In this **Homework 3** you will write a SAS IML code/function to implement Newton-Raphson (NR) method to solve non-linear equation.

For linear equation e.g. $f(x) = 2x + 5$ the solution is easy, it is $2x + 5 = 0$ or $x = -2.5$
So, Solution is that value of x for which $f(x) = 0$

For non-linear question this is not so straightforward. Here is some introduction to the above from Mathworld for non-linear equation.

Newton's method, also called the Newton-Raphson method, is a root-finding algorithm that uses the first few terms of the Taylor series of a function $f(x)$ in the vicinity of a suspected root. Newton's method is sometimes also known as Newton's iteration, although in this work the latter term is reserved to the application of Newton's method for computing square roots.

For $f(x)$ a polynomial, Newton's method is essentially the same as Horner's method.

The Taylor series of $f(x)$ about the point $x = x_0 + \epsilon$ is given by

$$f(x_0 + \epsilon) = f(x_0) + f'(x_0)\epsilon + \frac{1}{2} f''(x_0)\epsilon^2 + \ldots. \tag{1}$$

Keeping terms only to first order,

$$f(x_0 + \epsilon) \approx f(x_0) + f'(x_0)\epsilon. \tag{2}$$

This expression can be used to estimate the amount of offset ϵ needed to land closer to the root starting from an initial guess x_0. Setting $f(x_0 + \epsilon) = 0$ and solving (2) for $\epsilon \equiv \epsilon_0$ gives

$$\epsilon_0 = -\frac{f(x_0)}{f'(x_0)}, \tag{3}$$

which is the first-order adjustment to the root's position. By letting $x_1 = x_0 + \epsilon_0$, calculating a new ϵ_1, and so on, the process can be repeated until it converges to a fixed point (which is precisely a root) using

$$\epsilon_n = -\frac{f(x_n)}{f'(x_n)}. \tag{4}$$

Unfortunately, this procedure can be unstable near a horizontal asymptote or a local extremum. However, with a good initial choice of the root's position, the algorithm can be applied iteratively to obtain

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{5}$$

for $n = 1, 2, 3, \ldots$. An initial point x_0 that provides safe convergence of Newton's method is called an approximate zero.

You stop the iteration when $|x_{n+1} - x_n| < \varepsilon$, i.e. your two iterative evaluation of x_n is close.

Note that the equation (5) is the key element here. Your task is to Implement NR in IML.

You need to solve the root for two equations via IML. You need to provide in each case a starting value (x_0) and a convergence parameter (ε), typically 0.001 or a smaller value. Starting value could be any number. Try with few different choices to ensure in all cases you always get a single solution.

(1) $f(x) = x^3 + 5x - 3$, note, $f'(x) = 3x^2 + 5$

(2) $f(x) = e^{2x} - x - 6$, note, $f'(x) = 2e^{2x} - 1$

Print also your SAS output showing the root and each iteration. Once you find a solution check that if $f(x_n) \approx 0$ (i.e. very close to zero if not exactly zero) for that x_n.

Now use the same code to find,

Find, correct to 5 decimal places, the x-coordinate of the point on the curve $y = \ln x$ which is closest to the origin. Use the Newton Method.

Note, $f'(x) = \frac{1}{x}$ for, $y = f(x) = \ln(x)$, so you may need to update you code accordingly.

What you must provide, 1. SAS code for each case (as word/SAS file), 2. Final Solution to each equation in SAS output window, print x_0, ε, x_n and print final $|x_{n+1} - x_n| < \varepsilon$ to show convergence. Also show iteration number of convergence. However, please don't print all iterations as it could take many lines before convergence.

Total points 70 (20 for correct implementation of NR, 15 for each of the three problems correct solution and 5 for the cleverness and compactness of your code)

Hints: - You need to use some kind of loop structure within IML, which will be exited only when the condition $|x_{n+1} - x_n| < \varepsilon$ is true. However, be careful not to get stuck in an infinite loop. Therefore, you may want to give a "max iteration" (<5000) number to guard against that.

Note: - You should not use SAS built in function like "polyroot", "NLPNRA" or Proc FCMP etc.