Question 2

In this question, you will learn how to perform polynomial regression with linear regression tools using python.

Given train data $(x_1 = 0, y_1 = -1.25), (x_2 = 0.5, y_2 = -0.6), (x_3 = 2, y_3 = -4.85)$ and test data $(x_4 = -1, y_4 = -5.2)$, $(x_5 = 1, y_5 = -0.9)$, $(x_6 = 3, y_6 = -13)$:

a. Define a 3×1 two-dimensional matrix called X_train in which each row is an observation from the training data, and define a (one-dimensional) vector called y_train which contains the responses of these observations (in the same order). Repeat this process with the test data (X_test, y_test).
b. Calculate the regular LS estimators $\widehat{w}_0$, $\widehat{w}_1$ using only the training data with sklearn built-in functions. What are the predicted values for X_test?
c. What is the MSE of the regression on the train data? What is the MSE of the regression on the test data? What can you conclude from these values?
d. Write a function which receives a np-array of explanatory variables X and a np-array of responses Y and returns the least squares estimator using the closed-form expression we saw in class. There is no need to check that the input is valid. (don't forget to add ones!)
e. Plot the regression line in a dashed (--) black line. Scatter (with plt.scatter()) the points in the train data with marker='*' and scatter the points in the test data with marker='o'. You should use legend (for train and test data) and label the axes. The range in the x axis should be np.arange(-3, 5).
   Does it look like the regression fit the data?
f. We will now try to perform a 2nd degree polynomial regression, meaning we will assume now that $y_i \approx \gamma_0 + \gamma_1 \cdot x_i + \gamma_2 \cdot x_i^2$.
   1. If we would mark $z_i = (x_i, x_i^2)^T$, how can we write y_i in a linear form?
   2. Define a 3×2 matrix called Z_train in which the first column corresponds to x_i and the second column corresponds to x_i^2 for each x_i in the train data (in the same order as in section a.). Repeat this process with the test data (Z_test).

3. Use the function you wrote in section d. to calculate the LS estimators $\hat{\gamma} = (\hat{\gamma}_0, \hat{\gamma}_1, \hat{\gamma}_2)^T$ using only the training data (Z_train and y_train). What is the MSE of the regression on the train data? What is the MSE on the test data?
4. Plot the corresponding 2nd degree polynomial function in a red dashed line, alongside the original regression line in a black dashed line. Scatter the points in the training data (with marker='*'), as well as the points in the testing data (with marker='o'). You should use legend for both data type (train/test) and regression type (linear/polynomial) and label the axes. The range in the x axis should be np.arange(-3, 5). Name the plot 'Polynomial Regression vs. Linear Regression'.

   Which regression seems to perform better?

g. Which assumption did not hold, thus making the linear regression to fail?