SIT789 - Applications of Computer Vision and Speech Processing

Credit Task 9.2: Speaker recognition using GMMs

Speaker Recognition method:

```
likelihood=np.zeros(len(gmms))
def speaker_recognition(audio_file_name, gmms):
  f= mfcc_extraction(audio_file_name, hop_duration,num_mfcc)
  for i in range(len(gmms)):
    score = gmms[i].score(f)
    likelihood[i]=score.sum()

  speaker_id = np.argmax(likelihood)
  print(np.max(score))
  return speaker_id
```

Testing on all samples:

```
for i  in test_file_names:
  speaker_id = speaker_recognition(i, gmms)
  print(speakers[speaker_id])
```

```
Bassel
-98.2911389007375
Bassel
-106.55775570582445
Bassel
-111.71420782446337
Bassel
-111.99074803111284
Bassel
-125.43875064090845
Beady
-124.69144136354328
Beady
-122.71700677635731
Beady
-128.91289990464426
Beady
-132.2949551157578
Beady
-127.55702634111144
Beady
-118.97239918962177
Beady
-242.67219913887794
Arthur
-243.90147462912495
Bart
-215.62805098776028
Arthur
-212.75820057426122
Arthur
-195.44645800667126
```

Evaluation algorithm report:

```
train_s = []
test_s = []
for filename in train_file_names:
  train_s.append(speaker_recognition(filename, gmms))
for filename in test_file_names:
  test_s.append(speaker_recognition(filename, gmms))
train_s=np.array(train_s)
test_s=np.array(test_s)
train_s=train_s.reshape(-1,1)
test_s=test_s.reshape(-1,1)
svm_1=svm.SVC()
svm_1.fit(train_s,train_labels)

pred_labels = svm_1.predict(test_s)
print('SVM report:', np.sum(pred_labels ==test_labels)/len(test_labels))


ada= AdaBoostClassifier()
ada.fit(train_s,train_labels)

pred_labels1 = ada.predict(test_s)
print('AdaBoost report:', np.sum(pred_labels1 ==test_labels)/len(test_labels))
```

```
SVM report: 0.9142857142857143
AdaBoost report: 0.6228571428571429
```