Project 3 | Group 5

Step 0: Prepare Environment and Load Packages

Before you run next chunk, please follow the instructions to install all packages we need.

Pre-requirements:

numpy, random, pickle, time, xgboost, PIL, gist, csv, FFTW

(1) Install numpy, random, pickle

$ pip install numpy

$ pip install random

$ pip install pickle

(2) Install FFTW

FFTW download: http://www.fftw.org (http://www.fftw.org)

Install instruction: http://www.fftw.org/fftw3_doc/Installation-on-Unix.html (http://www.fftw.org/fftw3_doc/Installation-on-Unix.html)

$ ./configure --enable-single --enable-shared

$ make

$ sudo make install

(3) Install gist

Download lear_gist: https://github.com/tuttieee/lear-gist-python (https://github.com/tuttieee/lear-gist-python)

$ sudo python setup.py build_ext

$ python setup.py install

If fftw3f is installed in non-standard path (for example, HOME/local), use -I and -L options:

$ sudo python setup.py build_ext -I HOME/local/include -L HOME/local/lib

(4) Install xgboost

Instructions for Install XGBoost on Mac OSX : https://www.ibm.com/developerworks/community/blogs/jfp/entry/Installing_XGBoost_on_Mac_OSX?lang=en (https://www.ibm.com/developerworks/community/blogs/jfp/entry/Installing_XGBoost_on_Mac_OSX?lang=en)

You might encounter a problem when insert command "make -j4". Here is an efficeint way to solve the problem: https://stackoverflow.com/questions/36211018/clang-error-errorunsupported-option-fopenmp-on-mac-osx-el-capitan-buildin (https://stackoverflow.com/questions/36211018/clang-error-errorunsupported-option-fopenmp-on-mac-osx-el-capitan-buildin)

In [1]:

```
import GIST
import pandas as pd
import random
import pickle
import time
import xgboost
```

Step 1: Read Test Pictures Information

Before you run next chunk, please make sure you meet following requirements:

(1) Make sure path variable is where you store all your test images

(2) Make sure 5000 SIFT feature descriptors of your test images are stored in the data folder as feature_sift_test.csv

(3) Make sure label of your test images are stored in the data folder as label_test.csv

In [2]:

```
path = "/Users/siyi/Documents/Study-Columbia/17FALL/GR5243-Applied-Data-Science/
Project3/training_set/images2"
GIST.feature_output(path)
gist_new = pd.read_csv('feature.csv', skiprows=1, header = None).iloc[:, 1:]
sift_new = pd.read_csv('../data/feature_sift_test.csv').iloc[:, 1:]
label_new = pd.read_csv('../data/label_test.csv').iloc[:, 1]
feature = pd.concat([sift_new, gist_new], axis=1)
feature.columns = ['x' + str(i+1) for i in range(5000)] + ['f' + str(i+1) for i
in range(960)]
```

Step 2: XGBoost Model

In [3]:

```
# require X_test, y_test
X_test = feature
y_test = label_new
```

In [4]:

```
# load the baseline model
filename = '../output/model_baseline.sav'
xgb_1 = pickle.load(open(filename, 'rb'))

# load the tuned xgboost model
filename = '../output/model_tuned.sav'
xgb_2 = pickle.load(open(filename, 'rb'))
```

In [5]:

```
print("Baseline: ")
pred = xgb_1.predict(X_test)
y_label = y_test.values
print ('classification error=%f' % (sum([pred[i] != y_label[i] for i in range(le
n(y_label))]) / float(len(y_label)) ))
print ('You can check training time in the file xgboost_train.py.')

print("Tuned: ")
pred = xgb_2.predict(X_test)
y_label = y_test.values
print ('classification error=%f' % (sum([pred[i] != y_label[i] for i in range(le
n(y_label))]) / float(len(y_label)) ))
print ('You can check training time in the file xgboost_train.py.')
```

```
Baseline:
classification error=0.000000
You can check training time in the file xgboost_train.py.
Tuned:
classification error=0.000000
You can check training time in the file xgboost_train.py.
```

In [ ]:

```

```