Stockholms universitet

Computational Statistics
Advanced Course, 7.5 HEC, Spring 2022
Home Assignment 2 until March 11

Mahmood Ul Hassan and Frank Miller, Department of Statistics

March 4, 2022

You are supposed to solve these problems individually without any cooperation. Record your answers clearly. If you cite material from other sources, or use intellectual ideas or code from others, point out this clearly with stating the source. Please include your R computer code and relevant output in your solution. Send an email to `mahmood.ul-hassan@stat.su.se` with your solutions until the above deadline with a file name like `CompStat2022HA2March_[Your name].pdf` and extra file(s) with similar name with your computer code.

1 Two-dimensional minimization

Consider the two dimensional function

$$f(x, y) = \sin(x + y) + (x - y)^2 - 1.5x + 2.5y + 1$$

for $x \in [-1.5, 4]$ and $y \in [-3, 4]$. The aim is to search for the position (x^*, y^*) of the global **minimum** of f in the given range for x and y.

a. Make a contour plot for this function.

b. Search (x^*, y^*) using the Newton algorithm. Describe choices for starting value(s) and stopping rules.

c. With the search in b., a global minimum is not detected if it is on the boundary of the region $[-1.5, 4] \times [-3, 4]$. Investigate each of the four boundary lines separately with one-dimensional minimization to check if the global minimum is attained on the boundary.

2 Integration with Simpson's 3/8 rule

In the lecture and computer labs, Riemann, Trapezoidal, and Simpson's rule were discussed as specific Newton-Cotes integration rules. Another Newton-Cotes rule is the so-called "Simpson's 3/8 rule". There, the integral on each subinterval $[x_i, x_{i+1}]$ is approximated by

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{3h}{8} \left\{ f(x_i) + 3f(x_i + h/3) + 3f(x_i + 2h/3) + f(x_{i+1}) \right\},$$

where $h = x_{i+1} - x_i$.

a. Program an algorithm to do integration with the Simpson's 3/8 rule. The number of nodes n can be fixed; you do not need to increase it until a convergence criterion is met. Note that an algorithm for Simpson's rule was shown in the computer labs and you might use parts of it and modify it to the Simpson's 3/8 rule.

b. With your program from a., integrate the density ϕ of the standard normal distribution between -2 and 2 ($P(-2 \le X \le 2) = \int_{-2}^{2} \phi(x)dx$ for a standard normal random variable X). Use $n = 10$ and $n = 20$ nodes. Compare the result with using R's `pnorm`-function for determining $P(-2 \le X \le 2)$.

3 Gauss-Hermite integration

Nodes 6-11 and the weights for the 11-point Gauss-Hermite integration are given in the following table. The nodes are symmetric around 0 which means that nodes 1-5 are the negative values of nodes 7-11 with the same weight (e.g. $x_4 = -x_8 = -1.32656$, $A_4 = A_8 = 0.117228$).

i	Nodes x_i	Weights A_i
6	0	0.654759
7	0.656810	0.429360
8	1.32656	0.117228
9	2.02595	0.0119114
10	2.78329	0.000346819
11	3.66847	0.00000143956

a. Plot the weights versus the 11 nodes.

b. Compute the integrals $\int_{-\infty}^{\infty} f(x)\, dx$ of the following density functions $f(x)$ using this 11-point Gauss-Hermite integration. Theoretically, we know that the exact integral is 1 and the result shows you how good the 11-point Gauss-Hermite performs in this situation.

   (i) $f(x) =$ the density of the t-distribution with 3 degrees of freedom,

   (ii) $f(x) = \frac{1}{2}e^{-|x|}$ (double-exponential distribution),

   (iii) $f(x) =$ the density of the normal distribution $N(\mu, 1)$, where $\mu = 0, 1, 2, 3$.

   Comment on these results.

4 EM algorithm for mixture of normal components

In the lectures, an EM algorithm was presented for the case of a univariate normal mixture model with two components; you can find also an R-text-file `emalg.r` with the code in Athena.

a. Use the algorithm from the lecture as start and modify it for the case of three components, i.e. the mixture of three normal distributions. Important: Use this provided algorithm to start with and generalize it; do not write a completely new code.

b. Generate $n = 800$ random observations from a normal mixture with mixing parameters $p_1 = 0.2, p_2 = 0.3, p_3 = 0.5$, mean values $\mu_1 = 2, \mu_2 = 4, \mu_3 = 5.5$, and standard deviations $\sigma_1 = 1.2, \sigma_2 = 1, \sigma_3 = 0.8$.

c. Fit the data from b. to a normal mixture model with three components using your program from a. Check convergence of the results by providing plots of current estimates for the model parameters versus iteration-number. Which estimates do you get for the model parameters? Are the estimates close to the true values which you used for data generation?