster*," The American Naturalist 185, no. 6 (June 2015): 797-808.

Fit to prior knowledge?



ABC Sequential Monte Carlo [ABC Particle Filters]


Posterior
Prior

Rejection methods are inefficient if the prior and posterior are very different.

This is the motivation for using importance samplers (or MCMC)

What if we don’t have a good importance sampling distribution?....

Sequential Monte Carlo ABC [SMC-ABC]

Importance Sampling

- Parameter θ, prior π, importance sampling dist^n ξ, observed data D.
- Draw θ' from ξ and accept with prob. P(D | θ').
- If we accept θ', then add a mass of $\pi(\theta')/\xi(\theta')$ to the posterior at θ'.
- Result: f(θ | D).
- Rationale: try to use a ξ that is closer to the posterior f() than the prior π.
- Goal: More efficient sampler.

Importance Sampling

1. Parameter θ, prior π, importance sampling dist^n ξ, observed data D.
2. Draw θ' from ξ and simulate data D'. Accept θ' if D′=D (or S′=S, for summary stats S).
3. If we accept θ', then add a mass of $\pi(\theta')/\xi(\theta')$ to the posterior at θ'.
4. Return to 1.

Result: $f(\theta \mid D)$ [or $f(\theta \mid S)$].

Rationale: try to use a ξ that is closer to the posterior f() than the prior π.

Goal: More efficient sampler.

Sequential Monte Carlo: Intuition

- A form of importance sampling
- Moves through a series of ‘generations’
- Each generation has a set of N particles
- At each generation, perturb the N particles in the previous generation
  - Do it in a way such that the distribution of those particles will tend towards the posterior as we move through the generations.
- Note, this does not require us to come up with an importance sampling distribution.
- Goes by many names, e.g.,…“Particle Filtering”, “Sequential Importance Sampling”,…

SMC samplers - more formally:

Goal, to construct a sequence of distributions $f_t(x)$, $t = 1, \ldots, T$, with $x \in E$ (some space), where the final distribution f_T is some distribution of interest. (e.g. a posterior distribution, or the true location or path of a set of particles)

Start by sampling x_1, a set of N particles from f_1.
There are two components:
Mutation operator: $M_{t-1}(x_{t-1} \rightarrow x_t)$ [Note that this operator may depend on t]
Correction: particles will be re-weighted with respect to f_t (an importance weight)

We are using the empirical distribution obtained by observing the final N particles to approximate the desired distribution.

Del Moral, P., Doucet, A., Jasra, A.: Sequential Monte Carlo samplers. J. R. Stat. Soc. B **68**, 411–436 (2006)
Peters, G. W., Fan, Y., & Sisson, S. A. (2012). On sequential Monte Carlo, partial rejection control and approximate Bayesian computation. *Statistics and Computing*, *22*(6), 1209–1222. doi:10.1007/s11222-012-9315-y

In more detail:

- Start by sampling x_1, a set of N particles from f_1. (We might use the prior for f_1, say)
- Suppose that at time $t-1$, the distribution f_{t-1} can be approximated empirically by $g_{N,t-1}$, the empirical distribution formed from the N weighted particles.
  - First: these particles are first propagated to the next distribution f_t (approximated by $g_{N,t}$) using the mutation kernel $M_t(x_{t-1}, x_t)$
  - Second: Assign new weights $W_t = W_{t-1} \times w_t(x_{t-1})$, where W_{t-1} is the weight of a particle at time $t-1$ and w_t is the incremental weight (a function of the mutation operator).

- Performance is heavily dependent on the mutation operator M (just as with MCMC and the transition/proposal kernel q).
- M poorly chosen means perturbed x_t don't often fall into points well-supported by f_t
- Result: If so, the system degenerates into a few particles with high weight (rest have low weight) and the empirical distribution is a poor approximation of f_t.

Del Moral, P., Doucet, A., Jasra, A.: Sequential Monte Carlo samplers. J. R. Stat. Soc. B **68**, 411–436 (2006)
Peters, G. W., Fan, Y., & Sisson, S. A. (2012). On sequential Monte Carlo, partial rejection control and approximate Bayesian computation. *Statistics and Computing*, *22*(6), 1209–1222. doi:10.1007/s11222-012-9315-y

ABC SMC - Formally

Given a decreasing sequence of tolerance thresholds $\varepsilon_1, \ldots, \varepsilon_T$, observed data y, summary statistics S, distance metric ρ and model parameters θ (with prior distribution π):

1. At iteration t = 1,
for i =1,...,N,

until $\rho(S(x), S(y)) < \varepsilon_t$, sample $\theta_i^{(1)} \sim \pi(\theta)$ and simulate $x \sim p(x|\theta_i^{(1)})$.

Set $\omega_i^{(1)} = 1/N$.

Set τ^2_2 = twice the empirical variance of the $\theta_i^{(1)}$'s.

2. At iteration $2 \le t \le T$,
for i =1,...,N,

until $\rho(S(x), S(y)) < \varepsilon_t$

pick θ_i^* from the $\theta_j^{(t-1)}$'s with probabilities $\omega_j^{(t-1)}$;

generate $\theta_i^{(t)} \sim K(\theta|\theta_i^*;\tau_t^2)$ and $x \sim p(x|\theta_i^{(t)})$.

Set $\omega_i^{(t)} \propto \pi(\theta_i^{(t)}) / \Sigma_{j=1,\ldots,N}\, \omega_j^{(t-1)} K(\theta_i^{(t)} | \theta_j^{(t-1)};\tau_t^2)$.

Take τ^2_{t+1} as twice the weighted empirical variance of the $\theta^{(t)}$'s.

Beaumont, M. A. (2010). Approximate Bayesian Computation in Evolution and Ecology. *Annual Review of Ecology, Evolution, and Systematics*, *41*(1), 379–406. doi:10.1146/annurev-ecolsys-102209-144621

Toni T, Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, *6*, 187–202.

ABC SMC - Formally

Given a decreasing sequence of tolerance thresholds $\varepsilon_1, \ldots, \varepsilon_T$, observed data y, summary statistics S, distance metric ρ and model parameters θ (with prior distribution π):

1. At iteration t = 1,
for i =1,...,N,

sampled param value

simulated data

until $\rho(S(x), S(y)) < \varepsilon_t$, sample $\theta_i^{(1)} \sim \pi(\theta)$ and simulate $x \sim p(x|\theta_i^{(1)})$.

Set $\omega_i^{(1)} = 1/N$. ← the weight of the accepted param. value

Set τ^2_2 = twice the empirical variance of the $\theta_i^{(1)}$'s.

2. At iteration $2 \le t \le T$,
for i =1,...,N,
until $\rho(S(x), S(y)) < \varepsilon_t$

The perturbation:
e.g., K might be a Normal(θ_j, τ_t^2) distribution

Importance sampling weight

pick θ_i^* from the $\theta_j^{(t-1)}$'s with probabilities $\omega_j^{(t-1)}$;

generate $\theta_i^{(t)} \sim K(\theta|\theta_i^*;\tau_t^2)$ and $x \sim p(x|\theta_i^{(t)})$.

Set $\omega_i^{(t)} \propto \pi(\theta_i^{(t)}) / \Sigma_{j=1,\ldots,N} \omega_j^{(t-1)} K(\theta_i^{(t)} | \theta_j^{(t-1)};\tau_t^2)$.

Take τ^2_{t+1} as twice the weighted empirical variance of the $\theta^{(t)}$'s.

Beaumont, M. A. (2010). Approximate Bayesian Computation in Evolution and Ecology. *Annual Review of Ecology, Evolution, and Systematics*, *41*(1), 379–406. doi:10.1146/annurev-ecolsys-102209-144621

Toni T, Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, *6*, 187–202.

Adding partial rejection control. Informally:

- Start by sampling x_1, a set of N particles from f_1.
- Suppose that at time $t-1$, the distribution f_{t-1} can be approximated empirically by $g_{N,t-1}$ using N weighted particles.
  - First: these particles are first propagated to the next distribution f_t (approximated by $g_{N,t-1}$) using the mutation kernel $M_t(x_{t-1}, x_t)$
  - Second: Assign new weights $W_t = W_{t-1} w_t(x_{t-1})$, where W_{t-1} is the weight of a particle at time $t-1$ and w_t is the incremental weight (a function of the mutation operator).
  - Third: If the weight of a particle at distribution f_t falls below a finite threshold, $c_t/N > 0$, with some probability the particle is discarded and a new particle is generated from f_{t-1} (repeat until weight of new particle is $> c_{t,N}$).

For more details see:

Del Moral, P., Doucet, A., Jasra, A.: Sequential Monte Carlo samplers. J. R. Stat. Soc. B **68**, 411–436 (2006)

Peters, G. W., Fan, Y., & Sisson, S. A. (2012). On sequential Monte Carlo, partial rejection control and approximate Bayesian computation. *Statistics and Computing*, *22*(6), 1209–1222.

Application

- The repressilator (Elowitz & Leibler 2000) - a popular toy model for gene regulatory systems
  - Consists of 3 genes connected in a feedback loop
  - Each gene represses the next gene in the loop, and is repressed by the previous gene.
  - The model consists of six ordinary differential equations and four parameters.

Toni T, Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, *6*, 187–202.

Application

$$\frac{\mathrm{d}m_1}{\mathrm{d}t} = -m_1 + \frac{\alpha}{1+p_3^n} + \alpha_0, \qquad (3.7a)$$

$$\frac{\mathrm{d}p_1}{\mathrm{d}t} = -\beta(p_1 - m_1), \qquad (3.7b)$$

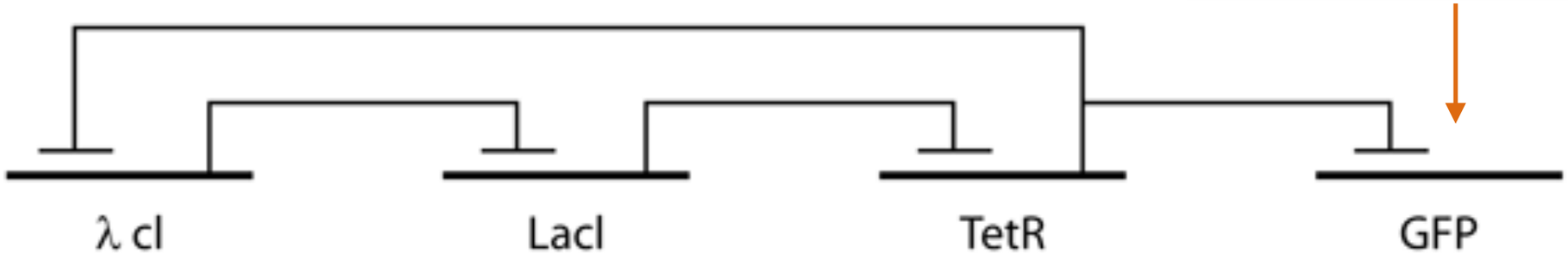
$$\frac{\mathrm{d}m_2}{\mathrm{d}t} = -m_2 + \frac{\alpha}{1+p_1^n} + \alpha_0, \qquad (3.7c)$$

$$\frac{\mathrm{d}p_2}{\mathrm{d}t} = -\beta(p_2 - m_2), \qquad (3.7d)$$

$$\frac{\mathrm{d}m_3}{\mathrm{d}t} = -m_3 + \frac{\alpha}{1+p_2^n} + \alpha_0, \qquad (3.7e)$$

