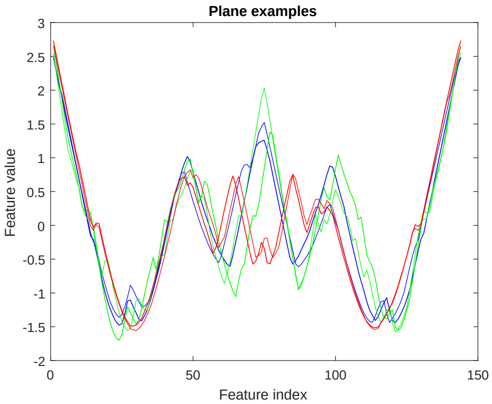
Prototypes and Matrix Relevance Learning in Complex Fourier Space

M. Straat, M. Kaden, M. Gay, T. Villmann, A. Lampe, U. Seiffert, M. Biehl, and F. Melchert

June 26, 2017

Overview

- A study of classification of time series.
- In Fourier space: Vectors in $\mathbb{C}^n$.
- Generalized Matrix Learning Vector Quantization (GMLVQ) on complex-valued data.
- Evaluation and interpretation of the Fourier-space classifiers.



Learning Vector Quantization (LVQ)

- Dataset of vectors $\boldsymbol{x}^m \in \mathbb{R}^N$, each carrying class label $\sigma^m \in \{1, 2, ..., C\}$
- Training: For each class σ, identify prototype(s) $\boldsymbol{w}^i \in \mathbb{R}^N$ in feature space that are typical representatives for that class.
- Aim: Classify novel vectors $\boldsymbol{x}^\mu$, assigning them to the class of the nearest prototype.

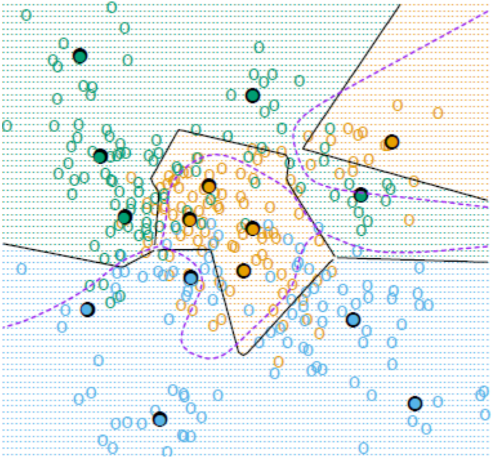


Figure: LVQ with 5 prototypes per class. Initialized with K-means on each class. *Black line*: Piece-wise linear decision boundary.

$d(\boldsymbol{x}, \boldsymbol{w}) = (\boldsymbol{x} - \boldsymbol{w})^T (\boldsymbol{x} - \boldsymbol{w})$, sq. Euclidean distance.

---

procedure LVQ
  for each training epoch **do**
    for each labeled vector $\{\boldsymbol{x}, \sigma\}$ **do**
      $\{\boldsymbol{w}^*, S^*\} \leftarrow \text{argmin}_i \{d(\boldsymbol{x}, \boldsymbol{w}^i)\}$
      $\boldsymbol{w}^* \leftarrow \boldsymbol{w}^* + \eta \Psi(S^*, \sigma)(\boldsymbol{x} - \boldsymbol{w}^*)$

$$\Psi(S, \sigma) = \begin{Bmatrix} +1, & \text{if } S = \sigma \\ -1, & \text{otherwise} \end{Bmatrix}$$

Classification of novel data point $\boldsymbol{x}^\mu$:

$$\text{Closest prototype } \{\boldsymbol{w}^*, S^*\} \leftarrow \underset{i}{\text{argmin}} \{d(\boldsymbol{x}^\mu, \boldsymbol{w}^i)\}$$

$$\text{Classify } \boldsymbol{x}^\mu \text{ in class } S^* : \{\boldsymbol{x}^\mu, \sigma^\mu = S^*\}$$

GMLVQ

- Learn feature relevance and adapt d accordingly.
  - Adaptive quadratic distance measure: $d_{\Omega}(\boldsymbol{x}, \boldsymbol{w}) = (\boldsymbol{x} - \boldsymbol{w})^T \boldsymbol{\Omega}^T \boldsymbol{\Omega}(\boldsymbol{x} - \boldsymbol{w})$.
- Update two prototypes upon presentation of $\{\boldsymbol{x}, \sigma\}$.
  - $\boldsymbol{w}^+$: Closest prototype of the same class as $\boldsymbol{x}$.
  - $\boldsymbol{w}^-$: Closest prototype of a different class than $\boldsymbol{x}$.

