This assignment is due **July 18, 2025**. You may work in teams of up to four people. If you work as a team, choose a team leader to sign-up your team on the homework 1 groups page of Bruin Learn. Please submit one html or pdf file. Upload your solutions to Bruin Learn. Show all your work. You may use the modules `numpy`, `pandas`, `scipy`, and `matplotlib`.

1. Prices at various strikes for call options on a stock which is currently trading at $100 are shown in the table. Each option is European and matures in three months or 0.25 years. The continuously compounded risk-free rate is 4%. Assume that the stock does not pay dividends.

Strike ($)	80	90	100	110	120
Call ($)	21.73	13.72	8.04	4.61	2.80

(a) Write a function, or use the function from class, that gives the price of a call option assuming the Black-Scholes framework. Use it to find the price of a call option with strike $95 if the volatility is $\sigma = 35\%$.

(b) Newton's method is an algorithm to find the roots of a differentiable function f. The first step is to make an initial guess x_0. Then

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

for $n = 1, 2, \ldots$. Apply Newton's method to find the volatility σ-values implied by the prices in the table above. Use the Black-Scholes call option formula. The σ-values should be different at each strike. Make sure your σ-values are accurate to at least two basis points, i.e. to the 100th of a percent.

(c) Use `root_scalar` in `scipy.optimize` to check your answers in (b).

(d) Plot the implied volatility σ-values you found in (b) or (c) as a function of the strikes K.

2. We want to estimate the area under the curve $y = e^{-\sqrt{x}}$ from $x = 1$ to $x = 100$.

(a) Approximate the area using a Riemann sum with

$$P = \left(1, 1 + \frac{99}{n}, 1 + \frac{99 \cdot 2}{n}, \ldots, 1 + \frac{99 \cdot (n-1)}{n}, 100\right).$$

Find the height of each rectangle using the left, middle, and right points of each subinterval determined by consecutive entries of P. Calculate the results for $n = 10,\ 25,\ 50,\ 75,$ and 100 rectangles. Save your results in a `pandas` data frame.

(b) In part (a), the Riemann sum was calculated using a uniform partition, which is not the most efficient way to approximate the area. This is because $e^{-\sqrt{x}} \approx 0$ for $x \gg 0$. The approximation will be better if we sample more small values of x and fewer large values. With this reasoning in mind repeat (a) but with

$$P = \left(1, 100^{1/n}, 100^{2/n}, \ldots, 100^{(n-1)/n}, 100\right).$$

(c) Use analytic techniques to find the exact area

$$\int_1^{100} e^{-\sqrt{x}}\, dx.$$

(d) Using `matplotlib.pyplot`, graph your results from parts (a) and (b) via the function `scatter`. Use `subplots` to create three subplots, one for left endpoints, midpoints, and right endpoints. In each subplot, draw a dashed horizontal red line using `axhline` to denote your result from part (c). Make sure to add labels and legends so that it is easy to identify all of your subplots' features.

3. You have graduated from the MFE program and landed yourself a nice job pricing annuities! For each of the following perpetuities (i.e. annuities with never-ending payments) find the price, or prove that the result would not converge. Derive your solutions analytically when possible. Assume one payment is made at the end of each year and a constant annual discount rate of 6.5%. Let us say that it is January 1, 2025, for simplicity.

(a) Payments of $10.

(b) Payments of the numerical value of the year, e.g. the perpetuity would pay $2025 this year.

(c) A payment of $1 this year, an increase of 10% per year for the next four years, and an increase of 4% per year after that. That is, a payment sequence of

$$\$1, \$1.10, \$1.10^2, \$1.10^3, \$1.10^4, \$1.10^4 \times 1.04, \$1.10^4 \times 1.04^2, \ldots.$$

(d) Payments of $\$k^k/k!$ in the k-th year, where 2025 corresponds to $k = 1$.