The University of British Columbia

Data Science 580 Modelling and Simulation I

Lab Assignment 1

In order to complete this lab, you will need to download and install R (and RStudio if you want it). It is recommended that you complete this step prior to the beginning of the lab session. Some assistance can be found at `http://rtricks4kids.ok.ubc.ca/RTrix/startup.php` If you have not used R before, I recommend that you start working through the materials listed in the pdf document contained there. Note that for the purpose of this lab, much of the R code just needs to be copied, pasted, and changed; there is no real programming element.

The following material reviews and extends some of what was covered in lectures. Read it and answer the questions that follow.

Testing Randomness

There are many tests for randomness. The main goals of these tests are to ensure that the output from a random number simulator are:

- reasonably close to uniformly distributed
- reasonably close to independent of each other

It is impossible to ensure that these two conditions will hold for a simulated sequence. It is not too hard to check the first condition, but the second condition can only be checked in an incomplete way.

Testing Uniformity

A simple test for the uniform distribution is based on the chi-square test.

(a) Divide the interval $[0, 1]$ into m equal subintervals. According to the uniform distribution, the probability that a uniformly distributed value would lie in one of the subintervals is $1/m$.

If n numbers are simulated, we would expect $E_i = n/m$ to lie in the ith subinterval.

The observed number of simulated values in the ith intervals can be counted: O_i.

(b) The chi-square test statistic is

$$x = \sum_{i=1}^{m} \frac{(O_i - E_i)^2}{E_i}.$$

x will be large if O_i and E_i differ a lot, i.e. if the uniform assumption does not hold. Otherwise, x is likely to be small.

If the uniform assumption holds, then the p-value for the goodness-of-fit test of uniformity is calculated as

$$P(X \geq x)$$

where X is chi-square distributed on $m - 1$ degrees of freedom.

(c) The test data can be extended to other discrete random variables. Here, m is the number of all possible values. There is a built-in `chisq.test()` function in R, but we will create a simpler one, specifically designed to test discrete random number generators: `rng.chisq()`. It takes arguments `x`, the data vector (simulated from a discrete random number generator):

```
rng.chisq <- function(x) {
# x is output from a discreat uniform pseudorandom number
n=length(x)
x=as.vector(table(x))#Oi
m=length(x)
p <- rep(1,m)/m
Ex <- n*p#Ei
chisq <- sum((x-Ex)^2/Ex)
pvalue <- 1-pchisq(chisq, m-1)
list(test.statistic=chisq, p.value=pvalue, df=m-1)
}
```

Example:

```
install.packages("purrr")
library(purrr)
x1=rdunif(150,b=80,a=55)
x2=rdunif(150,b=110,a=90)
x3=rdunif(150,b=85,a=55)
data=cbind(c(1:450),c(x1,x2,x3))# 450 simulated discrete numbers
rng.chisq(data[1:150,2])
```

If the p-value is not small, then there is very little evidence against the uniformity hypothesis and we conclude that the numbers are following a uniform distribution.

Testing Independence

Many tests have been devised to try to test independence.

In class, we saw that the autocorrelation function can be used to test linear dependence between lagged values. Try the `acf()` function, and look for spikes in the graphical output. These indicate linear dependence. This is studied in detail in a time series course.

1. Question 1 (4 marks). The number of heart beats per minute (BPM) was assumed to be a standard statistic in clinical practice. Participants were asked to sit down and rest, then cycle (light exercise) and rest again after exercise. Each phase (rest, exercise and rest) took a total of 150 minutes. We simulated 450 BPM according to Wallot et al. (DOI: 10.3389/fphys.2013.00211). First, read the simulated BPM data matrix using

```
Data=readRDS(file = "...path...to.../BPM.rds")
```

In the Data matrix, the first column is the number of minutes and the second column is the BPM. if you need to analyze the BPM between 10 and 20 minutes, you should use the

```
Data[10:20,2]
```

(a) (2 marks) Test the uniformity of the subsequence in the first phase. Report the p-value and write the conclusion clearly.

(b) (2 marks) Test the uniformity of the total sequence. Report the p-value and write the conclusion clearly.

2. Question 2 (3 marks). To forecast currency exchange rates, we examine their dependence structure. For example, we take the exchange rate of the Canadian dollar against the U.S. dollar from the data source:

   ```
   https://www.ofx.com/en-ca/forex-news/historical-exchange-rates/cad/usd/
   ```

   Read the saved rate data:

   ```
   Data=readRDS(file = "...path...to.../Exchanges.rds")
   ```

   Analyze the rate data in the last column Data[,4].

   (a) (2 marks) Show the autocorrelations for the first 5 lags.

   (b) (1 mark) Is the sequence dependent?

3. Question 3 (4 marks). To calculate the autocorrelation function of a DNA sequence, the symbols (A, T, C, G) must be converted to numerical values. It is important to note at this point that if we simply assign a numerical value to each nucleotide the resulting correlation will depend on the particular assignment, i.e., the mapping itself may lead to spurious results. This problem does not arise if the sequence is binary, i.e., assign 1 to one group and 0 to another.

   Read the txt file obtained from Dai et al. (https://doi.org/10.1002/jcc.20471):

   ```
   Data=read.table("...path...to.../DNA.txt", header = FALSE)
   ```

   Cross-correlation is defined by assigning different values to x_i and $x_{i+\ell}$. For example, $x_i = 1$ when A is found at position i and 0 at all other positions, and $x_{i+\ell} = 1$ when T is found at position $i+\ell$ and 0 at all other positions. Such an autocorrelation function will measure the statistical properties of the pairs (A,T) separated by a distance ℓ (Gene 300, 2002, 105-115).

   Given the numerical sequence $\{x_1, \ldots, x_n\}$, calculate the variance

   $$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} x_i^2 - \left(\frac{1}{n}\sum_{i=1}^{n} x_i\right)^2$$

   and the autocorrelation at distance ℓ by:

   $$C(\ell) = \frac{1}{\sigma^2}\left[\frac{1}{n-\ell}\sum_{i=1}^{n-\ell} x_i x_{i+\ell} - \frac{1}{(n-\ell)^2}\sum_{i=1}^{n-\ell} x_i \sum_{i=1}^{n-\ell} x_{i+\ell}\right].$$

   For example, calculate the variance and $C(1)$ for the pairs (A, T) in the Human sequence:

   ```
   n=nchar(Data[1,2])
   i=1;
   x=rep(0,n)
   while(i<(n-1)){
   if(substring(Data[1,2],i,i)=="A") x[i]=1
   if(substring(Data[1,2],i+1,i+1)=="T") x[i+1]=1
   i=i+2
   }
   ```

```
sigma2=sum(x^2)/n-(mean(x))^2
C1=(sum(x[1:(n-1)]*x[2:n])/(n-1)-mean(x[1:(n-1)])*mean(x[2:n]))/sigma2
C1
```

(a) (2 marks) Calculate $C(1)$ for the pairs (A, T) in all sequences:

(b) (2 marks) Calculate $C(1)$ for the pairs (A, C) in all sequences: