E9 231: Digital Array Signal Processing

Scribe: K. Murali Krishna
Dept. of EE & ECE
Indian Institute of Science
Bangalore 560 012, India
murali@ee.iisc.ernet.in

Class held on: 20 Nov 2008

Today
-Gradient descent algorithm
-LMS

1 Speepest Descent Algorithm

$$\begin{aligned} \xi(W) &= E\{(D - W^H X)(D^H - X^H W)\} \\ &= \sigma_d^2 - W^H S_{XD^H} - S_{XD^H}^H W + W^H S_X W \end{aligned} \tag{1}$$

differntiating w.r.t. W^H and set it equal to zero

$$\begin{aligned} S_X W_o &= S_{XD^H} = \mathbf{p} \\ \mathbf{p} &\triangleq S_{XD^H} \end{aligned} \tag{2}$$

assume S_X and $\mathbf{p}$ are known
Insteed of inverting S_X, will use a gradient search
Definition

$$\begin{aligned} W(k) &= W(k-1) - \alpha \nabla \xi(W)|_{W=W(k-1)} \\ W(0) &= \bar{0} \quad \text{or} \quad \frac{V_s}{N} \\ \nabla \xi(W) &= S_X W - \mathbf{p} \end{aligned} \tag{3}$$

consider the error vector

$$\begin{aligned} W_e(k) = W(k) - W_o W_e(k) &= W_e(k-1) - \alpha(S_X W(k-1) - S_X W_o) \\ &= (I - \alpha S_X) W_e(k-1) \end{aligned}$$

$S_X = u \Lambda u^H$ eigen decomposition of S_X

$$W_e(k) = u(I - \alpha\Lambda) u^H W_e(k-1)$$

$V_e(k) \triangleq u^H W_e(k)$

$$V_e(k) = (I - \alpha\Lambda) V_e(k-1) \tag{4}$$

$$
\begin{aligned}
[V_e(k)]_n &= (1-\alpha\lambda_n)[V_e(k-1)]_n \\
&= (1-\alpha\lambda_n)(1-\alpha\lambda_n)[V_e(k-2)]_n \\
&= (1-\alpha\lambda_n)^K[V_e(0)]_n
\end{aligned} \tag{5}
$$

$|1-\alpha\lambda_n| < 1, \qquad n = 1, 2, ..., N$
$0 < \alpha < \frac{2}{\lambda_{max}}$
Time constant $= \frac{1}{\alpha\lambda_n}$
Governed by $\frac{1}{\alpha\lambda_{min}}$
If the $\frac{\lambda_{max}}{\lambda_{min}}$ is large, it will take a long time to converge.

2 Stochastic Speepest Descent

-Widrow 1967

$$W(k) = W(k-1) - \alpha\nabla\xi(W)|_{W=W(k-1)} \tag{6}$$
$$\nabla\xi(W) = S_X W - \mathbf{p} \tag{7}$$

Need to estimate S_X and $\mathbf{p}$
Simple choices

- 1

$$\hat{S}_X = X_K X_K^H \tag{8}$$
$$\hat{S}_{XD^H} = X_K D_K^H \tag{9}$$

- 2

$$\hat{S}_X = \frac{1}{K}\sum_{k=1}^{K} X_k X_k^H \tag{10}$$
$$\hat{S}_{XD^H} = \frac{1}{K}\sum_{k=1}^{K} X_k D_k^H \tag{11}$$

Algorithm:

$$
\begin{aligned}
\hat{W}(k) &= \hat{W}(k-1) + \alpha(k)X_k(D^H(k) - X_k^H\hat{W}(k-1)) \\
\nabla\xi(W) &= X_k X_k^H W - X_k D_k^H
\end{aligned} \tag{12}
$$

$\tilde{Y}_p(k) \triangleq \hat{W}^H(k-1)X_k$
$e_p(k) \triangleq D(k) - \tilde{Y}_p(k)$

$$
\begin{aligned}
\hat{W}(k) &= \hat{W}(k-1) + \alpha(k)X_k e_p^H(k) \\
\hat{W}(0) &= \bar{0} \quad \text{or} \quad \frac{V_s}{N}
\end{aligned} \tag{13}
$$

$$\hat{S}_X = X_k X_k^H + \sigma_L^2 I \tag{14}$$

$$
\begin{aligned}
\hat{W}(k) &= \hat{W}(k-1) + \alpha(k)X_k e_p^H(k) - \sigma_L^2\alpha(k)\hat{W}(k-1) \\
&= (1-\sigma_L^2\alpha(k))\hat{W}(k-1) + \alpha(k)X_k e_p^H(k) \\
&= \beta(k)\hat{W}(k-1) + \alpha(k)X_k e_p^H(k)
\end{aligned} \tag{15}
$$

