Statistical Programming Assignment 1

Gordon Ross

October 22, 2018

- **Submission:** Only one submission attempt is allowed. The deadline is 23:59 on Friday November 2nd. Late submissions will incur a 15% penalty.

  To submit, create an R script called `matriculationnumberA1.R` where `matriculationnumber` refers to your matriculation number, and upload to the Assessment 1 section of Learn. Failure to give your script file the correct name will incur a 5% penalty. **You have to submit a single script file, i.e.,** `matriculationnumberA1.R`. Failure to comply with this will incur a 5% penalty. Your answer for each question must be included in a corresponding section of your R script file. For example, your answer/code for question 1.1 must be included in a section which looks like:

```
## ;;
## ---------------------------------------------
## Q1: -- add your code below
## ---------------------------------------------
## ;;

## 1.1
code goes here

## ---------------------------------------------
```

I will deduct 5% of marks for script files which are disorganised (e.g questions are not answered in numerical order, or where it is not clear which question a code fragment is answering) so please make sure your file has a sensible structure.

- **Guidance - Assessment criteria.**

  - ☐ A marking scheme is given. Additionally to the marking scheme, your code will be assessed according to the following criteria:
    - ☐ **Style**: follow `https://google.github.io/styleguide/Rguide.xml` with care;
    - ☐ **Writing of functions**: avoid common pitfalls of local vs global assignments; wrap your code in a coherent set of instructions and try to make it as *generic* as possible; Also, functions that are meant to be optimized with `optim` must be written accordingly, see `?optim`.
    - ☐ **Executability**: your code must be executable and should not require additional code in order to run. A common pitfall is failure to load `R` packages required by your code.

- **Deadline**: Friday November 2nd, 23:59.
- **Individual feedback** will be given.

Question 1

1. Exponential smoothing is a common approach for smoothing time series data to help extract the underlying trend. Suppose $X_1, \ldots, X_n$ are a sequence of time ordered observations. The Exponentially smoothed series is then $a_1, \ldots, a_n$ where:

$$a_1 = X_1$$
$$a_t = \lambda X_t + (1-\lambda)a_{t-1}, \quad t = 2, 3, 4, \ldots, n$$

where $0 < \lambda < 1$ is a user defined parameter. Consider the sequence

```
x <- c(2.53,  4.22,  1.15, 4.33,  7.31,  5.89,  2.74,
      8.69,  5.19,  8.38, 13.65, 13.23, 5.18, 14.61,
      14.78, 14.15, 20.19, 18.09, 17.57, 21.73)
```

Plot this data, and then on the same plot the Exponentially smoothed series for $\lambda = 0.2$ as a red line. Next, still on the same plot, plot the Exponentially smoothed series for $\lambda = 0.5$ as a blue line. Save your resulting plot (with all 3 lines) as a PDF file called matriculationnumberQ1a.pdf and upload it along with your submission script file (note: to save plots in R studio, use the Export tab that displays above the plot window) **(5 marks)**

2. Suppose $Y_!, \ldots, Y_n$ are i.i.d random variables with population mean μ and population standard deviation σ. Let $\bar{X}$ denote the sample mean. The Central Limit Theorem states that $\sqrt{n}(\bar{X} - \mu)/\sigma$ converges to a Normal(0,1) distribution as $n \to \infty$.

   Suppose that Y has an Exponential(1) distribution. Consider the sample size $n = 50$. Generate 1000 different data-sets with this sample size and plot the empirical density of $\sqrt{n}(\bar{X} - \mu)/\sigma$ in black, with the density function of a Normal(0,1) superimposed on the same plot in red. Save and upload your plot as a PDF file called matriculationnumberQ1b.pdf **(5 marks**)

Question 2

The kurtosis of a random variable Y with mean μ is a measure of how likely it is to generate extreme values. The Normal distribution has a kurtosis of 0, and so-called 'heavy tailed' distributions such as the Student-t and Cauchy have higher values. The kurtosis is defined as:

$$Kurt(Y) = \frac{E[(Y-\mu)^4]}{E[(Y-\mu)^2]^2} - 3$$

(note: the '-3' is a commonly-used convention so that the kurtosis of the Normal distribution works out to be 0)

1. Write a function in R which takes a data vector y and computes the sample kurtosis using the above formula. **(4 marks)**

2. Consider the following data:

   ```
   y <- c(-0.90, 0.47, 0.87, -5.37, -0.48,
          0.24, 0.71, 0.58, -0.54 ,-0.41,
          0.09, 0.32, 0.07, 1.70, -0.41,
          0.33, -0.72, -0.74, -0.35,  1.14)
   ```

   Use a bootstrap to compute a 95% confidence interval for the kurtosis. **(4 marks)**

3. Explain a method for implementing a hypothesis test where the null hypothesis is that the kurtosis is 0, and the alternative is that it is not 0. Hence draw a careful conclusion about the kurtosis of the above data. **(2 marks)**

Question 3

The 'rainforest' dataframe in the DAAG packages contains measurements for 4 different rainforest species .

