Statistical Inference 2 Summer Term 2025

Lecture: Prof. Dr. Göran Kauermann
Exercises: Sergio Buttazzo, Jan Anders
Tutorial: Eugen Gorich

Exercise Sheet 5

Exercise 5.1 - ABC

Sometimes, the exact likelihood of some obtained data can be unknown or difficult to calculate. Nevertheless, we want to be able to find the (posterior) distribution of the parameters of our data generating process. Approximate Bayesian Computation (ABC) is a way of computing the posterior in such settings, as long as data can be simulated using a set of proposed parameters from the prior. For example, some economists assume that random fluctuations on stock markets can be modelled using geometric Brownian motion, i.e. a random walk, which has parameters $\mu = 0$, σ and t. Geometric Brownian motion is a continuous-time stochastic process and is defined as $S_t = S_0 \cdot \exp\left(\left(\mu - \frac{\sigma^2}{2}\right) t + \sigma W_t\right)$, where μ is the drift, σ the volatility, and W_t is a standard Brownian motion. In discrete approximation, increments are simulated as $\Delta S_t \sim \mathcal{N}\left(\left(\mu - \frac{\sigma^2}{2}\right)\Delta t, \sigma^2 \Delta t\right)$.
In this exercise, we will consider the `EuStockMarkets` dataset that comes preloaded with base `R`. More precisely, we will try to model the noise in a vector of the daily closing price of the DAX in 1994.

(a) Fill the gaps in the code skeleton `Ex_5_1_skeleton.R` provided on Moodle to implement ABC to generate an approximate sample from the posterior distribution for μ, σ and t that best models the random behaviour of the DAX in 1994. Implement the following prior: $\mu \sim N(0,1)$, $\sigma \sim Unif(0,1)$, $t \sim Unif(0.001, 0.01)$.
As distance function, use the mean of the squared distances between the simulated and real data points. You may also try out other priors and distance metrics.

(b) Does the posterior distribution accurately capture our prior belief that this segment of the DAX follows a driftless geometric Brownian motion, i.e. $\mu^* = 0$? How could you test for this?

(c) Visually inspect results from your posterior by plotting the actual historic stock prizes next to a few simulated stock prizes, using the posterior means as parameters for the simulation. Do the simulations model the actual movement of the stock index? Why not?

(d) What happens if we lower the acceptance threshold ϵ? What distribution does our posterior take for $\epsilon \to \infty$?

Exercise 5.2 - Local Monte Carlo Sampling

We consider an unknown function $f : [0,1] \to \mathbb{R}$ that is expensive to evaluate. You are given a small number of $k = 1 \dots 8$ evaluations $(\boldsymbol{x_{known}}, \boldsymbol{y_{known}})$, where $y_k = f(x_k)$. The goal is to simulate plausible versions of f on a grid and quantify uncertainty about its shape.
The known observations of f are:

```
x_known <- c(0.1, 0.12, 0.18, 0.43, 0.51, 0.68, 0.86, 0.95)
y_known <- c(0.3750,  0.3015,  0.0927,  0.2374, -0.5391, -0.0649,  0.3946, -0.4626)
```

Define a grid $\boldsymbol{x}^* = (\tilde{x}_1, \dots, \tilde{x}_m)$ of $i = 1 \dots m$ equidistant points in $(a, b) \subset [0,1]$, such that all known x_k are contained in $\boldsymbol{x}^*$. We wish to simulate J possible discretized versions of f on $\boldsymbol{x}^*$. Let $\boldsymbol{y}^{*(j)} = (y_1^{*(j)}, \dots, y_m^{*(j)})$ denote the j-th simulation of f on the grid $\boldsymbol{x}^*$, for $j = 1, \dots, J$. Use the following scheme:

1. If $j = 1$, set $y_i^{*(j)}$ for each i by drawing i.i.d from a normal distribution with sample mean and variance estimated from $\boldsymbol{y_{known}}$.

2. For each simulation $j = 2, \dots, J$, loop over $\tilde{x}_i$, starting with $i = 1$. For each grid point $\tilde{x}_i \in \boldsymbol{x}^*$, set

$$y_i^{*(j)} = \begin{cases} y_k & \text{if } \tilde{x}_i = x_k \text{ for some known } (x_k, y_k), \\ \mathcal{N}(\mu_i^{(j)}, \sigma^2) & \text{otherwise,} \end{cases}$$

where

$$\mu_i^{(j)} = \begin{cases} y_{i+1}^{*(j-1)} & \text{if } i = 1, \\ y_{i-1}^{*(j)} & \text{if } i = m, \\ \frac{1}{2}(y_{i-1}^{*(j)} + y_{i+1}^{*(j-1)}) & \text{otherwise.} \end{cases}$$

In words, we use the most recent available neighbours in the simulation: for indices already filled in the current iteration j, use a value from $\boldsymbol{y}^{*(j)}$, otherwise use one from the previous simulation $\boldsymbol{y}^{*(j-1)}$.

(a) Implement the above procedure in R to simulate S versions of $f(\boldsymbol{x}^*)$ on the grid. What do you have to keep in mind to obtain S independent samples? Choose reasonable values for m, a, b, J, and σ.

(b) Use your function to generate at least $S = 250$ independent simulations $\boldsymbol{y}^{*(s)}$, $s = 1, \dots, S$. Plot all simulated curves $(\boldsymbol{x}^*, \boldsymbol{y}^{*(s)})$ as line plots in a single figure. Highlight the known data points (x_k, y_k) using distinct markers (e.g., dots). Comment on the resulting visualization. What does it reveal about the uncertainty in the function's values across different regions of the domain? How could you formally quantify the uncertainty at each grid point x_i^*?

(c) Suppose you reverse the simulation order and proceed in descending order through the x_i^*, so starting with $i = m$. Would the resulting simulated functions be statistically different? Why or why not?

Exercise 5.3 - True or False

True or **False**? Adequately justify your answers.

(a) Bayesian statistics can handle small sample sizes more effectively than frequentist statistics by incorporating prior information, while frequentist statistics requires larger sample sizes for reliable inference.

(b) Markov Chain Monte Carlo (MCMC) methods are Bayesian.

(c) Approximate Bayes Computation (ABC) is useful for obtaining an approximate posterior distribution for a parameter θ in situations where the model, i.e. the family of the data generating process, is not known.

(d) When performing ABC, the choice of the distance metric has a strong impact on both the quality of the approximation of the posterior as well as the number of iterations needed to obtain it.

(e) Bootstrap confidence intervals achieve the nominal coverage level (e.g. 95%) regardless of the underlying population distribution.