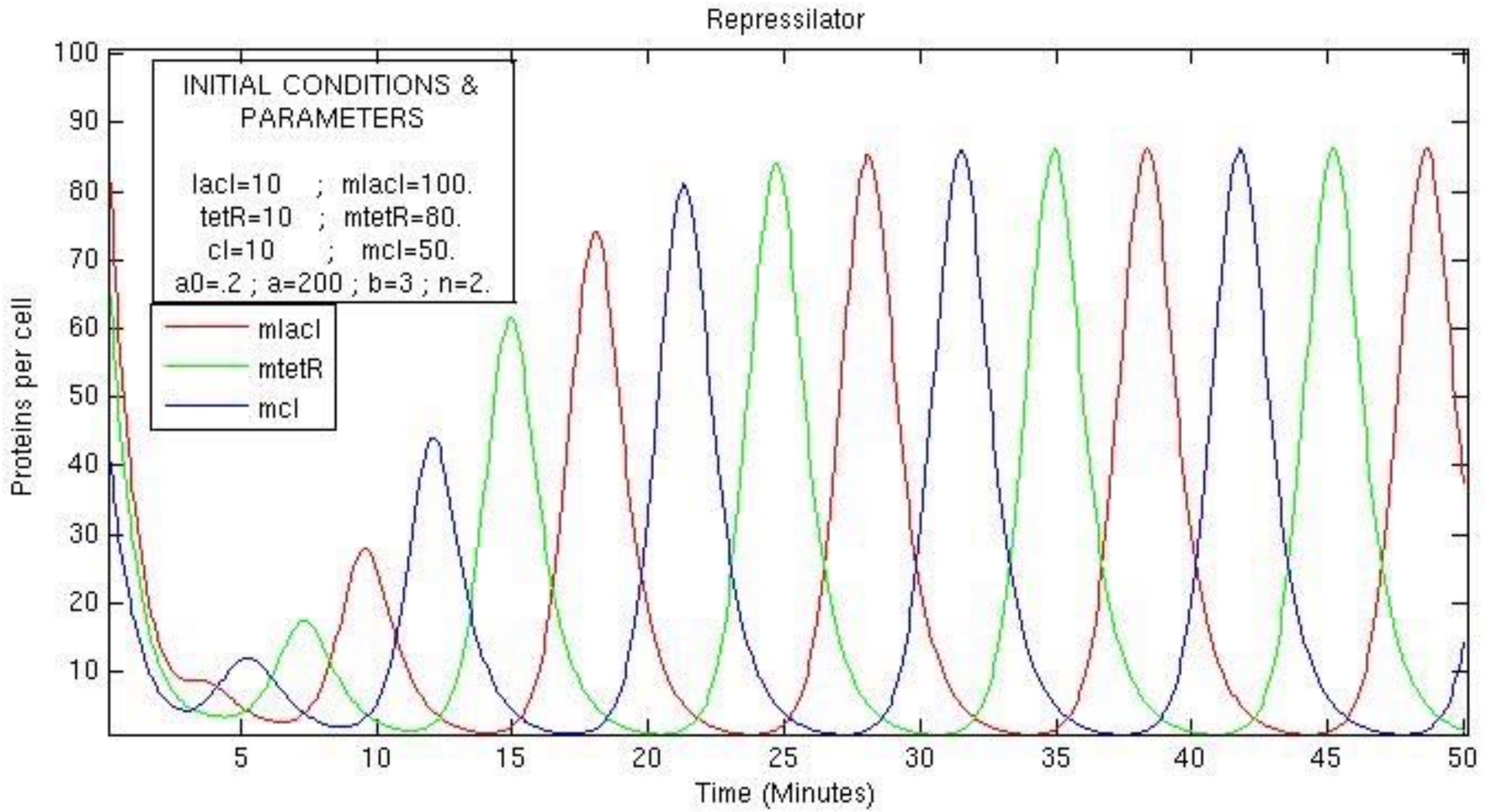
$$\frac{\mathrm{d}p_3}{\mathrm{d}t} = -\beta(p_3 - m_3). \qquad (3.7f)$$

Green fluorescent protein (used to measure expression)



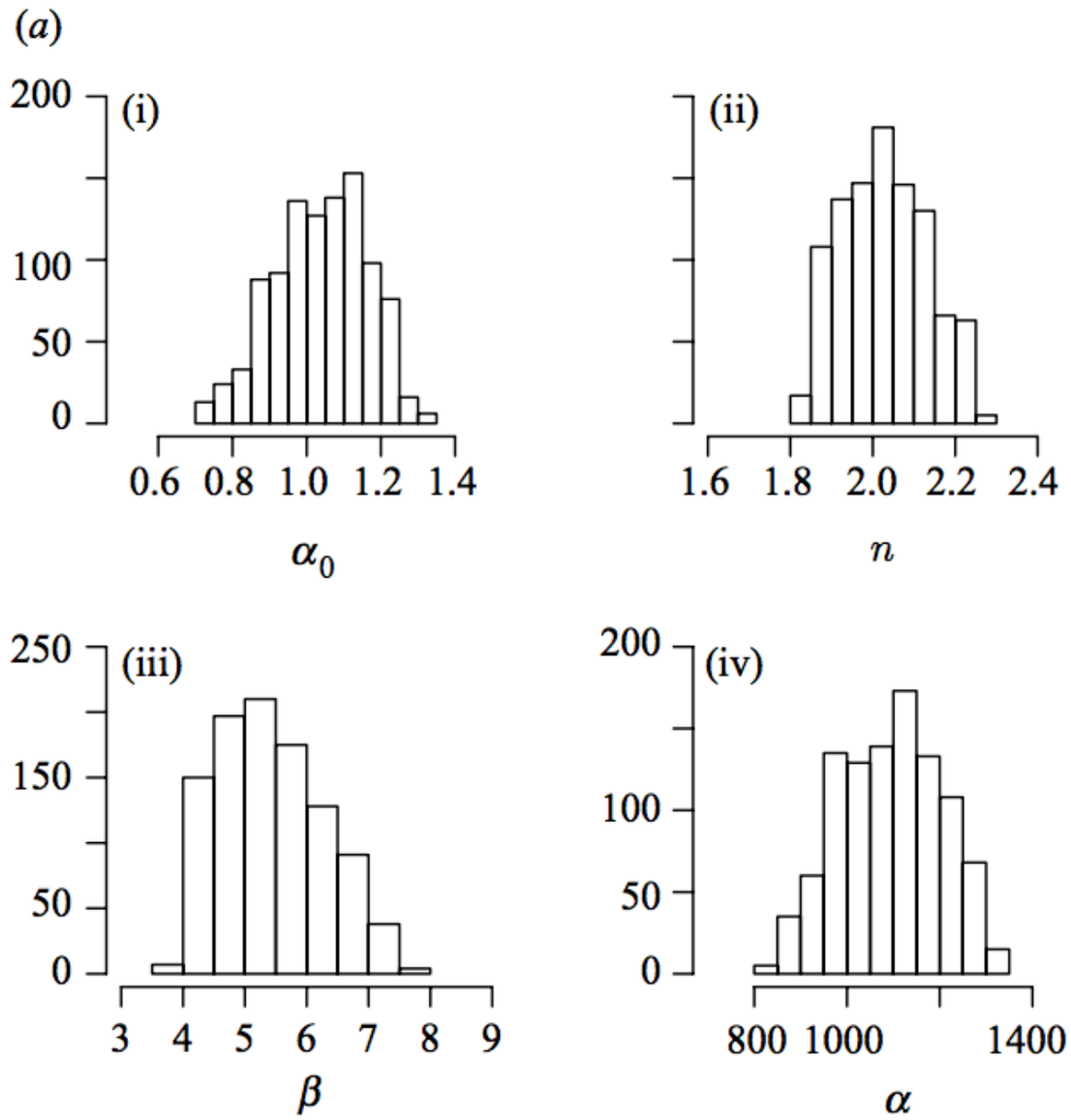
Toni T, Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, *6*, 187–202.

Application



Toni T, Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, *6*, 187–202.

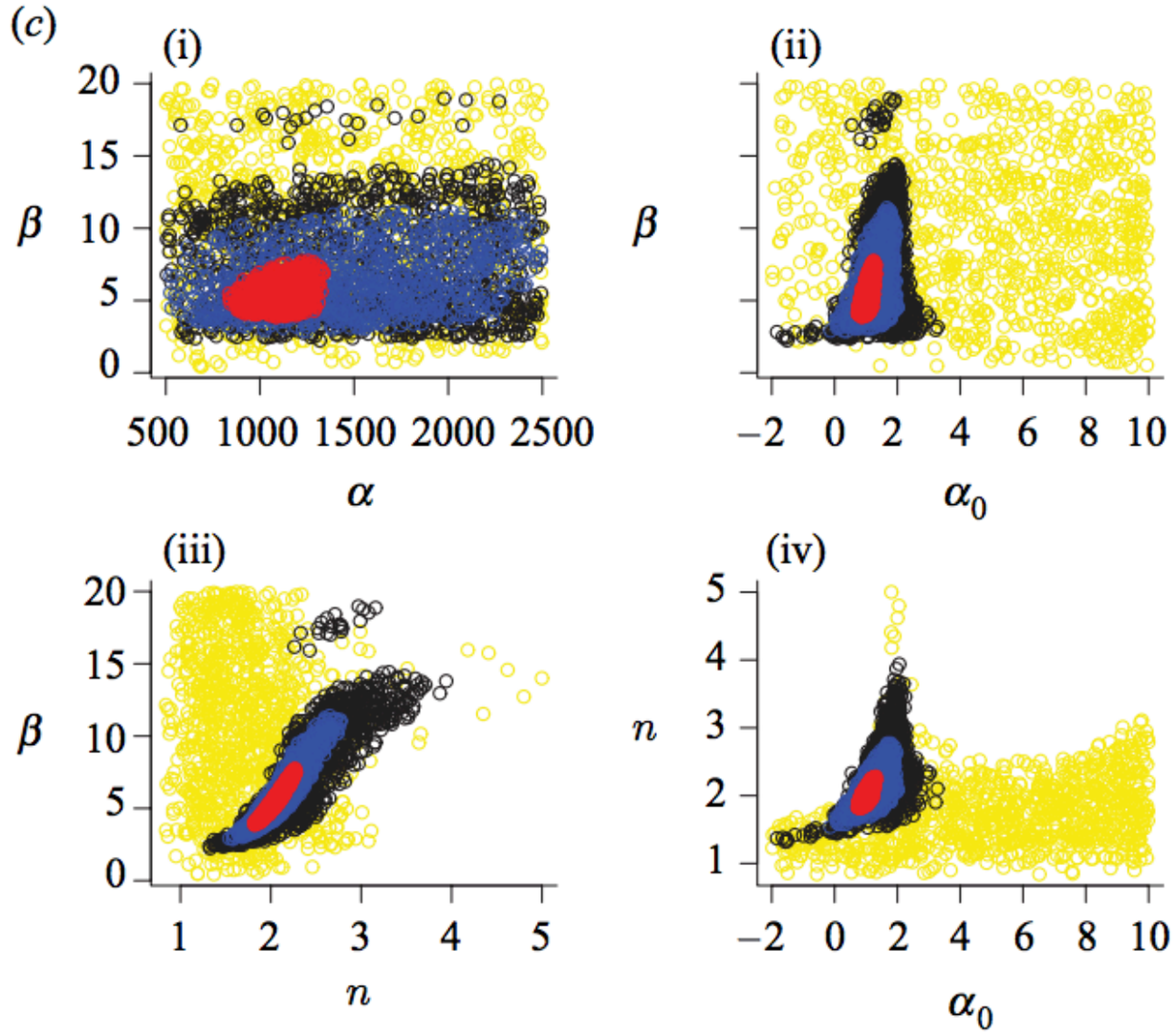
Results



Histogramṡ of the approximate marginal posterior distributions of parameters α_0, n, b and α of the deterministic repressilator model.

Toni T, Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, *6*, 187–202.

Results



The output (i.e. the accepted particles) as two-dimensional scatterplots. The particles from population 1 are in yellow, particles from population 4 in black, particles from population 7 in blue and those from the last population in red. Islands of particles are observed in population 4 and they can be explained by the multimodality of the fourth intermediate distribution

Regression - adjusted ABC

Regression - adjusted ABC

- Recall, ABC rejection:

Suppose:

Observed data D, and a set of (normalized) summary statistics $S=\{S_1,\ldots,S_n\}$

(And possibly a set of weights $w_1,\ldots,w_n$).

A model M with parameter(s) θ.

A *tolerance* ε

1. Sample θ from prior π.
2. Simulate D′ using θ. Calculate S′
3. Accept θ if $\sum_i w_i(S_i^o - S_i^s)^2 < \epsilon$
4. Return to 1.

Results: independent samples from something we will call $\varphi(\theta|D)$ [~$f(\theta|D)$].

So we accept all θ that generate data close enough to the observed data D, and then treat those θ equally when constructing the posterior distribution.

Regression - adjusted ABC

- Intuition:
  - Why treat all accepted θ equally?
  - The closer the simulated data are to the observed data, the better the generating θ are likely to approximate samples from f(θ|D) (rather than φ(θ|D))
  - So, construct the posterior distribution φ(θ|D) using a weight on each accepted θ that reflects how well the data it simulated matched the observed data….

Regression - adjusted rejection ABC

Suppose we have a set of statistics $S_1(y),\ldots,S_s(y)$ for observed data y.

