Assignment 1: Logistic Regression and Neural Networks

Homework assignments will be done individually: each student must hand in their own answers. Use of partial or entire solutions obtained from others or online is strictly prohibited. Electronic submission on Canvas is mandatory.

1. **Document Classification** (100 points) In this homework, you need to classify news articles into four categories: *World*, *Sports*, *Business*, and *Science/Technology* by building your own classifiers. The data provided is from AG news. Please follow the steps below:

   - (5pts) **Preprocess data**: Remove punctuation, stopwords and common words (e.g., "reuters","site"), irrelevant symbols, urls, and numbers if needed.
   - (7pts) **Feature selection:** Choose about 4000 words from the training data as input features (you can choose more or less total words); You can use document frequency to select the top 2000 words for each class.
   - (5pts) **Construct examples** Use the chosen words as features to build a TF-IDF feature vector for each news article in both training and testing.
   - (60pts) **Build** two classifiers (described below).
     1. (30 pts) Implement a Logistic Regression (LR) model with L_2 regularization:

        $$J = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ik}\log\left(\frac{\exp f_k}{\sum_{c=1}^{K}\exp f_c}\right) + \lambda\sum_{j=1}^{d} w_{kj}^2 \tag{1}$$

        – (10 pts) Given this formula, show the steps to derive the gradient of J with respect to $\mathbf{w}_k$ in your report.

        – (10 pts) Implement mini-batch gradient descent.

        – (10 pts) Implement stochastic gradient descent.
     2. (30 pts) Implement a Multilayer Perceptron (MLP) model with one hidden layer.

        – (10 pts) Implement the forward pass. Start with 50 neurons for the hidden layer. You can change this based on your validation errors.

        – (10 pts) Compute the gradients of J with respect to the parameters and implement back-propagation. You can use either SGD or mini-batch GD. Use the same cross-entropy loss as Eq. 1 (you can ignore the L_2 regularization)

        – (10 pts) Specify the settings of the model such as the network structure, the optimizer, the initial learning rate etc.
   - (8pts) Use **cross-validation** on the training data; Report the recall and precision for each category on the test and validation sets; Choose the best λ(in LR) and the number of neurons in the hidden layer (in MLP) using the validation set.
   - (5pts) **Plot** training loss and validation loss for LR and MLP.
   - (10pts) **Compare** the results of both classifiers in a table and provide an analysis for the results.

   Please follow the below instructions when you submit the assignment.

1. You are allowed to use packages for preprocessing text, TF-IDF, and plotting, but you are not allowed to use packages for implementing LR and MLP.
2. Your submission should consist of a zip file named Assignment1_LastName_FirstName.zip which contains:
   - a python file (.py) or jupyter notebook file(.ipynb). The file should contain the code and the output after execution (in comments if you submit a .py file). You should also include detailed comments.
   - a pdf file to show (1) the derivation steps of the gradient of J with respect to $\mathbf{w}_k$ in LR and (2) analysis on the results of both models (plots, tables, etc).