From Data to Insights - Exercise sheet 6

discussed May 31

May 24, 2024

1 Implementing the Metropolis-Hastings Algorithm

A) Warm-up:

Show that the two equations on slide 15 of lecture 11 are equivalent to each other.

B) Describe the Metropolis-Hastings algorithm in your own words.

C) Implementing an actual chain:

The 2D probability density

$$\begin{aligned} p(x,y) &= \tfrac{1}{4\pi}\exp\left(-\tfrac{1}{2}\left\{(x-\mu_{x,1})^2+(y-\mu_{y,1})^2\right\}\right) \\ &\quad +\tfrac{1}{4\pi}\exp\left(-\tfrac{1}{2}\left\{(x-\mu_{x,2})^2+(y-\mu_{y,2})^2\right\}\right) \end{aligned} \tag{1}$$

with $\mu_{x,1} = 0$, $\mu_{y,1} = 0$ and $\mu_{x,2} = 3$, $\mu_{y,2} = 3$ is just the averaged sum of two Gaussian PDFs. Implement the Metropolis-Hastings Algorithm for that distribution. You can use an isotropic Gaussian proposal distribution with standard deviations $\sigma_i = 0.3$.

2 Pseudo code for kernel density estimation (aka. exam preparation)

A) Consult the wikipedia page `https://en.wikipedia.org/wiki/Pseudocode` and describe in your own words the purpose and the characteristic features of *pseudo code*.

B) Assume that you have an array `random_draws` of independent random draws from some unknown PDF $p(x)$. Write pseudo code - **HANDWRITTEN, ON A SHEET OF PAPER** - that uses kernel density estimation to estimate that PDF. Assuming that $p(x)$ is not too far from Gaussian, what is the optimal kernel width that you should choose to minimize the error of your estimate?

C) Now turn your pseudo code into actual code, and use the data provided in `kde_N100.dat` of the lecture cloud as your array `random_draws`.

3 Discussion questions

A) Recall at least 3 potential problems of sampling a posterior density with the Metropolis-Hastings algorithm.

B) Recall: in what way is the task of kernel-density estimation another instance of the 'bias-versus-variance trade off'?