

Task_3

February 8, 2022

```
[51]: import pandas as pd
import numpy as np
import librosa as lr
import matplotlib.pyplot as plt
from scipy.signal.signaltools import wiener
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split
from sklearn.metrics import plot_confusion_matrix
from sklearn.metrics import f1_score, precision_recall_curve
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import cross_val_score
import pywt
```

[51]:

```
[60]: files = lr.util.find_files(r'C:
    ↳\Users\markm\Desktop\project\2022-knu-bioacoustics\bones_data')
files = np.asarray(files)
len(files)
```

[60]: 238

```
[37]: #initialize audio data
output_list = [0]*118 + [1]*120
```

```
[38]: targetframe = pd.DataFrame({'path':files, 'output':output_list})
```

```
[39]: class AudioFeatures:
    def __init__(self, files, path_column, sample_rate=16000):
        self.files = files
        self.path_column = path_column
        self.sample_rate = sample_rate
        self.audios = []
        self.features = []
```

```

def extract_all_features(self):
    self.load()
    self.energy_normalize()
    self.get_features()
    return self.create_features_df()

def load(self):
    for path in range(len(self.path_column)):
        audio, sample_rate = lr.load(self.path_column.values[path], sr =
→self.sample_rate)
        self.audios.append(audio)
def energy_normalize(self):
    for i, audio in enumerate(self.audios):
        w_audio = wiener(audio)
        current_energy = np.sum(w_audio ** 2)
        self.audios[i] = w_audio / np.sqrt(current_energy)
def get_features(self):
    for audio in self.audios:
        stft = np.abs(lr.stft(audio))
        result = np.array([])
        chroma = np.mean(lr.feature.chroma_stft(S=stft, sr=self.sample_rate).
→T, axis=0)
        result = np.hstack((result, chroma))
        mel = np.mean(lr.feature.melspectrogram(audio, sr=self.sample_rate).
→T, axis=0)
        result = np.hstack((result, mel))
        contrast = np.mean(lr.feature.spectral_contrast(S=stft, sr=self.
→sample_rate).T, axis=0)
        result = np.hstack((result, contrast))
        cA, cD = pywt.dwt(audio, 'db1')
        result = np.hstack((result, [cA.mean(), cD.mean()]))
        self.features.append(result)
def create_features_df(self):
    column_names = []
    for j in range(147):
        if j < 12: column_names.append('chroma_' + str(j))
        elif j >= 12 and j < 140: column_names.append('mel_' + str(j-12))
        else: column_names.append('contrast_' + str(j-140))
    column_names.append('wavelet_cA')
    column_names.append('wavelet_cD')
    featuresframe = pd.DataFrame(self.features, columns = np.
→array(column_names))
    featuresframe['path'] = files
    return featuresframe

