Lab 2

PHONEME dataset
Classification with MAP criterion
PCA and MDA feature selection

Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Introduction. Phoneme dataset

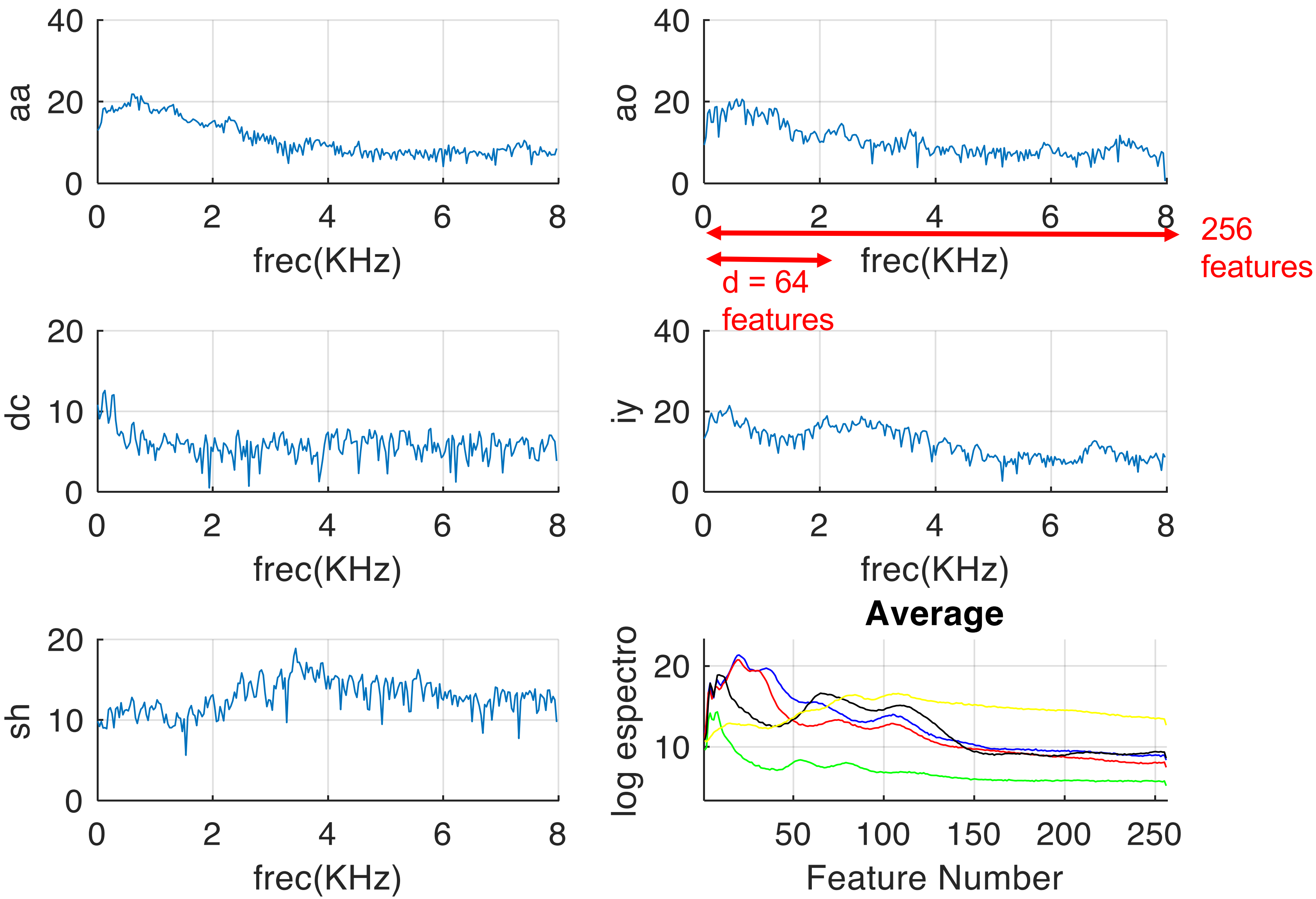
Objectives:

- Work with a real dataset
- Split the dataset into training and test subsets
- Dimensionality reduction (feature selection) using PCA and MDA. Application to sinthetic and real datasets

PHONEME dataset:

