# UNIVERSITY OF OSLO

## Faculty of mathematics and natural sciences

Exam in: STK4051/STK9051 — Computational statistics

Day of examination: Thursday June 8th, 2023

Examination hours: 15.00 – 19.00.

This problem set consists of 6 pages

Permitted aids: None

Note the final sheet contains definitions of densities that are used in problems.

### Problem 1

Monte Carlo sampling is a technique for computing high dimensional integrals, such as

$$\int_{R^n} h(\boldsymbol{x})g(\boldsymbol{x})d\boldsymbol{x} \qquad (1)$$

a) Given a set of samples $(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_B)$ from the distribution $g(\boldsymbol{x})$. How would you approximate the integral in (1)? What is the condition required for convergence?

Using the samples from $g(\boldsymbol{x})$, we are now interested in computing the integral

$$\int_{R^n} h(\boldsymbol{x})f(\boldsymbol{x})\, d\boldsymbol{x} \qquad (2)$$

b) Develop a formula using samples from $g(\boldsymbol{x})$ to approximate the integral in (2). What is the condition needed for this modified scheme to converge. Finally, assume $q(\boldsymbol{x})$ to be proportional to $f(\boldsymbol{x})$, i. e. $q(\boldsymbol{x}) = c \cdot f(\boldsymbol{x})$ where c is unknown. How would you then perform the estimate? Which names do we use for these two approaches?

In a simulation study we want to compute the probability of exceedance, $P(X > x_0)$ in a standard normal distribution. We propose two different algorithms:

**Algorithm 1:**
for $i = 1, \dots, B$

1) Sample $u_i \sim$ Uniform$[u, 0,1]$

2) Compute $x_i = \Phi^{-1}(u_i)$ ,

with $\Phi^{-1}(\cdot)$ being the quantile function of the standard normal distribution.

Define the approximate exceedance probability as follows:

$$P_1(x_0) = \frac{1}{B}\sum_{i=1}^{B} I(x_i > x_0) \quad (3)$$

**Algorithm 2:**
for $i = 1, \dots, B$

1) Sample $u_i \sim$ Uniform$[u, 0,1]$

2) Compute $x_i = x_0 - \log u_i$

Define the approximate exceedance probability as follows (see page 6 for definitions of distributions in the expression):

$$P_2(x_0) = \frac{1}{B}\sum_{i=1}^{B} \frac{\phi(x_i; 0,1)}{\text{Exp}(|x_i - x_0|; 1)} \quad (4)$$

c) Explain what happens in step 2 for both algorithms, i.e. What is the distribution of $x_i$ in the two algorithms? Argue why both methods approximate the desired exceedance probability.

Figure 1 below shows the estimate as a function of $x_0$, using B=1000 in both algorithms. The two algorithms have been run independently for several values of $x_0$to create the figure.

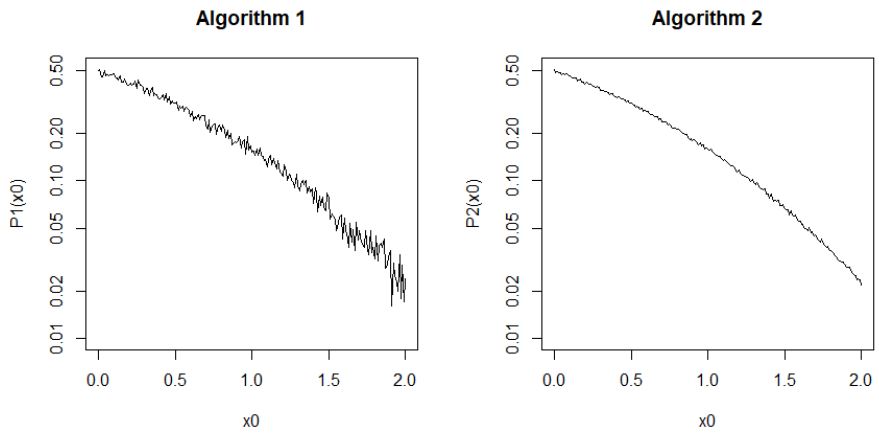


Figure 1: Estimation of exceedance probability using algorithm 1(left) and algorithm 2(right).

d) Which algorithm would you prefer? Argue the case both from the figure and from a theoretical perspective. The estimates contain some jitter creating a rugged appearance. Which well-known phenomenon creates this jitter? How would you change your selected algorithm so that the function get a smoother appearance?