1. Given observation y, repeat the following until M points have been generated:
   a. Draw $\theta_i \sim \pi(\theta)$.
   b. Simulate $x_i \sim p(x \mid \theta_i)$.
2. Compute k_j, the empirical standard deviation of the $S_j(x)$.
3. Define $\rho(S(x), S(y))$: $\sqrt{\sum_{j=1}^{s} (S_j(x)/k_j - S_j(y)/k_j)^2}$. ← Normalize the summary stats.
4. Choose tolerance ϵ such that the proportion of accepted points $P_\epsilon = N/M$.
5. Weight the simulated points $S(x_i)$ using $K_\epsilon(\rho(S(x_i), S(y)))$, where

$$K_\epsilon(t) = \begin{cases} \epsilon^{-1}\left(1 - (t/\epsilon)^2\right) & t \le \epsilon \\ 0 & t > \epsilon \end{cases}.$$

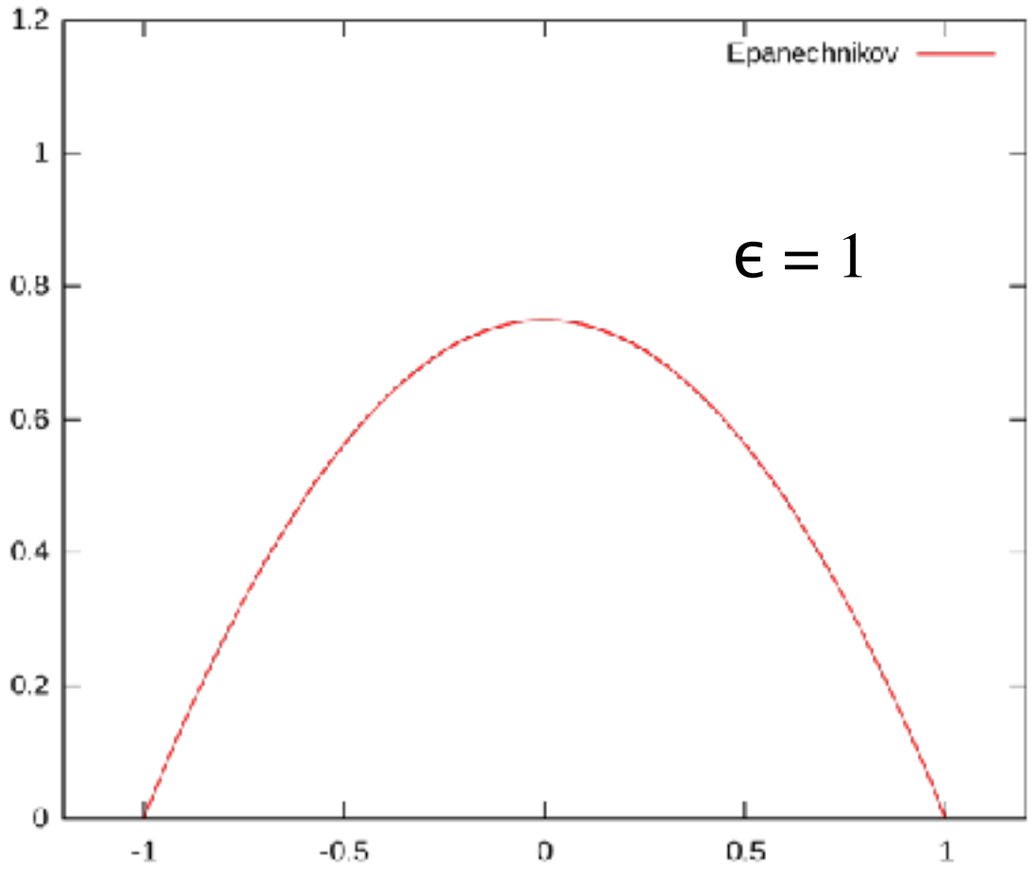
← weight according to distance

6. Apply weighted linear regression to the N points that have nonzero weight to obtain an estimate of $\hat{E}[\theta \mid S(y)]$.
7. Adjust $\theta_i^* = \theta_i - \hat{E}[\theta \mid S(x_i)] + \hat{E}[\theta \mid S(y)]$. ← regression adjustment
8. The θ_i^*, with weights $K_\epsilon(\rho(S(x_i), S(y)))$, are taken to be random draws from an approximation to the posterior distribution $p(\theta \mid y)$.

Epanechnikov kernel

5. Weight the simulated points $S(x_i)$ using $K_\epsilon(\rho(S(x_i), S(y)))$, where

$$K_\epsilon(t) = \begin{cases} \epsilon^{-1}\left(1 - (t/\epsilon)^2\right) & t \le \epsilon \\ 0 & t > \epsilon \end{cases}.$$

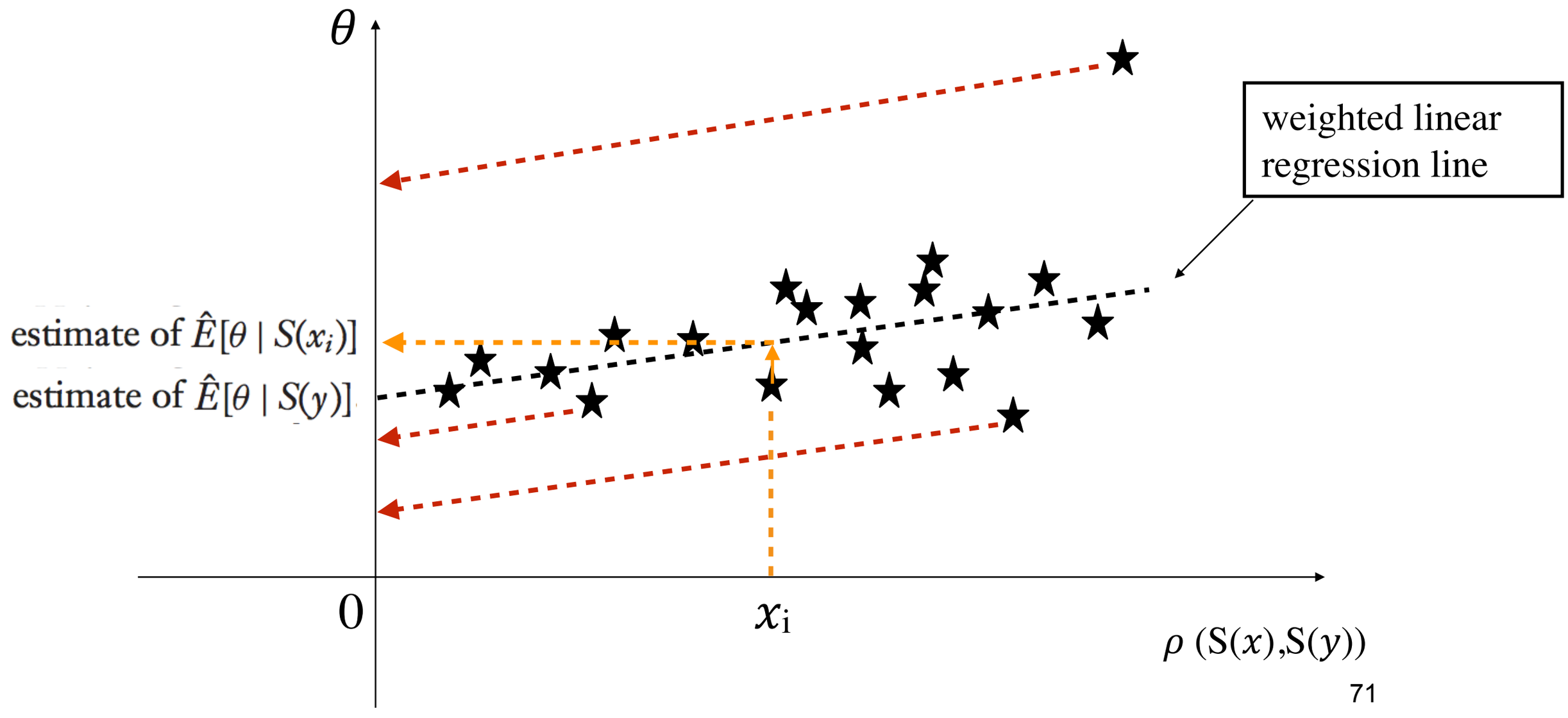


y-axis is the weight assigned to each data-point, as a function of its distance to the observed data (x-axis)

Regression adjustment

6. Apply weighted linear regression to the N points that have nonzero weight to obtain an estimate of $\hat{E}(\theta \mid S(x))$.
7. Adjust $\theta_i^* = \theta_i - \hat{E}[\theta \mid S(x_i)] + \hat{E}[\theta \mid S(y)]$.

★ = accepted data-points



Application

Data:

Gene frequencies at 8 loci on the Y chromosome.

445 males taken from a number of different populations around the world.

Originally from Perez-Lezaun et al. (1997) and Seielstad et al. (1998).

Model (after Pritchard et al. 1999):

Exponential population growth following an ancestral population of constant size N_A chromosomes.

Growth begins t_g generations from the present time.

Current population size is $N_0 = N_A \exp(rt_g)$, where r is the population growth rate per generation.

Coalescent model.

Summary statistics (after Pritchard et al. 1999):

(1) The mean (across loci) of the variance in repeat numbers;

(2) The mean effective heterozygosity (i.e., the probability of two randomly drawn chromosomes differing at a particular locus, averaged across loci);

(3) The number of distinct haplotypes in the sample.

Application

Compare three analyses:

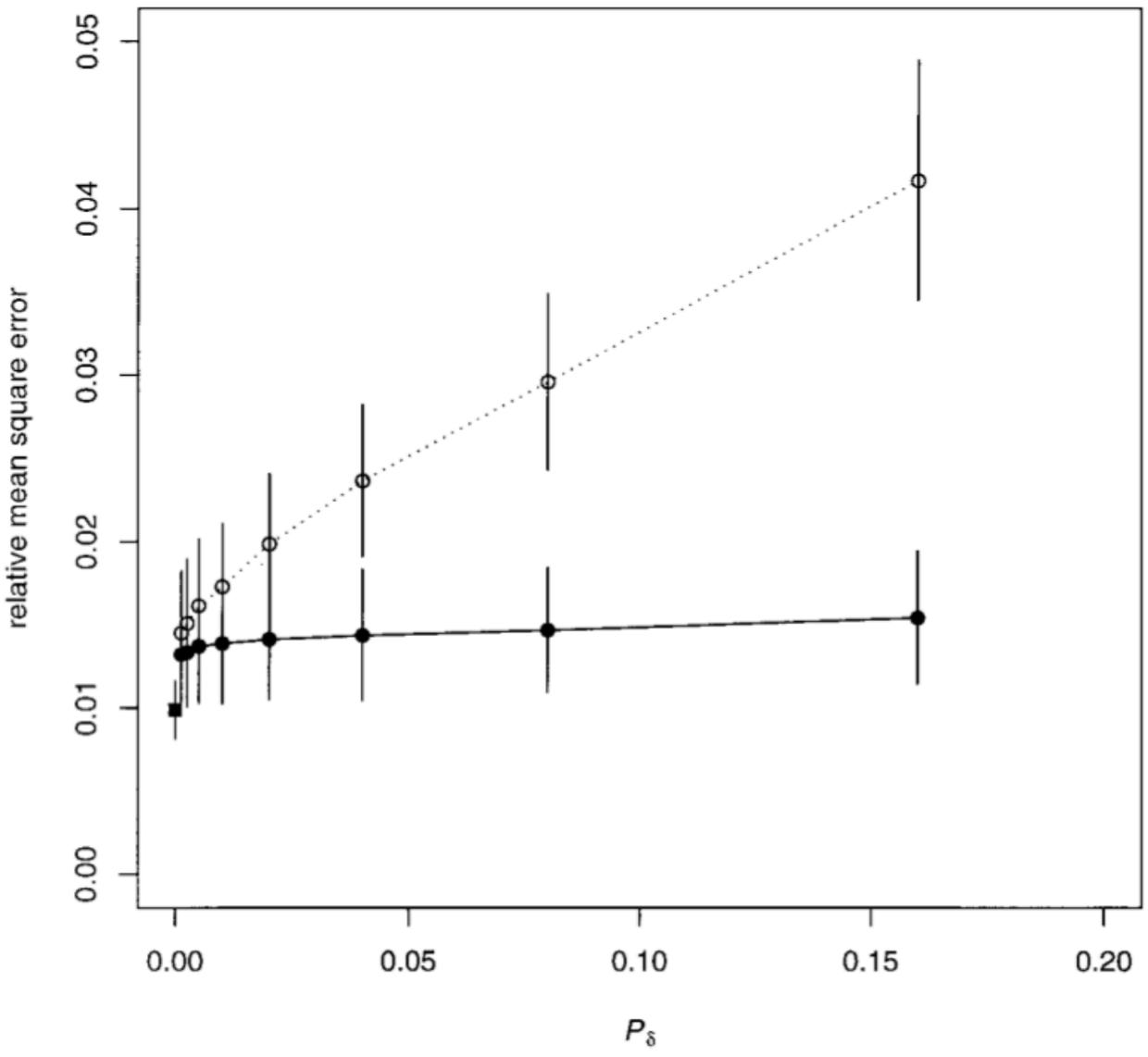
1) Standard rejection method (c.f. Pritchard 1999)
2) Regression-adjusted rejection (c.f. Beaumont et al, 2002)
3) Exact MCMC solution (which is possible in this simple case, but would have been intractable for larger problems.

In addition, they consider how much it helps to add two extra summary statistics:

4) A multivariate version of kurtosis (the 'peakiness' of the repeat number distribution)
5) A measure of mean linkage disequilibrium (correlation between loci) averaged across loci

Results 1

Constant population-size model - so the only parameter is the mutation rate θ.

