# Machine *Learning*

# Gradient Descent Homework

**Mostafa S. Ibrahim**
*Teaching, Training and Coaching for more than a decade!*

*Artificial Intelligence & Computer Vision Researcher*
*PhD* from Simon Fraser University - Canada
*Bachelor / MSc* from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

# Problem #1: Generalization for Gradient Descent

- We implemented the gradient descent for 2 variables (x, y)
- Do simple changes to let the code process the entire vector of variables, rather than just two variables
  - Test it using the two functions provided
  - Compare the results with those of the old program
- Create file: **gradient_descent_vec.py**
- **def** gradient_descent(**fderiv, inital_start**, step_size = 0.001, precision = 0.00001, max_iter = 1000):
  - inital_start: numpy array of D values
  - fderiv: function can take a vector of D values and returns D gradients

# Problem #2: Function of 3 Variables

- Consider the function: f(x, y, z) = sin(x) + cos(y) + sin(z)
- Explore several initial starting points, and report a list of **minimum** values for this function using gradient descent
- Import; don't copy!
    - **from** gradient_descent_vec **import** gradient_descent
- As an example, one of the outputs should:
  Initially start at [1. 2. 3.5]
  ends at point: [0.22457445 2.67782963 4.21311734]
  with a minimum value of -1.5496155799445872
- Recall derivative of **sin(x)** = cos(x), and derivative of **cos(x)** = -sin(x)
    - Or use an online calculator

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."