Statistical Programming Assignment 2

Gordon Ross

November 19, 2018

- **Submission:** Only one submission attempt is allowed. The deadline is 23:59 on Monday December 3rd. Late submissions will incur a 15% penalty.

  To submit, create an R script called `matriculationnumberA2.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**: Monday December 3rd, 23:59.
- **Individual feedback** will be given.

Please answer **all** three questions. The first question is a fairly straightforward test of Monte–Carlo integration. The last 2 questions are more conceptually challenging, and will apply computational techniques to less artificial problems than we have seen in lectures.

Question 1

Use Monte Carlo integration with 100,000 random numbers to evaluate the following integrals. In your script file, report both your code, the estimated integral value, and the Monte Carlo error.

- $\int_1^3 x^2 e^{\left(-(x-2)^2\right)} dx$, using a $N(2,1)$ proposal distribution [5].
- $\int_1^5 y^5 \log(y) dy$ using a Uniform(1,5) proposal [5].

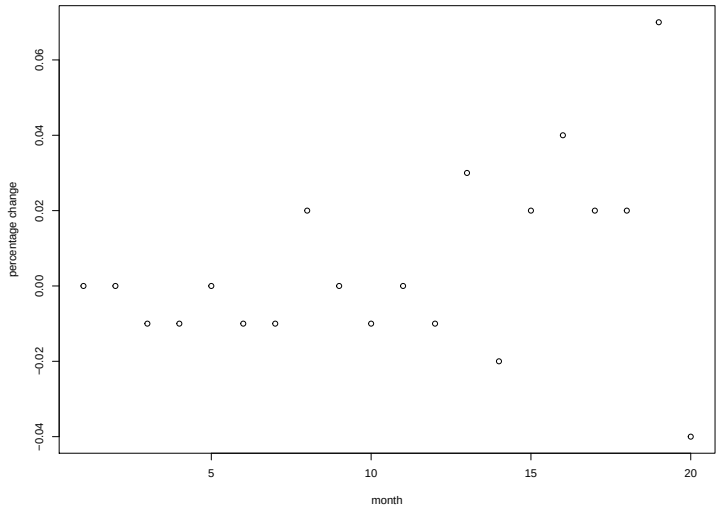
It is well known that π is the solution to the following integral:

$$\int_0^1 \frac{4}{1+x^2} dx$$

Use Monte Carlo to approximate this integral, for sample sizes (i.e. the number of random numbers) $N \in (10, 100, 1000)$. For each value, also compute the Monte Carlo approximation error . Write the values and the errors in your script files [5].

Question 2

The below plot shows the monthly percentage returns to a financial asset over a 20 month period. The numbers generating the plot are shown beneath it.




```
y <- c( 0.48,  0.50, -0.86, -0.83, -0.32, -1.30, -1.42,
        1.74, -0.29, -1.31, -0.07, -1.22,  3.24, -1.97,
        1.81,  4.00,  1.87,  1.50,  6.81, -4.14)
```

In finance, it is often useful to know whether there has been a change in the variance over time, i.e. some point k such that the variance of observations $y_1, \ldots, y_k$ is equal to σ_1^2 and the variance of observations $y_{k+1}, \ldots, y_n$ is equal to σ_2^2, where $\sigma_1^2 \neq \sigma_2^2$ (note that by convention, y_k belongs to the pre-change segment).

1. Assuming that the observations are Gaussian, describe how an F-test could be used to test whether a change has occurred at location $k = 10$. Clearly state the null and alternative hypotheses. (Write your words as a comment in your R script). [2]

2. Implement this test in R and make a conclusion based on your p-value (remember that the var.test() function carries out the F test). [3]

3. In practice, we do not know which specific k to test for (i.e. we do not know in advance where the change occurred). Instead, we wish to estimate which value of k the change occurred at. One approach for this is to perform the F test at every possible value of $k \in (2, 3, \ldots, n-2)$, so 17 tests in total given 20 observations (note that we need at least 2 observations in each segment to compute the variance, hence why we do not consider $k \in \{1, n-1, n\}$).

   In other words, for each value of $k \in (2, 3, \ldots, n-2)$, split the observations into the sets $y_1, \ldots, y_k$ and $y_{k+1}, \ldots, y_n$, then perform an F-test and record the p-value.

   After carrying out these 17 tests, the best estimate of k will be the value of k which gives the lowest p-value since this provides the most evidence for a difference in variance. Perform this procedure in R and hence determine which value of k is most likely to be the change point. in the above data. [10]

4. Next, we need to determine whether the change point we found is statistically significant, i.e. is there really evidence to suggest that there is a change in variance at the value of k you found above? Unfortunately, we cannot just check whether the p-value of the F test at this point is less than 0.05 because we did not just perform one test, we performed 17 tests and chose the lowest p-value. This multiple testing issue means that a more sophisticated procedure is necessary.

   Instead we can use a variant of permutation testing. Let the null hypothesis be that there is no change point anywhere, i.e. that all observations have the same distribution. Let the alternative be that there is a change point at some unknown value of k. If the null hypothesis is true, we can rearrange the 20 observations in any order we like. For each rearrangement, compute the minimum p-value over all 17 F tests, and hence approximate the distribution of the minimum p-value under the null hypothesis. Plot this distribution, and hence conclude whether there is evidence to suggest that a change has occurred in the given sequence (i.e. check if your minimum p-value from part 3. above is in the lower 5th quintile of p-values from this null distribution). [10].