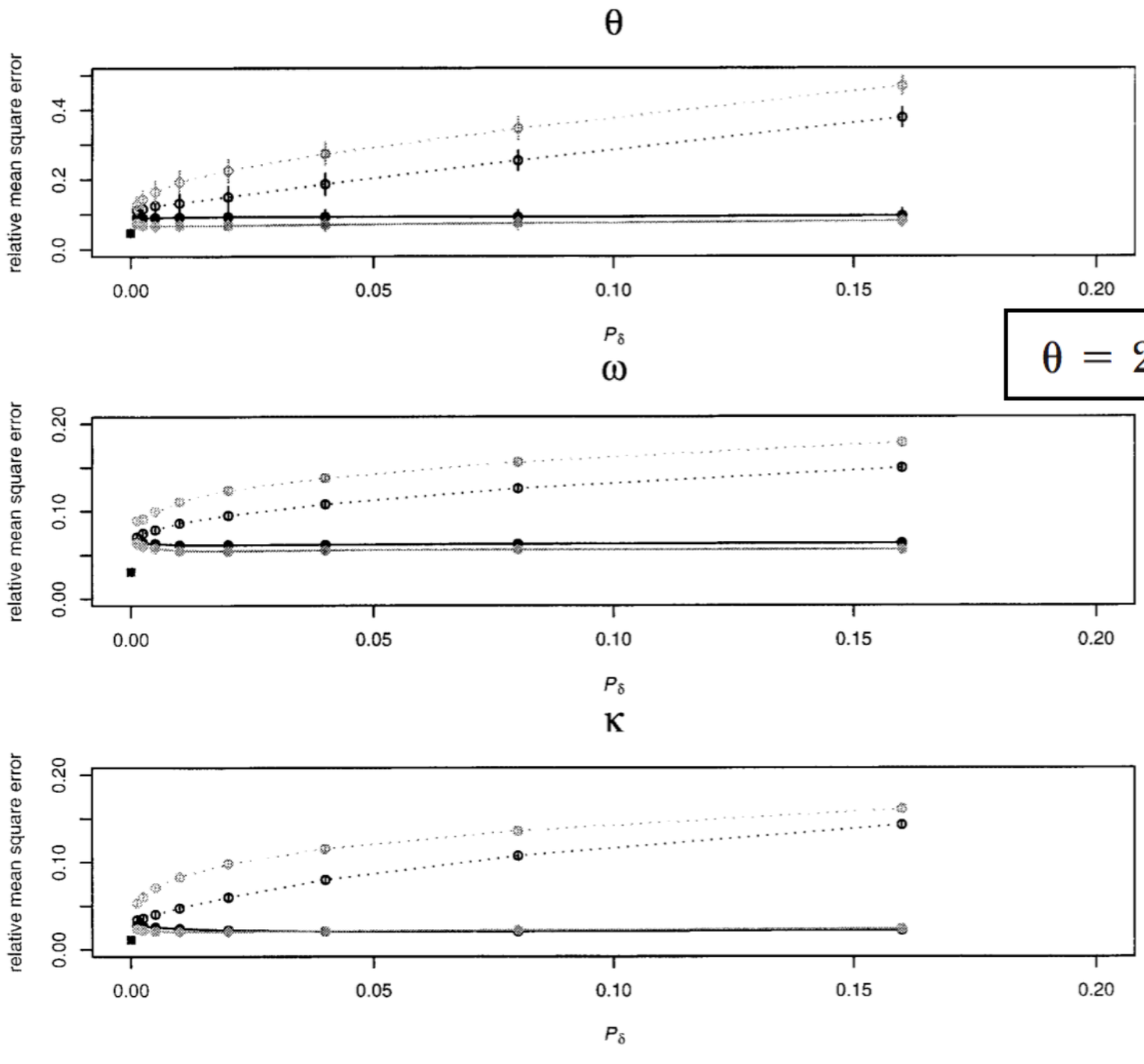


P_δ: the proportion of data-points accepted

FIGURE 1.—A plot of the RMSE in estimates of θ against a measure of tolerance P_δ, as defined in the text. Estimates using the rejection method are shown as a dotted line and those from the regression method as a solid line. The RMSE for the MCMC method is shown by the solid square at $P_\delta = 0$. Standard errors are shown as vertical bars.

Results 2

Population growth model - so now there are 3 parameters



$$\theta = 2N_A\mu, \quad \omega = rN_A, \quad \kappa = rt_g$$

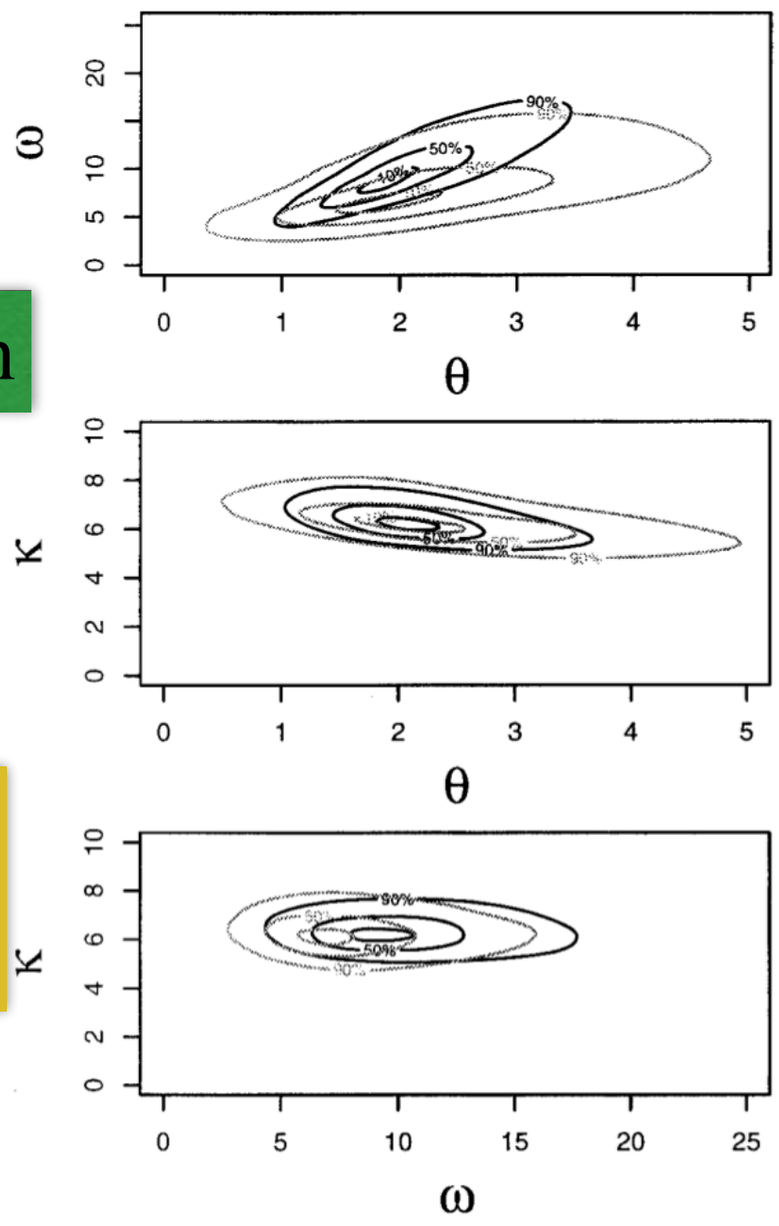
FIGURE 2.—A plot of the RMSE in estimates of θ, ω, and κ against a measure of tolerance P_δ, as defined in the text. Solid lines were obtained using three summary statistics. Shaded lines were obtained using five summary statistics. Other details are as for Figure 1.

Results 2

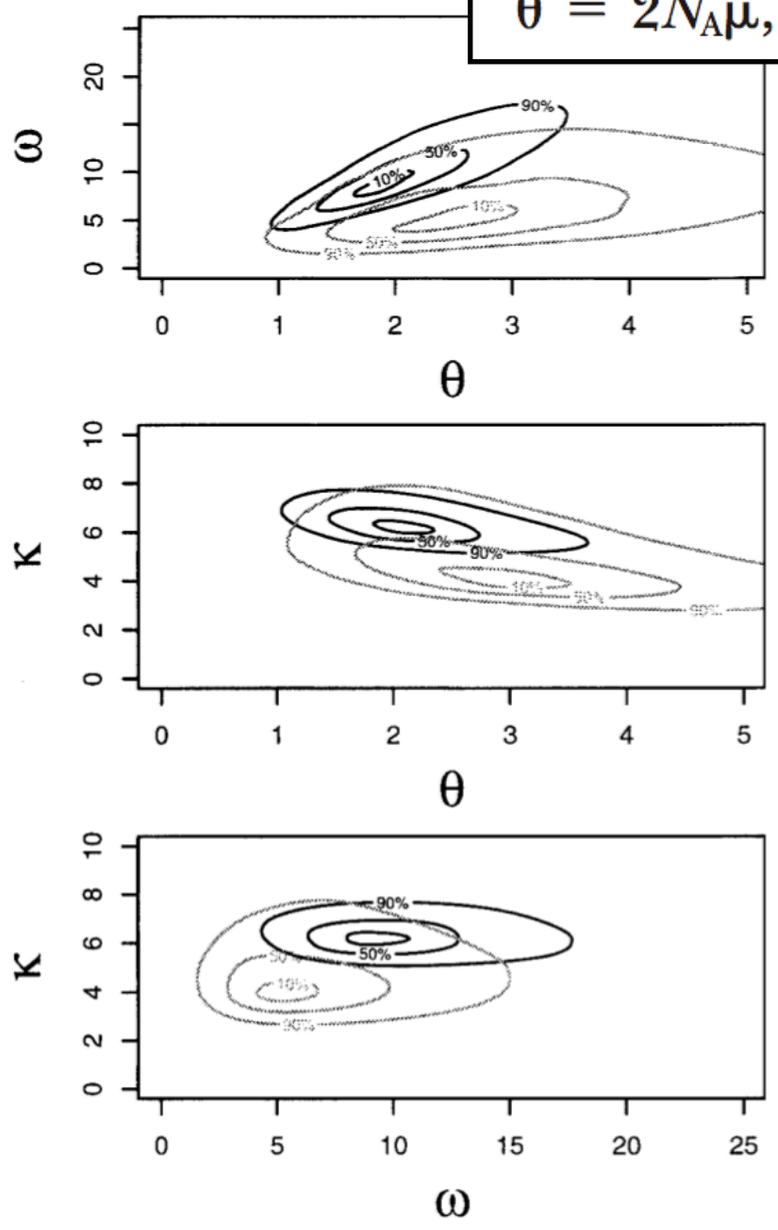
Population growth model - so now there are 3 parameters

$$\theta = 2N_A\mu, \quad \omega = rN_A, \quad \kappa = rt_g$$

Regression



Rejection



ABC: grey lines, both sides

MCMC posterior: solid line (both sides)

FIGURE 4.—Plots of the joint posterior densities for the three pairs of parameters among θ, ω, and κ, estimated by MCMC, regression, and rejection methods. Densities estimated by the regression method are shown on the left, and those estimated by the rejection method are on the right. A tolerance of 0.08 was used to compare the regression and rejection methods. The 10, 50, and 90% highest posterior density contours are shown. Those estimated by MCMC are shown as solid lines, and those estimated from summary statistics are shown as shaded lines.

Regression adjustment - summary

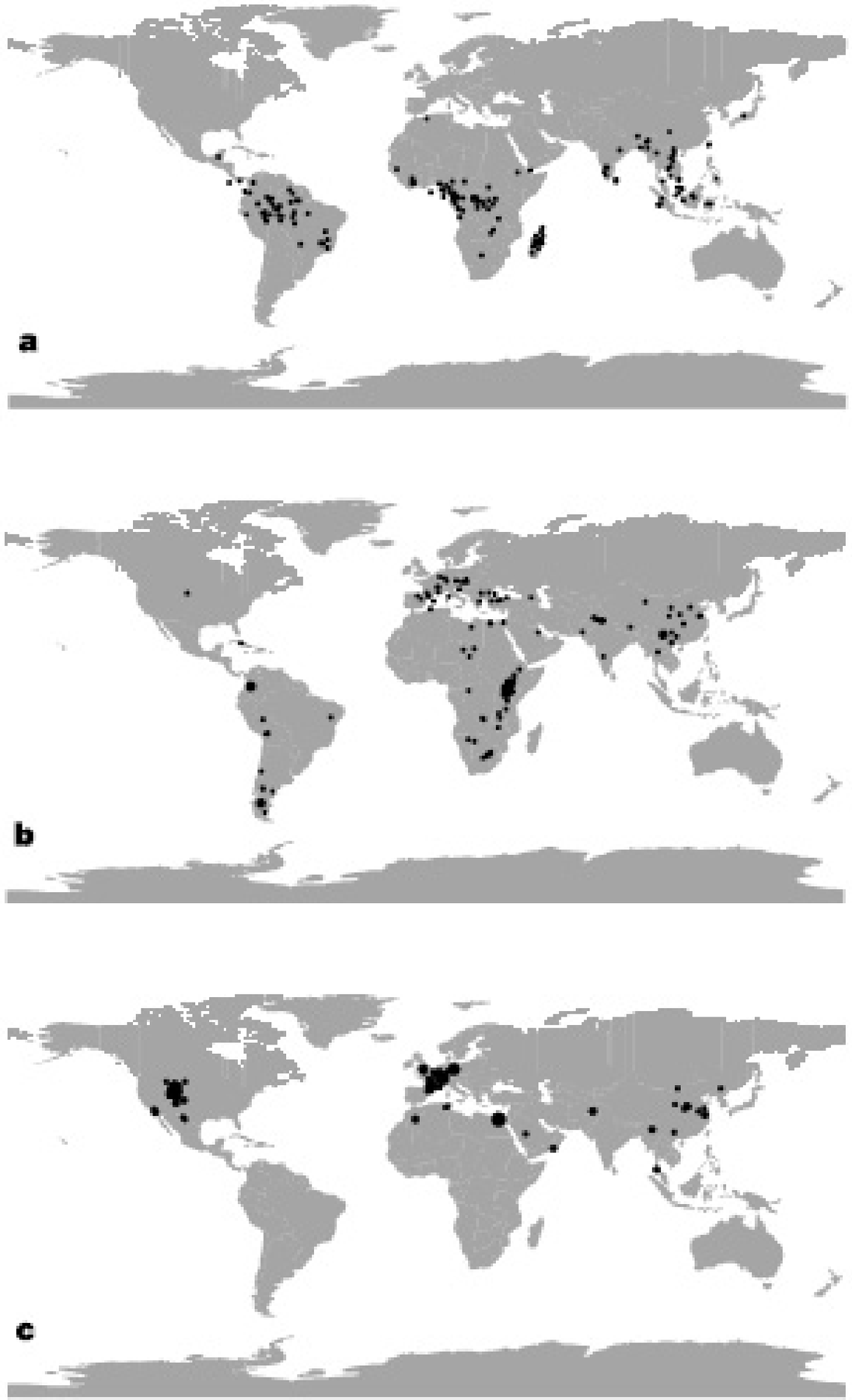
- Shows significant gain in performance over standard (ABC) rejection.
- Was later generalized to non-linear weighted regression (Blum, M. G. B., & François, O. (2010). Non-linear regression models for Approximate Bayesian Computation. *Statistics and Computing*, *20*(1), 63–73.).
- Statistics you use do matter.
- Normalize your summary statistics!
- When you can produce an explicit, exact solution (e.g., via MCMC) you should do so.

