Senior Thesis
Week 2 R Assignment

Background:
Last week we examined some of the benefits of simulation methods. This week we want to build on that framework and also introduce the idea of linear models. Often, our first attempt at modeling a continuous response is to say that it is related to the predictors through a linear model (linear in the parameters):

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

where y_i is the response for the i-th subject, β_0 and β_1 are parameters of interest, x_i is the known covariate vector for the i-th subject, and ϵ_i is a random variable that captures error in the observed response. This allows us to model the variability we see in the response for subjects with the same value for the covariate. Since there is only a single predictor here, this is referred to as simple linear regression.

Given a vector of responses $\mathbf{y}$ and a vector of covariates $\mathbf{x}$, we typically estimate the parameters in the model by finding the vector $\boldsymbol{\beta}$ that minimizes

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i\right)^2$$

which gives us the so called Least Squares (LS) estimators of β_0 and β_1. In MA382, you will see that if we assume that $\epsilon_i \sim N(0, \sigma^2)$ for some unknown variance σ^2, then it turns out that the least squares estimate of β_1, lets call it $\widehat{\beta}_1$, follows a normal distribution. That is, the sampling distribution (the way that this estimate varies from one sample to the next) can be modeled using a normal distribution. This is an extension of the Central Limit Theorem.

However, this assumes that the distribution of the error terms is Normal. In this assignment, you will examine how robust the sampling distribution is to this assumption.

Assignment:
You have been provided with an R script that outlines my general approach to this problem, but nearly all code has been removed.

1. For the first problem, write an `R` function that takes two parameters — a vector y, and a vector x — and returns the least squares estimate of the slope from a regression of y on x. In particular, your code should have a form like:

   ```
   fct.name <- function(y,x){ some code here }
   ```

   While `R` has a lot of built-in capability, we will occasionally need to build our own functions, and this is practice making a simple one. However, inside your function, make use of as many of `R`'s built in capability as you can find. For example, the `lm()` performs a regression and estimates the parameters.

2. Consider a fixed covariate vector $\mathbf{x} = (1, 2, \ldots, 10)^\top$ and generate $m = 1000$ replicate datasets, each with $n = 10$ observations according to the following linear model:

   $$y_i = 5x_i + \epsilon_i$$

   where ϵ_i follows an Exponential distribution with rate parameter 1. So, when you are done, you should have 1000 vectors, each of size 10 representing the response vector.

3. For each of your response vectors from the previous part, run a regression of that response vector on the covariate vector and estimate the slope (use your function from part (a)). Then, create a histogram to examine the sampling distribution of the resulting estimates.