```

```
[40]: AF = AudioFeatures(files, targetframe.path)
featuresframe = AF.extract_all_features()
featuresframe.head()
```

```
E:\Anaconda\lib\site-packages\scipy\signal\signaltools.py:1475: RuntimeWarning:
divide by zero encountered in true_divide
```

```
res *= (1 - noise / lVar)
```

```
E:\Anaconda\lib\site-packages\scipy\signal\signaltools.py:1475: RuntimeWarning:
invalid value encountered in multiply
```

```
res *= (1 - noise / lVar)
```

```
[40]: chroma_0 chroma_1 chroma_2 chroma_3 chroma_4 chroma_5 chroma_6 \
0 0.632079 0.529796 0.545900 0.505762 0.517238 0.553138 0.481255
1 0.594664 0.466061 0.493480 0.454627 0.496202 0.534354 0.469201
2 0.624867 0.486126 0.502495 0.464540 0.492716 0.548291 0.493582
3 0.589826 0.438803 0.471486 0.462928 0.547951 0.605022 0.505054
4 0.643045 0.570014 0.523268 0.516824 0.600332 0.681074 0.527220

chroma_7 chroma_8 chroma_9 ... contrast_0 contrast_1 contrast_2 \
0 0.637571 0.516364 0.537102 ... 17.337019 24.908236 30.876651
1 0.685500 0.508953 0.491237 ... 18.635553 26.206566 30.159046
2 0.669318 0.530182 0.502750 ... 19.255570 25.967992 30.345703
3 0.725907 0.519484 0.463850 ... 18.775563 23.933087 29.808447
4 0.713372 0.569016 0.631481 ... 17.722464 23.870836 28.071499

contrast_3 contrast_4 contrast_5 contrast_6 wavelet_cA wavelet_cD \
0 18.964371 18.914433 18.760497 13.073751 0.000081 1.513449e-09
1 19.636376 17.346977 18.997591 13.265693 0.000020 -1.966753e-07
2 19.313075 18.645173 19.179313 13.232155 0.000037 2.318190e-07
3 19.991412 18.331584 18.510110 13.105874 0.000031 2.108760e-07
4 19.540765 19.294940 18.441520 12.708895 0.000001 2.092521e-07

path
0 C:\Users\markm\Desktop\bones_data\Maria_hand_f...
1 C:\Users\markm\Desktop\bones_data\Maria_hand_f...
2 C:\Users\markm\Desktop\bones_data\Maria_hand_f...
3 C:\Users\markm\Desktop\bones_data\Maria_hand_f...
4 C:\Users\markm\Desktop\bones_data\Maria_hand_f...
```

```
[5 rows x 150 columns]
```

```
[44]: merged_dataframe = pd.merge(featuresframe, targetframe, on='path')
merged_dataframe.shape
```

```
[44]: (238, 151)
```

```
[13]: merged_dataframe.head()
```

```
[13]: chroma_0 chroma_1 chroma_2 chroma_3 chroma_4 chroma_5 chroma_6 \
0 0.637388 0.537756 0.557660 0.514624 0.523419 0.558437 0.488089
1 0.602096 0.476943 0.508802 0.466705 0.505800 0.542673 0.477051
2 0.629650 0.493800 0.514160 0.474915 0.500193 0.553764 0.500037
3 0.596047 0.447029 0.482445 0.472345 0.554453 0.610405 0.511621
4 0.647448 0.575883 0.530943 0.523121 0.604158 0.683875 0.532571

chroma_7 chroma_8 chroma_9 ... contrast_1 contrast_2 contrast_3 \
0 0.641312 0.522931 0.545891 ... 24.984384 31.481056 19.048178
1 0.689177 0.517138 0.501273 ... 26.042326 30.234271 19.435843
2 0.672046 0.535884 0.511762 ... 25.901820 30.880173 18.952288
3 0.729326 0.525377 0.472263 ... 23.941591 29.841835 19.782401
4 0.717968 0.574256 0.636712 ... 23.872444 28.500406 19.350508

contrast_4 contrast_5 contrast_6 wavelet_cA wavelet_cD \
0 18.013709 18.269762 22.978542 0.000356 1.998557e-06
1 16.493433 18.029449 24.579640 0.000079 -9.652582e-07
2 17.886660 18.502711 23.257113 0.000169 9.261599e-07
3 17.547886 18.057543 24.362124 0.000120 1.334696e-06
4 18.433448 18.128938 22.580220 0.000007 3.142485e-07

path output
0 C:\Users\markm\Desktop\bones_data\Maria_hand_f... 0
1 C:\Users\markm\Desktop\bones_data\Maria_hand_f... 0
2 C:\Users\markm\Desktop\bones_data\Maria_hand_f... 0
3 C:\Users\markm\Desktop\bones_data\Maria_hand_f... 0
4 C:\Users\markm\Desktop\bones_data\Maria_hand_f... 0
```

[5 rows x 151 columns]

```
[57]: X_train, X_test, y_train, y_test = train_test_split(merged_dataframe.
    →drop(columns = ['path', 'output']),
                                merged_dataframe.output,
    →test_size=0.3, random_state=3)
parametr = {'n_estimators':range(5,41,5), 'max_depth':range(1,7),
    'min_samples_leaf':range(1,8), 'min_samples_split':range(2,4)}
voice_rf = RandomForestClassifier(random_state=0)
search = GridSearchCV(voice_rf, parametr, n_jobs=-1, cv=5)
search.fit(X_train,y_train)
best_voice_forest = search.best_estimator_
```

```
[58]: predictions = best_voice_forest.predict(X_test)
scores = cross_val_score(best_voice_forest, merged_dataframe.drop(columns =
    →['path', 'output']),
                                merged_dataframe.output, cv=5)
print(f'Accuracy: {(predictions == y_test.to_numpy()).sum()}/{len(predictions)}')
print('Cross-validation score: ', scores.mean())
```

Accuracy: 68/72

Cross-validation score: 0.8236702127659574

```
[59]: plot_confusion_matrix(best_voice_forest, X_test, y_test)  
      plt.show()
```