Cost one example $\boldsymbol{x}^m$

$$e^m = \frac{d_{\Omega}[\boldsymbol{w}^+] - d_{\Omega}[\boldsymbol{w}^-]}{d_{\Omega}[\boldsymbol{w}^+] + d_{\Omega}[\boldsymbol{w}^-]} \in [-1, 1] \,.$$

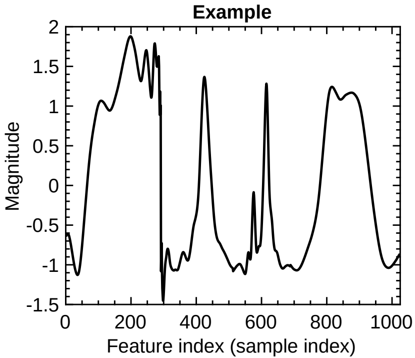
Learning is minimization of the cost with gradient descent:

$$\boldsymbol{w}^{\pm} \leftarrow \boldsymbol{w}^{\pm} - \eta_w \nabla_{\boldsymbol{w}^{\pm}} e^m$$
$$\boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega} - \eta_{\Omega} \nabla_{\boldsymbol{\Omega}} e^m$$

Time series

$$f(t) \to f(i\Delta T), i = 0, 1, ..., N-1$$

- Vectors $\boldsymbol{x} \in \mathbb{R}^N$.
- Temporal order of dimensions.



Training in coefficient space

Approximate $f(t) = \sum_{i=1}^{n} c_i g_i(t)$:

- Using Chebyshev basis.
- Using Fourier basis: $\boldsymbol{x} \in \mathbb{R}^N \rightarrow \boldsymbol{x_f} \in \mathbb{C}^n$.
- Prototypes $\boldsymbol{w}^i \in \mathbb{C}^n$ and relevances $\boldsymbol{\Lambda}$ Hermitian.

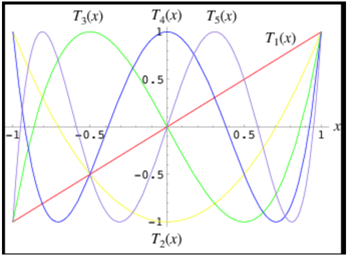


Figure: 5 Chebyshev basis functions

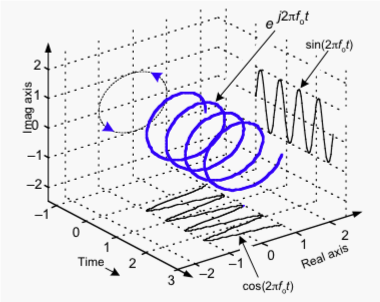


Figure: Fourier complex sinusoid

F. Melchert, U. Seiffert, and M. Biehl, Polynomial Approximation of Spectral Data in LVQ and Relevance Learning, in Workshop on New Challenges in Neural Computation 2015

Fourier: Time $\leftrightarrows$ Frequency

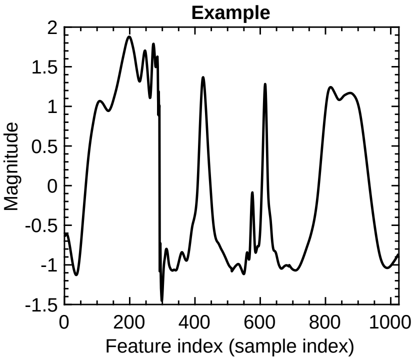
Matrix $\boldsymbol{F} \in \mathbb{C}^{n \times N}$ with rows $e^{-j2\pi kn/N}$, $k = 0, 1, 2, ..., N-1$.

