MA930 Data Analysis & Machine Learning

Lecture 6: Advanced Bayesian inference for time-series data

Haoran Ni

Recap from last time

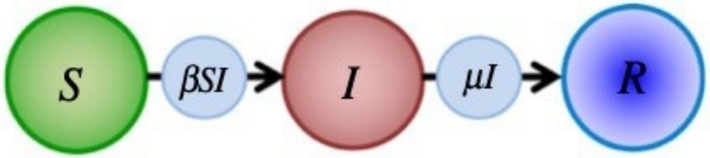
- **Curve fitting:**
  - Linear fit
  - Polynomial fit
- **Stochastic time-series models:**
  - Autoregressive models
  - Bayesian inference for autoregressive models

Outline

- Example: Bayesian inference for epidemiological time series
- Markov chain Monte Carlo (MCMC)
- Approximate Bayesian Computation (ABC)

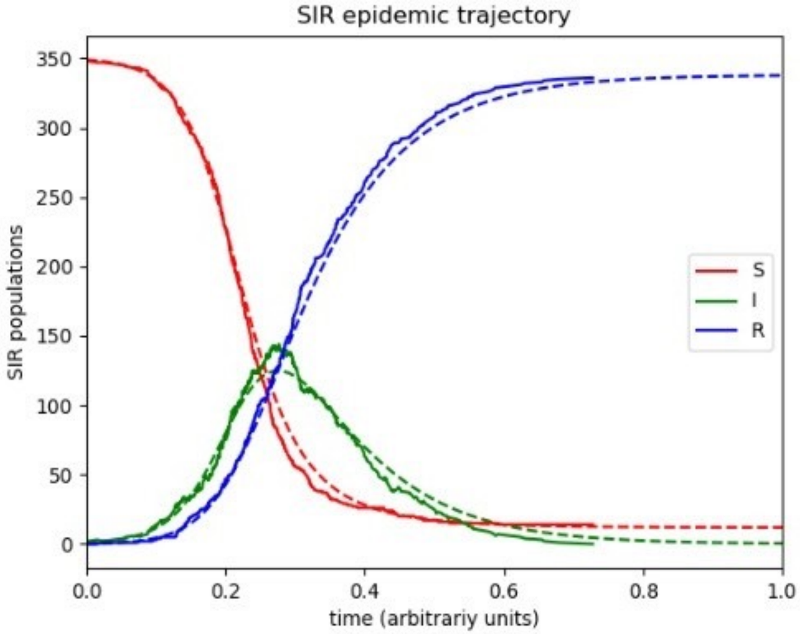
Overall goal: Understand that different advanced methods of Bayesian inference exist, and understand when ABC and MCMC should be used (so that you can learn relevant methods if ever required in your research).

Example: Bayesian inference for epidemiological time series

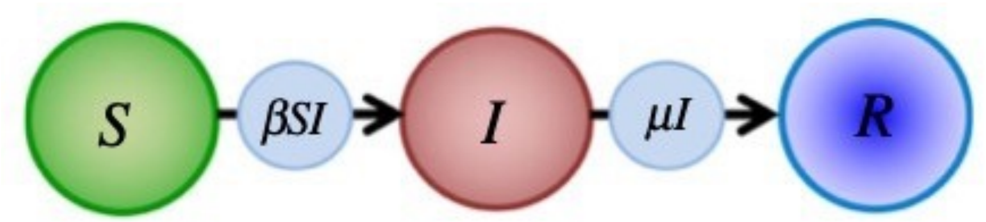


Simulation

Parameter inference

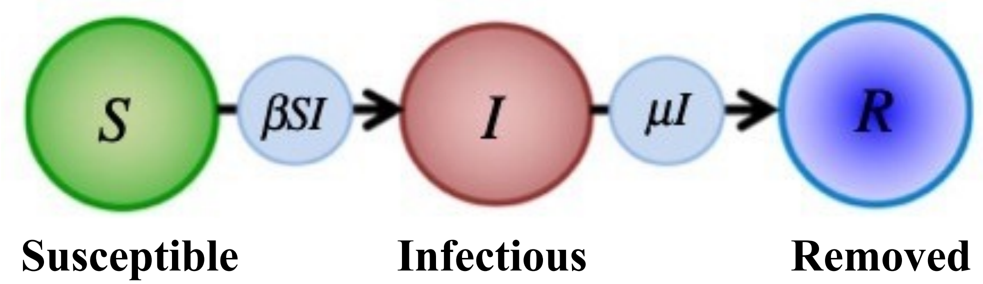


Example: Bayesian inference for epidemiological time series



Susceptible **Infectious** **Removed**

Example: Bayesian inference for epidemiological time series



Susceptible **Infectious** **Removed**

- To simulate a stochastic version of this model (using the Gillespie direct method):

