MATH4007 COMPUTATIONAL STATISTICS

Assessed Coursework 2 — 2022/2023

Your work should be submitted electronically via the module's Moodle page by **15:00 Thursday 11th May 2023**. Since this work is assessed, your submission must be entirely your own work (see the University's policy on Academic Misconduct). Submissions up to five working days late will be subject to a penalty of 5% of the maximum mark per working day.

Submission requirements

The submission should be uploaded electronically via the submission box on Moodle, and contain:

1. A pdf file containing any computational results (plots/relevant output) and discussion. This can be produced using e.g. R Markdown, or by copying output into a Word document. Please convert any documents **to pdf** for uploading.

2. A pdf of your theoretical working. A scan of handwritten work is fine, but you could also typeset using Latex if you prefer. If it's more convenient, you can combine this and the above part into one, e.g. if you wish to put everything in one Latex document, but this is not required.

3. An R script file, i.e. **with a .r extension** containing your R code. This should be clearly formatted, and include *brief* comments so that a reader can understand what it is doing. The code should also be ready to run without any further modification by the user, and should reproduce your results (approximately, for simulation-based results).

Please make sure that all required working, results, details of implementation and discussion are contained in **components 1 and 2** of the above list and not in the script file. The work will be assessed based on the working, output and discussion in these components, and the script file will only be used for verification of results. The exception is for the R code itself, whereby it is sufficient to say "refer to script file" where a question asks you to write R code.

A complete submission consists of all the files in your final submission. The submission time of the work will be based on the time at which the submission is complete, i.e. all files are uploaded. **Please carefully check after uploading your work that the files you upload are the correct ones.** Updates to any part of the submission after the deadline will be considered a new submission and late penalties will be applicable.

Questions

1. In general, suppose a method for constructing a confidence interval for some quantity of interest θ exists. The *coverage* of the method is the actual probability (often expressed equivalently as a percentage) that, given a random sample of data, the confidence interval constructed from this sample will contain the true value of θ. The *nominal coverage* is the confidence level we choose, e.g. we often choose to construct 95% confidence intervals. We'd like the coverage of the procedure to be equal to the nominal coverage - e.g. when we construct 95% confidence intervals, we'd like the coverage to be 0.95 (or 95%), so that intervals will contain the true value of θ 95% of the time. If the coverage and the nominal coverage are equal, then the procedure is said to have *exact coverage*.

   In some specific cases, we can construct intervals which have exact coverage. For instance, when data are normally distributed, the usual confidence interval for the mean is derived straight from the true sampling distribution of the sample mean (the estimator), and hence has exact coverage. Usually though, intervals are constructed from approximations, using e.g. asymptotic theory. We can investigate the coverage properties of a proposed method using simulation.

   Here, we'll investigate the coverage of bootstrap confidence intervals, where θ is the true value of a probability density function and $\hat{\theta}$ is the kernel density estimator.

   Consider the mixture density

$$f(x; \mu_0, \mu_1, p) = (1-p)f_0(x; \mu_0) + pf_1(x; \mu_1), \quad x \in \mathbb{R},$$

   where $p \in (0,1), \mu_0 \in \mathbb{R}, \mu_1 \in \mathbb{R}$ are parameters and f_0 and f_1 are normal densities with variance 1 and mean μ_0 and μ_1 respectively. Let $p = 0.3$, $\mu_0 = 0$, $\mu_1 = 3$. Now, let θ be the true value of the density at $x = 0$ (i.e. $\theta = f(0; \mu_0 = 0, \mu_1 = 3, p = 0.3)$. Then, given a sample of n data points from f, $\hat{\theta}(h)$ is the kernel density estimate of θ using a bandwidth of h (we'll use the standard normal kernel throughout, which is the default for the `density` function in `R`).