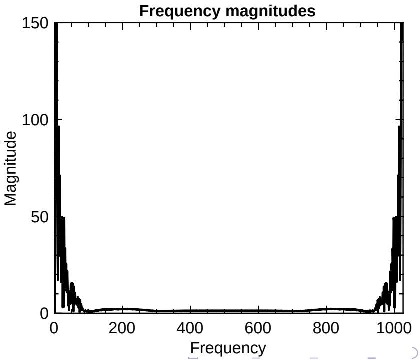
Forward (DFT):

$$\boldsymbol{x}_f = \boldsymbol{F}\boldsymbol{x} \in \mathbb{C}^n$$

Backward (iDFT):

$$\boldsymbol{x} = \frac{1}{N}\boldsymbol{F}^H \boldsymbol{x}_f \in \mathbb{R}^N$$





GMLVQ complex-valued data

Quadratic distance measure

$$d_{\mathbf{\Lambda}}[\boldsymbol{x}_f, \boldsymbol{w}_f] = (\boldsymbol{x}_f - \boldsymbol{w}_f)^H \mathbf{\Omega}^H \mathbf{\Omega} (\boldsymbol{x}_f - \boldsymbol{w}_f) \in \mathbb{R}_{\geq 0} \, .$$

Cost one example $\boldsymbol{x}_f^m$

$$e^m = \frac{d_{\mathbf{\Lambda}}[\boldsymbol{w}_f^+] - d_{\mathbf{\Lambda}}[\boldsymbol{w}_f^-]}{d_{\mathbf{\Lambda}}[\boldsymbol{w}_f^+] + d_{\mathbf{\Lambda}}[\boldsymbol{w}_f^-]} \in [-1, 1] \, .$$

Compute gradients w.r.t. $\boldsymbol{w}_f^+$, $\boldsymbol{w}_f^-$ and $\mathbf{\Omega}$ for learning:

$$\nabla_{\boldsymbol{w}_f^+} e^{\mu} = \frac{\partial e^{\mu}}{\partial d_{\mathbf{\Lambda}}^+} \frac{\partial d_{\mathbf{\Lambda}}}{\partial \boldsymbol{w}_f^+}$$

Wirtinger derivatives

- $f(z) : \mathbb{C} \to \mathbb{R}$.
- Operators $\frac{\partial}{\partial z} = \frac{1}{2}\left(\frac{\partial}{\partial x} - i\frac{\partial}{\partial y}\right)$ and $\frac{\partial}{\partial z^*} = \frac{1}{2}\left(\frac{\partial}{\partial x} + i\frac{\partial}{\partial y}\right)$
- $f(z) = z \cdot z^*$, then $\frac{\partial f}{\partial z} = z^*$ and $\frac{\partial f}{\partial z^*} = z$.

Wirtinger gradients

$\frac{\partial}{\partial \mathbf{z}} = \left(\frac{\partial}{\partial z_1} \quad , \ldots, \quad \frac{\partial}{\partial z_N}\right)^T$ and $\frac{\partial}{\partial \mathbf{z}^*} = \left(\frac{\partial}{\partial z_1^*} \quad , \ldots, \quad \frac{\partial}{\partial z_N^*}\right)^T$

Using the Wirtinger gradient:

$$\frac{\partial}{\partial \mathbf{z}^*}(\mathbf{z}^H \boldsymbol{A}\mathbf{z}) = \boldsymbol{A}\mathbf{z}$$

M. Gay, M. Kaden, M. Biehl, A. Lampe, and T. Villmann, "Complex variants of GLVQ based on Wirtinger's calculus"

Learning rules

Complex-valued GMLVQ (Wirtinger)

$$\nabla_{\boldsymbol{w}_f^*} d_{\boldsymbol{\Lambda}}[\boldsymbol{x}_f, \boldsymbol{w}_f] = -\boldsymbol{\Omega}^H \boldsymbol{\Omega}(\boldsymbol{x}_f - \boldsymbol{w}_f)\,,$$

$$\nabla_{\boldsymbol{\Omega}^*} d_{\boldsymbol{\Lambda}}[\boldsymbol{x}_f, \boldsymbol{w}_f] = \boldsymbol{\Omega}(\boldsymbol{x}_f - \boldsymbol{w}_f)(\boldsymbol{x}_f - \boldsymbol{w}_f)^H\,.$$

Relevance matrix $\boldsymbol{\Lambda} = \boldsymbol{\Omega}^H \boldsymbol{\Omega}$ is Hermitian.