- Assume that events occur at exponentially distributed time intervals. The time until the next event therefore follows an exponential distribution with rate parameter $\beta SI + \mu I$

- The probability that the next event is an infection is:

$$\text{Prob(infection)} = \frac{\beta SI}{\beta SI+\mu I}. \text{ Similarly, Prob removal} = \frac{\mu I}{\beta SI+\mu I}$$

Example: Bayesian inference for epidemiological time series

1. Initialise the number of individuals in each of the S, I and R classes in the model, and set the outbreak time $t = 0$.
2. Steps 2-4 should be repeated while the outbreak is still ongoing (i.e. $I > 0$). First calculate two random numbers r_1, r_2 each uniformly distributed in $(0, 1)$.
3. Calculate the time of the next event from an exponential distribution. Set
$$t = t + \frac{1}{\beta SI + \mu I} \ln(1/r_1) \, .$$
4. Choose whether the next event is an infection event or removal event. If
$$r_2 < \frac{\beta SI}{\beta SI + \mu I}$$
, then the next event is an infection event, and so set S = S – 1 and I = I + 1. Otherwise set I = I – 1 and R = R + 1.

Example: Bayesian inference for epidemiological time series

For a ***complete dataset*** arising from a simulation of the stochastic SIR model, the data is then of the form:

Time (*t*)	***S***	***I***	***R***
0	999	1	0
0.24	998	2	0
0.36	998	1	1
0.76	997	2	1
1.01	996	3	1
	Etc.		

Example: Bayesian inference for epidemiological time series

- The likelihood function is given by a product over all events. Sim. to last time, if x_1 represents the initial conditions (e.g. (S,I,R) = (999,1,0) at t=0), then x_2 represents whatever happens next (e.g. next event is (999,1,0) → (998,2,0) at $t = 0.24$) etc.
- If there are 4 states (i.e. 3 events + initial condition), then:

$L(\beta, \mu) = \Pr(x_1, x_2, x_3, x_4 | \beta, \mu) = f(x_4|x_3) f(x_3|x_2) f(x_2|x_1)$.

- If event *i* is an infection event, then

$f(x_i|x_{i-1})$

$$= \frac{\beta S_{i-1} I_{i-1}}{\beta S_{i-1} I_{i-1} + \mu I_{i-1}} \times (\beta S_{i-1} I_{i-1} + \mu I_{i-1}) \exp(-(\beta S_{i-1} I_{i-1} + \mu I_{i-1})(t_i - t_{i-1}))$$

where e.g. I_{i-1} represents the number of infected individuals immediately before event *i*.

Example: Bayesian inference for epidemiological time series

- The likelihood function is given by a product over all events. Sim. to last time, if x_1 represents the initial conditions (e.g. (S,I,R) = (999,1,0) at t=0), then x_2 represents whatever happens next (e.g. next event is (999,1,0) → (998,2,0) at $t = 0.24$) etc.
- If there are 4 states (i.e. 3 events + initial condition), then:

$$L(\beta,\mu) = \Pr(x_1, x_2, x_3, x_4 | \beta, \mu) = f(x_4|x_3)f(x_3|x_2)f(x_2|x_1).$$

- If instead event *i* is a removal event, then

$$f(x_i|x_{i-1})$$

$$= \frac{\mu I_{i-1}}{\beta S_{i-1} I_{i-1} + \mu I_{i-1}} \times (\beta S_{i-1} I_{i-1} + \mu I_{i-1}) \exp(-(\beta S_{i-1} I_{i-1} + \mu I_{i-1})(t_i - t_{i-1}))$$

where e.g. I_{i-1} represents the number of infected individuals immediately before event *i*.

Example: Bayesian inference for epidemiological time series

Exercise 6.1. Bayesian inference for stochastic SIR model
Run a simulation of the stochastic SIR model, starting with one infected individual out of a population of 1000 individuals (the rest of whom are susceptible), using the parameter values: β=0.0003, μ=0.1.Plot S vs t, I vs t and R vs t.

