5. (Programming Assignment) Let $\mathbf{x} \in \mathbb{R}^n$ and $z \in \mathbb{R}$ be zero-mean independent Gaussian random variables with covariance matrices $\mathbf{I}$ and σ^2, respectively. That is, $\mathbf{x} \sim \mathcal{N}(0, \mathbf{I})$ and $z \sim \mathcal{N}(0, \sigma^2)$. Define $y = \theta^{\mathbf{T}}\mathbf{x} + \theta_0 + z$. In this assignment, we want to use stochastic gradient descent (SGD) to compute a linear regression model between $\mathbf{x}$ and y. Write a Python code to do the following:

   (a) Let $n = 4$, $\sigma^2 = 1/4$, $\theta = [1, 1/2, 1/4, 1/8]^T$ and $\theta_0 = 2$. Generate $m = 10,000$ i.i.d. *training* samples from $\mathbb{P}_{X,Y}$. That is $\{(\mathbf{x}^{(1)}), y^{(1)}), ..., (\mathbf{x}^{(m)}, y^{(m)})\}$.

   (b) Use SGD with a batch size of 10 to estimate model parameters. Plot the Mean-Squared Error (MSE) vs. the number of iterations.

   (c) Generate m new i.i.d. *test* samples from $\mathbb{P}_{X,Y}$. Use estimated parameters to compute the MSE on the test set.

   (d) Repeat parts (a)-(c) using $m = 10$. How do training and test errors change? Why?