Real-valued GMLVQ

$$\nabla_{\boldsymbol{w}} d_{\boldsymbol{\Lambda}}[\boldsymbol{x}, \boldsymbol{w}] = -2\boldsymbol{\Omega}^T \boldsymbol{\Omega}(\boldsymbol{x} - \boldsymbol{w})\,,$$

$$\nabla_{\boldsymbol{\Omega}} d_{\boldsymbol{\Lambda}}[\boldsymbol{x}, \boldsymbol{w}] = \boldsymbol{\Omega}(\boldsymbol{x} - \boldsymbol{w})(\boldsymbol{x} - \boldsymbol{w})^T\,.$$

Relevance matrix $\boldsymbol{\Lambda} = \boldsymbol{\Omega}^T \boldsymbol{\Omega}$ is symmetric (also Hermitian).

After each epoch, normalize $\boldsymbol{\Lambda}$ such that $\mathrm{tr}(\boldsymbol{\Lambda}) = 1$.

The testing scenarios

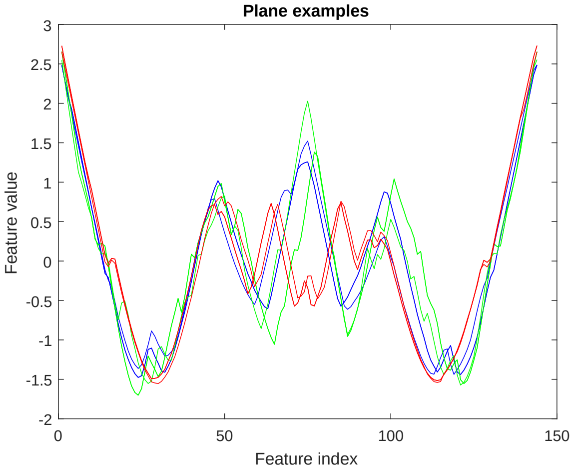
1. GMLVQ in original time domain on vectors $\boldsymbol{x} \in \mathbb{R}^N$.
2. GMLVQ (Wirtinger) in complex Fourier space on vectors $\boldsymbol{x}_f \in \mathbb{C}^n$ with $n = [6, 11, ..., 51]$.
3. GMLVQ in Fourier space on vectors $\boldsymbol{x}_f \in \mathbb{R}^{2n}$, real and imaginary concatenated.
4. GMLVQ on smoothed time domain vectors $\hat{\boldsymbol{x}} \in \mathbb{R}^N$.

Before training...

- All dimensions z-score transformed.
- One prototype per class. Initialization prototype class i: $\boldsymbol{w}^i \approx \text{mean}(\{(\boldsymbol{x}, y) | y == i\})$.
- $\Lambda = cI$.

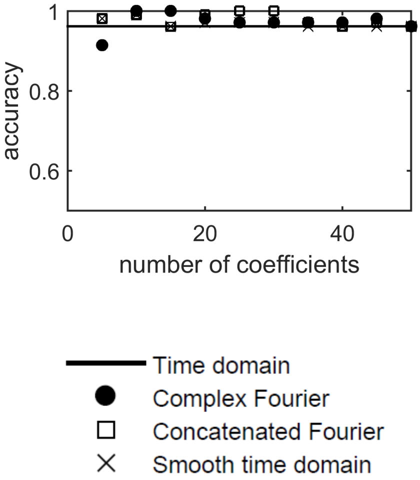
Plane dataset

- 210 labeled vectors $(\mathbf{x}, y) \in R^{144} \times \{1, 2, ..., 7\}$
- 105/105 train/val vectors.