Further example (read at your leisure)

ABC Application - fossil record

- Fossil data from multiple primate species
- Can recognize the species from which each fossil is drawn (i.e. fossils ⇒ #species)
- Oldest known fossil primates are from the basal Eocene epoch (54–55 million years [Myr] ago)
- When did the primates diverge from other placental mammals?

[Soligo, Christophe, et al. "New light on the dates of primate origins and divergence." *Primate origins: adaptations and evolution*. Springer US, 2007. 29-49.]



- Mid-range of geographical distribution for individual modern and fossil primate species. **a**, Modern and sub-fossil primates (170 species obtained from Wolfheim's review of the distribution of modern primates) **[current]**. **b**, Fossil species for the Late Pleistocene to the Late Oligocene (167 species) **[older]**. **c**, Fossil species from the Early Oligocene to the Early Eocene **[oldest]** (196 species). (The database of fossil primates was compiled from a large number of published sources. A full list of references can be found at http://www.unizh.ch/anthro/ Main/Who/Soligo/supinfo2.html).

- Figure from Tavaré et al. NATURE, VOL 416, 726-729 APRIL 2002

Table 1. Data for the primate fossil record. References can be found in the supplemental material in (14).

Epoch	Bin k	Time T_k	Observed number of species (D_k)
Late Pleistocene	1	0.15	19
Middle Pleistocene	2	0.9	28
Early Pleistocene	3	1.8	22
Late Pliocene	4	3.6	47
Early Pliocene	5	5.3	11
Late Miocene	6	11.2	38
Middle Miocene	7	16.4	46
Early Miocene	8	23.8	36
Late Oligocene	9	28.5	4
Early Oligocene	10	33.7	20
Late Eocene	11	37.0	32
Middle Eocene	12	49.0	103
Early Eocene	13	54.8	68
Pre-Eocene	14		0

Time is measured in millions of years.
Dinosaurs became extinct about 65Myrs ago.

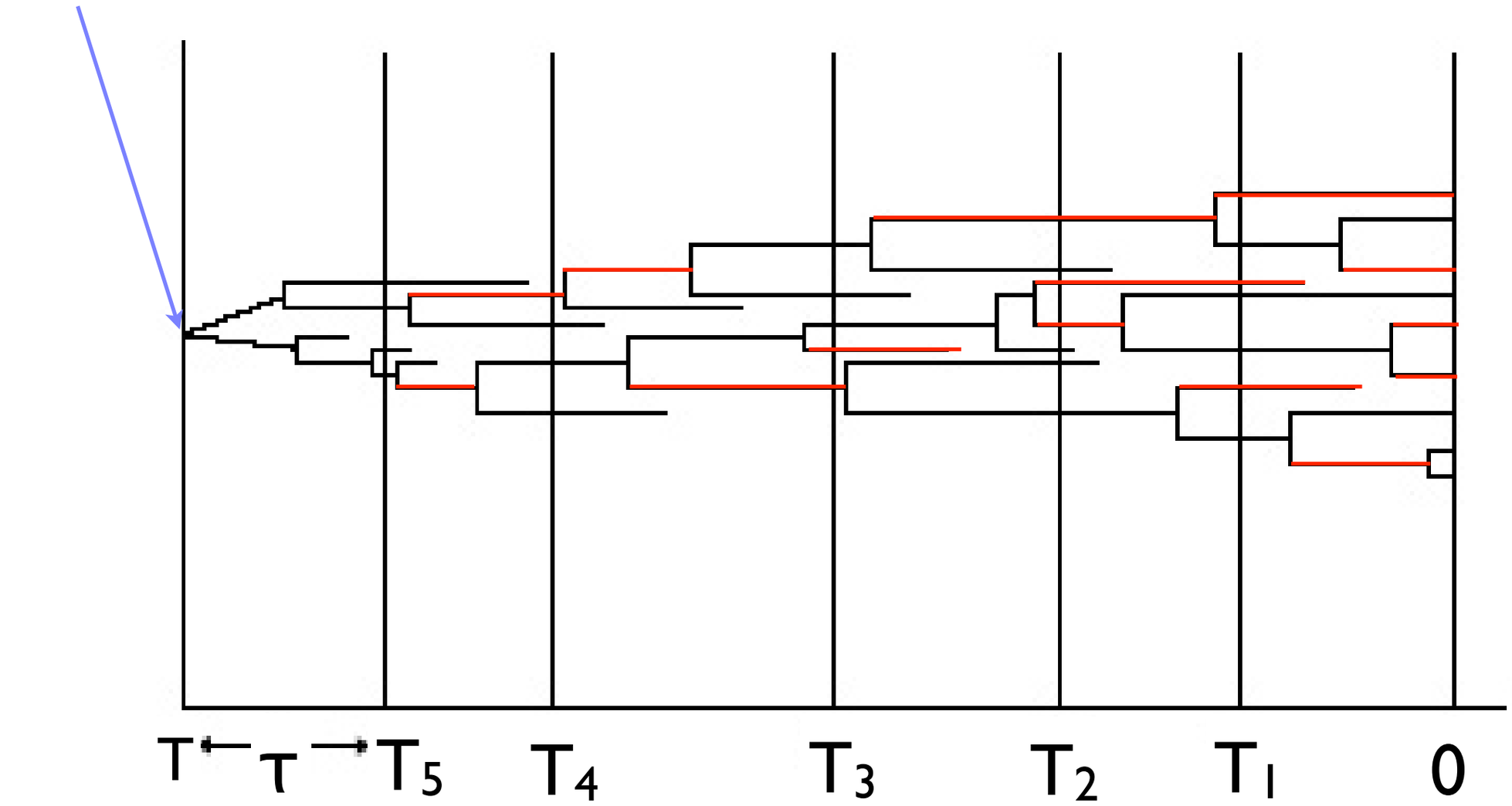


Fig. 1. An illustration of the stochastic model of fossil finds. Bases of 5 stratigraphic intervals at T_1 , , . . . , T_5 Myr ago are shown along the x-axis. The temporal gap between the base of the final interval and the point at which the two founding species originate is denoted by τ . Red lines indicate species found in the fossil record. Time 0 is the present day.

Model, M

- Speciation modeled by non-homogeneous Markov birth-and-death process.
- Suppose we start with one species at time 0 (the MRCA).
- Measure time in Myrs (millions of years).
- Species go extinct at rate λ. (So lifetime ~ exp(λ) and has mean 1/λ)
- When species ‘dies’ at time *u*, it is replaced by an average of *m(u)* new species. [non-homogeneous]

Model, M

- Let Z_t be the number of species existing at time *t*, then its expectation is given by:

$$EZ_t = 2\exp\left\{\lambda\int_0^t (m(u)-1)du\right\}$$

- [Result uses existing theory for birth-and-death processes, of which there is much].

- Divide time into k intervals
- Interval k starts at time T_k
- T_1 is youngest interval, T_k is oldest (and contains first known primate fossil)
- MRCA occurs at time $T=T_k+\tau$ (goal is to estimate this time)

- Assume that, conditional on the number of distinct species N_j that lived in the j^{th} stratigraphic interval, the number of species D_j actually found in the fossil record in this interval is a binomial random variable with parameters N_j and α_j, $j = 1, 2, \ldots, k+1$.

- The D_j are assumed to be conditionally independent given the N_j.

- The parameter α_j gives the probability of sampling a fossil of species in the j^{th} stratigraphic interval. (This allows for different probs. of finding fossils left during different eras.) *Independent of death during the interval*.

Rejection method

- Let $D = (D_1, \ldots, D_{k+1})$ be the species counts observed in the k+1 stratigraphic intervals
- Write θ for the vector of parameters of the process, one of which is τ, the temporal gap
- Likelihood $P(D|\theta)$ is difficult to compute (because the tree is unobserved), but simulation is extremely straightforward: Simulate a speciation tree and observe how many species are present in each time interval
- Once the N_j are simulated, the D_j are binomial samples with parameters N_j and α_j.

Distance measure $\rho(D,D')$

Real data Simulated data

- The counts $D_1, \ldots, D_{k+1}$ can be represented as the total number of fossils found:

  $D_+ = D_1 + \cdots + D_{k+1}$,

- and a vector of proportions

  $(Q_1, \ldots, Q_{k+1}) := (D_1/D_+, \cdots, D_{k+1}/D_+)$

- So use the (arbitrary) metric:

$$\rho(D, D') = \left| \frac{D'_+}{D_+} - 1 \right| + \frac{1}{2} \sum_{j=1}^{k+1} | Q_j - Q'_j |$$

"The first term measures the relative error in the total number of fossils found in a simulated data set and the actual number, while the second term is the total variation distance between the two vectors of proportions."

Model details

- Modeled sampling fractions α_j as $\alpha_j=\alpha p_j$, where the p_j are known proportions and α is estimated from the data.

Table 2. Sampling proportions p_j

j	1	2	3	4	5	6	7	8	9	10	11	12	13	14
p_j	1.0	1.0	1.0	1.0	0.5	0.5	1.0	0.5	0.1	0.5	1.0	1.0	1.0	0.1

Rejection method approach

- $\theta=(\tau,\alpha)$.
- Priors: $\tau \sim Unif(0,100)$, $\alpha \sim Unif(0,0.3)$
- distance tolerance = 0.3
- 2000 acceptances

Posterior distributions

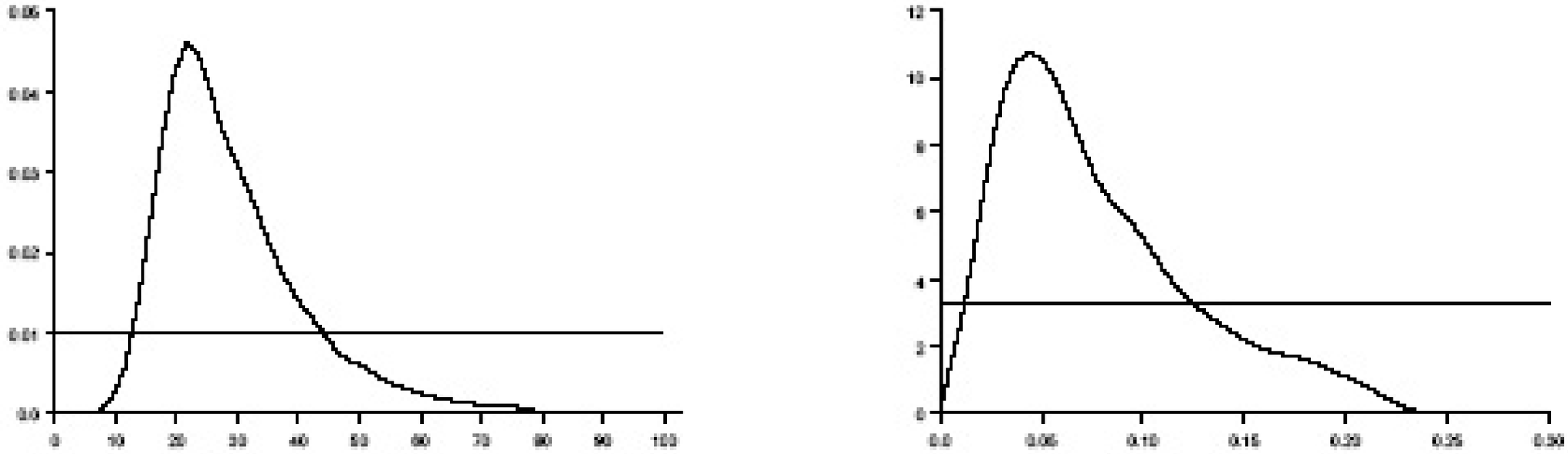


Fig. 2. Left panel: posterior for τ. Right panel: posterior for $\bar{\alpha}$. Horizontal lines show prior density.