   The objective is to perform a simulation study to investigate how the coverage of the bootstrap confidence interval for θ depends on h, by repeatedly simulating samples from f, constructing bootstrap intervals for θ, and computing how many intervals contain the true value of θ. Specifically, do this as follows:

   - Use a sample size of $n = 100$ throughout.
   - For a fixed value of h, do the following:
     - Simulate $n = 100$ data points from f.
     - Use the nonparametric bootstrap, with statistic $\hat{\theta}(h)$ to produce a 95% confidence interval for θ using this simulated sample. Store this interval.
     - Repeat these two steps to produce an appropriately large number of intervals, each based on a different simulated sample from f.
     - Using the true value of θ to test whether each interval contains the true value, determine the (estimated) coverage of the procedure for this value of h.
   - Do this for various h, and report and discuss your findings. You may use the `density` command to compute $\hat{\theta}$ within the bootstrap algorithm, but you must code the actual bootstrap procedure yourself (and not use `boot` or similar).

   [16]

2. Data are available for the number of admissions to four different Accident and Emergency (A&E) departments at four different hospitals, on five different nights. Let y_{ij} denote the number of admissions to A&E department i on the j^{th} night, $i = 1, \ldots, 4$, $j = 1, \ldots, 5$, and let $\boldsymbol{y}$ denote the vector of all 20 observations. The data are given in the following table.

A&E department	Admissions
1	5 4 7 2 3
2	2 1 5 5 5
3	9 7 6 7 10
4	8 4 13 11 5

A possible Bayesian hierarchical model to describe these data is the following:

$$y_{ij}|\lambda_i, \mu \sim \mathrm{Po}(\lambda_i), \ \mathrm{i} = 1, \ldots, 4, \ \mathrm{j} = 1, \ldots, 5 \quad (1)$$

$$\lambda_i|\mu \sim \mathrm{Exp}(\mu), \ \mathrm{i} = 1, \ldots, 4 \quad (2)$$

where $\mathrm{Po}(\lambda)$ denotes the Poisson distribution with mean λ and $\mathrm{Exp}(\mu)$ denotes the exponential distribution with rate parameter μ, i.e. if $Z \sim \mathrm{Exp}(\mu)$ then the density is $\mu e^{-\mu z}$. In (1) and (2), the variables on the left of a conditioning sign are assumed conditionally independent of each other, given the variables on the right of the conditioning sign. E.g. Given μ, the λ_i are independent and identically distributed as $\mathrm{Exp}(\mu)$.

Thus, each λ_i represents the rate of admissions per night to department i under the Poisson model (and hence also the mean number of admissions per night), and it is of interest to perform inference for the λ_i. To complete the specification, the following prior distribution on μ is used:

$$\pi(\mu) \propto \frac{1}{\mu}.$$

ASIDE: *This hierarchical model allows each group (department) to have its own parameter λ_i, but at the same time, these λ_i come from a common population of all the possible λ values, controlled by the parameter μ — thus, all the observed data points contribute information to learning about μ, and hence the λ_i from other groups.* END OF ASIDE.

Tasks

(a) Derive the joint posterior distribution $\pi(\boldsymbol{\lambda}, \mu|\boldsymbol{y})$ (up to proportionality), where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_4)$.

(b) Describe fully the Gibbs sampler to sample from $\pi(\boldsymbol{\lambda}, \mu|\boldsymbol{y})$, i.e. derive the relevant full conditional distributions and describe the steps of the algorithm.

(c) Write R code to implement your Gibbs sampler.

(d) Run your sampler, and produce suitable evidence to show you can be confident it is performing reasonably.

Question continues on next page

(e) Use the output of your sampler to produce suitable plots and numerical summaries in order to compare $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, and hence comment briefly on any differences between night admission rates at the various A&E departments.

(f) Consider making predictions for a new night in A&E department 4, denoted by y^*. Explain how you can use the output of your sampler to simulate samples from the predictive distribution $\pi(y^*|\boldsymbol{y}) = \int \pi(y^*|\lambda_4)\pi(\lambda_4|\boldsymbol{y})d\lambda_4$.

HINT: You do not need to attempt to work out any integrals theoretically. This is purely about simulation. How can you use your samples to simulate from $\pi(y^*, \lambda_4|\boldsymbol{y})$, and how does this help?

(g) Implement this to obtain samples of y^* from $\pi(y^*|\boldsymbol{y})$, where y^* is the quantity defined in (f). Use your samples of y^* to estimate the probability that the number of admissions to A&E department 4 on this new night is at least 10.

[24]