For that simulation, plot the joint log-likelihood function for β and μ (consider possible values for β between 0.0002 and 0.0004 and μ between 0.05 and 0.15), the joint likelihood function, and plot the marginal densities.

N.b. If the disease in your simulation dies out straight away, you may have to rerun your code until you have a simulation that does not die out. As in the last lecture, numerics are improved by calculating the log-likelihood and exponentiating at the end.

Markov chain Monte Carlo

Markov chain Monte Carlo

Markov chain Monte Carlo

- Calculating the likelihood function at *every* possible parameter value can be slow (particularly when estimating large numbers of parameters simultaneously, or when considering a wide parameter range).

- For that reason, efficient ways to explore the parameter space and narrow in on the correct parameter estimate are required.

- Markov chain Monte Carlo (MCMC): For when the likelihood function can be written down, but the parameter space to explore is large.

Markov chain Monte Carlo

- Suppose that we wish to estimate the value of β in exercise 5.7 (and that the value of μ is known exactly). We use the following algorithm:

1. Make an initial guess: $\beta^{(0)} = 0.0005$
2. Sample a new candidate guess: $\beta' = \beta^{(0)} + \boxed{U(-0.0001, 0.0001)}$
3. Calculate the likelihood $L^{(0)} = L(\beta^{(0)})$ and $L' = L(\beta')$
4. Decide on the next value in the chain, $\beta^{(1)}$. To do this, then:
    - If $L' \geq L^{(0)}$, then set $\beta^{(1)} = \beta'$
    - If $L' < L^{(0)}$, then set $\beta^{(1)} = \beta'$ with probability $L' \,/\, L^{(0)}$; Otherwise set $\beta^{(1)} = \beta^{(0)}$
5. Repeat from step 2 to find $\beta^{(2)}$ from $\beta^{(1)}$

The proposal distribution. It is arbitrary (but it should be symmetric, to allow the parameter space to be explored without bias)

Markov chain Monte Carlo

- Suppose that we wish to estimate the value of β in exercise 5.7 (and that the value of μ is known exactly). We use the following algorithm:

1. Make an initial guess: $\beta^{(0)}= 0.0005$
2. Sample a new candidate guess: $\beta' = \beta^{(0)} + U(-0.0001,0.0001)$
3. Calculate the likelihood $L^{(0)} = L(\beta^{(0)})$ and $L' = L(\beta')$
4. Decide on the next value in the chain, $\beta^{(1)}$. To do this, then:
    - If $L' \geq L^{(0)}$, then set $\beta^{(1)} = \beta'$
    - If $L' < L^{(0)}$, then set $\beta^{(1)} = \beta'$ with probability $L' \,/\, L^{(0)}$; Otherwise set $\beta^{(1)} = \beta^{(0)}$
5. Repeat from step 2 to find $\beta^{(2)}$ from $\beta^{(1)}$

The accept-reject step. The chain either stays where it is, or updates to the candidate value