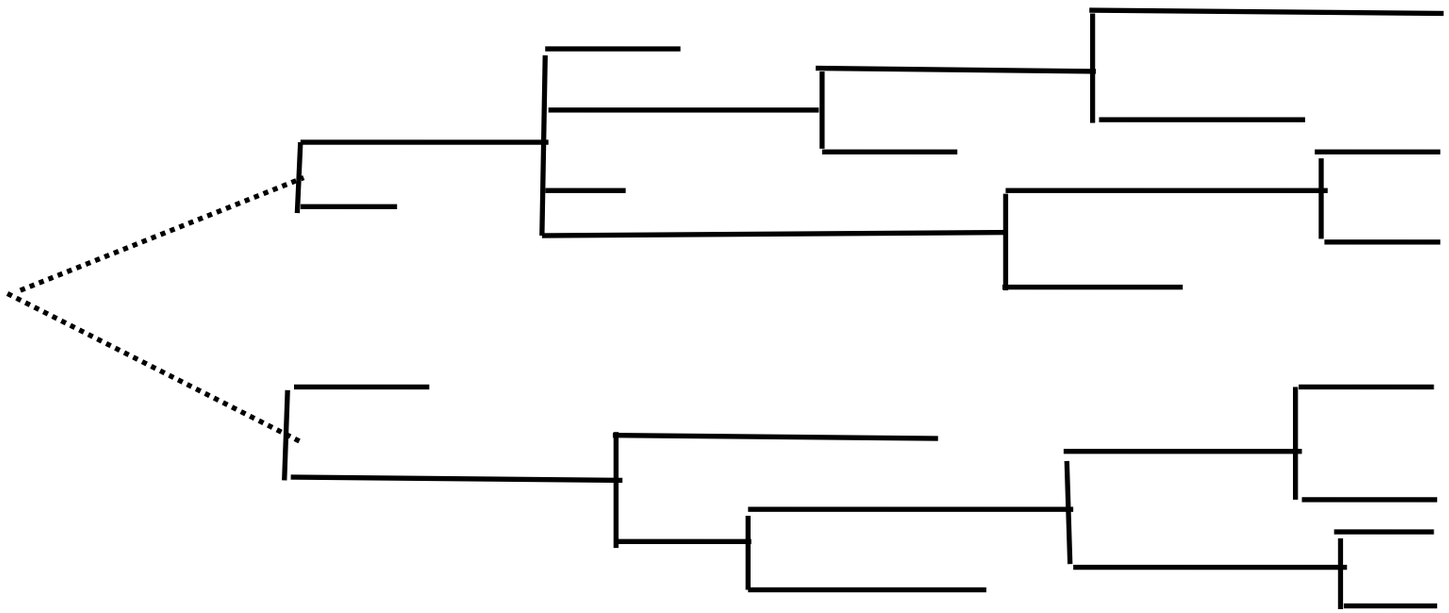
95% CI for TMRCA is (69.1,115.8)

Full MCMC version

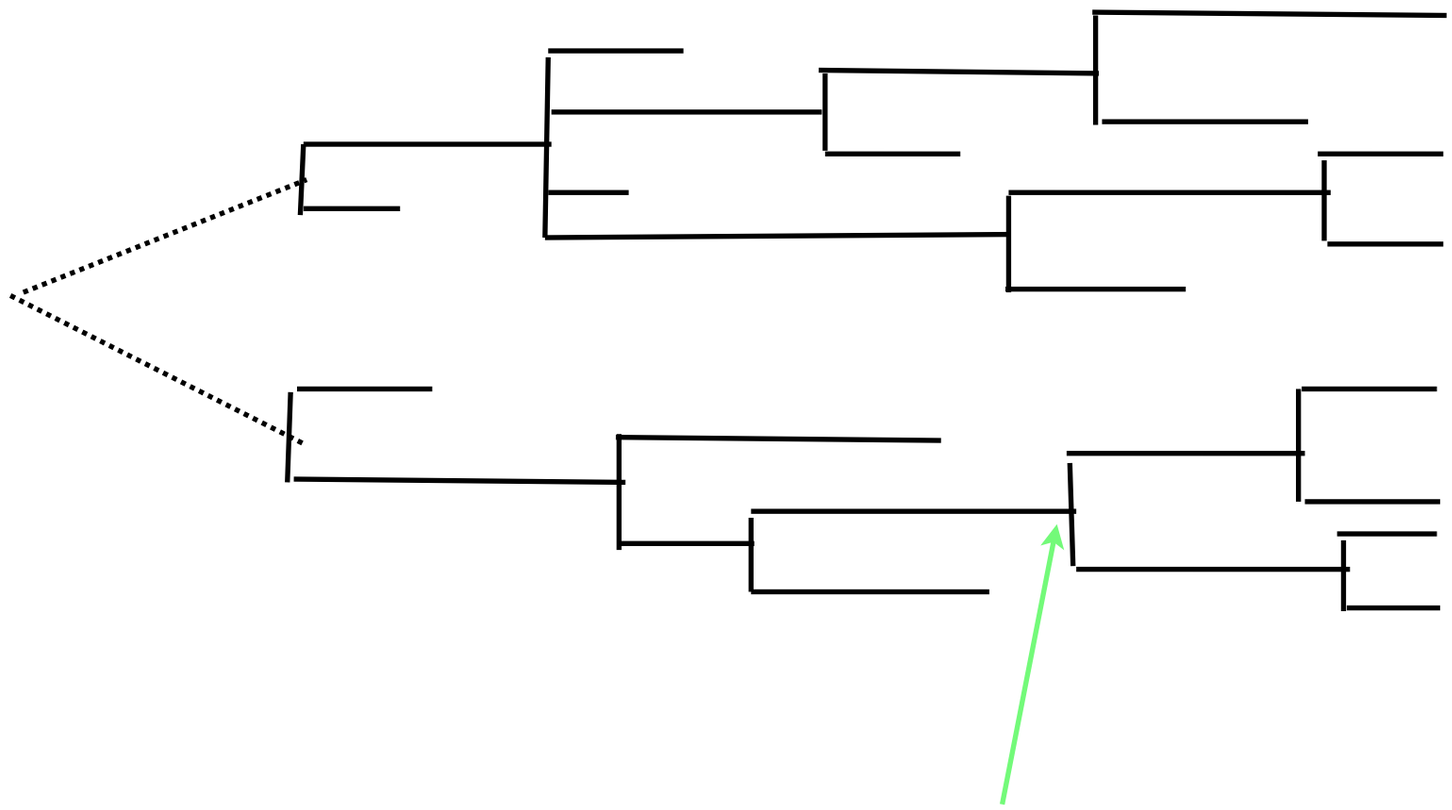
- Statespace: $\theta \in \Theta$, where $\theta = \{\tau, \alpha, \text{full tree topology}\}$
- Proposal kernel: 4 moves -

  1. 'Death/Birth' move
  2. 'Strip' move
  3. 'Scale' move
  4. Change τ and/or α

‘Death/Birth’ move

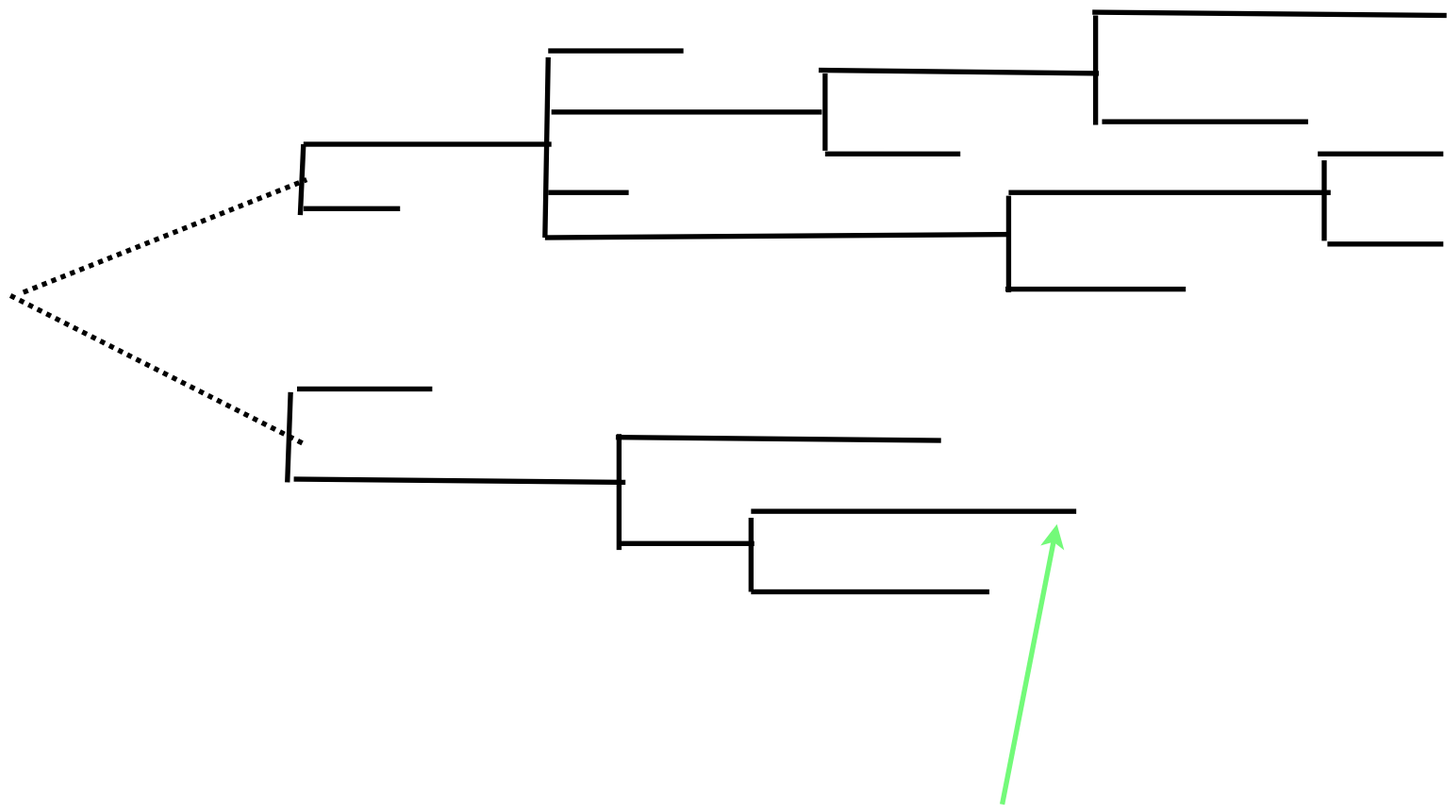


‘Death/Birth’ move



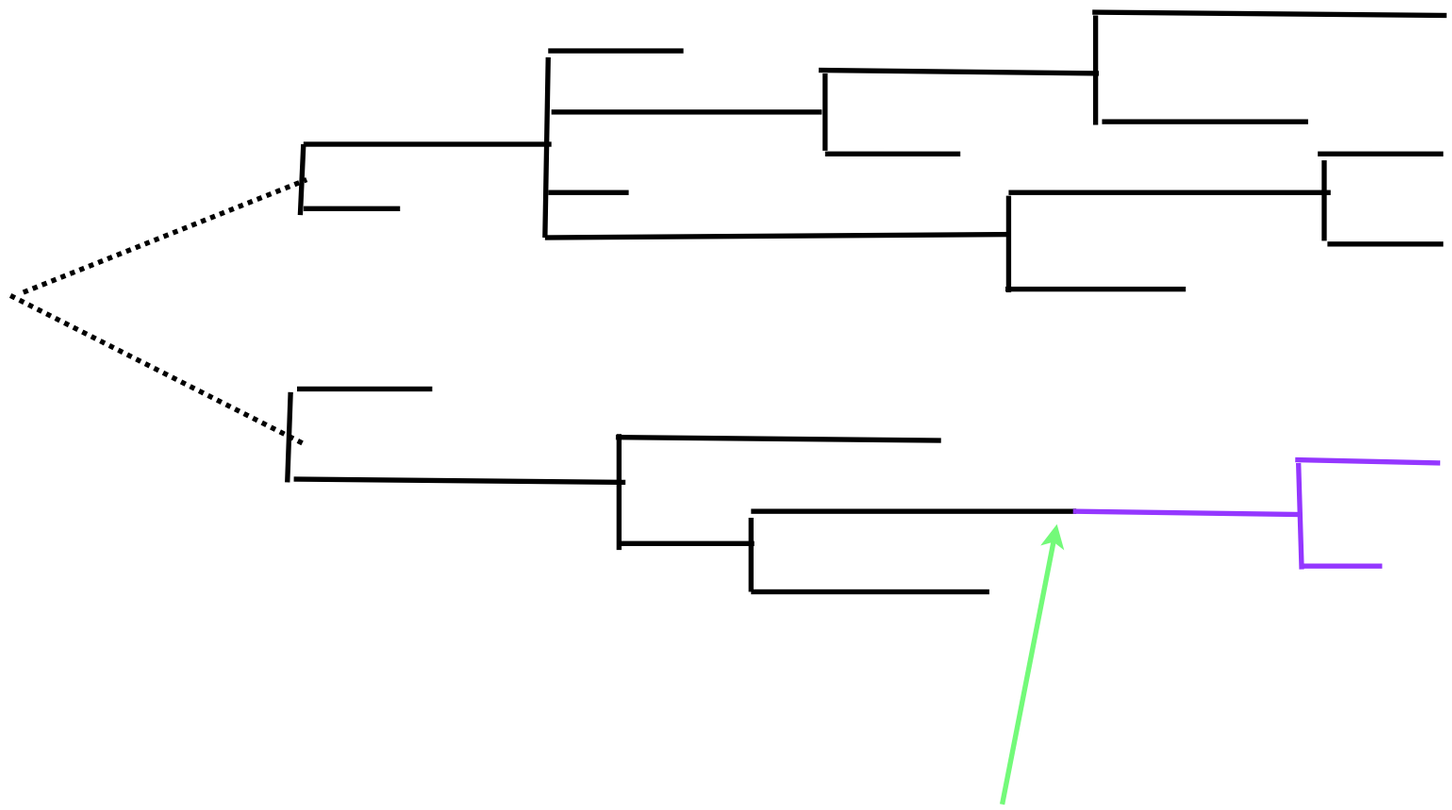
Pick a node, delete everything that follows and generate new trajectory (according to birth/death process rules)

