Programming Assignment 2: 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.
- You can only use **numpy**, **matplotlib** and are not allowed to use **NLTK, scikit-learn or any other machine learning toolkit**.
- **Submit your code as one notebook file (.ipynb) on LMS. The name of file should be your roll number.**
- Deadline to submit this assignment is: **Sunday 14th March, 2020 11:59 p.m.**

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 evaluation metrics 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.

Dataset Preprocessing:

You'll represent each review by the 6 features $x_1 \ldots x_6$ and 1 class label y as shown in the table below:

Feature	Definition	Comment
x_1	count(positive words) ∈ review	Positive lexicon is provided
x_2	count(negative words) ∈ review	Negative lexicon is provided
x_3	Star Rating (1-10 scale)	Mentioned in filename
x_4	log(word count of review)	
x_5	1 if "no" ∈ review, 0 otherwise	
x_6	1 if "!" ∈ review, 0 otherwise	
y	1 if positive, 0 otherwise	Mentioned in directory name

Implementation:

Implement Logistic Regression 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:

- Preprocessing function
- Sigmoid function
- Cross-entropy loss function
- Gradient Descent (both stochastic and batch)
- Prediction function that predict whether the label is 0 or 1 for test reviews using learned logistic regression
- Report evaluation with both stochastic and batch gradient descent

Use the procedural programming style and comment your code thoroughly (just like programming assignment 1).

Evaluation:

You are required to provide a confusion matrix (like the one below) with values obtained by running your Logistic Regression classifier on test set. Also report Precision, Recall, Accuracy and F1 score.

		gold labels	
		pos	neg
system/ classifier output	pos	t_p	f_p
	neg	f_n	t_n