Face and speech classification

SUR - Machine Learning and Recognition

Samuel Kuchta xkucht11
Marián Zimmermann xzimme03

Obsah

1 Solutions

1.1 Audio-Based Speaker Recognition Systems

1.1.1 Core Components

The system implements two distinct models for speaker recognition:

- **per class Gaussian Model (Misleadingly called GMM in code)**:

$$p(x|\lambda_s) = \mathcal{N}(x|\mu_s, \Sigma_s) \tag{1}$$

- **KMeans with Speaker Mapping**:

$$\hat{y} = \underset{s}{\text{argmax}} \sum_{i=1}^{K} I(c_i = s) \tag{2}$$

1.1.2 Feature Extraction Pipeline

- **Signal Processing**:
  - MFCC extraction (13 coefficients)
  - Delta (Δ) and double-delta ($\Delta\Delta$) coefficients
- **Augmentation**:
  - Gaussian noise: $\tilde{x} = x + \epsilon,\ \epsilon \sim \mathcal{N}(0, 0.005)$
  - Random gain: $\tilde{x} = x \cdot 10^{(g/20)},\ g \sim U(-10, 10)$
- **Temporal Aggregation**:
  - Mean & standard deviation
  - 25th/75th percentiles

1.1.3 Performance Metrics

Metric	**Single-Gaussian**	**KMeans**
Accuracy on dev	45.16% (28/62)	22.58% (14/62)

1.2 CNN for images

First step was creating simple CNN with 4 layers. 2 convolutionals for feature extraction and 2 fully connected for classification with RelU as activation function and cross-entropy as loss function. It was trained and evaluated on given data, with 20 epochs, as the loss was not decreasing much further. The accuracy on validation data was 40%. Low accuracy was probably cause due to small amount of data.

This problem was adressed by adding cross-validation. Althought due to high amount of classes and small amount of data it proved not to be effective, so it was not used.

To further improve the generalization of the model, data augmentation was added. Different augmentations were tested:

- Gaussian noise
- Brightness
- Rotation

Each image in the given datasets was augmented by these augmentations. After observing the loss and accuracy rotating augmentation was removed as it was greatly reducing accuracy of the model, given the fact that the image had to be either resized or parts of the image had to be completely cut. So it was proceeded only with Gaussian noise and Brightness, which was the model able to learn.

Final step was adding PCA and LDA, which also turned out to be a problem, as it lead to bad generalization, once again, the high amount of classes might be the problem.

1.2.1 Improvement suggestions

It would be possible to increase accuracy by creating a pipeline, where at first would be detected the gender of the target.

2 Installation and run

Before installing there should be data prepared in following directories relative to root of unpacked zip:

- `/train` - target training data
- `/dev` - non target training data
- `/eval` - evaluation set

Additionaly, due to size of the models, they need to be downloaded from cloud and put into root directory https://nextcloud.fit.vutbr.cz/s/eEEtBce3GwxKnw2.

To install and reproduce results:

- Install libraries with `pip install -r requirements.txt`
- Run face CNN prediction using `python face_detection.py --eval`

