Part 2f: Sequential simulation

Textbook: pp. 48-49

Introduction to sequential sampling

- The joint density can be factored as a product of conditional densities.
- In some cases it is easier to simulate from conditional densities.
- A sample from the joint distribution can be produced via sequential sampling from conditional densities.

An algorithm for sequential simulation

Algorithm 2.11 (componentwise simulation)

input:

marginal density p_{X_1}

conditional densities $p_{X_i|X_1,\dots,X_{i-1}}$ for $i = 2, 3, \dots, n$

randomness used:

samples from p_{X_1} and $p_{X_i|X_1,\dots,X_{i-1}}$

output:

a sample $(X_1, \dots, X_n) \sim p$

1: generate $X_1 \sim p_{X_1}$

2: **for** $i = 2, 3, \dots, n$ **do**

3: generate $X_i \sim p_{X_i|X_1,\dots,X_{i-1}}(\cdot \,|\, X_1, \dots, X_{i-1})$

4: **end for**

5: return $(X_1, \dots, X_n)$

The joint distribution

Lemma 2.12 The joint density of the vector $(X_1, X_2, \dots, X_n)$ produced by algorithm 2.11 is:

$$P_{X_1,\dots,X_n}(x_1,\dots,x_n)$$
$$= P_{X_1}(x_1)\cdot P_{X_2|X_1}(x_2|x_1)\cdots P_{X_n|X_{n-1},\dots X_1}(x_n|x_{n-1}\cdots x_1)$$

$n=2 \Rightarrow \quad f_{X_1,X_2}(x_1,x_2) = f_{X_1}(x_1)\cdot f_{X_2|X_1}(x_2|x_1)$

Assume for n

$$f_{X_1,\dots,X_n}(x_1,\dots,x_n) = f_{X_1}(x_1)\cdot f_{X_2|X_1}(x_2|x_1)\cdots f_{X_n|X_{n-1},\dots,X_1}(x_n|x_{n-1},\dots,x_1)$$

$n+1 \Rightarrow \quad f_{X_1,\dots,X_{n+1}}(x_1,\dots,x_{n+1}) = f_{X_1,\dots,X_n}(x_1,\dots,x_n)\cdot f_{X_{n+1}|X_n,\dots,X_1}(x_{n+1}|x_n,\dots,x_1)$

$$= f_{X_1}(x_1)\cdots f_{X_n|X_{n-1},\dots,X_1}(x_n|x_{n-1},\dots,x_1)\cdot f_{X_{n+1}|X_n,\dots,X_1}(x_{n+1}|x_n,\dots,x_1)$$

Simulating the AR(1) process

The AR(1) process is defined via the recursion:

$$X_n = a_1 \cdot X_{n-1} + \epsilon_n,$$

where $X_1 \sim N(0, \tau^2)$ and $\epsilon_i \sim N(0, \sigma^2)$, $2 \le i \le n$, are independent of each other.

$$f_{X_n | X_{n-1}, \ldots X_1}(x_n | x_{n-1}, \ldots, x_1) = f_{X_n | X_{n-1}}(x_n | x_{n-1}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_n - a_1 x_{n-1})^2}{2\sigma^2}}$$

1: generate $x_1 \sim N(0, \tau^2)$

2: for $i = 2, 3, \ldots, n$

3: generate $x_i \sim N(a_1 x_{n-1}, \sigma^2)$

4: end for

5: return $(x_1, \ldots, x_n)$