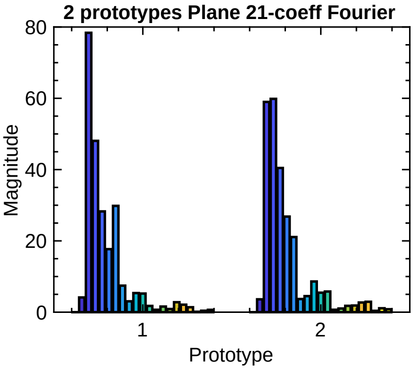
Plane - Classification performance

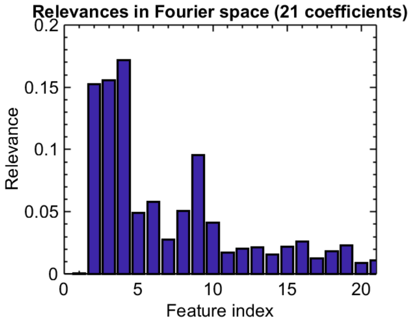
Accuracies of the 4 testing scenarios on validation set



Interpreting the classifier

- Prototypes $\boldsymbol{w}_f^i \in \mathbb{C}^n$
- Matrix $\boldsymbol{\Lambda}_f$ is Hermitian: $\boldsymbol{\Lambda}_f = \boldsymbol{\Lambda}_f^H$

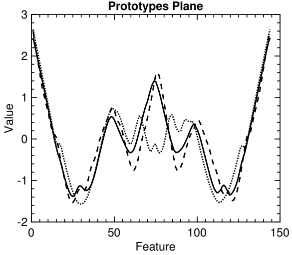


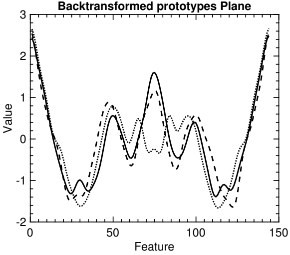


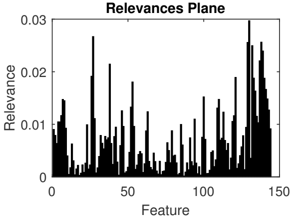
- Map prototypes to time domain with iDFT: $\boldsymbol{w}^i = \frac{1}{N}\boldsymbol{F}^H \boldsymbol{w}_f^i$.
- Relevance matrix to time domain:
  $d[\boldsymbol{x}_f, \boldsymbol{w}_f] = (\boldsymbol{x} - \boldsymbol{w})^H \boldsymbol{F}^H \boldsymbol{\Lambda}_f \boldsymbol{F} (\boldsymbol{x} - \boldsymbol{w})$.

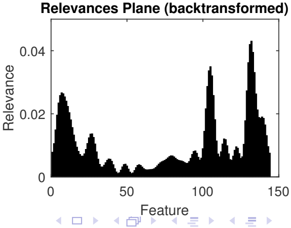
Plane - Prototypes and feature relevance

