Biostatistics 615 Learning Exercise #12 (10 pts)

Due by November 12th 2024 (Tuesday) 11:59pm. Use Gradescope (via Canvas) to submit an R file.

- Your submission should only contain one `R` file named `rmvnorm.R` that contains a function named `rmvnorm(n, p, rho)`.
- 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 within the time limit for each test case, you will lose the points for those test cases. Note that the official solution finishes much faster than the test cases, so this should be a reasonable time limit.

Problem 1 - Multivariate Normal Sampling (10 pts)

Suppose $(X_1, \ldots, X_p)^{\mathrm{T}} \sim N_p\{0, \Sigma(\rho)\}$, which is a p-dimensional multivariate normal distribution with mean zero and covariance matrix $\Sigma(\rho) = \{\sigma_{i,j}(\rho)\}_{p\times p}$, where for $i, j \in \{1, \ldots, p\}$,

$$\sigma_{i,j}(\rho) = \exp\left\{p\log(\rho)|l_i - l_j|^{1.99} - |\cos(l_i)| - |\cos(l_j)|\right\},$$

where $p \geq 2$, $\rho \in (0, 1)$ and $l_i = (i-1)/p$.

Write an R script `rmvnorm.R` which contains a function `rmvnorm(n, p, rho)`. Given an integer $p(2 \leq p \leq 200)$ and a real number $\rho \in (0, 1)$ as the function arguments draws n random samples, which should be at least **100,000** in the evaluation, from $N_p\{0, \Sigma(\rho)\}$.

Using those random samples, your program should estimate and return a numeric vector of size 3 that contains the following three values in order:

1. $\mathrm{E}\{\max_j X_j\}$
2. $\mathrm{E}\left\{\sqrt{\sum_{j=1}^{p} X_j^2}\right\}$
3. $\Pr(X_1 X_2 > 0.5\rho)$

Note that all the output values only need one valid precision digit in your code (do not round the values yourself).

Example output of running the test code is given below:

```
> print(rmvnorm(100000, 80, 0.42), digits=1)
[1] 0.7 3.8 0.2
> print(rmvnorm(100000, 103, 0.63), digits=1)
[1] 0.7 4.3 0.1
```

For all problems, you can include as many functions as you want as long as your code contains the required function.

There are specific requirements in the implementation:

- You are NOT allowed to use any functions outside the `base` and `stats` package in your implementation. Use `help(...)` to check whether a function you want to use belongs to the allowed packages or not.
- Your answer should be accurate up to 1 significant digits for each output values.
- Each test case must finish within 10 seconds.

You do not need to implement error handling for malformed arguments in this function.