



Audio & Speech

Week4: Baby Sounds Challenge

Prof. Chi-Chun Lee, Yi-Wen Liu

TAs: 楊晶宇、林蔭澤、鄭語芳、張繭云

EE3662: Digital Signal Processing Lab

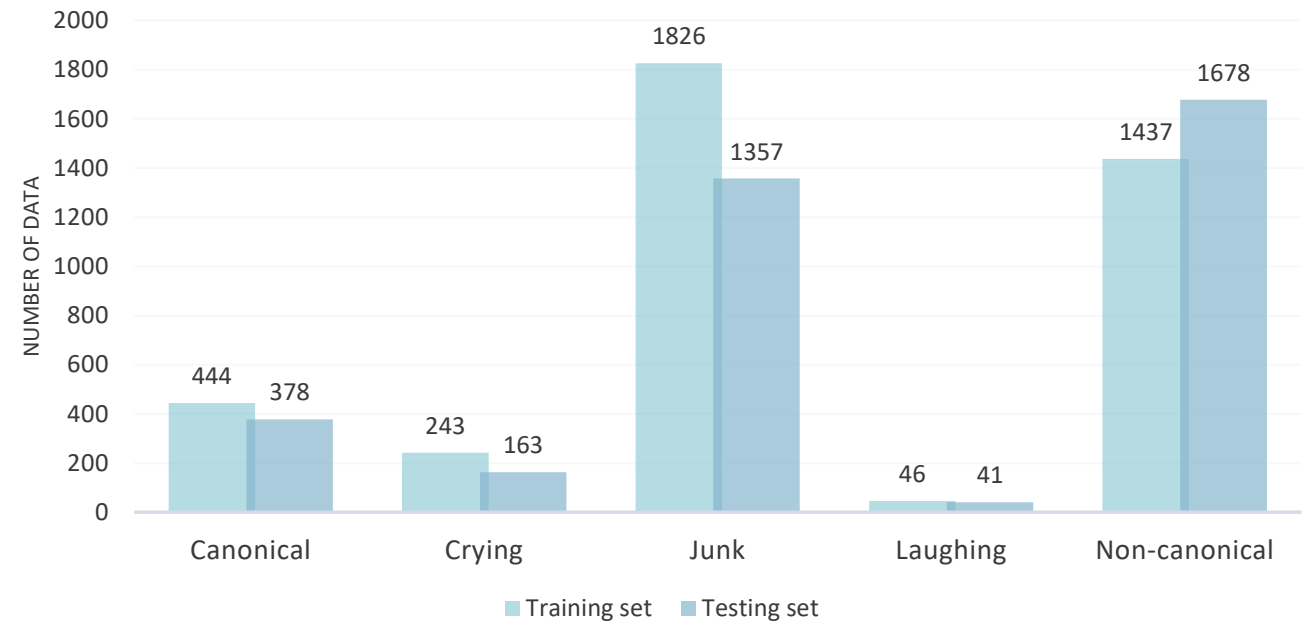
#Lab4– Oct. 17, 2022



Dataset Description

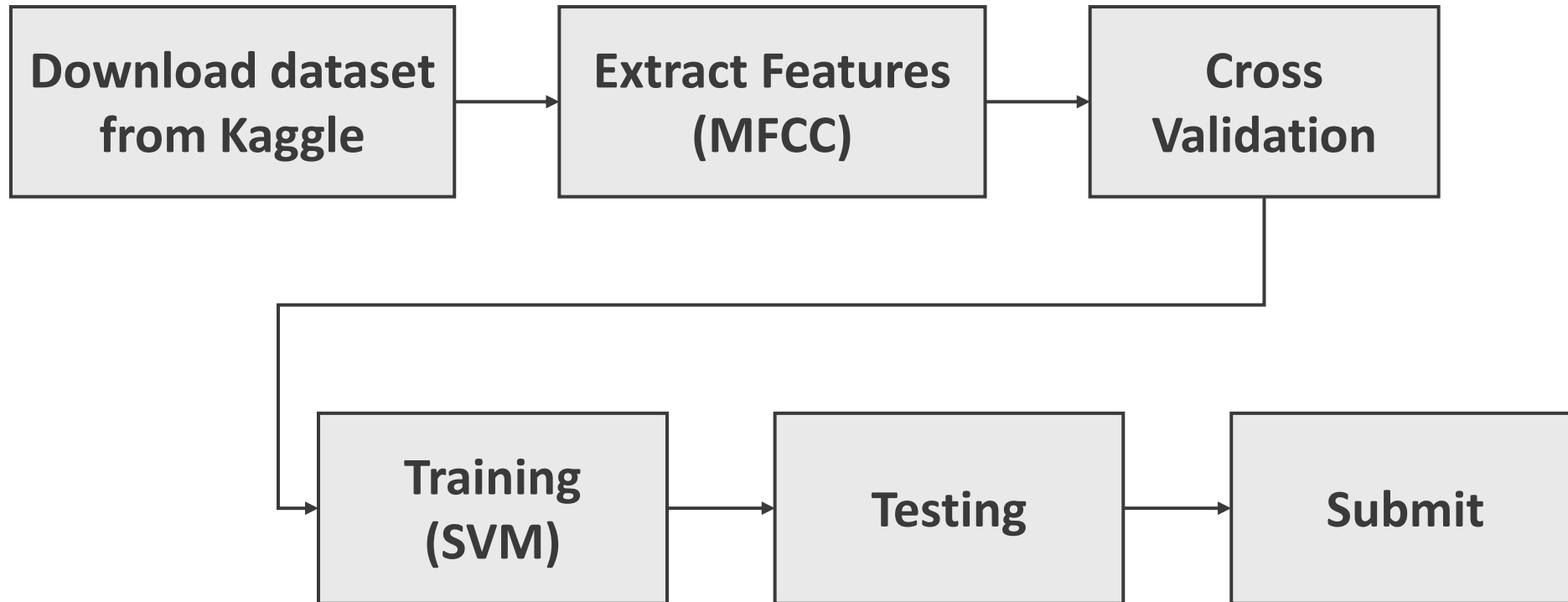


- **Baby Sounds Challenge**
 - The INTERSPEECH 2019 Computational Paralinguistics Challenge
- **Datasets**
 - Training set: 3996 files
 - Testing set: 3617 files
- **Download Datasets**
 - Kaggle → Data





Flow Chart





Kaggle



- Join this link and create your own account:
 - <https://www.kaggle.com/t/5109c86420ca6360dfb1b2e891fc3bc7>
- Create your Team name: using your student ID

Community Prediction Competition

2022DSPLab : Detecting Baby Sounds

Competition for NTHU 11110 EE366200 DSP Lab

1 teams · 10 days to go

Overview Data Code Discussion Leaderboard Rules **Team** Submissions **Submit Predictions** ...

Overview

Description

Evaluation

Competition for EE3662 11110 DSP Lab Audio and Speech Part

In this competition, you are asked to predict the most possible type of baby sounds from the audio clips. The training data are 3996 audio clips with labels. You need to predict the labels for the dev data.

Your goal is basically using MFCC and SVM to train a model to predict the types of baby sounds. The method you used is not limited.

You need to describe your method step by step in the report.

Launch a day ago

Close 10 days

Teams 10 days to go

Overview Data Code Discussion Leaderboard Rules **Team** Submissions **Submit Predictions** ...

Your Team

Everyone that competes in a Competition does so as a team - even if you're competing by yourself. [Learn more.](#)

General