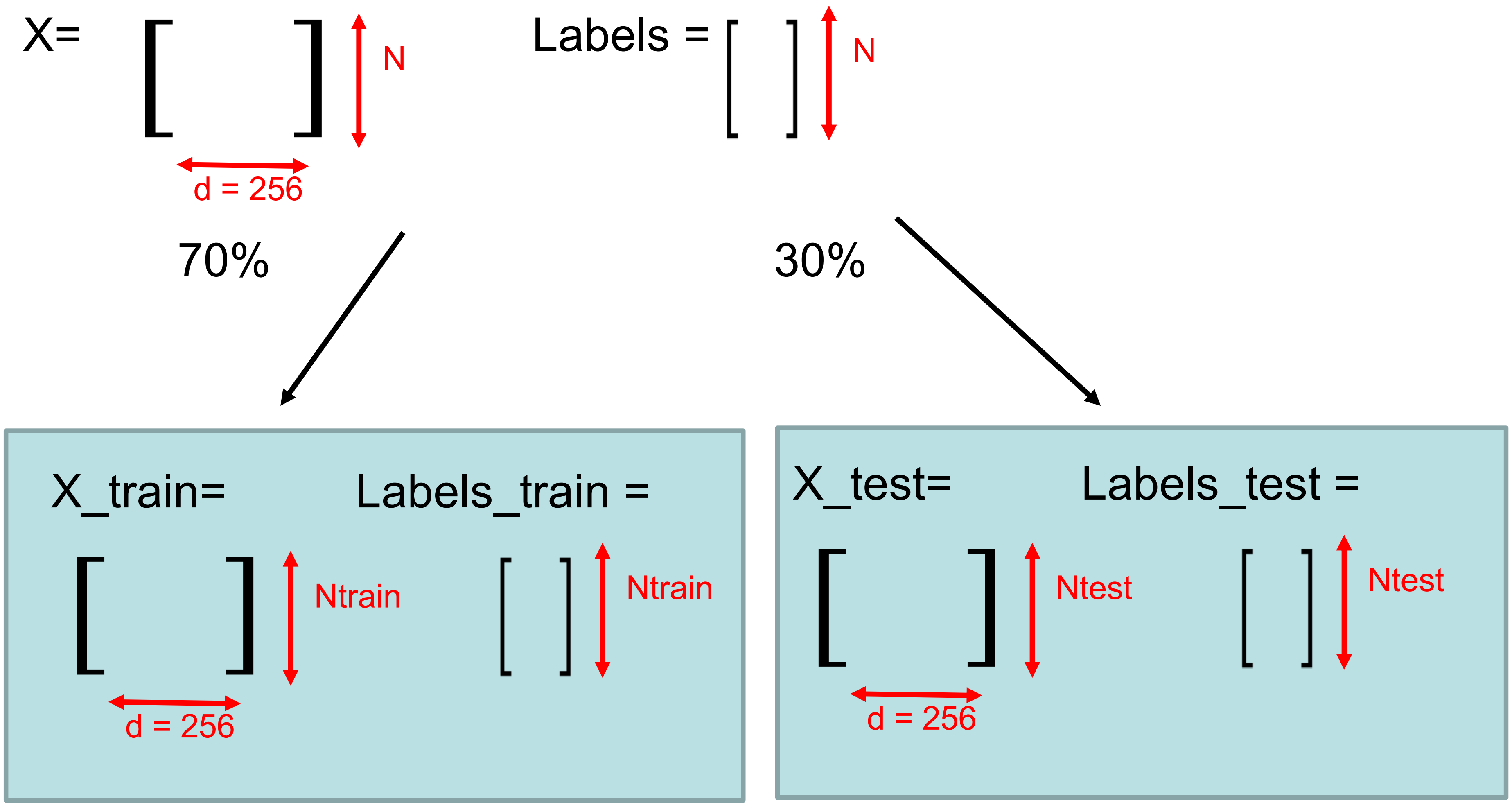
- Each vector has been obtained computing $log\left(\left|TF\left(x(n)\right)\right|^2\right)$ where the sequence $x(n)$ corresponds to part of a recording of a phoneme at a sampling rate of 16 kHz.
- Vectors correspond to 5 posible phonemes or classes:
  'aa' (695) 'ao'(1022) 'dcl'(757) 'iy'(1163) 'sh'(872).
- For each vector we initially have 256 features, corresponding to the spectrum between 0 and 8 kHz.
- In the first part of Lab2 we will work just fith the first 64 samples (frequencies 0 to 2 kHz).

