T-61-181 Biomedical Signal Processing

Sections 4.5.3 – 4.6.2

Learning basis vectors and weights for EEG signals

vibhor kumar

Finding optimal basis function to denoise

1. Karhunen-Loeve Expansion
2. Interprtaion as Linear , Time-Variant Filtering
3. Damped Sinusoids : Finding Optimal function

Adaptive Analysis Using Basis Functions

1. Instantaneous LMS algorithm
2. Block LMS algorithm

Karhunen-Loeve Expansion – optimal basis function

To design the basis function φ_k such that the signal part is efficiently represented witha small number of functions

Derivation:

Decompostion of signal x in to two sums representing signal and noise

$$\mathbf{x} = \sum_{k=1}^{K} w_k \varphi_k + \sum_{k=K+1}^{N} w_k \varphi_k = \hat{\mathbf{s}} + \hat{\mathbf{v}}.$$

Aim is to find set of φ_k such that resembles original signal $\mathbf{s}$ as closely as possible.

Can be done by minimising the noise power estimate:

$$\mathcal{E} = E\left[\hat{\mathbf{v}}^T \hat{\mathbf{v}}\right] = E\left[(\mathbf{x} - \hat{\mathbf{s}})^T (\mathbf{x} - \hat{\mathbf{s}})\right]$$

PROOF

considering that the actual signal is composed of $\mathbf{x} = \mathbf{s} + \mathbf{v}$ the noise power estimate becomes.

$$\mathcal{E} = E\left[(\mathbf{s} - \hat{\mathbf{s}})^T (\mathbf{s} - \hat{\mathbf{s}})\right] + 2E\left[(\mathbf{s} - \hat{\mathbf{s}})^T \mathbf{v}\right] + E\left[\mathbf{v}^T \mathbf{v}\right]$$

$E[\mathbf{v}\mathbf{v}^T]$ = $\mathbf{R}_v$ = $\sigma_v^2 \mathbf{I}$ as noise is zero-mean with correlation matrix $\sigma_v^2 \mathbf{I}$

$$\begin{aligned} E\left[(\mathbf{s} - \hat{\mathbf{s}})^T \mathbf{v}\right] &= 0 - E\left[\left(\sum_{k=1}^{K} w_k \varphi_k\right)^T \mathbf{v}\right] \\ &= -E\left[\left(\sum_{k=1}^{K} (\varphi_k^T \mathbf{x}) \varphi_k\right)^T \mathbf{v}\right] \\ &= -E\left[\left(\sum_{k=1}^{K} \varphi_k^T (\mathbf{s} + \mathbf{v}) \varphi_k\right)^T \mathbf{v}\right] \end{aligned}$$

as signal noise are assumed to be uncorrelated.

$$\begin{aligned} &= -\sum_{k=1}^{K} \mathrm{tr}\left(E\left[\mathbf{v}^T \varphi_k \varphi_k^T \mathbf{v}\right]\right) \\ &= -\frac{K}{N} E\left[\mathbf{v}^T \mathbf{v}\right] \\ &= -K \sigma_v^2. \end{aligned}$$

$$\mathcal{E} = E\left[\hat{\mathbf{v}}^T\hat{\mathbf{v}}\right] = E\left[(\mathbf{s}-\hat{\mathbf{s}})^T(\mathbf{s}-\hat{\mathbf{s}})\right] + (N-2K)\sigma_v^2.$$

Since noise power is not dependent on φ_k . The minimisation of $\mathcal{E}$ is equivalent to making $\hat{\mathbf{s}}$ resemble **s**

Optimal basis function and correlation matrix of data

We start frim the equation of noise power estimate again:

$$\mathcal{E} = E\left[\hat{\mathbf{v}}^T \hat{\mathbf{v}}\right] = E\left[\left(\sum_{k=K+1}^{N} w_k \varphi_k\right)^T \left(\sum_{l=K+1}^{N} w_l \varphi_l\right)\right] = \sum_{k=K+1}^{N} E\left[w_k^2\right]$$

$$= \sum_{k=K+1}^{N} \varphi_k^T \mathbf{R}_x \varphi_k$$

since $w_k^2 = \varphi_k^T \mathbf{x}\mathbf{x}^T \varphi_k = \varphi_k^T \mathbf{R}_x \varphi_k$