TEAM NAME

TA_test

This name will appear on your team's leaderboard position.






Kaggle (Download Datasets)



- Click *Data*

Baby_Data (2 directories, 1 files) ↓ 🔍 >

| | | |
|--|--|--|
|  wav_dev 3617 files |  wav_train 3996 files |  label_raw_train.csv 99.76 kB |
|--|--|--|

Data Explorer
87.42 MB

- ▾ Baby_Data
 - wav_dev
 - wav_train
 - ▮ label_raw_train.csv
 - ▮ upload_sample.csv

Summary

- 7615 files
- 4 columns

↓ **Download All**



Kaggle (Upload results)



- Click *submit predictions*
- Upload your results.csv
 - Results format:

| | A | B |
|----|----------------|---------------|
| 1 | file_name | Prediction |
| 2 | devel_0001.wav | Junk |
| 3 | devel_0002.wav | Junk |
| 4 | devel_0003.wav | Junk |
| 5 | devel_0004.wav | Non-canonical |
| 6 | devel_0005.wav | Junk |
| 7 | devel_0006.wav | Non-canonical |
| 8 | devel_0007.wav | Junk |
| 9 | devel_0008.wav | Crying |
| 10 | devel_0009.wav | Junk |

You can refer to upload_sample.csv


[Overview](#) [Data](#) [Notebooks](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [Host](#) [My Submissions](#) [Submit Predictions](#)

```
> kaggle competitions submit -c competition-for-ee3662-dsp-lab-audio-and-speech -f submission.csv -m "Message"
```

Make a submission for [Woan-Shiuan Chien](#)

You have 5 submissions remaining today. This resets 11 hours from now (00:00 UTC).













Step 1
Upload submission file



File Format
Your submission should be in CSV format. You can upload this in a zip/gz/rar/7z archive, if you prefer.

Number of Predictions
We expect the solution file to have 3617 prediction rows. This file should have a header row. Please see sample submission file on the [data page](#).