2.1 Run audio models

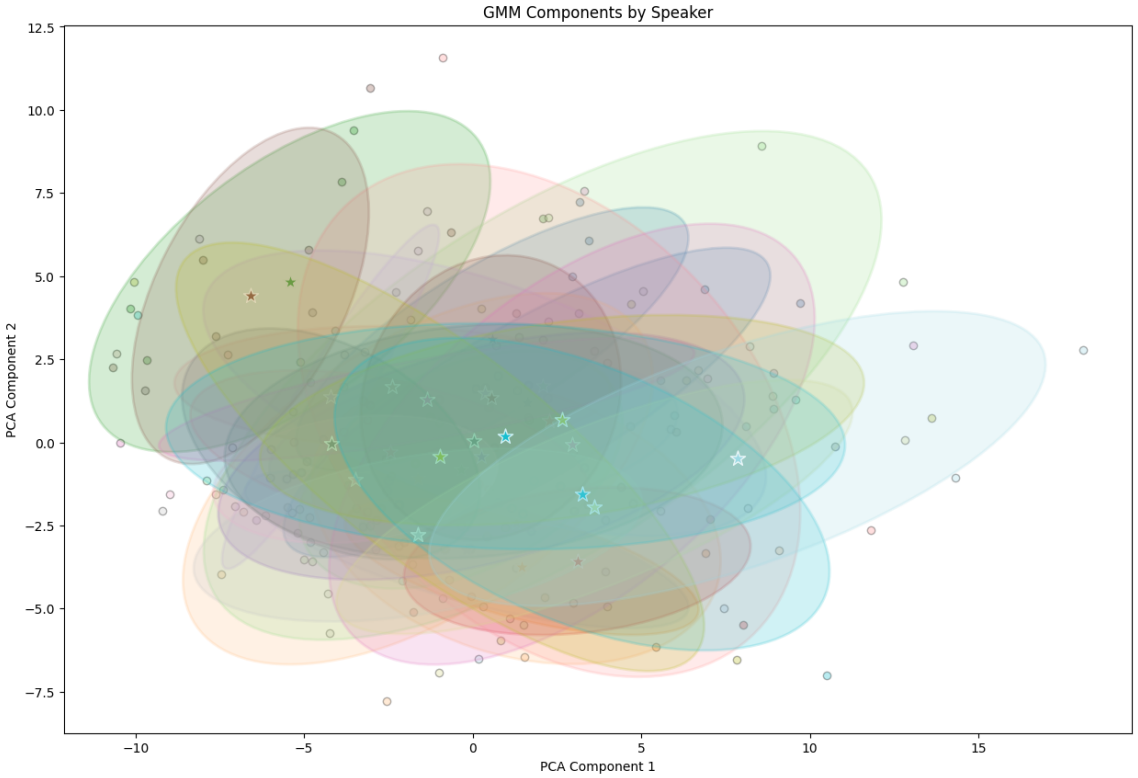
```
$ python audio.py --train_folder <folder>
                  --test_folder <folder>
                  --audio_gmm / --audio_kmeans
                  [--augment]
                  [--train]

# examples:
--train_folder train --test_folder dev --audio_gmm --train --augment
--train_folder train --test_folder eval --audio_kmeans > kmeans.txt
```