Time domain training vs. 21 coefficient Fourier space



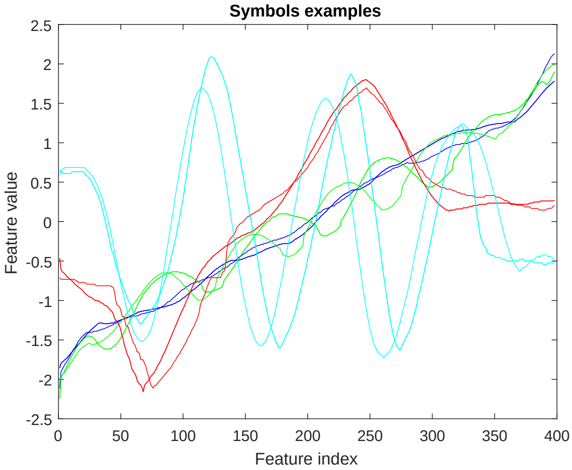




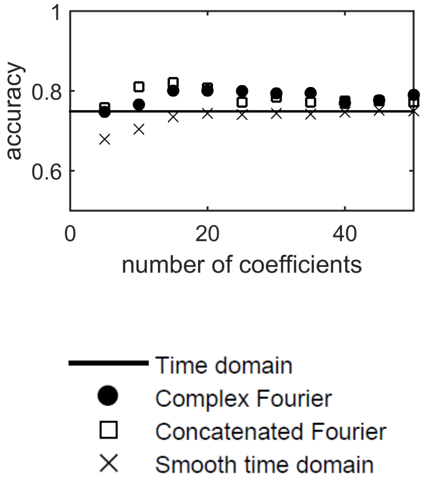


Symbols dataset

- 1020 feature vectors $(\mathbf{x}, y) \in \mathbb{R}^{398} \times \{1, 2, ..., 6\}$
- 25/995 train/validation vectors.

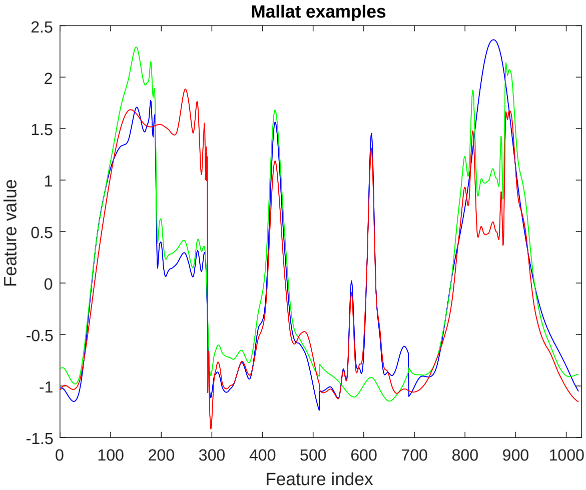


Accuracies of the 4 testing scenarios on validation set

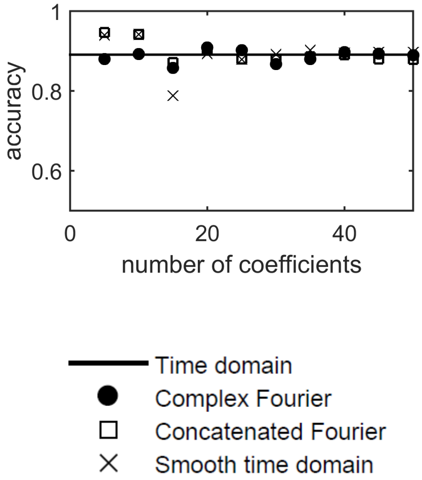


Mallat dataset

- 2400 feature vectors $(\mathbf{x}, y) \in \mathbb{R}^{1024} \times \{1, 2, ..., 8\}$
- 55/2345 train/validation vectors.

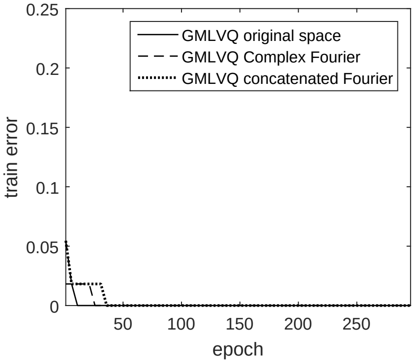


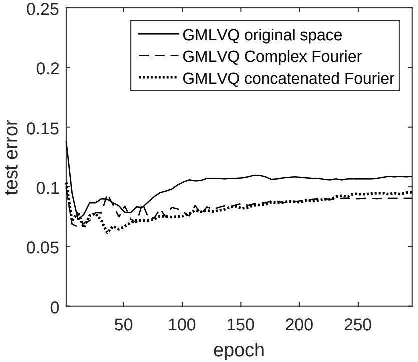
Accuracies of the 4 testing scenarios on validation set



Mallat - Classification error curves

Error development on the training and validation set





Discussion

Learning in complex Fourier-coefficient space...

- can be an effective method for classification of periodic functional data.
- can provide an efficient low-dimensional representation.
- has the potential to improve classification accuracy.

For future research: How to obtain close to optimal accuracy with the least number of adaptive parameters.