EECS 461 / ECE 523 - MACHINE LEARNING
Assignment 4

Image Recognition

In this assignment you are going to perform some basic image recognition tasks. The image dataset we have provided you contains images (in jpeg format) of dimensions 120x128, with the training set consisting of 315 images and the test set consisting of 90 images.

In each image, the subject has the following characteristics:

- Name – name of the subject
- Direction Faced – left, right, straight, up
- Emotion – happy, sad, neutral, angry
- Eyewear – open, sunglasses

Each image follows the naming convention "*name_directionFaced_emotion_eyewear.jpg*"

Direction Faced Analysis

Create X_train using images in the *TrainingSet* folder. You will use PIL module to open jpeg files:

```
from PIL import Image
image_array = np.array(Image.open('sample_image.jpeg').convert('L'))
```

Each image has shape of 120x128. Flatten each image array to a vector of dimensions 1x15360. Label of the image will be maintained from the file name.
Create y_train_directionfaced using images' file names. For instance, if the file name is *aaa_right_neutral_eyewear.jpg,* then the label of the image is 'right'. Use the following dictionary to encode directions into a numerical format:

```
direction_encode = {'right': 0, 'left': 1, 'up': 2, 'straight': 3}
```

At the end, X_train will be a numpy array that contains 315 images of dimensions 1x15360 and y_train_directionfaced array will contain 315 encoded image labels.

Create X_test and y_test_directionfaced arrays using the TestSet folder.

a. (20 points) Train a Random Forest classifier (with parameters of random_state=0) on the training dataset and time how long it takes, then evaluate the resulting model on the test set. Return the trained model, time of training, and accuracy on the test set in a pickle format as ***part_a.pkl***.

b. (25 points) Use PCA to reduce the training dataset's dimensionality, with an explained variance ratio of 95%. Train a new Random Forest classifier on the reduced dataset and see how long it takes. Was training much faster? Return the trained model, time of training, and accuracy on the test set in a pickle format as ***part_b.pkl***.

Emotion Analysis

You will again use X_train and X_test in this part but this time you will use emotion as a label. For instance, if the file name is *aaa_right_neutral_eyewear.jpg,* then the label of the image is 'neutral'. Create y_train_emotion and t_test_emotion according to emotion label. Use the following dictionary to encode emotions into a numerical format:

```
emotion_encode = {'neutral': 0, 'happy': 1, 'angry': 2, 'sad': 3}
```

c. (20 points) Train a Logistic Regression classifier (with parameters of multi_class="multinomial", solver="lbfgs", random_state=0) on the training dataset and time how long it takes, then evaluate the resulting model on the test set. Return the trained model, time of training, and accuracy on the test set in a pickle format as ***part_c.pkl***.

d. (25 points) Use PCA to reduce the training dataset's dimensionality, with an explained variance ratio of 95%. Train a new Logistic Regression classifier on the reduced dataset and see how long it takes. Was training much faster? Return the trained model, time of training, and accuracy on the test set in a pickle format as ***part_d.pkl***.

Saving Results

In order to save your answers in pickle format, you will use the scikit-learn's joblib as follows:

```
clf = anyModel()
from sklearn.externals import joblib
joblib.dump(clf, 'filename.pkl')
```

Assignment Submission (10 points)

At the end, you have 4 pickle files. Put these pickle files and your .py file into a folder and name the folder as **<Your first name>_<Your last name>_Assignment4**, then compress the file.

- Don't use Turkish characters or space in the folder name.

(5 points) Proper folder name.
(5 points) Proper format of files & file names.