1. Install the DAAG package using install.packages() and load it into R **(3 marks)**

2. Use the bootstrap to construct a confidence interval for the difference in the mean value of 'wood' between the **B. myrtifolia** and **Acmena smithii** species. **(7 marks)**

Question 4

If Y_i has a Poisson(λ_i) distribution then its density function is:

$$p(Y_i|\lambda_i) = \frac{\lambda_i^{Y_i}}{Y_i!}e^{-\lambda_i}$$

Suppose that we have a situation where we believe the variables $Y_1, \ldots, Y_n$ follow a Poisson distribution. For each variable, we have an associated predictor X_i and would like to model Y as a function of X. This leads to the Poisson regression model where:

$$\lambda_i = e^{\beta_0 + \beta_1 X_i}$$

1. Write an R function which takes a parameter vector (β_0, β_1) and two vectors y and x and returns the log-likelihood of the Poisson regression model. **(6 marks)**

2. A group of bacteria are stored at -70 degrees Celsius. It is believed that the number of bacteria will decay over time. Download the file counts.csv from Learn and read this data into R (do not type it in manually!). The first column denotes the time each measurement was taken at, and the second column is the associated count. Use the optim() function to fit a Poisson regression model to this data where X is time, and Y is the count. Hence estimate the parameters β_0 and β_1. **(6 marks)**

3. Use your fitted model to make a prediction for the count at time $t = 20$ **(2 marks)**

4. Use bootstrap to find a 95% confidence interval for this prediction. **(6 marks)**

Question 5

(1) Probability integral transform - inverse sampling

The probability density function of the **Rayleigh** distribution with scale parameter $\sigma > 0$ is given by

$$f(x) = \frac{x}{\sigma^2}e^{-x^2/(2\sigma^2)} \qquad x \geq 0, \sigma > 0.$$

Write a function in `R` called `rRayleigh` that simulates a random sample of size n from the Rayleigh(σ). Your code should contain the function, appropriately indented and commented, as well as a call to the function that is used to simulate 1000 random variates from the Rayleigh distribution with $\sigma = 1$.

(**5 marks**)

(b) Probability integral transform - inverse sampling

The cumulative distribution function of the **generalised extreme value** distribution (short-hand GEV) with parameters $\mu \in \mathbb{R}$ (location), $\sigma > 0$ (scale) and $\xi \in \mathbb{R}$ (shape) is given by

$$G(x) = \exp\left[-\left\{1+\xi\left(\frac{x-\mu}{\sigma}\right)\right\}_+^{-1/\xi}\right], \tag{1}$$

where $x_+ = \max(x, 0)$.

Write a function in `R` called `rgev` that takes as inputs `n` , `mu`, `sigma` and `xi`, and returns n random variates from the GEV(μ, σ, ξ). Your function should treat the limiting case

$$\lim_{\xi \to 0} G(x) = \exp\left\{-\exp\left(-\frac{x-\mu}{\sigma}\right)\right\}, \qquad x \in \mathbb{R},$$

separately, possibly with an if else statement that reads

```
[]{r}
if(abs(xi) < tol.xi.limit)
{

} else
```

where `tol.xi.limit` is also function input that can be defaulted to `5e-2`.

Your answer should contain the code of the function as well as three function calls that are used to generate

- 1000 random variates from the GEV$(0, 1, 0)$;
- 1000 random variates from the GEV$(0, 1, 1)$;
- 1000 random variates from the GEV$(0, 1, -1/2)$.

(**5 marks**)

(c) Rejection sampling

Consider the **Beta**(α, β), distribution on $[0,1]$ with density given by

$$f(x) = \frac{1}{\text{Be}(\alpha, \beta)} x^{\alpha-1}(1-x)^{\beta-1}, \qquad x \in [0,1], \alpha, \beta > 0.$$

Is it always possible to construct a rejection sampling algorithm that generates Beta(α, β) variates with a Uniform$(0,1)$ proposal? Write a function in `R` that is called `rbeta.rs` that takes as inputs `n`, `alpha` and `beta` and returns n random variates from the Beta(α, β) using rejection sampling with Uniform$(0,1)$ proposal but returns the error message `"Invalid parameter values"` when such a scheme is not feasible.

Your answer should contain the code of the function as well as a call to the function that is used to generate 1000 Beta$(2,1)$ random variates.

(**10 marks**)

Question 6

Let X_1, X_2, X_3 be a random sample from a Cauchy$(\theta, 1)$ distribution with density function:

$$p(X|\theta) = \frac{1}{\pi} \frac{1}{1+(x-\theta)^2}$$

1. Derive the log-likelihood function. (**3 marks**)
2. Find the maximum likelihood estimate of θ for the observations $Y_1 = 0, Y_2 = 4, Y_3 = 8$ (**2 marks**)