Feature Engineering

Papangkorn Inkeaw, Ph.D.

Feature Extraction for Time Series

Lab 8

Load dataset

Dataset: Acted Emotional Speech Dynamic Database (http://m3c.web.auth.gr/research/aesdd-speech-emotion-recognition/)

- Import libraries

```
import pandas as pd
import numpy as np
from scipy.io import wavfile
import scipy.io
import tsfresh
import matplotlib.pyplot as plt
from pyts.transformation import ShapeletTransform
```

- Read a wav file

```
samplerate, data = wavfile.read(r'ActedEmotionalSpeechDynamicDatabase/anger/a01 (1).wav')
print(data.shape)
```

Extract Global Features using `tsfresh` library

- Extract first-order statistics features

```
mean_f = tsfresh.feature_extraction.feature_calculators.mean(data)
std_f = tsfresh.feature_extraction.feature_calculators.standard_deviation(data)
skewness_f = tsfresh.feature_extraction.feature_calculators.skewness(data)
max_f = tsfresh.feature_extraction.feature_calculators.maximum(data)
min_f = tsfresh.feature_extraction.feature_calculators.minimum(data)
first_order_fv = np.array([mean_f, std_f, skewness_f, max_f, min_f])
```

- Extract autocorrelation features

```
def autocorrelation(data, max_lag = 10):
    auto_corr_fv = []
    for lag in range(2, max_lag+1):
            tmp = tsfresh.feature_extraction.feature_calculators.autocorrelation(data, lag)
            auto_corr_fv.append(tmp)
    auto_corr_fv = np.array(auto_corr_fv)
    return auto_corr_fv
autocorr_fv = autocorrelation(data)
```

Extract Global Features using `tsfresh` library

- Extract Fourier coefficients

```
def Fourier_coefficients(data, max_coeff = 10):
    coeff_real = []
    coeff_imag = []
    for coeff in range(0, max_coeff+1):
            tmp = tsfresh.feature_extraction.feature_calculators.fft_coefficient(data,[{"coeff":
coeff,"attr":"real"}, {"coeff": coeff,"attr":"imag"}])
            tmp_l = list(tmp)
            coeff_real.append(tmp_l[0][1])
            coeff_imag.append(tmp_l[1][1])
    auto_corr_fv = np.array(auto_corr_fv)
    coeff_real = np.array(coeff_real)
    coeff_imag = np.array(coeff_imag)
    fft_fv = np.concatenate((coeff_real, coeff_imag), axis=None)
    return fft_fv
fourier_coeff_fv = Fourier_coefficients(data)
```

Extract Features for Classification

- List all Wav files in all subfolder in the corpus

```
filenames = []
y = []  #class labels list
class_no = 1
for dirName in glob.glob("ActedEmotionalSpeechDynamicDatabase/*/"):
     for imgFile in glob.glob(dirName+"*.wav"):
               filenames.append(imgFile)
               y.append(class_no)
     class_no = class_no + 1
```

- Retrieve each wav file and extract feature

```
feature_len = 9
X = np.zeros((len(y),feature_len), dtype=float)
i = 0
for wavFile in filenames:
     s, data = wavfile.read(wavFile)
     X[i,:] = autocorrelation(data)
     i = i + 1
```

Your work!

1. Use the Acted Emotional Speech Dynamic Database
2. Extract feature vectors of samples on the dataset
3. Split the dataset into training and test dataset (Hint: use *train_test_split* method in sklearn library)
4. Construct a classifier using the training samples for identifying emotion type
5. Evaluate the performance of the classifier on the test set
6. Submit your program to the assignment submission system (http://hw.cs.science.cmu.ac.th/).

Note:

- Put your name and student ID in the first cell using comment tag.
- Name your python notebook file with the pattern Lab_08_XXXXXXXXX.py (XXXXXXXXX is your student ID)

References & Study Resources

- http://m3c.web.auth.gr/research/aesdd-speech-emotion-recognition/
- https://tsfresh.readthedocs.io/en/latest/
- https://pyts.readthedocs.io/en/stable/index.html