Problem Set 3

YSC4204 – Statistical Computing

Due on: Tuesday, 20 September, 23:59

Problem 1

R's exponential random number generator

In class we have been wondering why R's core team decided to use the Ahrens-Dieter algorithm for exponential random numbers instead of the more straightforward inverse transform method.

(a) Find a user forum where we can ask questions about the R core team's rationale. Briefly explain in your homework solution why this forum is appropriate.

(b) Draft a professionally phrased post. A reader should understand that you are aware of run-time and numerical accuracy issues.

Please do not post your question yet! Instead, I will combine your solutions into a proposed post. We will briefly discuss it in class before posting it together.

(5 points total. 2 for (a), 3 for (b).)

Problem 2

Assessing a proposed alternative to the Box-Muller method

Rizzo claims on page 58 of her book:
"If $U, V \sim Unif(0, 1)$ are independent, then

$$Z_1 = \sqrt{-2 \log U} \cos(2\pi V),$$
$$Z_2 = \sqrt{-2 \log V} \sin(2\pi U)$$

are independent standard normal variables".
The second line is probably a typo because in the Box-Muller method $Z_2 = \sqrt{-2 \log U} \sin(2\pi V)$ (i.e. U and V are swapped). However, taking literally what she wrote, is her claim correct?
Here are a few hints to reach an answer. If two random variables Z_1 and Z_2 are independently standard normally distributed, then the pair (Z_1, Z_2) should come from a bivariate standard normal distribution. First make a scatter plot to gain some intuition. Then run a hypothesis

test for a bivariate normal distribution. I leave it up to you how you want to test the data, but I found this link useful:

`https://cran.r-project.org/web/packages/MVN/vignettes/MVN.pdf`

Your solution to this problem includes commented R code, a scatter plot and an explanation how you reach your conclusion.

(5 points.)

Problem 3

Relating `rnorm()` to `runif()`

(a) Here are a few random numbers generated by R.

```
> RNGkind("default", "Box-Muller")
> set.seed(536)
> runif(4)
[1] 0.3285004 0.3659592 0.9251519 0.6418518
> set.seed(536)
> rnorm(4)
[1] -0.6713455 1.2489044 0.8394670 -0.4267210
```

Based on the *uniform* random numbers generated above, explain the generated *normal* random numbers. In your answer, refer to the code in `sexp.c`.

(b) Here are more random numbers.

```
> RNGkind("default", "default")
> set.seed(536)
> format(runif(4), digits = 22)
[1] "0.3285003982018679380417" "0.3659591851755976676941" "0.9251519178505986928940"
[4] "0.6418517879210412502289"
> set.seed(536)
> format(rnorm(4), digits = 22)
[1] "-0.4440577932828237983642" " 1.4406055141632929661455"
[3] "-1.1768451363657648212069" " 0.3563781940832911887540"
```

Based on the *uniform* random numbers generated above, explain the first two generated *normal* random numbers (i.e. including the trailing digits). Again refer to the code in `sexp.c`.

(4 points total. 2 each for (a) and (b).)

Problem 4

Binary representation of floating point numbers

On page 320, Rizzo shows the following R output.

```
> (.3 - .1)
[1] 0.2
> (.3 - .1) == .2
[1] FALSE
> .2 - (.3 - .1)
[1] 2.775558e-17
```

(a) Derive the exact binary representation of 0.1, 0.2 and 0.3. In your answer, you will need the geometric series formula: $\sum_{i=0}^{n} x^i = \frac{1-x^{n+1}}{1-x}$ if $x \neq 0$.

(b) Can you explain the last line of output?

(4 points total. 2 each for (a) and (b))

Problem 5

Avoid procrastination and cramming before the midterm exam (Friday, 30/09).

Review already now the course material that we covered before recess week. In your teams, come up with one question (or two, if you prefer) that you want to see addressed during the class on Wednesday, 28 September.

(2 points.)