Homework 3: Emotion Recognition

For this assignment you will work on emotion recognition in speech. The speech segments are extracted from the Emotional Prosody Speech and Transcripts (https://catalog.ldc.upenn.edu/LDC2002S28). In the speech files folder, there are 2,324 WAV files, and all files are named with the format "speaker_session_emotion_start-time_content.wav". These files are from 7 speakers (cc, cl, gg, jg, mf, mk, mm), labeled with 15 emotions (anxiety, boredom, cold-anger, contempt, despair, disgust, elation, happy, hot-anger, interest, neutral, panic, pride, sadness, shame).

What to submit

Code:

- Feature extraction
- Classification

Data:

- You don't have to submit any data, but please make sure that all features used in the experiments can be reproduced by running the code.

Report:

- 3 sections, described below: (1) feature analysis, (2) classification experiments, (3) error analysis

README:

- Documentation of your code

Note: This assignment will be graded based on your report. However, we will check the code and ensure that it is consistent with your report. Make sure that your report is easily reproducible from your code -- include any necessary instructions to run your code in your README.

1. Feature Analysis (40 points)

Extract six features from each speech segment:

- The min, max, mean of pitch
- The min, max, mean of intensity

Resources: Praat
Parselmouth (a Python library for Praat)
Note: For Praat-based pitch extraction, please set the pitch range as 75~600 Hz, and use autocorrelation as the analysis method. Please use only the left channel (channel 1) for analysis in this section.

Since each speaker naturally has a different pitch range and other voice qualities like intensity, you need to normalize the features accordingly by speaker. There are at least two ways you may want to normalize: Z-score normalization over the individual speaker is one common method. Another method you may want to try is normalizing by means of the individual speaker's neutral utterances; just subtract the mean of the feature value for the neutral utterances from the feature value of each of the utterances.

You need to turn in plots of the mean and standard deviation of each feature for all of the 15 emotion classes. Please also specify for each plot whether it was created a) without normalization; b) with normalization (and if so which type of normalization you used). Specifically, create 2 plots for each feature, one without normalization, and one with normalization. In each plot, visualize the mean and standard deviation values (over all speech files of each emotion) for all emotions. You can use graphs with error bars to visualize mean and standard deviation, as illustrated in the examples in *Figure 1*. This will result in (6 x 2 = 12) plots. Then tell us what you learn from these plots. Please report and discuss at least 5 interesting observations.

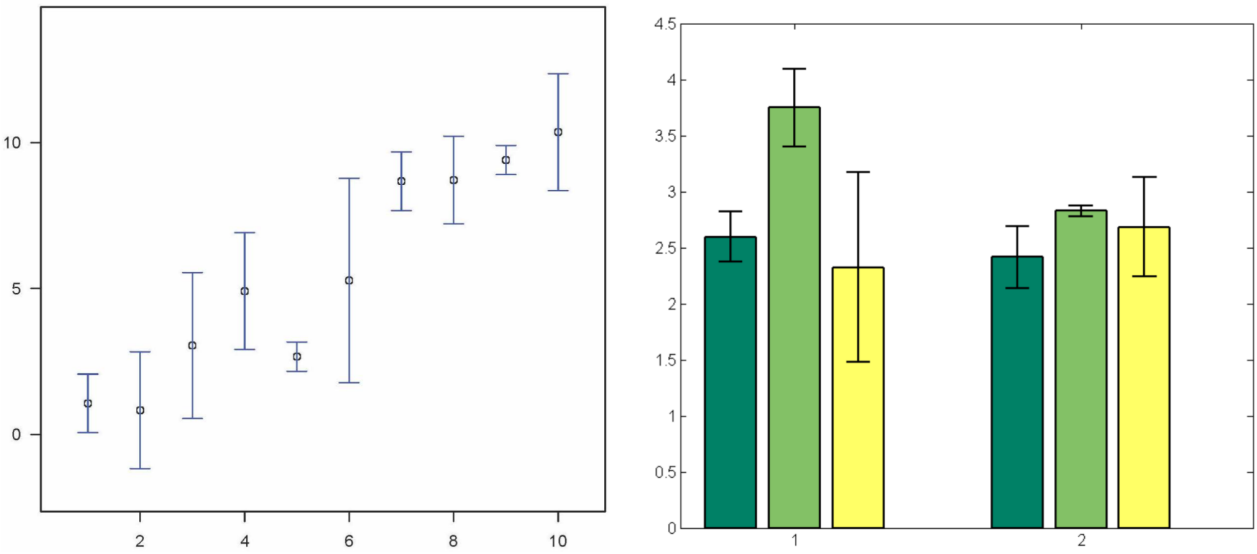

Figure 1: Examples of error bars

2. Classification Experiments (30 points)

Extract a set of acoustic-prosodic features using the openSMILE toolkit. Normalize your extracted features and use leave-one-speaker-out cross validation to predict the emotion categories. Leave-one-speaker-out cross validation means, for each speaker S, train on all other six other speakers combined and test on S. Then, average the results (precision, recall, and F1) for each emotion as your final overall result. Also report your average score over all emotions. Submit results of the 7 experiments (screenshots or cut-and-paste is fine) and the final average results.

Resources:
openSMILE
Scikit-learn
Note: Please check the documentation page for detailed installation instructions:
Here is a quick guide:

```
git clone https://github.com/audeering/opensmile.git
cd opensmile/
bash build.sh
```

Feature extraction: under `opensmile` directory, run

```
./build/progsrc/smilextract/SMILExtract -C config_path -I input_path -O
output_path
```

We recommend using the configuration file provided in the toolkit distribution to extract “The INTERSPEECH 2009 Emotion Challenge feature set” as a start. The config file can be found at

```
./config/is09-13/IS09_emotion.conf
```

You can train a multiclass classifier and use the classification report function in sklearn to generate the results. Regarding the classifier, you can use either a traditional machine learning model, such as random forest and SVM, or a neural network model. Report the type and structure of the model you use. Please avoid excessive tuning of the hyperparameters of the classifier you use, since you want to avoid overfitting the dataset.

You will be able to get the full credit for accuracy (10 points) if your model reaches aggregated average accuracy (over all emotions and speakers) of 0.14 or higher, or aggregate average F1 score of 0.13 or higher. You will receive (up to 3 points) bonus credit if your model reaches aggregate average accuracy of 0.24 or higher, or aggregate average F1 score of 0.23 or higher.

3. Error analysis (30 points)

Analyze the errors made by your best performing leave-one-speaker-out experiment, i.e. the best results you got for one of the 7 speakers.

Which class(es) were easiest to predict? Why do you think they were easy? Which were the most difficult? Why do you think they were difficult?

Based on this analysis, what ideas do you have to further improve your classifier?