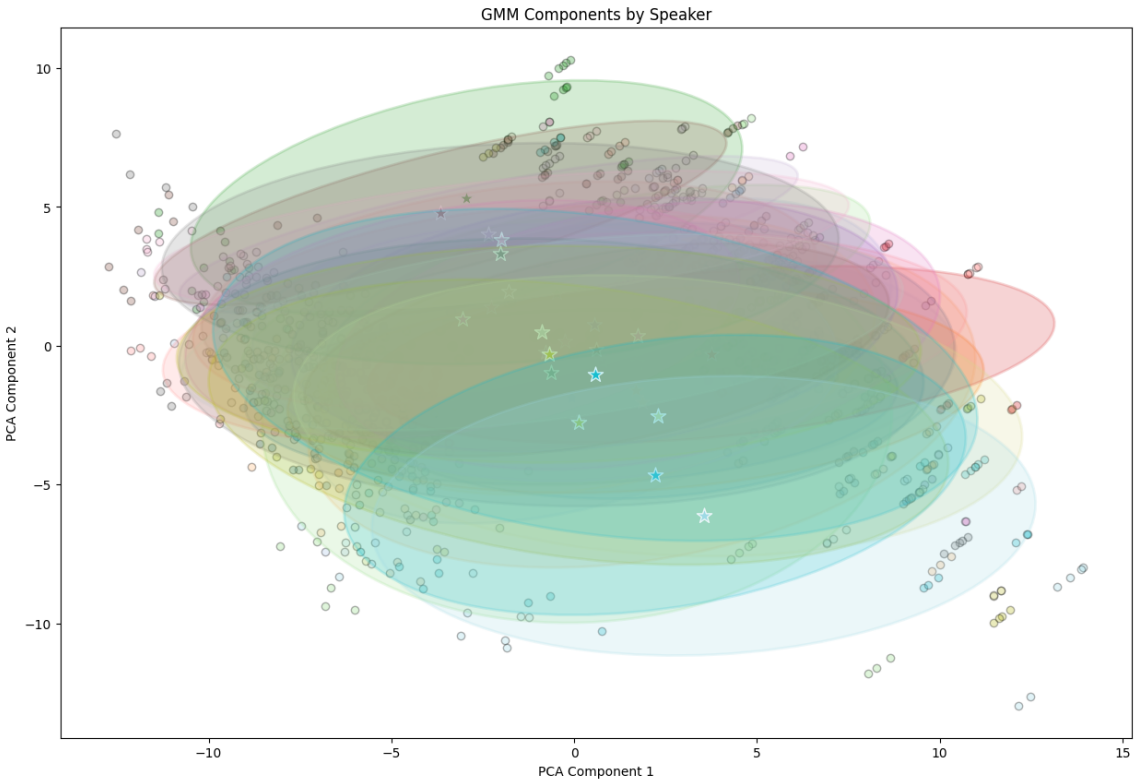
2.2 CNN

To train and save the model you can run `python face_detection.py -train` in `src` directory. The model will be saved in `src` directory. To augment the data you can run `python augmentation.py input_file output_file -noise -brightness -rotate` and choose type of augmentation.