‘Death/Birth’ move



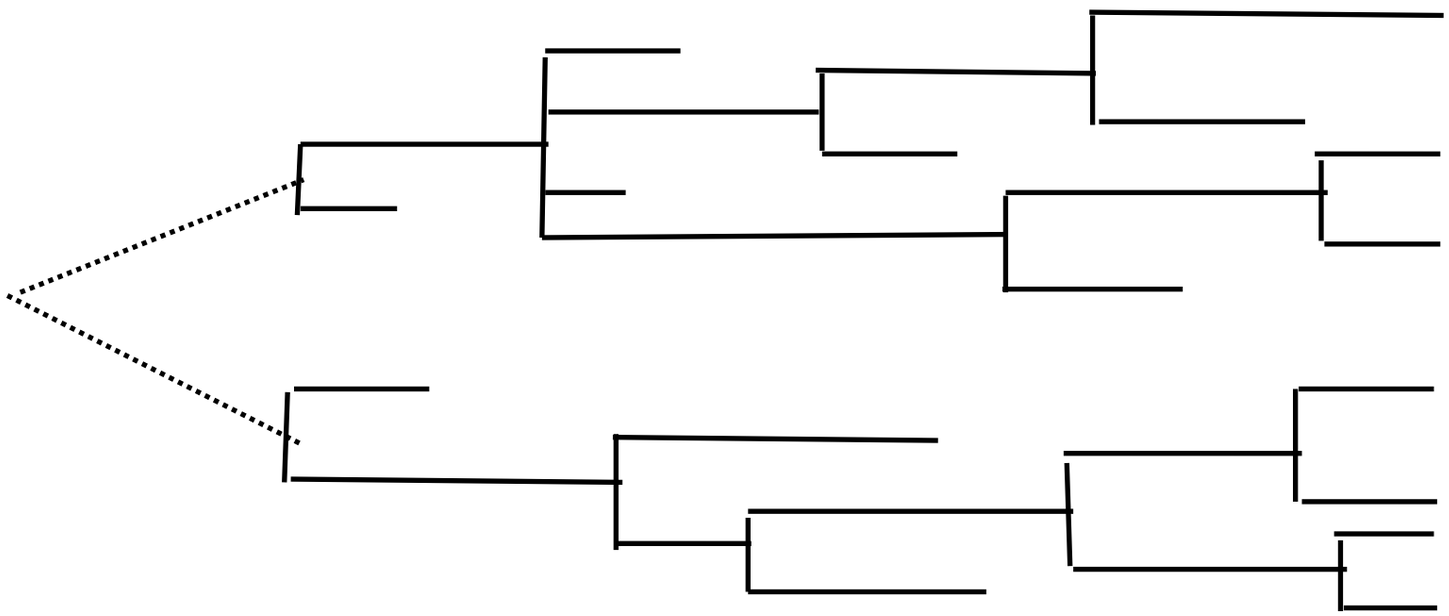
Pick a node, delete everything that follows and generate new trajectory (according to birth/death process rules)

‘Death/Birth’ move



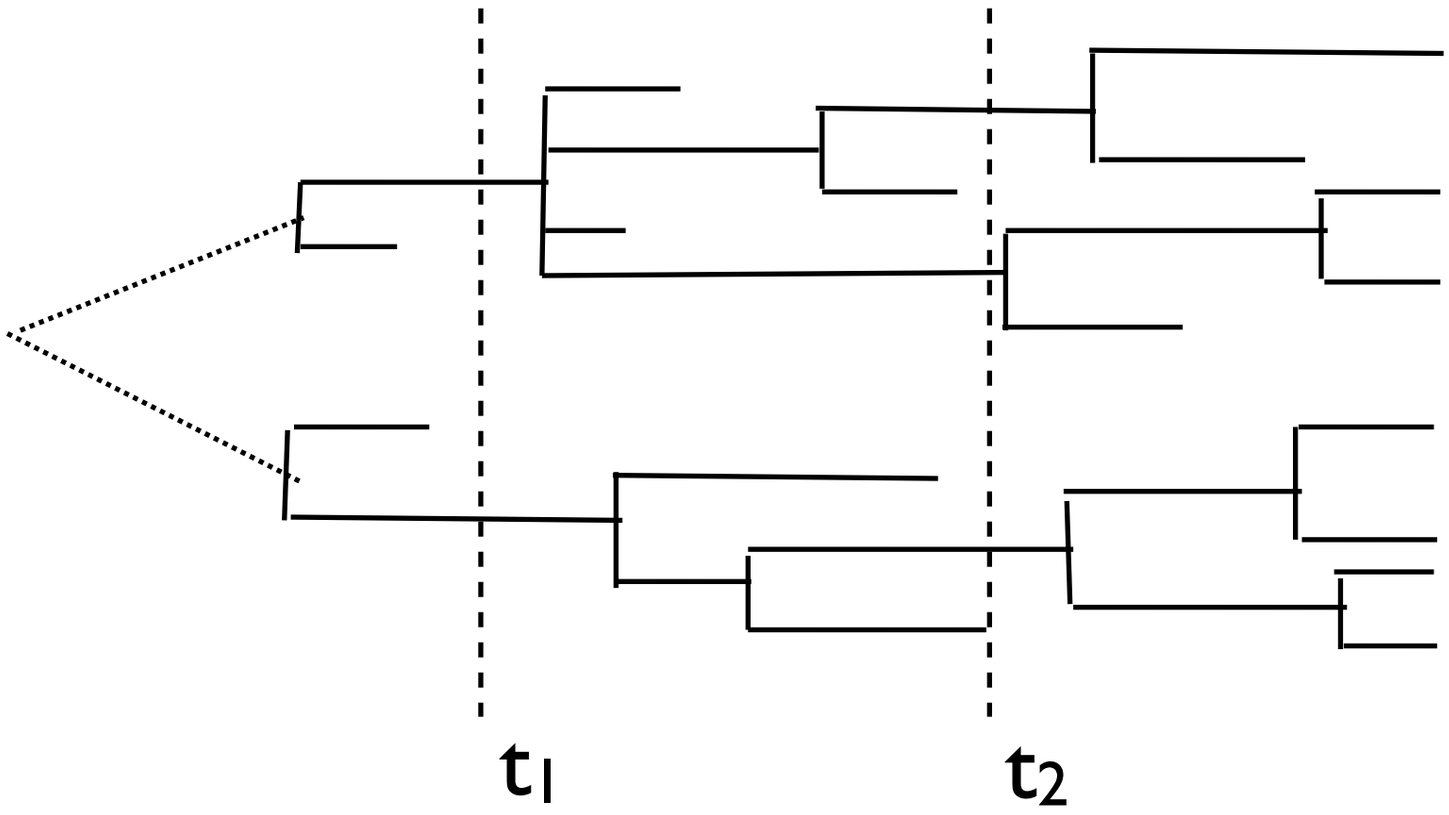
Pick a node, delete everything that follows and generate new trajectory (according to birth/death process rules)

‘Strip’ move



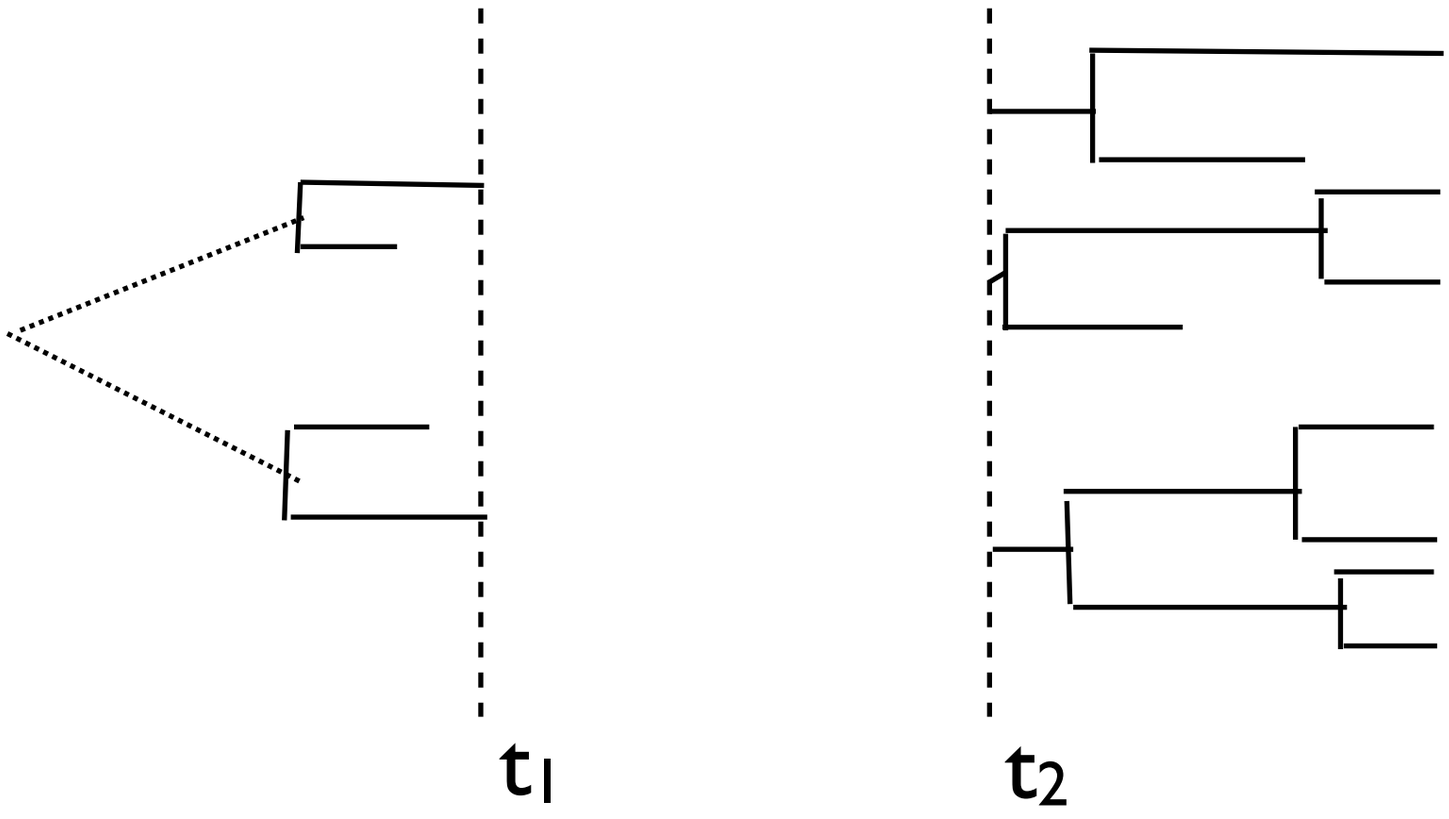
Pick two random times, delete everything in-between and generate a new trajectory to replace the deleted piece