3 Griffiths LMS, or Steered Direction LMS

$S_{XD^H} = E\{X_k D_k^H\}$ is assumed to be known

$$\begin{aligned} X_k &= V_s F_k + N_k \\ D_k &= F_k \\ E\{X_k D_k^H\} &= \sigma_s^2 V_s \end{aligned} \tag{16}$$

$$\begin{aligned} \hat{W}(k) &= \hat{W}(k-1) + \alpha(k)(\sigma_s^2 V_s - X_k X_k^H \hat{W}(k-1)) \\ \text{diagonal loading} &\qquad X_k X_k^H + \sigma_L^2 I \\ \hat{W}(k) &= \rho(k)\hat{W}(k-1) + \alpha(k)(\sigma_s^2 V_s - X_k \hat{Y}_p^H(k) \end{aligned} \tag{17}$$

Comparision:

- LMS

$$\hat{W}_{LMS}(k) = \hat{W}_{LMS}(k-1) + \alpha(k) X_k [D^H(k) - X_k^H \hat{W}_{LMS}(k-1)] \tag{18}$$

- Computationally simple.
- Take long time to converge.

- RLS

$$\hat{W}_{RLS}(k) = \hat{W}_{RLS}(k-1) + \phi^{-1}(k) X_k [D^H(k) - X_k^H \hat{W}_{RLS}(k-1)] \tag{19}$$

- Require more computations compared to LMS.
- Take less time to converge.

4 Detection of signal subsppace dimension

$$\begin{aligned} X_k &= V(\psi) F_{s,k} + N_k \\ S_x &= V S_f V^H + \sigma_\omega^2 I \end{aligned} \tag{20}$$

want to detect $d = rank(S_f)$
Eigen decomposition of S_x

$$\begin{aligned} S_x &= u_s \lambda u_s^H + \sigma_\omega^2 u_N u_N^H \\ \lambda_1 \geq \lambda_2 \geq ......\lambda_d > \lambda_{d+1} &= \lambda_{d+2} = ... = \lambda_N \end{aligned} \tag{21}$$

In practice, estimate of $\hat{S}_x$ and get
$\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq ...... \geq \hat{\lambda}_d \geq \hat{\lambda}_{d+1} \geq .... \geq \hat{\lambda}_N$
Sequential hypothesis test
$H_0 :\quad (N-d)$ smallest eigen values are equal
$H_1 :\quad (N-d-1)$ smallest eigen values are equal
-Anderson, 1963

1. $K \gg N$, the noise eigenvalues cluster around σ_ω^2
   $\hat{S}_x = \frac{1}{K}\sum_{k=1}^{K} X_k X_k^H$

2. In particular $(\hat{\lambda}_n - \sigma_\omega^2) \sim O(\frac{1}{sqrtK})$
   $n = (d+1), (d+2), ..., N$

3. A sufficient statistic for detecting d is
   $L_d(d) = K(N-d)\ln\left(\frac{\frac{1}{N-d}\sum_{k=d+1}^{N}\hat{\lambda}_k}{(\prod_{k=d+1}^{N}\hat{\lambda}_k)^{\frac{1}{n-d}}}\right)$

4. $2L_d(d) \sim \chi^2((N-d)^2-1)$
   $\nu(d) \triangleq 2L_d(d)$
   $f_{\nu,d}(\nu) = \frac{\nu^{\frac{1}{2}((N-d)^2-1)-1}}{2^{\frac{(N-d)^2-1}{2}}\Gamma(\frac{(N-d)^2-1}{2})}e^{-\nu/2}$

5. Procedure: Fix a confidence level (say 99 %)
   compare τ_0 such that $\int_{\tau_0}^{\infty} f_{\nu,0}(r)dr = 0.99$
   $\nu(0) < \tau_0$, set $\hat{d} = 0$ else compare τ_1 such that $\int_{\tau_1}^{\infty} f_{\nu,1}(r)dr = 0.99$
   $\nu(1) < \tau 1 \quad \text{set} \quad \hat{d} = 1,$
   else......