The chain stays at more likely values of β for longer

Markov chain Monte Carlo

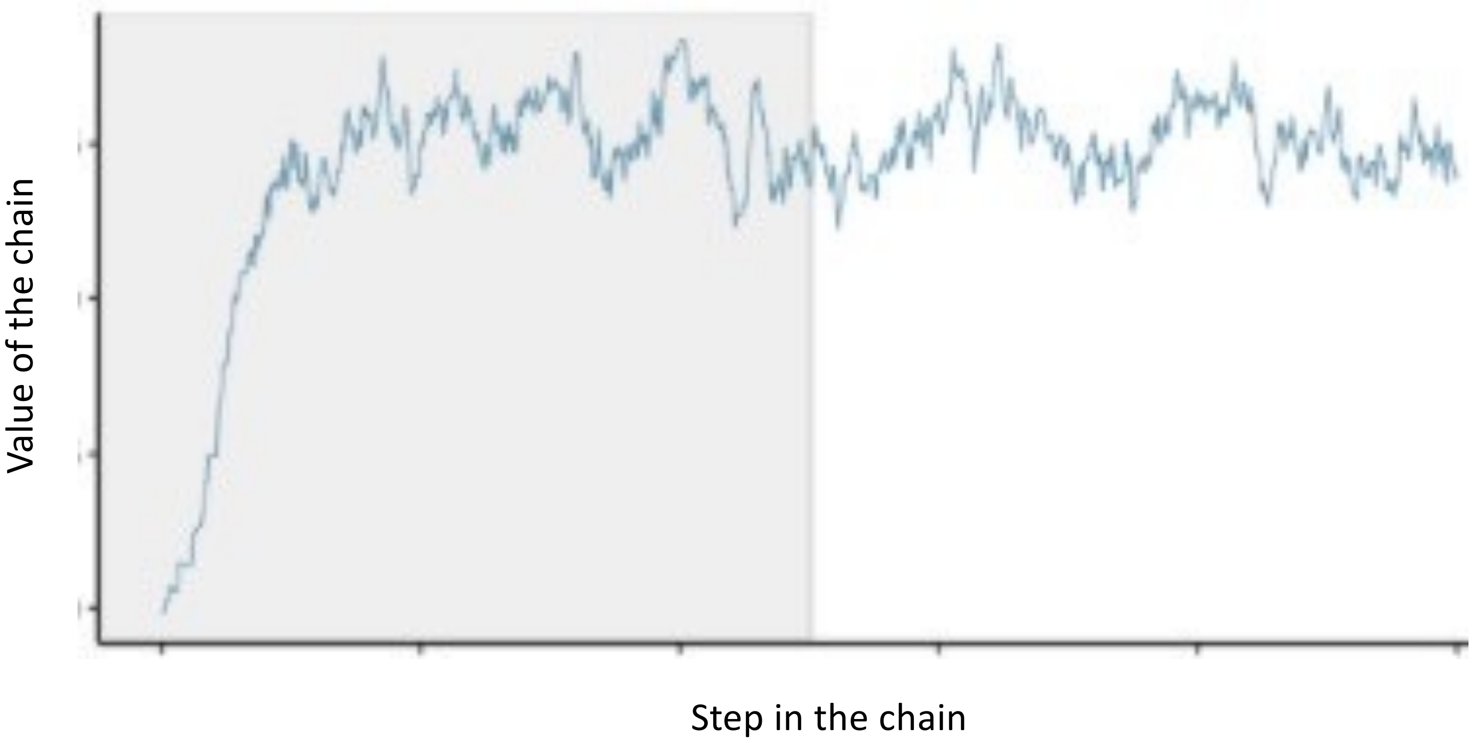
- Suppose that we wish to estimate the value of β in exercise 5.7 (and that the value of μ is known exactly). We use the following algorithm:

1. Make an initial guess: $\beta^{(0)} = 0.0005$
2. Sample a new candidate guess: $\beta' = \beta^{(0)} + U(-0.0001, 0.0001)$
3. Calculate the likelihood $L^{(0)} = L(\beta^{(0)})$ and $L' = L(\beta')$
4. Decide on the next value in the chain, $\beta^{(1)}$. To do this, then:
   - If $L' \geq L^{(0)}$, then set $\beta^{(1)} = \beta'$
   - If $L' < L^{(0)}$, then set $\beta^{(1)} = \beta'$ with probability $L' \,/\, L^{(0)}$; Otherwise set $\beta^{(1)} = \beta^{(0)}$
5. Repeat from step 2 to find $\beta^{(2)}$ from $\beta^{(1)}$

Again, better to work with log-likelihoods where possible for numerical stability: then exponentiate the log probability!