‘Strip’ move



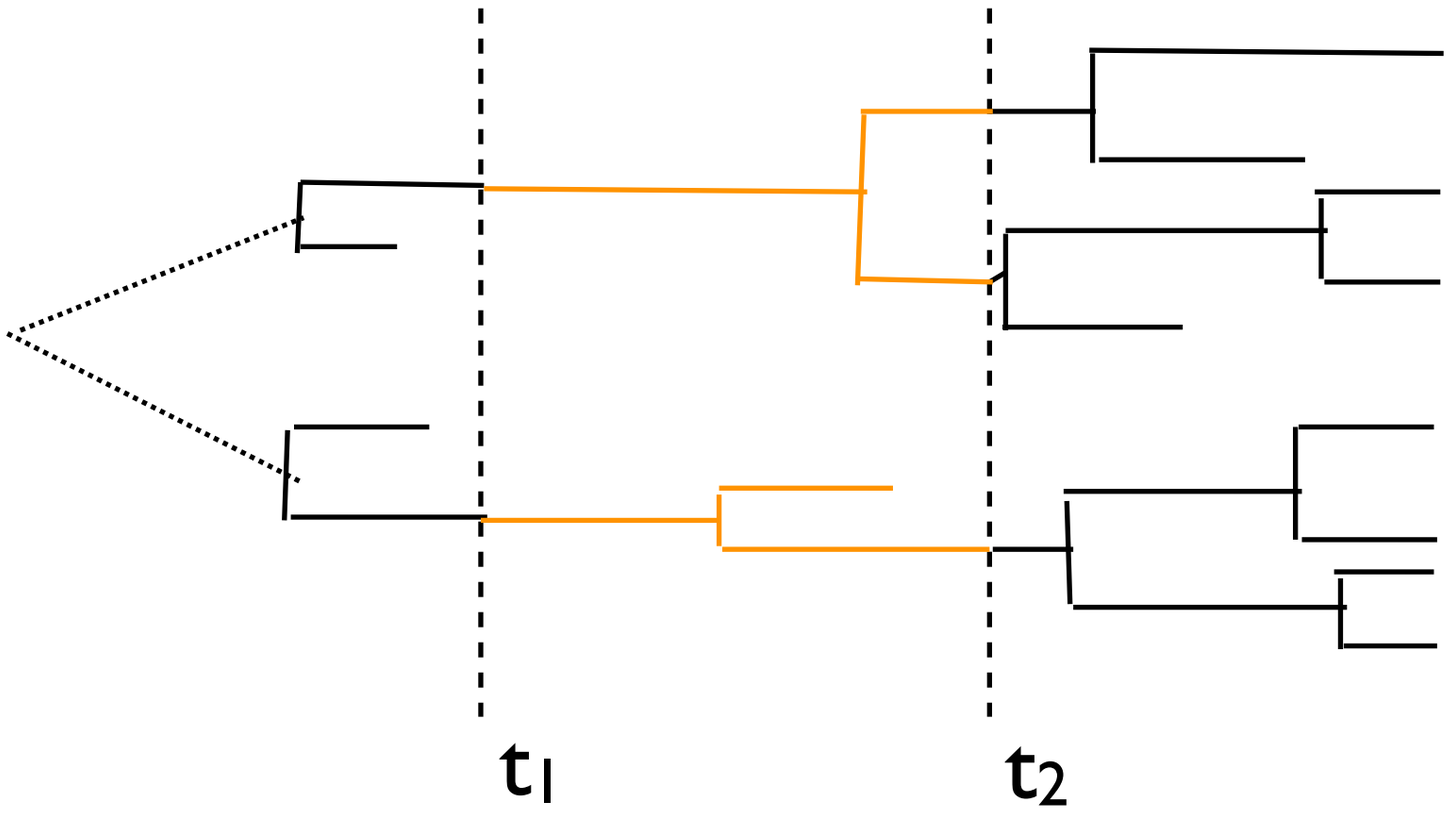
Pick two random times, delete everything in-between and generate a new trajectory to replace the deleted piece

‘Strip’ move



Pick two random times, delete everything in-between and generate a new trajectory to replace the deleted piece

‘Strip’ move

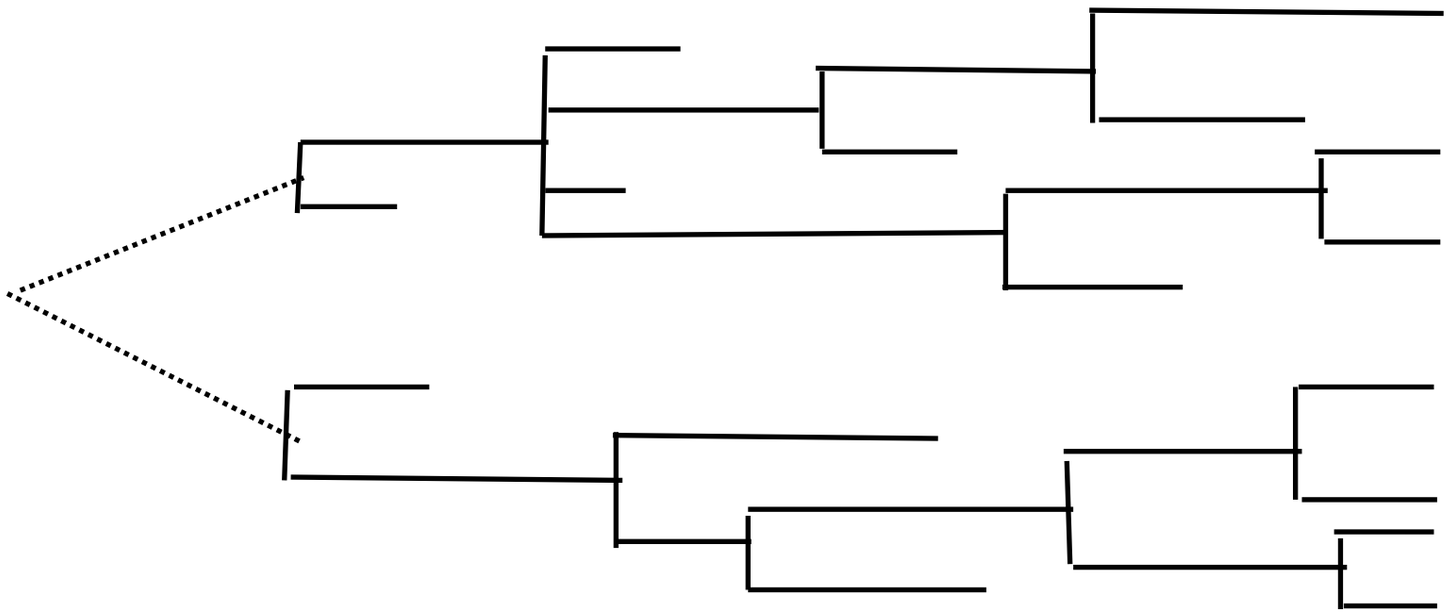


Pick two random times, delete everything in-between and generate a new trajectory to replace the deleted piece

How do they do that?

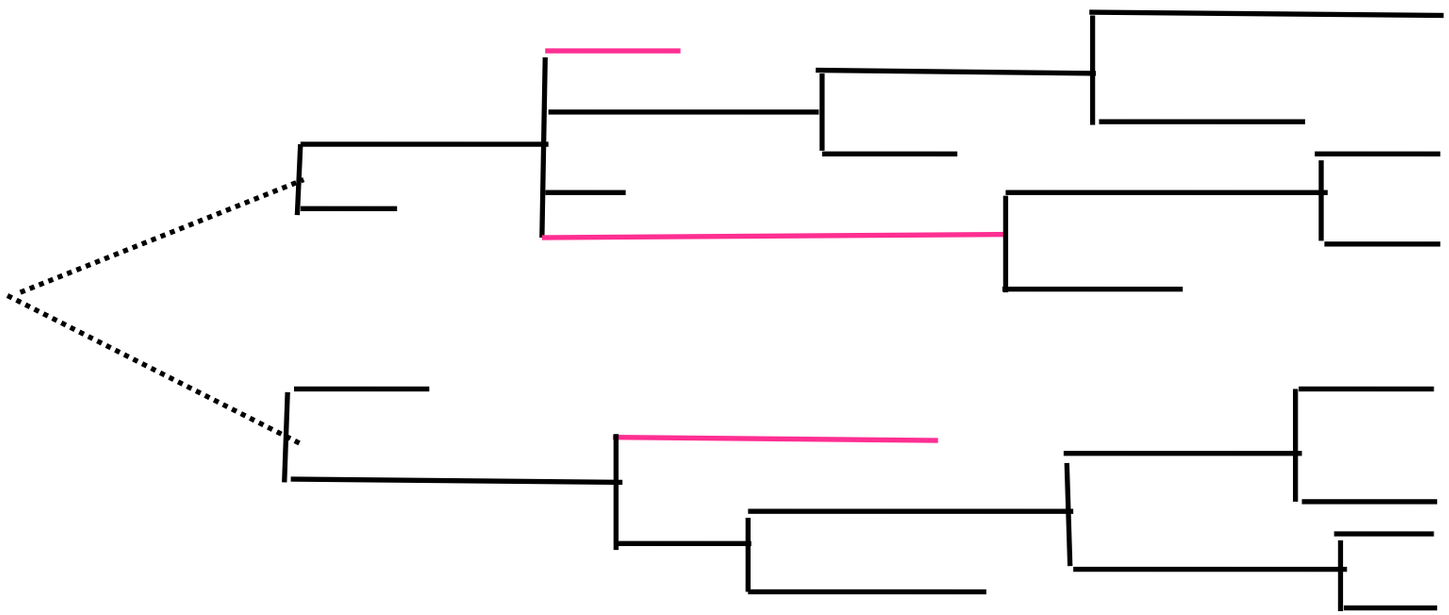
- Need to generate realization of B&D process with N_1 lines at t_1 and N_2 lines at t_2
- Use simulation:
  - In some cases, can simulate conditional on N_1, t_1, N_2 and t_2
  - In general, they use a rejection method ⇒ inefficient

‘Scale’ move



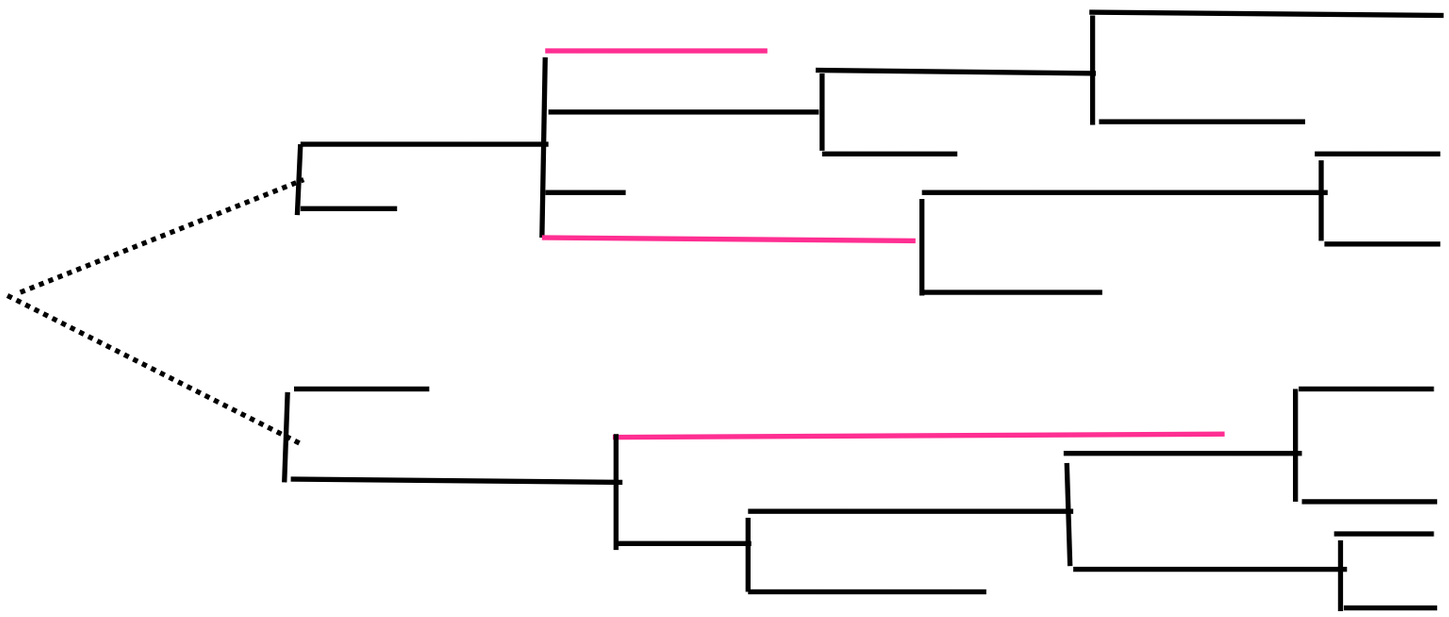
Update a fraction of the branch lengths

‘Scale’ move



Update a fraction of the branch lengths

‘Scale’ move



Update a fraction of the branch lengths

Change to τ &/or α

- These are updated along with the other moves.
- e.g. α is updated along with each birth/death move (add a Gaussian r.v.)

Results

- 1200 observations, by sampling every 250 iterations
- divergence time interval of (68.1, 99.1) Myr

Table 4. Summary statistics for τ, $\bar{\alpha}$ from MCMC

	τ	$\bar{\alpha}(\%)$
25th percentile	20.8	4.2
median	25.6	5.3
mean	26.6	6.1
75th percentile	31.3	7.2

Recall: TMRCA = 54 Myrs + τ
Result of earlier analysis: 95% CI for TMRCA is (69.1,115.8)

END