## Problem 2

Consider a model from waiting time analysis, where we record the waiting time between two events. When the events occur according to a constant rate, the waiting times will follow an exponential distribution $\text{Exp}(x;\lambda) = \text{Erlang}(x;1,\lambda)$, with $\lambda$ being a parameter. If the sensor which record the events occasionally malfunction and miss out of one event, then the recorded waiting time is the sum of two exponential distributions, which follows an Erlang distribution $\text{Erlang}(x;2,\lambda)$. For simplicity, we will disregard the possibility of missing out on multiple events in succession. Let C denote which class a recorded data come from, if $C = 1$ then it comes from the $\text{Exp}(x;\lambda)$ distribution, if $C = 2$ then it comes from the $\text{Erlang}(x;2,\lambda)$ distribution. We do not know which class observations belong to.

Consider the following model for class and waiting time:

$$\text{Prob}(C_i = 1) = p, \;\; p(x_i|C_i = 1) = \text{Exp}(x;\lambda)$$
$$\text{Prob}(C_i = 2) = 1 - p, \;\; p(x_i|C_i = 2) = \text{Erlang}(x;2,\lambda)$$

a) Write down the likelihood for $p$, and $\lambda$ for the waiting time observations according to this model. An implementation of the quasi-Newton algorithm BFGS is applied to find the Maximum Likelihood Estimate of $\theta = (p,\lambda)$. The results for 10 random start points are shown in Table 1. What is special with quasi-Newton methods in comparison with Newton methods? Comment on the results in Table 1. Is there an issue with the results? What could you do to avoid this problem in the code?

Table1: Each column presents the results of one run of the BFGS algorithm, the different runs have different start points. Bottom row shows the estimate of the likelihood. Row two and three shows the estimated values after convergence.

| Run | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| MLE p | 1,110 | 0,731 | 1,110 | 1,110 | 0,732 | 1,110 | 0,732 | 0,731 | 0,731 | 1,110 |
| MLE $\lambda$ | 1,461 | 2,083 | 1,461 | 1,461 | 2,082 | 1,461 | 2,083 | 2,083 | 2,083 | 1,461 |
| log ML | -503,85 | -503,05 | -503,85 | -503,85 | -503,05 | -503,85 | -503,05 | -503,05 | -503,05 | -503,85 |

b) Rather than optimizing the likelihood, we could use the EM algorithm. Introduce $C_i$ as a hidden variable and write down the complete log-likelihood $l(p, \lambda | \boldsymbol{x}, \boldsymbol{C})$. Define the Q function and show that it becomes:

$$Q(p, \lambda | p^{(t)}, \lambda^{(t)})$$

$$= \sum_{i=1}^{n} P(C_i = 1 | x_i, p^{(t)}, \lambda^{(t)}) [\log p + \log \lambda - \lambda x_i] \quad (5)$$

$$+ \sum_{i=1}^{n} P(C_i = 2 | x_i, p^{(t)}, \lambda^{(t)}) [\log(1-p) + 2 \log \lambda + \log (x_i) - \lambda x_i]$$

Derive the expression for $P(C_i = 1 | x_i, p, \lambda)$.

c) Derive the formula for updating $p$ and $\lambda$ in the EM algorithm and write down the full EM algorithm for the problem. (If you did not get an expression for $P(C_i = 1 | x_i, p, \lambda)$, just assume this can be evaluated) Which property does the EM algorithm have in terms of convergence.

d) How would you apply a non-parametric bootstrap to evaluate the sample variance in the estimator from the EM algorithm? You can assume that you already have a function which returns the estimates of $(p, \lambda)$, from a data set $\boldsymbol{x}$. Figure 2 show the results from a bootstrap applied to the estimates of $(p, \lambda)$. Discuss the results.

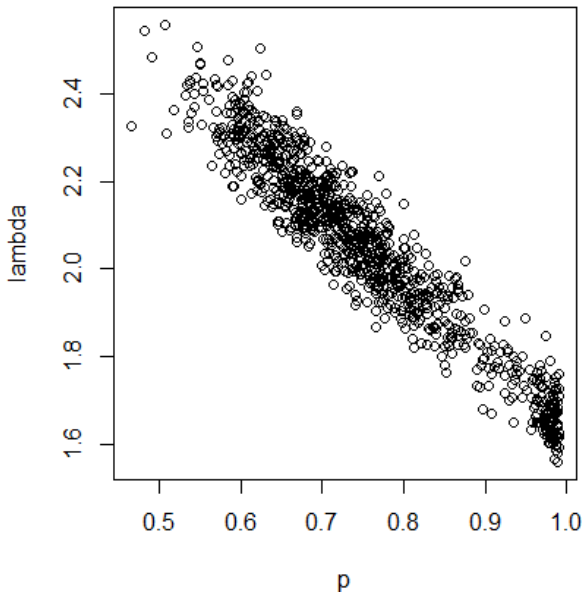


Figure 2: Results for bootstrapping uncertainty in the MLE using EM algorithm.