Minimisation by the use of Lagrange multiplier to ensure that orthonoramilty of φ is mentained. we get the function to be minimised as :

$$\mathcal{L} = \sum_{k=K+1}^{N} \varphi_k^T \mathbf{R}_x \varphi_k + \sum_{k=K+1}^{N} \lambda_k (1 - \varphi_k^T \varphi_k)$$

lagrange multipliers

MSE as Sum of Eigen values

The minimization of the above function with lagrange multipliers can eb done by taking gradient with respect to φ_k and setting it to zero, which gives:

$$\nabla_{\varphi_k} \mathcal{L} = \mathbf{R}_x \varphi_k - \lambda_k \varphi_k = 0.$$

or

$$\mathbf{R}_x \varphi_k = \lambda_k \varphi_k. \quad k = K+1, \ldots, N$$

so mean square error can be represented in terms of lagrange coefficients or eigen values λ_k

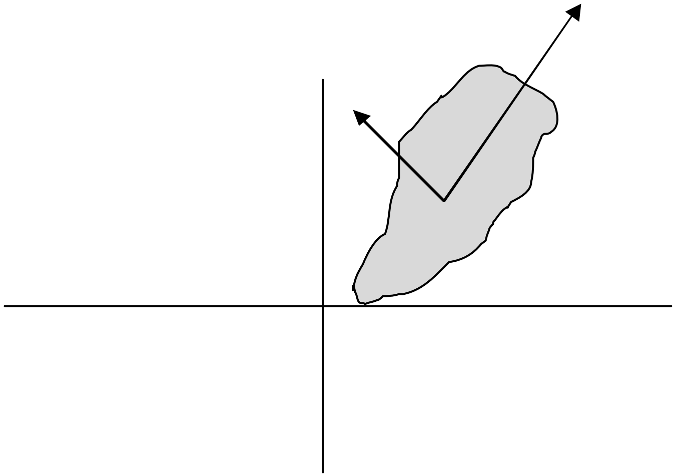
$$\mathcal{E} = \sum_{k=K+1}^{N} \varphi_k^T \mathbf{R}_x \varphi_k$$
$$= \sum_{k=K+1}^{N} \varphi_k^T (\lambda_k \varphi_k) = \sum_{k=K+1}^{N} \lambda_k$$

So is minmised when *N- K* smallest eigenvalues are chosen

or

When signal $\hat{s}$ is represented in terms of basis vector corresponding to largest K eiegen vales

Interpretations



It can be viewed as finding direction of maximum variance in n dimensional space

Interpretation as Linear, Time Variant Filtering

when signal is estimated as means of different basis function i.e.

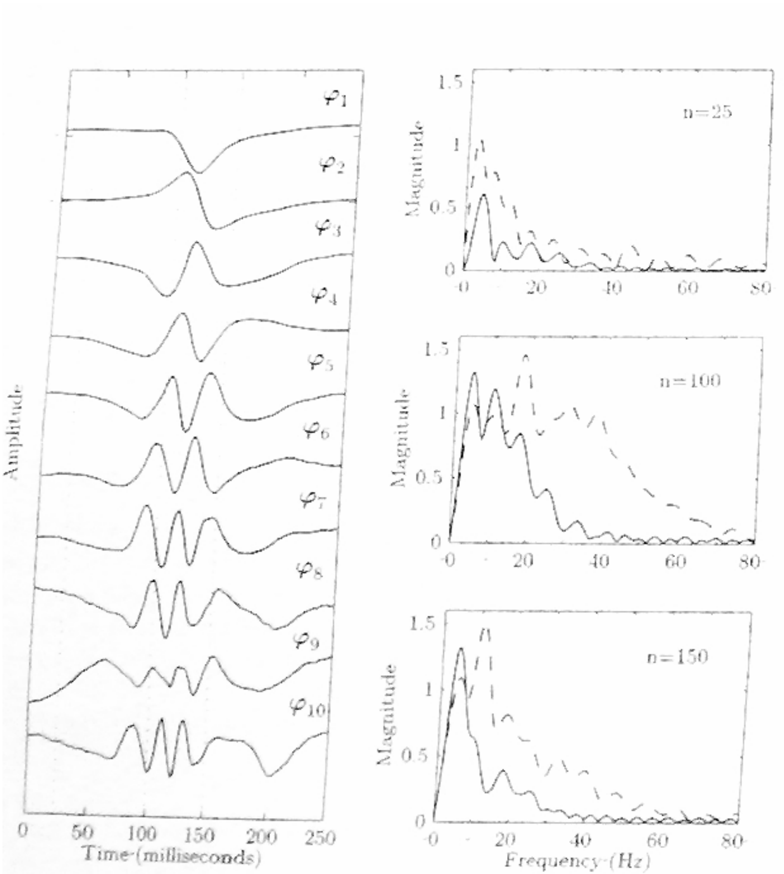
$$\hat{s}_t = \Phi_s \Phi_s^T x_t = \sum_{k=1}^{K} \sum_{l=0}^{N-1} \varphi_k(n) \varphi_k(l) x_t(l)$$

$$= \sum_{l=0}^{N-1} g(l,n) x_t(l), \quad n = 0, \ldots, N-1$$

where

$$g(l,n) = \sum_{k=1}^{K} \varphi_k(l) \varphi_k(n), \quad l, n = 0, \ldots, N-1$$

