Biostatistics 615 Learning Exercise #2 (10 pts)

Due by September 10th 2024 (Tuesday) 11:59pm. Use Gradescope (via Canvas) to submit an R file.

- Your submission should only contain one `R` file named `my_pexp.R` that contains a function named `my_pexp(x, rate, lower.tail, log.p)`.

- Your code will be evaluated in Gradescope using 10 different test cases using an automated script. Full credit will be given if your code passes all test cases.

- You are allowed to submit multiple times before the deadline, but only the last submission will be graded. Automated feedback will be provided for each submission.

- You need to implement the function to work with arbitrary (valid) input values beyond the 10 cases tested. If you tweak your implementation so that your functions works specifically for the test cases, you will not receive any credit.

- Implement your function as efficient as you can. If your program does not finish after running for 1.0 seconds, you will lose the points for those test cases. Note that the official solution finishes within 0.2 seconds for any test case, so this should be a reasonable time limit.

Problem 1 - `my_pexp.R` (10 pts)

Write a function `my_pexp(x, rate, lower.tail, log.p)` that should perform identically to the `pexp(x, rate, lower.tail, log.p)` function available in `stats` package in R. However, you are not allowed to use the `pexp()` or any other functions outside `base` package in your implementation.

The function should take the following arguments:

- `x`: a numeric quantile value.

- `rate`: a numeric value of rate parameter.

- `lower.tail`: a logical value indicating whether to compute $\Pr(X \leq x)$ (if `TRUE`) or $\Pr(X > x)$ (if `FALSE`).

- `log.p`: a logical value indicating whether to return the log probability.

Let x represent `x` and λ represent `rate`. When `lower.tail` is `TRUE` and `log.p` is `FALSE`, the function should return the following value:

$$\Pr(X \leq x) = 1 - \exp(-\lambda x)$$

When `lower.tail` is `FALSE`, It should return $\Pr(X > x)$ instead. When `log.p` is `TRUE`, it should return the log probability.

It is important to maintain the precision of the output. Make sure that the output of your function matches the output of the `pexp()` function across a range of input values. It may be helpful to try `help("exp")` and `help("log")` in R to understand various options to compute the exponential and logarithm without losing precision.

Note that you are NOT allowed to use the `pexp()` function or any other functions outside the `base` package in your implementation. Use `help(...)` to check whether a function belongs to the `base` package or not.

You may assume that the input values are always valid, with $x > 0$ and $\lambda > 0$ (i.e. no need to check for invalid input values).