Machine Learning (PG) M2021
Assignment-3

Max Marks: 150 **Due Date**: 11/11/2021, 11:59 PM

Instructions

- The assignment is to be attempted in groups of atmost 2 students.
- You can only use python as the programming language.
- You are free to use math libraries like `Numpy`, `Pandas`, `SciPy` etc.; any library is allowed for visualizations; and utility libraries like `os`, `pickle`, etc. are fine.
- Usage instructions regarding the other libraries is provided in the questions. **Do not use any ML module that is not allowed.**
- Create a '.pdf' report that contains your approach, pre-processing, assumptions etc. Add all the analysis related to the question in the written format in the report, anything not in the report will not be marked. Use plots where ever required. **Submit code, readme and report files in ZIP format with naming convention** `A3_rollno1_name1_rollno2_name2.zip`. This nomenclature has to be followed strictly. (If only one member: `A3_rollno_name.zip`)
- Remember to **turn in** after uploading on Google Classroom. No justifications would be taken regarding this after the deadline.
- Start the assignment early. Resolve all your doubts from TAs in their office hours **two days before the deadline.**
- You should be able to replicate your results during the demo, failing which will fetch zero marks.

---

1. (50 points) Download IMDB Moview Review dataset and use it for the following experiments. Split the dataset in a 80 : 20 split where 80% of the data will be used for training and the rest 20% will be used for testing.
    1. A word cloud is an image composed of words used in a particular text, in which the size of each word indicates its frequency or importance. Create a word cloud of the positive and negative reviews separately to see the prevalent words present in them. **(5 pts.)**
    2. Perform basic text preprocessing for all the reviews. Preprocessing should involve the following- converting review to lower case, removing punctuations, removing noisy texts like URLs, removing stopwords. Draw the word cloud again and report the differences observed, if any. **(10 pts.)**

3. Get Word2Vec word-vectors for each word present in each review (You can use the python library `Gensim` for this). Get a single vector representation for each review by taking the average of the vectors of all words present in it. Draw a 2D PCA plot using these vectors and see if the classes are separable or not. **(20 pts.)**
4. Use SVM from sklearn to learn a classifier for these reviews. Play around with different kernels and report all the results and analysis in the report. **(10 pts.)**
5. Visualise the decision boundary obtained by the best model on the PCA plot and comment on it. **(5 pts.)**

2. (50 points) Download The Oxford-IIIT Pet Dataset and use it for the following experiments. Split the dataset in a 80 : 20 split where 80% of the data will be used for training and the rest 20% will be used for testing. Reduce the size of the dataset images to a standard (32,32) for all images in train and test. Choose hyper-parameters in a systematical manner.

   1. (a) Perform PCA using sklearn on the dataset such that 90% of the total variance is retained - feature descriptor **(5 pts.)**
      (b) Combine Canny Edge Detection (CED) and color histogram (must be implemented from scratch) as a whole ie., (CED + color hist) - feature descriptor **(20 pts.)**

      Now perform the following for both these feature descriptors. (a) + (b)
   2. Visualize the 2D t-SNE plot and state your observations. Added, visualize any 5 images from features extracted using CED. **(10 pts.)**
   3. Use GridSearchCV (cv=5) to find the best parameters (C, kernel, in case of gaussian kernel) of SVM using the train set. Report the accuracies (train, test) and the run times on the best parameters obtained. State your observations (if any) on the obtained best parameters. **(10 pts.)**
   4. Develop a new training set by extracting the support vectors from the SVM fitted in (3). Now fit another SVM with the new training set and report the accuracies(train, test). Compare the accuracies from (3) and (4). State your observations. **(5 pts.)**

3. (50 points) Dataset: data. Use five fold cross validation (from scratch). (Dataset is small with just 3 labels, hence five fold cross validation is feasible to implement. Not implementing with five folds will fetch 0 marks.)
   You can only use the fit() from the sklearn for the SVM. Other tasks are to implemented from the scratch using the attributes of the model created. You can write a class that can utilize the fit() of the sklearn, and other methods of the class (such as predict()) can use the resultant attributes to achieve the other tasks.
   (1) Visualize the (whole dataset) and state your observations. **(5 pts.)**
   (2) Use a SVM with RBF kernel to classify this dataset through one vs rest approach. Perform the grid search for parameters C and to obtain their optimal values (only on 1 fold). Report the accuracy over the five folds along with the mean accuracy. Also, report the mean class accuracy. **(20 pts.)**

(3) Use a SVM with RBF kernel to classify this dataset through one vs one approach. Perform the grid search for parameters C and to obtain their optimal values (only on 1 fold). Report the accuracy over the five folds along with the mean accuracy. Also, report the mean class accuracy. **(20 pts.)**
(4) For the optimal values of the parameters obtained in (2) and (3), use the sklearn to make the predictions on the test set. Is there any deviation in the performance? **(5 Pts.)**
The OVO and OVR also have to implemented from the scratch.