Interpretation in frequency domain



orthonormal basis functions

Instantaneous frequency responses

Modeling with Damped Sinusoids

Sinusoidal basis function with exponential damping can be used to make a variety of basis functions, such as data x can be represented as :

phase

$$x(n) = \sum_{k=1}^{K} w_k e^{\rho_k n} e^{j(\omega_k n + \phi_k)}$$

frequency

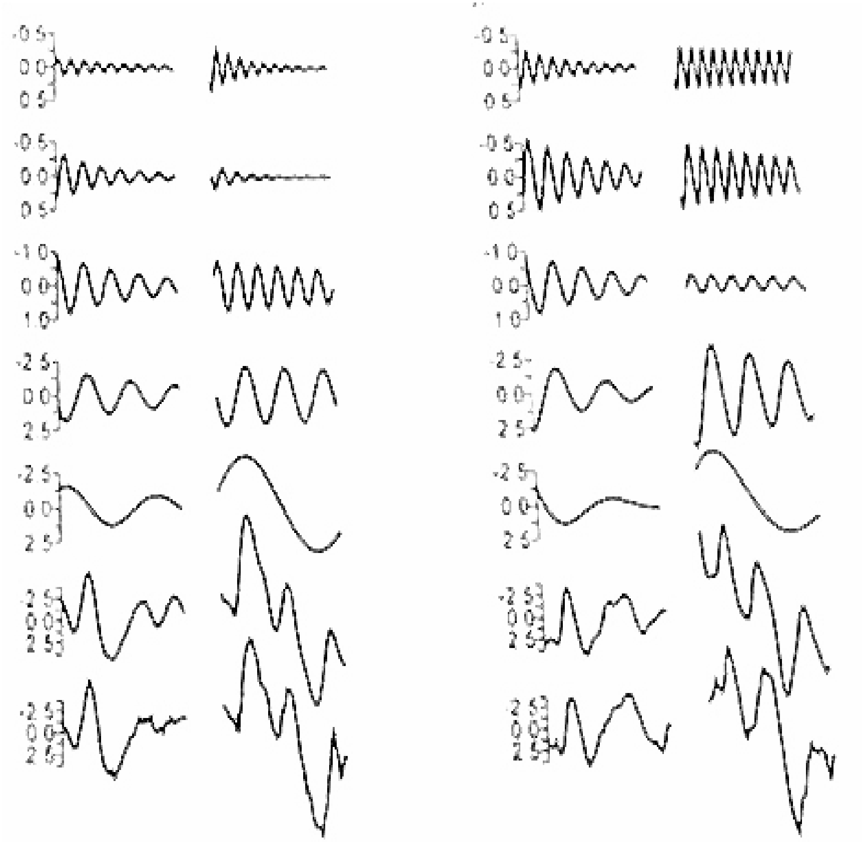
amplitude

Damping factor

Since x(n) is generally real valued signal, so data can be represented by damped cosines because sum of conjugate complex pairs is cosine.

$$x(n) = \sum_{k=1}^{K/2} 2 w_k e^{\rho_k n} \cos(\omega_k n + \phi_k)$$

Pros and cons of damped sinusoid as basis function



.Damping add an extra degree of freedom , so fewer basis function represent the data

No orthonormality

Non linearity of equations makes it tough to find the solutions for best basis function

Prony Method

The model $$x(n) = \sum_{k=1}^{K} u_k e^{\rho_k n} e^{j(\omega_k n + \phi_k)}$$

Can be represented as $$x(n) = \sum_{k=1}^{K} h_k z_k^n.$$ where $$h_k = u_k e^{j\phi_k}$$ $$z_k = e^{\rho_k + j\omega_k}$$

This model can be viewed as homogenous soultion to a linear difference equation with fixed parameters. i.e.