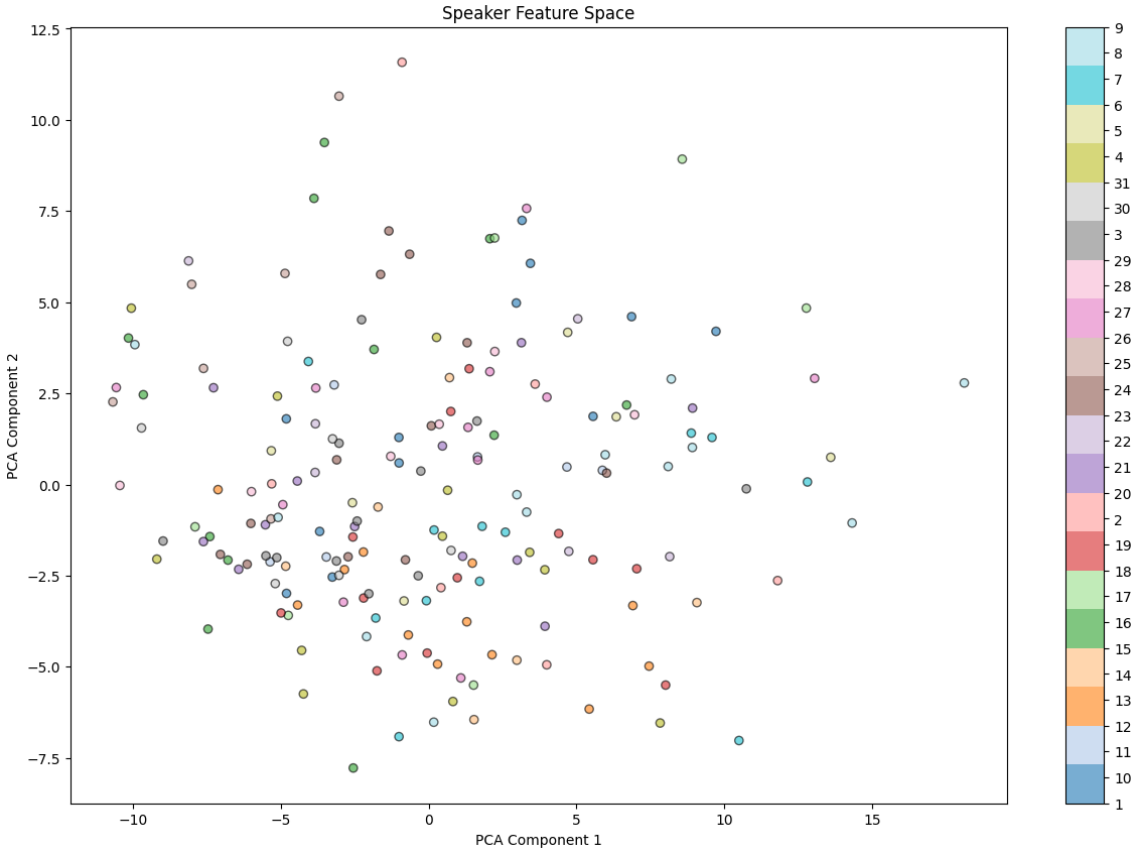
3 Figures



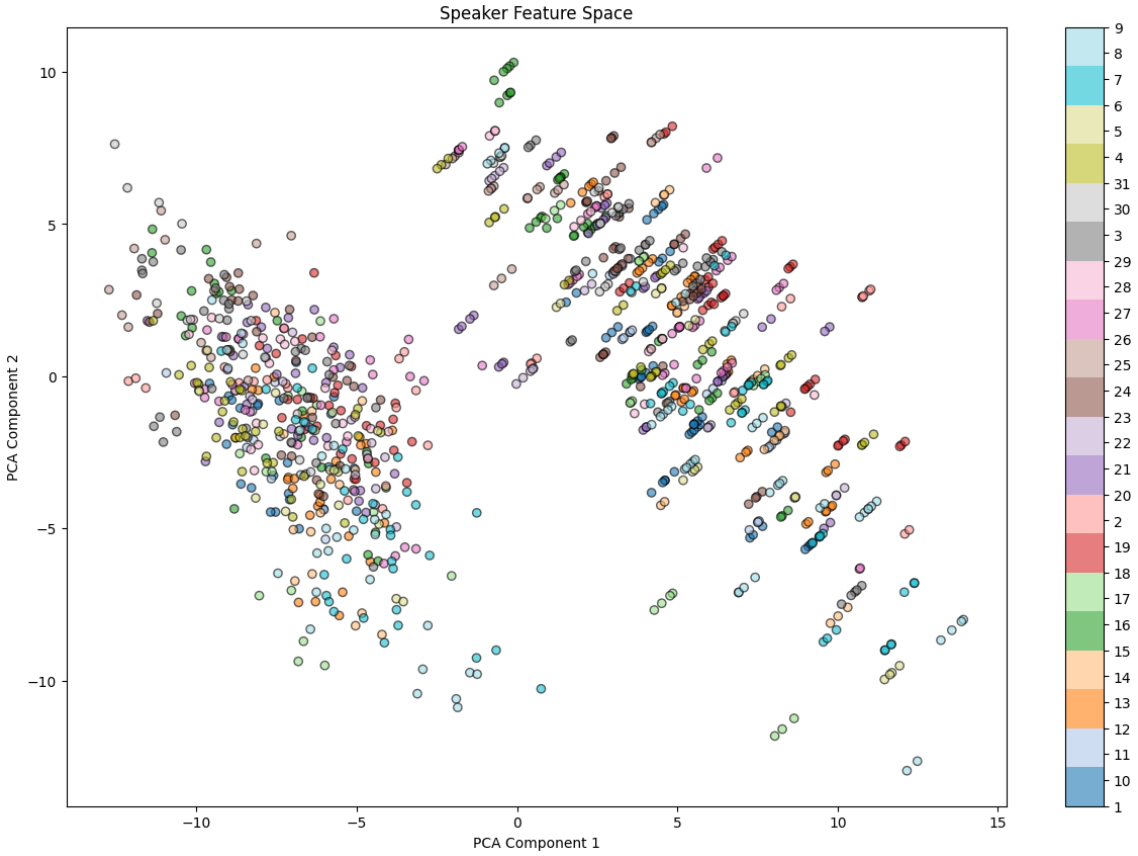
Obrázek 1: GMM components.