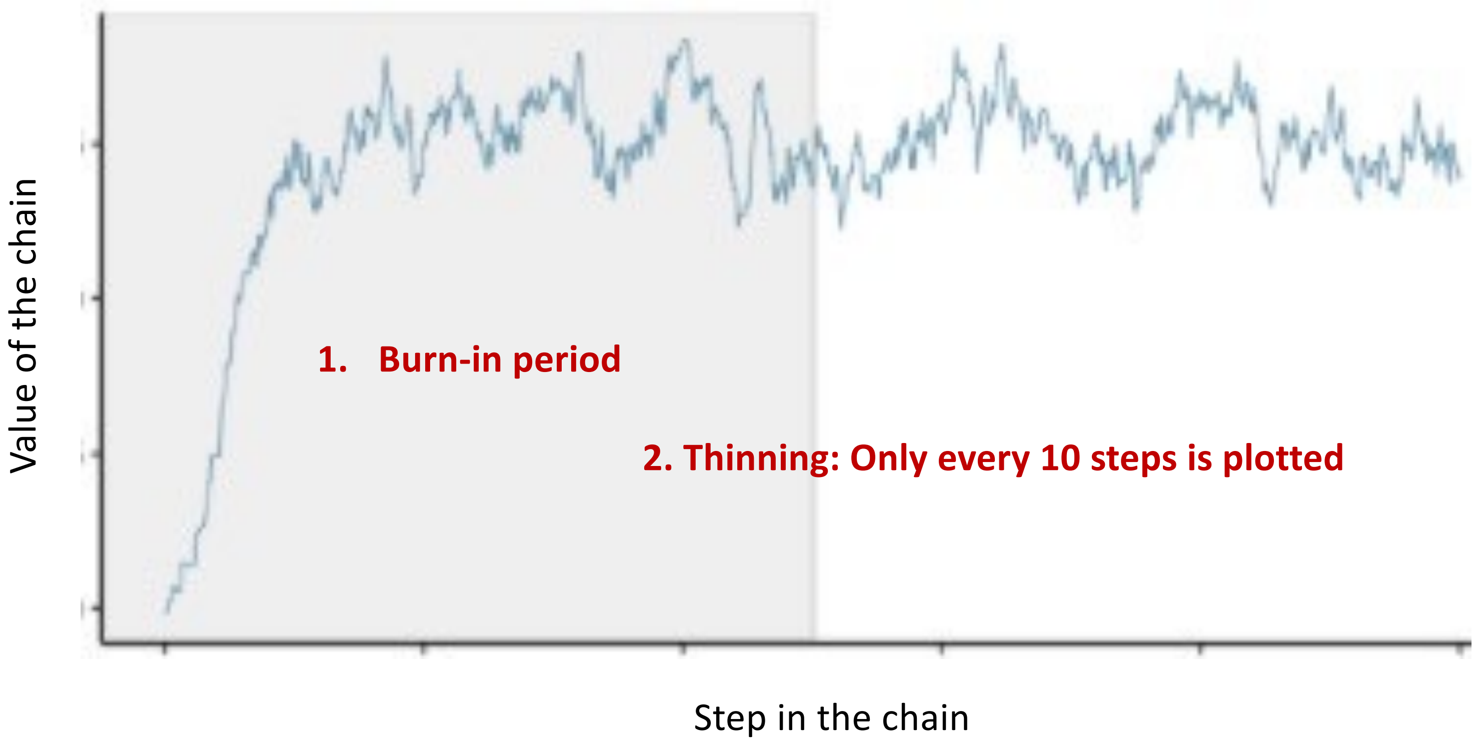
Markov chain Monte Carlo

- The output of an MCMC algorithm typically looks a bit like this:



Markov chain Monte Carlo

- The output of an MCMC algorithm typically looks a bit like this:



1. Burn-in period

2. Thinning: Only every 10 steps is plotted

Exercise 6.2. MCMC for stochastic SIR model

Run a simulation of the stochastic SIR model, starting with one infected individual out of a population of 1000 individuals (the rest of whom are susceptible), using the parameter values: β=0.0003, μ=0.1. Plot S vs t, I vs t and R vs t. This is the dataset.

Assume that the value of β is unknown (but that μ is known exactly). For the dataset, code up the Metropolis algorithm described earlier to estimate β.

i) Run the chain for 10,000 steps, and plot the value of the chain every 10 steps (the trace plot) – giving 1,000 potential estimates of β

ii) Discard the first 10% of the chain as burn in, and plot a histogram of the second 90% of the values plotted above (this is an approximation of the likelihood – which becomes exact as the number of steps tends to infinity (beyond the scope of this course!))