Example: one vector per class



Dataset partition into training and test sets

Generated variables:



Classifier design

Design of a linear (LC) and a quadratic (QC) clasifier

- Dataset with d=256 (or 64) features
- Reduced datset using manual selection of 2 features
- Reduced dataset (dimensión d') using PCA and MDA

Dimensionality reduction

Objective:

- Reduce the number of features (assuming column vectors) :

$$\mathbf{x}_k \ (d \text{ características}) \quad \Rightarrow \quad \mathbf{z}_k = \underset{d'\times d}{\mathbf{W}^T} \mathbf{x}_k \ (d' \text{ características})$$

  - **Be careful !!! In the lab we will work with raw-vectors, so we need to compute the right product of W**

- **The dimensionality reduction helps to**
  - Simplify the classifier structure
  - Reduce the computational cost
  - Remove redundant information

- **Take into account…**
  - The reduction matrix **W** must be created using the training dataset

Scatter matrix

$$\mathbf{m}_i = \frac{1}{N_i}\sum_{\mathbf{x}\in D_i}\mathbf{x}$$ *Mean of samples from class i*

$$\mathbf{m} = \frac{1}{N}\sum_{\mathbf{x}\in\{D_1,\dots,D_c\}}\mathbf{x} = \frac{1}{N}\sum_{i=1}^{c} N_i\mathbf{m}_i$$ *Mean of all samples*

$$\mathbf{S}_T = \sum_{\mathbf{x}\in\{D_1,\dots,D_c\}}(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T$$ *Total data dispersion*

$$\mathbf{S}_T = \underbrace{\sum_{i=1}^{c}\sum_{\mathbf{x}\in D_i}(\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^T}_{\mathbf{S}_C} + \underbrace{\sum_{i=1}^{c}\sum_{\mathbf{x}\in D_i}(\mathbf{m}_i-\mathbf{m})(\mathbf{m}_i-\mathbf{m})^T}_{\mathbf{S}_B}$$

Sum of intra-class scatter matrices

Inter-class scatter matrix

PCA (Principal Component Analysis)

Objective:

- Maximize: $trace\left(\mathbf{W}^T \mathbf{S}_T \mathbf{W}\right)$
- Constraints: $\mathbf{w}_i^T \mathbf{w}_i = E$

Solution (Matlab function pca.m):

- Columns of **W**: eigenvalues associated to the largest eigenvalues d' of $\mathbf{S}_T$:

$$\mathbf{S}_T \mathbf{w}_i = \lambda_i \mathbf{w}_i$$

Problem:

- PCA minimizes the approximation squared error but it does not guarantee the separability of the clases

PCA (Principal Component Analysis)

PCA Transformation

- Obtention of the PCA matrix from the training dataset

W_pca = $[\mathbf{w}_1, \mathbf{w}_2, .., \mathbf{w}_d]$ (pca.m)

- Projection to a smaller dimensión d' using **W** _red= $[\mathbf{w}_1, \mathbf{w}_2, .., \mathbf{w}_{d'}]$

W _red = **W_pca** (:,1..*d*')

- Transformation of training and test datsets:

```
X_train_pca=X_train*W_red;
X_test_pca=X_test*W_red;
```

- Representation of training and test dataset for linear and quadratic classifiers, varying the dimensionality of the feature space, from d'=1 to d'=d.

MDA (Multiple Discriminant Analysis)

Objective:

- Maximize intra-class separability while minimizing the inter-class scatter
- We measure the separability and scatter using the ellipsoid volumes, assuming data Gaussianity

Formulation:

- Maximization: $$\mathbf{W} = \arg\max_{\mathbf{W}} \frac{\left|\mathbf{W}^T \mathbf{S}_B \mathbf{W}\right|}{\left|\mathbf{W}^T \mathbf{S}_C \mathbf{W}\right|}$$

Solution (Matlab function mda_clp.m):

- d' ≤ min(d,c-1) (c: number of classes)
- **W columns:** eigenvectors associated to the largest eigenvalues:

$$\mathbf{S}_B \mathbf{w}_j = \sigma_j \mathbf{S}_C \mathbf{w}_j \qquad \Rightarrow \qquad \mathbf{S}_C^{-1} \mathbf{S}_B \mathbf{w}_j = \sigma_j \mathbf{w}_j$$

Lab2

Part1:

- Use Phoneme dataset (c=5 clases, d=256 features)
- Train Lc and Qc classifiers using the first d=64 features
- Train Lc and Qc classifiers using d'=2 manually selected features

Part2:

- Use synthetic Gaussian datasets (c=3 clases, d=3 features) for different SNR values
- Train Lc and Qc classifiers using all the features
- Train Lc and Qc classifiers after dimensionality reduction using PCA and MDA

Part3:

- Use Phoneme dataset
- Train Lc and Qc classifiers using d' features seleted with PCA and MDA
- Show Lc and Qc training/test error curves for varying number of features selected with PCA and MDA