Question 3

Statistical methods can be used to determine the (unknown) author of an unidentified piece of writing. This is known as stylometry. Research has

"a", "all", "also", "an", "and", "any", "are", "as", "at", "be", "been", "but", "by", "can", "do", "down", "even", "every", "for", "from", "had", "has", "have", "her", "his", "if", "in", "into", "is", "it", "its", "may", "more", "must", "my", "no", "not", "now", "of", "on", "one", "only", "or", "our", "shall", "should", "so", "some", "such", "than", "that", "the", "their", "then", "there", "things", "this", "to", "up", "upon", "was", "were", "what", "when", "which", "who", "will", "with", "would", "your"

Figure 1: The 70 grammatical words that characterise writing style

shown that people differ in how frequently they use basic grammatical English words such as 'a' and 'the'. These differences are quite small (perhaps one person only uses the word 'the' 1% more often than another person does) but they do exist, and can be identified given a large enough sample of a person's writing. As such, if we are given a large sample of a person's writing and a new text that has an unknown author, then it is possible to statistically test whether the person wrote it simply by counting up the number of times these basic grammatical words appear in the new text, and comparing it to their writing sample

This question will explore an example of this technique. You will download a text file which contains counts of how often each of 3 different authors used the 70 grammatical word from Figure 1. You will then use this to determine which of the 3 was the most likely author of a new piece of writing that has an unknown author. The three authors in question are Agatha Christie (British crime noveltist), Charles Dickens, and George R R Martin (author of Song of Ice and Fire/Game of Thrones)

First download the 'authorship.csv" file from the course webpage and load it into R. This contains a 3x71 matrix of counts, where the rows correspond to each of the above three authors in order. The columns are counts of how many times each author used each of the words from Figure 1, summed up over all of their published books. So for example, the first column of the first row counts how many times Agatha Christie used the word 'a' in her published novels, while the 2nd column of the third row counts how many times George R R Martin used the word 'all'. The 71st column counts the number of non-grammatical words each author used (i.e. every word which wasnt one of the 70 in this list). The total number of words used by each

authors are equal to the row sums, i.e. 3,808,305 for Christie, 3,825,393 for Dickens, and 1,753,671 for George R R Martin.

1. Download the authorship.csv file and then load it into R, as a numerical matrix of counts [3].

2. For each author i, we have a 71 element vector corresponding to the number of times they used each word. We will model this as a Multinomial(θ_i) distribution. The Multinomial distributionn is a generalisation of the Binomial distribution, where a random variabel can take one of K different outcomes (in this case, $K = 71$).

   For a particular author i, the unknown parameter θ_i is a 71 element vector, $\theta_i = (\theta_{i,1}, \theta_{i,2}, \ldots, \theta_{i,71})$. Suppose that $y_i = (y_{i,1}, y_{i,2}, \ldots, y_{i,71})$ is the vector of counts (i.e. $y_{1,1}$ is how many times author 1 used the word 'a' and so on). Then the maximum likelihood estimate of θ_i is:

   $$\hat{\theta}_{i,k} = \frac{y_{i,k}}{\sum_{j=1}^{71} y_{i,j}}$$

   (i.e. the MLE for the proportion of times each word is used by author i is simply the empirical proportion of times they used that word).

   Write R code to compute the maximum likelihood estimate of the 71-element θ_i vector for each of the three authors and report each one as a seperate commented line in your script file [5].

3. Download the 'unknowntext.csv' file from the course website and load it into R [2].

4. The unknowntext.csv file contains an extract of 10,000 words taken from a novel written by one of these three authors. As above, this is a 71 element vector which counts how many times each of the above grammatical words were used. We will try to determine which author wrote it by testing which of the estimated $\hat{\theta}_i$ parameters it is consistent with. This can be done using hypothesis testing, i.e. for each θ_i we will test:

   $$H_0 : p(z) = Multinomial(\theta_i)$$
   $$H_1 : p(z) \neq Multinomial(\theta_i)$$

where $z = (z_1, \ldots, z_{71})$ is the word counts for the unknown text. First, normalise this vector so that it sums to 1 by dividing each element by 10,000. Next, we define the test statistic:

$$T_i = \sum_{k=1}^{71} (z_k - \theta_{i,k})^2$$

where z_k is the normalised count for the k^{th} word. Compute this test statistic for all 3 authors and write down the values in your script file. [5]

5. T_i essentially measures the distance between the new text, and the parameter for each author. As such for each author i, we will reject the null hypothesis if $T_i > \gamma_i$ and conclude that this author did not write the text. We need to choose γ_i in order to make the Type 1 error equal to the usual 0.05 (so that we only mistakenly reject the null 5% of the time, if the author really did write the text).

   For each author, use Monte Carlo simulation to find the appropriate value of γ_i. You can do this by simulating sample data under the assumption that the null hypothesis is true, computing the test statistic for each simulated piece of data, and then defining γ_i to be the 95th quantile of these simulated test statistics. This means that if the null hypothesis is true, only 5% of observations simulated from the $Multinomial(\theta_i)$ distribution will be greater than this value of γ_i.

   In other words: for each i, simulate a large number (e.g. $S = 100,000$) of observations from the Multinomial(θ_i) with 71 categories and 10,000 words (i.e. equal to the number of words in the unknown text). Compute the test statistic T_i above for each simulated observation. Then, define γ_i to be the $0.95S$ largest of these values (similar to bootstrapping) [10]

6. Based on the above, compute which of the three null hypotheses are rejected and hence determine the most likely author of the unknown text. [5]