Programming for Data Science
Assignment 2

October 29, 2021

Dr Phil Smith

This assignment is worth 25% of the module mark. It is based upon the topics that you have been learning so far. In particular, this will focus on the concept of Classes. Each programming question should be submitted in a separate .py file. e.g Q1.py, Q2.py etc. All files and functions in the file must be given docstrings describing what you have done and code should be suitably formatted on penalty of losing marks.

1. (10 marks) Write a program that reads a comma separated value file of IMDB's top 1000 films and stores them as objects in Python. You should define the classes for this program as follows:

   - **Films**: This class will hold an instance variable that is a list containing *Film* objects.

     The **Films** class has the following methods:

     - *readFile*: This takes a single parameter that is the string path to the location of the file containing film information. The file will be stored as an instance variable
     - *createFilmsList*: This method will access the instance variable containing the file information and append *Film* objects to the list created in the constructor. This should not create a Film object for the header in the file.
     - *getFilmsList*: This takes no parameters and returns the instance variable containing the list of Film objects.
   - **Film**: This class contains the data about a film. In the constructor it will be passed the title, year of release and runtime. These will be stored as instance variables. The Film class will also contain *getter* methods for each of the instance variables. e.g *getTitle()* etc.

   To test that this program works, write and call a main method that creates a Films object that will hold the data about the IMDB Top 1000 films (from imdb_top_1000.csv.) Following this, print to the screen the string representations of the films for the following two test cases: 1. those that have a runtime of less than 70 minutes. 2. any films published after 2019. Output should be in the following format for each Film:

   > "The Shawshank Redemption was released in 1994 and has a runtime of 142 minutes."

2. (10 marks) In this question you will write a program to analyse the sentiment of a piece of text (whether a text conveys a positive or negative opinion).

   There are two parts to this program. First, write a class called **SentimentLexicon**. In this class there should be an instance variable initialised in the constructor that is a dictionary. In this class, write a method that takes two parameters, paths to both of the word lists, and reads in both of these files. Each line of each file should be read and stored into the dictionary instance variable as follows: the key for the dictionary is the word and the value is a score that represents the sentiment of a word. A word from the positive word list file should have a value of 1 and a value from the negative word list file should have a value of -1.

   Next you should create a class called **Classifier**. The instance variable for this should be an instantiated SentimentLexicon object. The Classifier class should have a method called *classify*, that takes a single parameter, which is a text string. For each word in the string, this should be looked up in the sentiment lexicon and a score calculated by summing the values for each word. If the score overall is positive a value of 1 should be returned from the classify method. If overall the score is negative, a value of -1 should be returned. If the score was 0, then this should be returned (a 0 is indicative of no sentiment).

   To test this program, write and call a main method that creates a Classifier object and iteratively classifies each of the following sentences (leaving punctuation in):

   - I love Python.
   - Python is the language I love!
   - The iPhone is clearly not the most terrible and worst phone ever. It is the best.

   For each sentence that is classified, store the results in separate dictionaries as follows and print them out:

   ```
   {'text':'I love Python', 'sentiment':1}
   ```

3. (5 marks) The final five marks are reserved for creative points in this assignment. Extend the functionality of either one of the programs that you wrote for question 1 or question 2. Marks will be awarded for aspects such as the novelty, complexity and usefulness of the extension that you have written. Submit the updated program as Q3.py and outline in a docstring at the top of the file what you have done to extend the functionality.