$$x(n) + a_1 x(n-1) + \ldots + a_k x(n-k) = 0$$

In prony's original method it assumed that number of available samples is equal to unknown parameters, so the difference equation is valid for n = k, . . . 2k-1

The difference equation can be represented as k x k matrix equation as:

$$\begin{bmatrix} x(K-1) & x(K-2) & \cdots & x(0) \\ x(K) & x(K-1) & \cdots & x(1) \\ \vdots & \vdots & \ddots & \vdots \\ x(2K-2) & x(2K-3) & \cdots & x(K-1) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_K \end{bmatrix} = - \begin{bmatrix} x(K) \\ x(K+1) \\ \vdots \\ x(2K-1) \end{bmatrix}$$

On solving the above equation we get values of $a_1 \ldots a_k$ which can be used to find the roots of the polynomial

$$A(z) = \sum_{l=0}^{K} a_l z^{K-l} = \prod_{k=1}^{K} (z - z_k)$$

The roots can be used to determine the parameters $h_1, \ldots h_k$ from the eq.

$$\begin{bmatrix} z_1^0 & z_2^0 & \cdots & z_K^0 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{K-1} & z_2^{K-1} & \cdots & z_K^{K-1} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_K \end{bmatrix} = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(K-1) \end{bmatrix}$$

Prony method for real problems: least square Prony method

When we want to use less parameters *k* than available number of observations *N* we have relax the requirement of difference equation as given below

$$x(n) + a_1 x(n-1) + . . . + a_k x(n-k) = e(n)$$ ***error***

And to minimise the error e(n) w.r.t the parameters $a_1, . . . a_k$

The parameters h1 . . .hk is determined by minimizing the function:

$$\|\mathbf{x}_N - \mathbf{Z}_N \mathbf{h}_K\|_2^2 = (\mathbf{x}_N - \mathbf{Z}_N \mathbf{h}_K)^H (\mathbf{x}_N - \mathbf{Z}_N \mathbf{h}_K)$$

And the least square solution comes out to be

$$\mathbf{h}_K = (\mathbf{Z}_N^H \mathbf{Z}_N)^{-1} \mathbf{Z}_N^H \mathbf{x}_N.$$

Adaptive analysis using Basis Functions

Instanteneous LMS Algorithm

The observed signal is made from concatenation of successive Evoked potentials $\mathbf{x}_i = [x((i-1)N), \ldots x(iN-1)]^T$

$$x(n) = x_{\lfloor \frac{n}{N} \rfloor + 1}\left(n - \left\lfloor \frac{n}{N} \right\rfloor N\right)$$

The siganl estimate is calcukated from truncated series expansion $\varphi_s(n)$ but this time with time varying weight vector ***w(n).*** For this purpose minimisation of MSE is done at every sample. For this the differenital of the function

$$\mathcal{E}_{\mathbf{w}}(n) = E\left[(x(n) - \varphi_s^T(n)\mathbf{w}(n))^2\right]$$

is taken to yeild : - $\nabla_{\mathbf{w}}\mathcal{E}_{\mathbf{w}}(n) = -2E[(x(n) - \varphi_s^T(n)\mathbf{w}(n))\,\varphi_s(n)]$

using the gradient to update weights and using mean μ instead of expectation the update formula is :

