Programming Assignment 3: Logistic Regression

Instructions:

- The aim of this assignment is to give you an initial hands-on regarding real-life machine learning application.
- Use separate training and testing data as discussed in class.
- You can only use Python programming language and Jupyter Notebook.
- There are two parts of this assignment. In part 1, you can only use **numpy**, **scipy**, **pandas, matplotlib** and are not allowed to use **NLTK, scikit-learn or any other machine learning toolkit**. However, you have to use **scikit-learn** in part 2.
- **Carefully read the submission instructions, plagiarism and late days policy at the end of assignment.**
- Deadline to submit this assignment is: **Sunday 15th November, 2020.**

Problem:

The purpose of this assignment is to get you familiar with sentiment classification. By the end of this assignment you will have your very own "Sentiment Analyzer". You are given with Large Movie Review Dataset that contains separate labelled train and test set. Your task is to train a Logistic Regression classifier on train set and report accuracy on test set.

Dataset:

The core dataset contains 50,000 reviews split evenly into 25k train and 25k test sets. The overall distribution of labels is balanced (25k pos and 25k neg). There are two top-level directories [train/, test/] corresponding to the training and test sets. Each contains [pos/, neg/] directories for the reviews with binary labels positive and negative. Within these directories, reviews are stored in text files named following the convention [[id]_[rating].txt] where [id] is a unique id and [rating] is the star rating for that review on a 1-10 scale. For example, the file [test/pos/200_8.txt] is the text for a positive-labeled test set example with unique id 200 and star rating 8/10 from IMDb.

Preprocessing:

In the preprocessing step you're required to remove the stop words and punctuation marks and other unwanted characters from the reviews and convert them to lower case. You may find the string and regex module useful for this purpose. A stop word list is provided with the assignment statement.

Feature Extraction:

In the feature extraction step you can you'll represent each review by the 3 features x_0, x_1, x_2 and 1 class label y as shown in the table below:

Feature	Definition	Comment
x_0	1	bias term
x_1	count(positive words) ∈ review	Positive lexicon is provided
x_2	count(negative words) ∈ review	Negative lexicon is provided
y	1 if positive, 0 otherwise	Mentioned in directory name

Part 1:

Implement Logistic Regression from scratch keeping in view all the discussions from the class lectures. Feel free to read Chapter 5 of Speech and Language Processing book to get in-depth insight of Logistic Regression classifier. Specifically, you'll need to implement the following:

- Sigmoid function
- Cross-entropy loss function
- Batch Gradient Descent
- Prediction function that predict whether the label is 0 or 1 for test reviews using learned logistic regression (use the decision threshold of 0.5)
- Evaluation function that calculates classification accuracy and confusion matrix on test set (the expected accuracy on the test set is around 72%)
- Report plots with no. of iterations/ epochs on x-axis and training/ validation loss on y-axis.

Use the procedural programming style and comment your code thoroughly (just like programming assignment 2).

Part 2:

Use scikit-learn's Logistic Regression implementation to train and test the logistic regression on the provided dataset. Use scikit-learn's accuracy_score function to calculate the accuracy and confusion_matrix function to calculate confusion matrix on test set.

Submission Instructions:

Submit your code both as notebook file (.ipynb) and python script (.py) on LMS. The name of both files should be your roll number. If you don't know how to save .ipynb as .py see this. **Failing to submit any one of them will result in the reduction of marks**.

Plagiarism Policy:

The code MUST be done independently. Any plagiarism or cheating of work from others or the internet will be immediately referred to the DC. If you are confused about what constitutes plagiarism, it is YOUR responsibility to consult with the instructor or the TA in a timely manner. No "after the fact" negotiations will be possible. The only way to guarantee that you do not lose marks is “DO NOT LOOK AT ANYONE ELSE'S CODE NOR DISCUSS IT WITH THEM”.

Late Days Policy:

The deadline of the assignment is final. However, in order to accommodate all the 11th hour issues there is a late submission policy i.e. you can submit your assignment within 3 days after the deadline with 25% deduction each day.