Approximate Bayesian Computation

Approximate Bayesian Computation

Approximate Bayesian Computation

- For some stochastic time series models, it may be difficult (or impossible) to write down the likelihood function

- To estimate parameters in a Bayesian framework, an approach for approximating the likelihood is needed

- If you are able to *simulate* the model, then it is possible to use Approximate Bayesian Computation (ABC)

- Idea: Simply run the model lots of times for different parameter values, and see which parameter values lead to simulations that look like the data

Approximate Bayesian Computation

- Unfortunately, it is unlikely that any simulation looks *exactly* like the data

- For that reason, ABC rejection sampling involves the following steps:

  1. Pick parameter values at random (possibly from the prior, if want to obtain a posterior)

  2. Run a model simulation

  3. If the simulation is within ε of the data (for some notion of distance between the simulation result and the data), then store those parameter values

  4. Repeat steps 1-3 until you have a large number of “matching” parameter sets, and use those sets to construct a distributional (normalised) estimate of the parameter values

Approximate Bayesian Computation

- ABC rejection sampling:

  1. Pick parameter values at random (possibly from the prior, if want to obtain a posterior)

  2. Run a model simulation

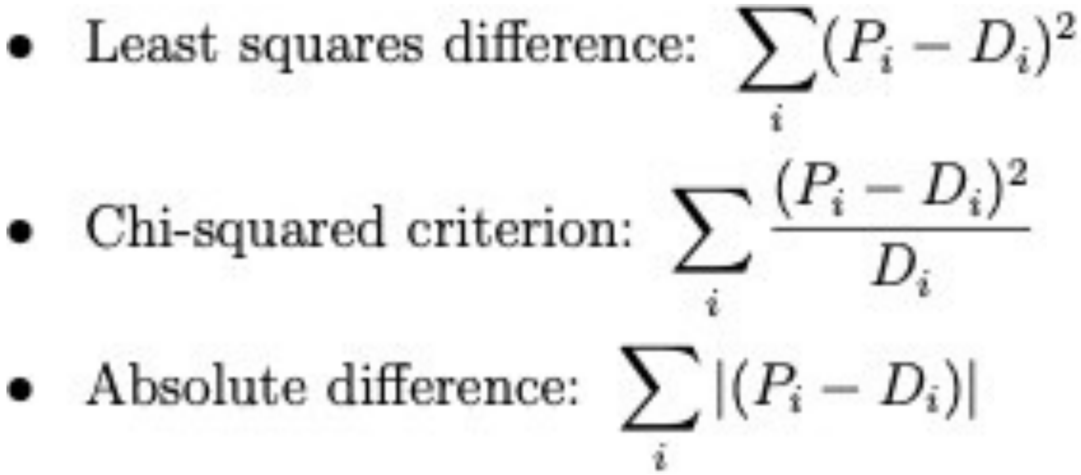
  3. If the simulation is within ε of the data (for some notion of distance between the simulation result and the data), then store those parameter values

  4. Repeat steps 1-3 until you have a large number of "matching" parameter sets, and use those sets to construct a distributional (normalised) estimate of the parameter values

If you run an infinite number of simulations, then in the limit $\varepsilon \to 0$ you retrieve exactly the likelihood function (this is because the probability of ***exactly*** retrieving the data is simply $\Pr(\text{data} \mid \text{parameters})$, which is the likelihood).

Approximate Bayesian Computation

Possible distance metrics include:



- Least squares difference: $\sum_i (P_i - D_i)^2$
- Chi-squared criterion: $\sum_i \frac{(P_i - D_i)^2}{D_i}$
- Absolute difference: $\sum_i |(P_i - D_i)|$

Approximate Bayesian Computation

Exercise 6.3. Bayesian inference using ABC for stochastic SIR model

Download the code from Teams that runs a simulation of the stochastic SIR model and records the value of I at $t = 0, 10, 20, 30, \ldots, 100$ (or write your own code).

Run one simulation (the dataset) using the parameter values: $\beta = 0.0003$, $\mu = 0.1$.

Pretend that the underlying value of β is unknown. Modify the code to run a large number of simulations to estimate the value of β from the dataset using ABC. Specifically, fit the model to data on I vs t, by repeatedly sampling possible values of β from the range [0.0002, 0.0004], assuming that the value of μ is known exactly. For the distance metric, use the chi-squared criterion with $\varepsilon = 50$ (and aim for 40,000 matching simulations).

n.b. In theory, you could estimate β and μ simultaneously (although this would require even more simulations...!)

Approximate Bayesian Computation

Summary:

- A range of Bayesian approaches exist for estimating parameters of simulation models from time series data
- If the likelihood function is known and the parameter space to explore is small −> calculate likelihood/posterior directly
- If the likelihood function is known but the parameter space is large −> MCMC
- If the likelihood function is unknown or complex to calculate −> ABC
- Many other approaches exist, but these are simple versions of some of the most common...