Step 2
Describe submission

Briefly describe your submission



Lab12 Grading








Kaggle Challenge Results (40%) + Report (60%)



Kaggle Challenge Grading



- **Baseline:** Using MFCC and SVM → **63.23%**

| # | Team | Members | Score | Entries | Last | Code | Join |
|---|--|---|---------|---------|------|------|------|
|  | score_90.csv | | 0.69809 | | | | |
| 1 | TA_test |  | 0.69809 | 2 | 1d | | |
|  | Your Best Entry! Your submission scored 0.61736, which is not an improvement of your previous score. Keep trying! | | | | | | |
|  | score_80.csv | | 0.64445 | | | | |
|  | score_70.csv | | 0.61736 | | | | |



Challenge Hints



- Kaggle has maximum daily submissions (5 times one day), so you can use SVM and Cross Validation in Lab3 to verify the training data
- Evaluation is based on **Mean F1-Score**
- Try different parameters (kernel, c, degree) to tune your model
- Try different features, statistics functions or classifiers to achieve higher performance

The evaluation metric for this competition is **Mean F1-Score**. The F1 score, commonly used in information retrieval, measures accuracy using the statistics precision p and recall r.

Precision is the ratio of true positives tp to all predicted positives tp + fp. Recall is the ratio of true positives tp to all actual positives tp + fn. The F1 score is given by:

$$[F1 = 2 \frac{p \cdot r}{p + r} \text{ where } p = \frac{tp}{tp + fp}, r = \frac{tp}{tp + fn}]$$

The F1 metric weights recall and precision equally, and a good retrieval algorithm will maximize both precision and recall simultaneously. Thus, moderately good performance on both will be favored over extremely good performance on one and poor performance on the other.

| | | 真實狀況 | |
|------|--------|-----------------------|----------------------|
| | | 事實為真 | 事實為假 |
| 預測狀況 | 預測為 陽性 | tp | fp (Type I error) |
| | 預測為 陰性 | fn (Type II error) | tn |



Challenge Rules



- Don't create another account to get more submission chances!
- Don't submit results from your classmates. We will run your code and compare the result with your score on leaderboard.

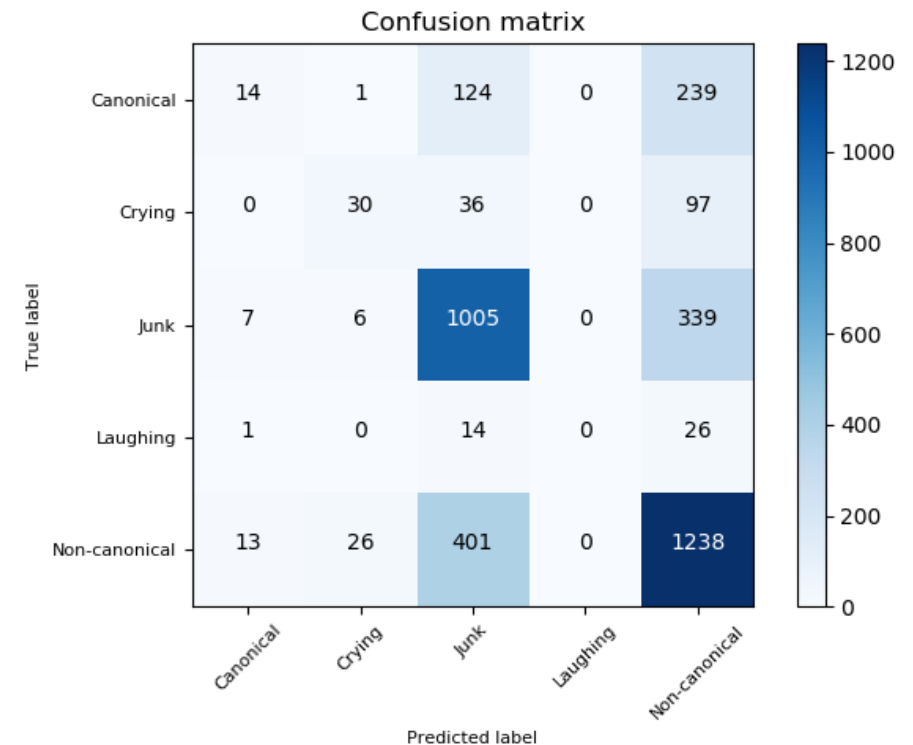


Report



- Try to explain your accuracies from your results table
- Using your best results to plot the confusion matrix (lab11)

| | Baby Sounds (%) | | | | | |
|--------------|-----------------|--------|-------|----------|---------------|-------|
| | Canonical | Crying | Junk | Laughing | Non-canonical | UAR |
| SVM (rbf) | 3.7 | 18.4 | 74.06 | 0 | 73.78 | 63.23 |
| SVM (linear) | | | | | | |
| SVM (poly) | | | | | | |
| ... | | | | | | |
| ... | | | | | | |





Report



- (Bonus) Analyses

1. Try to explain why some categories perform worse? The data distribution seems quite imbalanced, try to solve the problems and compare the experimental results.
2. What other problems did you meet? What is your solution? Any interesting findings?