Project

Project - Initial Release

Domain: Face recognition

Tasks: 1. Classify photos as "George Bush" "Not George Bush". 2. Classify photos as "Serena Williams" "Not Serena Williams".

Data:

https://drive.google.com/open?id=12fAqxqP8F3mqMqikbMbQgVnxtG1A0RwN (470 MB)

64x64, gray-scale photos. There are 13,233 photos total.

X.csv: all photos, where each row correspond to a photo, and columns correspond to pixels. You can reshape each row array into a 64x64 2D array and visualize the photo if you like.

y_bush_vs_others.csv: Labels for task 1

y_williams_vs_others.csv: Labels for task 2

Performance metrics: accuracy (I expect it to be pretty much meaningless for these tasks), AUC, Precision for class 1, Recall for class 1, F1 for class 1. Try playing with the decision threshold to increase precision, recall, and/or F1.

Classifiers: any scikit-learn classifier you like, except neural networks at the moment. Only scikit-learn classifiers; no other ML/image packages at the moment. You have to try at least KNN and SVMs with various kernels.

Submit: Nothing at the moment. Simply play with the data and various classifiers.

Project Phase 1

Do the following steps, once for the "Bush" and once for the "Williams" dataset.

1. Load the dataset.
2. Pick a classifier (more on this later).
3. Perform three-fold cross validation using the following code: stratified_cv_results = cross_validate(clf, X, y, cv=StratifiedKFold(n_splits = 3, shuffle=True, random_state = lastfourdigitsofyourA#), scoring=('precision', 'recall', 'f1'), return_train_score=False). You may also set n_jobs parameter if it helps you perform experiments faster.
4. Record results in a Google Sheet, using the following format: https://docs.google.com/spreadsheets/d/1ZLs7NI9eilqIJxiDRYT-cej5pgVzkgzZUKeK47AvzxU/edit?usp=sharing Make a copy of the sheet and fill it.
5. Put the results of the mean F1 results for n_neighbors=1, n_neighbors=3, n_neighbors=5, and best SVC result you got to a list and pickle it.
6. Create a pdf of the google sheet.

Classifiers you need to experiment with:

1. KNeighborsClassifier, with n_neighbors = 1, 3, and 5.
2. SVC with various C values, kernels, and various parameters for those kernels (such as gamma for rbf, degree for poly, etc).

In the Google sheet, report all KNeighborsClassifier results. For SVC, list all the parameter settings that you tried that resulted in a mean F1 score, and then report the parameter settings and results for the best result you got, in terms of mean F1.

Here are the files you need to submit:

1. bush.pickle
2. bush.pdf
3. williams.pickle
4. williams.pdf

Any questions about the format of the Google sheet? Please ask on Piazza.

Project Phase 2

Do the following steps, once for the "Bush" and once for the "Williams" dataset.

1. Load the dataset.
2. Fit and transform the full dataset using PCA. https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html You'll need to experiment with the parameters (at least with the n_components parameter for sure; you might want to experiment with the other parameters as well) to get a good result for the following steps.
3. Perform the following steps once for the "Bush" and once for the "Williams" target.
4. Pick a classifier (experiment with KNeighborsClassifier and SVC).
5. Perform three-fold cross validation using the following code: stratified_cv_results = cross_validate(clf, X, y, cv=StratifiedKFold(n_splits = 3, shuffle=True, random_state = lastfourdigitsofyourA#), scoring=('precision', 'recall', 'f1'), return_train_score=False). You may also set n_jobs parameter if it helps you perform experiments faster. In this case, X is the PCA transformed data, with good parameter settings.
6. Record only the mean F1 result in a Google Sheet, using the following format: https://docs.google.com/spreadsheets/d/1pf9WymvfcoaS0RxSgGuM6pNa25g4lKePxwUaA_ZxWTE/ Make a copy of the sheet and fill it.
7. Put the best mean F1 results for KNeighborsClassifier and SVC into a list of size 2 and pickle it with the appropriate name.
8. Create a pdf of the google sheet.

Your task is to experiment with a number of PCA parameters and classifier parameters to achieve the best mean F1 results you can get. Note that the PCA parameter settings that work for KNeighborsClassifier might not work well for SVC, and similarly, the best parameters for Bush might not be the best parameters for Williams.

Here are the files you need to submit:

1. bush.pickle
2. bush.pdf
3. williams.pickle
4. williams.pdf

Any questions? Please ask on Piazza.

Note: Considering to submit late? Please see slide 16 of the syllabus.

Project Phase 3

In this phase of the project, you'll need to develop and learn a deep learning model of the Bush and Williams splits (separately, of course).

Many of us do not have access to powerful computing; and hence, instead of three-fold cross-validation, we will perform a train-test split evaluation.

Repeat the following once for Bush and once for Williams dataset.

Use sklearn.model_selection.train_test_split to split your data into a train test split. Use test_size = 1/3, random_state = lastfourdigitsofyourA#, shuffle = True, and stratify is turned on appropriately (this is important).

Design a deep learning model; for image classification, it is best to use CNNs and maxpooling; CNNs followed by maxpooling, and then CNNs and maxpooling, etc. Most models use one or two dense layers at the end, before the output layer. The output layer has to be a single node, with a sigmoid activation.

Note that, you might need to reshape your input to a 64x64 and then use a 2D-CNN, instead of using an array 4096 followed by a 1D-CNN.

You don't have a lot of training data; so, you probably want to keep your structure somewhat simple.

Report F1 results on both "train" and "test" splits.

For submission, you'll need five files; two for Bush, two for Williams, and one pdf file. One is the model file, saved using Keras's save model function: https://keras.io/getting-started/faq/#how-can-i-save-a-keras-model. One is the pickle file, that has an array of two results [F1_on_train, F1_on_test].

1. bush.model
2. bush.pickle
3. williams.model
4. williams.pickle
5. phase4.pdf

Phase4.pdf file should contain a description of the structure of your Bush model and your Williams model (Note that these models can be but do not have to be same). We prefer drawings of your model structure, that include the type of the layers, and the dimensions of the layers, etc.

Any questions? Please ask on Piazza.

Note: Considering to submit late? Please see slide 16 of the syllabus.