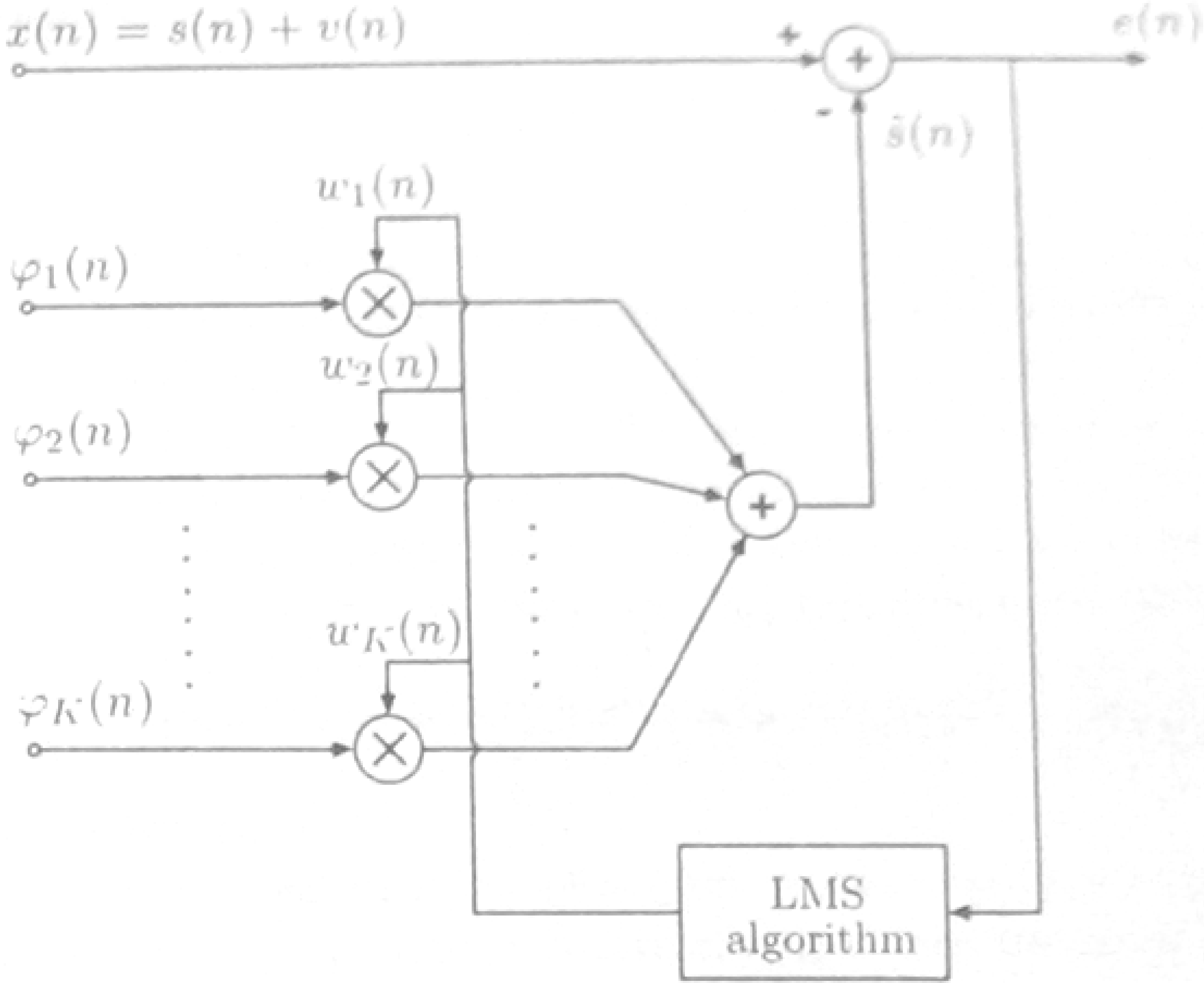
$$\begin{aligned}\mathbf{w}(n+1) &= \mathbf{w}(n) + \mu(x(n) - \boldsymbol{\varphi}_s^T(n)\mathbf{w}(n))\boldsymbol{\varphi}_s(n) \\ &= (\mathbf{I} - \mu\boldsymbol{\varphi}_s(n)\boldsymbol{\varphi}_s^T(n))\mathbf{w}(n) + \mu x(n)\boldsymbol{\varphi}_s(n)\end{aligned}$$

which when expressed in terms of initial weight vector ***w(0)*** can be written as

$$\mathbf{w}(n) = \mathbf{F}_0(n-1)\mathbf{w}(0) + \mu \sum_{m=0}^{n-1} x(m)\mathbf{F}_{m+1}(n-1)\boldsymbol{\varphi}_s(m)$$

where

$$\mathbf{F}_m(n) = \begin{cases} \prod_{j=m}^{n} (\mathbf{I} - \mu\boldsymbol{\varphi}_s(j)\boldsymbol{\varphi}_s^T(j)) & n \geq m \\ \mathbf{I} & m < n. \end{cases}$$



A representation of adaptive analysis using LMS algorithm

The Block LMS algorithm

Its a extension of singla trial LMS method using truncated series expansion.

the error function is quiet simmilar

$$\mathcal{E}_{\mathbf{w}_{i-1}} = E\left[(\mathbf{x}_i - \Phi_s \mathbf{w}_{i-1})^T (\mathbf{x}_i - \Phi_s \mathbf{w}_{i-1})\right]$$

but $\boldsymbol{w}_{i-1}$ is used instead of $\boldsymbol{w}_i$

so the updateing of weights is done by

$w_i = (1-\mu)w_{i-1} + \mu F^T_s\, x_i$ or $w_i = w_{i-1} + \mu F^T_s\,(x_i – F_s w_{i-1})$