## Problem 3

A nuclear power plant has 10 pumps. We are interested in the failure of the pumps, and model this using a poison process. We assume that each pump has its own failure rate $\lambda_i$, The failure rates for the individual pumps are independent, but they have the same prior distribution: $\text{Gamma}(\lambda_i; \alpha, \beta)$. We assume α to be fixed, but β to be random coming from a $\text{Gamma}(\beta;\ \gamma, \delta)$, with γ and δ being fixed numbers. Our data are obtained by observing each pump a period $t_i$, and count the number of failures $x_i$ in this period. We are interested in making parameter inference for the β and the individual failure rates, $\lambda_i, i = 1, \ldots 10$.

a) Argue that the shape of the posterior is known to proportionality by the expression:
$p(\boldsymbol{\lambda}, \beta | \alpha, \gamma, \delta, \boldsymbol{x}, \boldsymbol{t})$

$$\propto \frac{\delta^\gamma \beta^{\gamma-1}}{\Gamma(\gamma)} \exp(-\delta\beta) \prod_{i=1}^{10} \frac{\beta^\alpha\, \lambda_i^{\alpha-1}}{\Gamma(\alpha)} \exp(-\lambda_i \beta) \prod_{i=1}^{10} \frac{(\lambda_i t_i)^{x_i}}{x_i!} \exp(-\lambda_i t_i)$$

Derive/identify the distributions needed for performing the Gibbs sampler, that is:

$$p(\ \lambda_i\ |\alpha, \beta, \gamma, \delta\ \boldsymbol{t}, \boldsymbol{x}, \boldsymbol{\lambda_{-i}})\ \text{ for } i = 1, \ldots, 10$$

and

$$p(\ \beta | \alpha, \gamma, \delta, \boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{t})$$

Here we use the notation that $\boldsymbol{\lambda_{-i}}$ contains the individual failure rates apart from $\lambda_i$.

b) An alternative to the Gibbs-Sampler is the Metropolis Hastings algorithm. In this algorithm a central quantity is the Metropolis Hastings ratio:

$$R_{MH} = \frac{f(x^*)g(x|x^*)}{f(x)g(x^*|x)}. \qquad (6)$$

Define the terms on the right-hand side in equation (6). Provide a detailed description of the steps of the Metropolis-Hastings algorithm. Give an expression for the transition kernel in the Metropolis-Hastings algorithm for (for $x \neq x^*$).

c) To converge in distribution to the target distribution $f(x)$, we need the transition kernel $P(x, x^*)$ to satisfy the fixpoint equation:

$$f(x^*) = \int f(x)P(x, x^*)dx \qquad (7)$$

Show that if the transition kernel satisfies the detailed balance criterion:

$$f(x)P(x, x^*) = f(x^*)P(x^*, x)\ , \qquad (8)$$

then this is sufficient to satisfy the fixpoint equation in (7). Which additional requirement(s) is needed for a Markov chain Monte Carlo method to converge to its stationary limit distribution?

## Definitions of densities

Density of Gaussian/Normal distribution:

$$\phi(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right), x \in \mathbb{R}$$

Density of Exponential distribution:

$$\text{Exp}(x;\lambda) = \begin{cases} \lambda\exp(-\lambda x) & x > 0 \\ 0 & \text{else} \end{cases}$$

Density of Erlang distribution:

$$\text{Erlang}(x;k,\lambda) = \begin{cases} \frac{\lambda^k x^{k-1}}{(k-1)!}\exp(-\lambda x) & x > 0 \\ 0 & \text{else} \end{cases}$$

Density of Gamma distribution:

$$\text{Gamma}(x;\alpha,\beta) = \begin{cases} \frac{\beta^\alpha x^{\alpha-1}}{\Gamma(\alpha)}\exp(-\beta x) & x > 0 \\ 0 & \text{else} \end{cases}$$

Point probability of Poisson distribution:

$$\text{Poisson}(x;\lambda t) = \frac{(\lambda t)^x}{x!}\exp(-\lambda t), \quad \text{for } x = 0,1,2,\ldots$$