CSE440: Natural Language Processing II

Lab Assignment 3

1. Given the following dataset of 2D points:

Index	X	Y
1	1.0	9.9
2	2.0	8.0
3	3.0	6.1
4	4.0	3.9
5	5.0	2.0

Write a Python program using Pandas, NumPy, and matplotlib to: fit a polynomial regression model of degree 2 ($\hat{y} = ax^2 + bx + c$) using gradient descent.

Epochs = 1000
Learning Rate = 0.0001
Initial value of a = 0, b = 0, c=0

At every 100 epochs, print the values of a, b, c, and plot the polynomial curve ($\hat{y}$) along with the original data points to visualize how the model fits.

2. Load California Housing dataset from sklearn.datasets. Preprocess the data using Pandas: normalize the features, handle missing values if any and others. Use Keras to implement a shallow neural network with the following fixed hyperparameters:

Number of Hidden layer = 1
Hidden units: 31
Activation: ReLU
Optimizer: Adam
Loss: Mean Squared Error (MSE)
Epochs: 250

Train the model and print the accuracy on a test set. (Split the dataset into 80:20)
Include the model summary and training vs test loss plot.

3. Repeat **Task 2**, but this time build a deep neural network using the following hyperparameters:Train the model and print the accuracy on a test set. (Split the dataset into 80:20)
Include the model summary and training vs test loss plot.

Number of Hidden layer = 3
Hidden units: 64, 32 and 16 units respectively.
Dropout: 0.2 on hidden layer 1 and 2.
Activation: ReLU
Optimizer: Adam
Loss: Mean Squared Error (MSE)
Epochs: 200

4. Use the Iris dataset from sklearn.datasets for classification. Follow these steps:

1. Load the Iris dataset and preprocess it (split the data into a training set and a test set with an 80:20 ratio).
2. Build a neural network for classification with the following parameters:
    - **Number of Hidden Layers**: 3
    - **Units in Hidden Layers**: 64 units in the first hidden layer, 32 units in the second, and 16 units in the third.
    - **Dropout**: Apply a dropout rate of 0.2 to the first two hidden layers.
    - **Activation Function**: Use ReLU for the hidden layers and Softmax for the output layer.
    - **Optimizer**: Use the Adam optimizer.
    - **Loss Function**: Use Categorical Cross-Entropy for classification.
    - **Epochs**: Set the number of epochs to 200.
3. Train the model and evaluate its accuracy on the test set.
4. Include the model summary and plot the training vs. test loss graph.