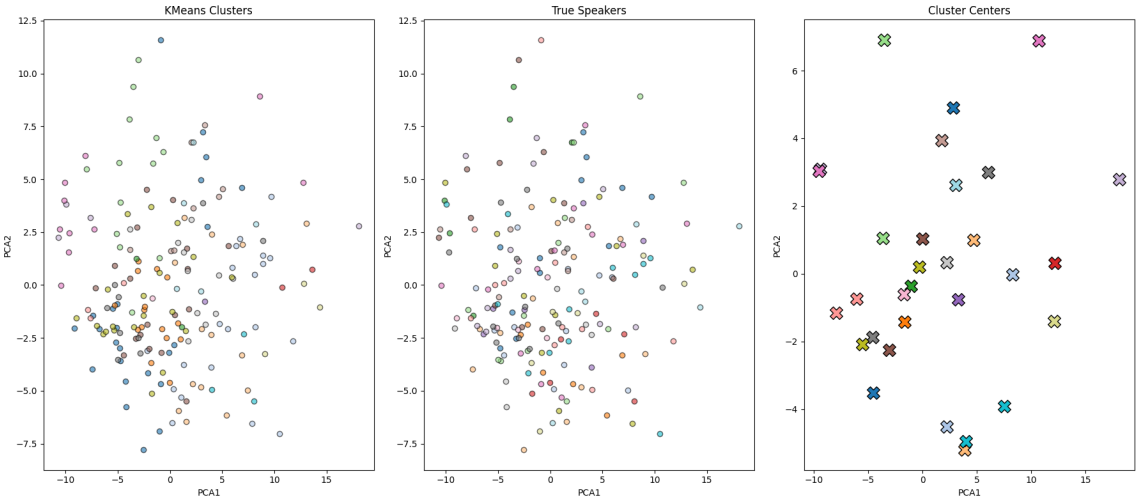
Obrázek 2: GMM components augmented.



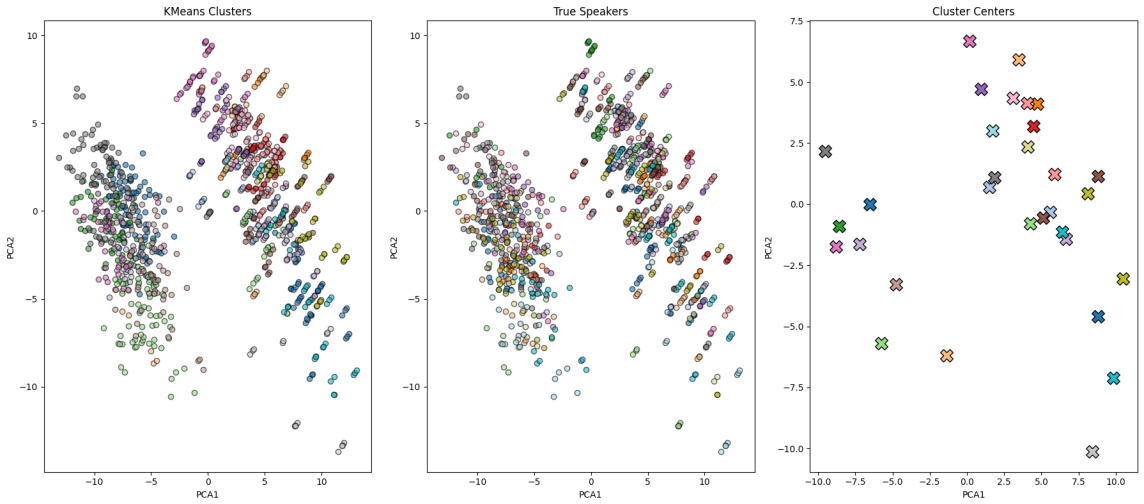
Obrázek 3: GMM features.



Obrázek 4: GMM features augmented.



Obrázek 5: K-Means cluster analysis.



Obrázek 6: K-Means cluster analysis augmented.