✖

Week 4 Quiz

<div style="text-align:center">

Back to Week weekNumber

</div>

✔   10/10 points
earned (100%)

Quiz passed!

✔   1 / 1
points

1.
What is produced at the end of this snippet of R code?

```
1  set.seed(1)
2  rpois(5, 2)
```

🔘   A vector with the numbers 1, 1, 2, 4, 1

△

Correct
Because the `set.seed()' function is used, `rpois()' will always output the same vector in this code.

⚪   It is impossible to tell because the result is random

⚪   A vector with the numbers 1, 4, 1, 1, 5

⚪   A vector with the numbers 3.3, 2.5, 0.5, 1.1, 1.7

✔   1 / 1
points

2.
What R function can be used to generate standard Normal random variables?

⚪   pnorm

⚪   dnorm

🔘   rnorm

△

Correct
Functions beginning with the `r' prefix are used to simulate random variates.

⚪   qnorm

✔ 1 / 1
points

3.
When simulating data, why is using the set.seed() function important? Select all that apply.

☐ It ensures that the sequence of random numbers is truly random.

Un-selected is correct

☐ It ensures that the random numbers generated are within specified boundaries.

Un-selected is correct

☐ It can be used to generate non-uniform random numbers.

Un-selected is correct

☐ It ensures that the sequence of random numbers starts in a specific place and is therefore reproducible.

Correct

✔ 1 / 1
points

4.
Which function can be used to evaluate the inverse cumulative distribution function for the Poisson distribution?

○ ppois

○ dpois

○ rpois

◉ qpois

Correct
Probability distribution functions beginning with the `q' prefix are used to evaluate the quantile (inverse cumulative distribution) function.

✔ 1 / 1
points

5.

What does the following code do?

```
1   set.seed(10)
2   x <- rep(0:1, each = 5)
3   e <- rnorm(10, 0, 20)
4   y <- 0.5 + 2 * x + e
```

○   Generate data from a Poisson generalized linear model

◉   Generate data from a Normal linear model

Correct

○   Generate random exponentially distributed data

○   Generate uniformly distributed random data

✔   1 / 1
       points

6.
What R function can be used to generate Binomial random variables?

○   pbinom

○   dbinom

◉   rbinom

Correct

○   qbinom

✔   1 / 1
       points

7.
What aspect of the R runtime does the profiler keep track of when an R expression is evaluated?

○   the global environment

○   the working directory

○   the package search list

◉   the function call stack

Correct

---

1 / 1
points

8.

Consider the following R code

```
1   library(datasets)
2   Rprof()
3   fit <- lm(y ~ x1 + x2)
4   Rprof(NULL)
```

(Assume that y, x1, and x2 are present in the workspace.) Without running the code, what percentage of the run time is spent in the 'lm' function, based on the 'by.total' method of normalization shown in 'summaryRprof()'?

O   23%

◉   100%

Correct

When using `by.total' normalization, the top-level function (in this case, `lm()') always takes 100% of the time.

O   50%

O   It is not possible to tell

---

1 / 1
points

9.

When using 'system.time()', what is the user time?

O   It is a measure of network latency

O   It is the "wall-clock" time it takes to evaluate an expression

◉   It is the time spent by the CPU evaluating an expression

Correct

O   It is the time spent by the CPU waiting for other tasks to finish

---

1 / 1
points

10.

If a computer has more than one available processor and R is able to take advantage of that, then which of the following is true when using 'system.time()'?

O   user time is 0

O   user time is always smaller than elapsed time