• wiener 필터 모델.

: 목표출력과 실제출력이 가장 유사하게 되는 가중치 값을 구하는 것.

* winner 필터는 신호의 잡음을 제거하기 위한 선형 필터 모델

* 손실함수는 제곱평균 에러를 사용한다. $L = \frac{1}{2} E(e^2)$

$e = d - y$

d : 목표출력

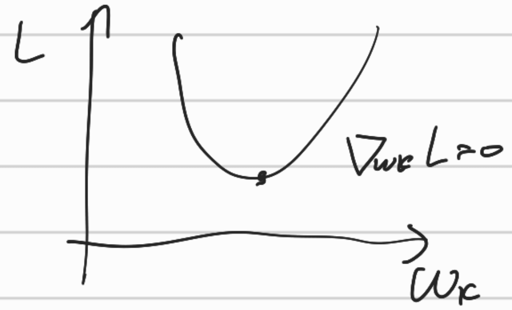
y : 실제출력

• 각 값이 등장할 확률이 동일 ⇒ 기댓값 = 평균값

각 값이 등장할 확률이 다름 ⇒ 기댓값 ≠ 평균값

$$
\begin{aligned}
L &= \frac{1}{2} E(e^2) \\
&= \frac{1}{2} E\left((d-y)^2\right) \\
&= \frac{1}{2} E\left(d^2 - 2dy + y^2\right) \\
&= \frac{1}{2} E(d^2) - E(dy) + \frac{1}{2} E(y^2) \\
&= \underbrace{\frac{1}{2} E(d^2)}_{\text{목표값의 제곱}} - \sum_{i=1}^{p} w_i \underbrace{E(x_i d)}_{\text{목표값과 입력신호의 상관성}} + \frac{1}{2} \sum_{i=1}^{p} \sum_{j=1}^{p} w_i w_j \underbrace{E(x_i x_j)}_{\text{입력 신호간의 자기 상관성}}
\end{aligned}
$$

L은 w_k에 대해 2차함수 → 극점이 존재.



⇒ $\nabla_{w_k} L = 0$ 이며

⇒ $\nabla L = -E(x_k \cdot d) + \sum_{i=1}^{p} w_i E(x_i x_k) = 0$

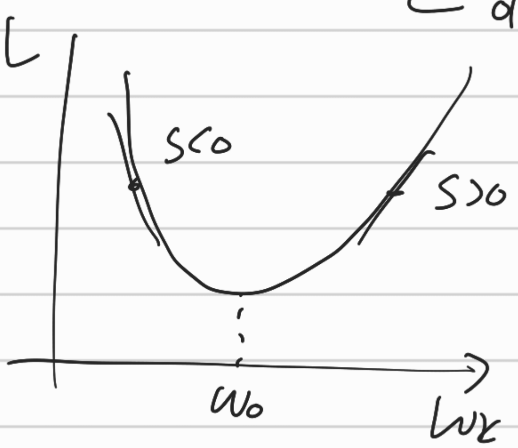
↳ winner - Hopf 방정식

· Winner-Hopf 방정식의 행렬 표현.

$$\begin{pmatrix} E(x_1x_1) & E(x_1x_2) & \cdots & E(x_1x_p) \\ \vdots & & & \vdots \\ E(x_px_1) & E(x_px_2) & \cdots & E(x_px_p) \end{pmatrix} \begin{pmatrix} w_{01} \\ w_{02} \\ \vdots \\ w_{0p} \end{pmatrix} = \begin{pmatrix} E(x_1d) \\ E(x_2d) \\ \vdots \\ E(x_pd) \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} w_{01} \\ w_{02} \\ \vdots \\ w_{0p} \end{pmatrix} = \begin{pmatrix} E(x_1x_2) & \cdots & E(x_1x_p) \\ \vdots & & \vdots \\ E(x_px_1) & \cdots & E(x_px_p) \end{pmatrix}^{-1} \cdot \begin{pmatrix} E(x_1d) \\ \vdots \\ E(x_pd) \end{pmatrix}$$

역이 없을 경우로...



→ 기울기를 이용하여 w_0를 구하자.
(gradient descent sheme)

1. 초기 가중치 값에 대해 함수 기울기 계산.
2. 1의 기울기 반대 방향으로 가중치 값을 변경
3. 변경된 가중치에 대해 e 계산
4. e 변화가 없을 때까지 반복.

· 선형 뉴런 : L이 2차함수. ∴ 최소값이 1개

· 비선형 뉴런 : 극소점 여러개 존재.

S층 결정이 사용하는 데이터의 성격에 따라 분류.

- Steepest Gradient descents
  : 통계적인 특별까지 고려한 오류함수 최소화 방법.
  : 파라미터 w_i에 대해 $\frac{\partial L}{\partial w_i}$을 구한 뒤 파라미터를 갱신.
  : Error Surface 에서 Steepest descent 경로를 따라 이동

서서히 가장 경사가 급한 쪽으로 향하고, 오차값 대비 최소점 도달.

for $t = 1 \sim N$

$$\nabla_{w_k} L^t = -E[x_k d_t] + \sum_{i=1}^{p} w_i^t \underbrace{E(x_i x_k)}_{i,\, t \text{ 개체의 포함.}}$$

$$w_k^{t+1} = \underset{\text{기존의 바로 계산값}}{w_k^t} - \underset{\text{비례상수}}{\eta}\, \underset{L \text{ 함수의 기울기}}{\nabla_{w_k} L^t}$$

(Widrow & Hoff 학습법)

- Delta rule. (Error correction learning rule)
  : 통계적 특별을 배제해서 기존 계산법을 단순화한 근사적 방법.
  에러 표면에서 Steepest descent 경로 추정을 지그재그 형태로 이동

  - 최소제곱 (LMS) 에러 사용은 그대로 둔다.
  에러는 기대값 대신 순간값 사용.

- 선형 뉴런의 경우

$$L = \frac{1}{2}(d-y)^2 \leftarrow y = \sum w_i x_i, \quad \frac{\partial y}{\partial w_i} = x_i$$

$$\frac{\partial L}{\partial w_i} = -(d-y)x_i$$

$$\Delta w_i = -\eta \frac{\partial L}{\partial w_i} = \underset{\text{학습률.}}{\eta}\, \underset{\text{출력에러}}{(d-y)}\, \underset{\text{입력벡터.}}{x_i}$$

$$\therefore w_i = w_i + \Delta w_i$$

- 비선형회귀

$$L = \frac{1}{2}\left(d - \varphi(w^T x)\right)^2$$

$$\frac{\partial L}{\partial w_2} = -\left(d - \varphi(w^T x)\right) \cdot \varphi'(w^T x) \cdot x_2$$

$$\Delta w_2 = -\eta \frac{\partial L}{\partial w_2}$$

$$w_2 = w_2 + \Delta w_2$$

$$\begin{pmatrix} 3 & 1 & 1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix} = 3 + 2 + 2 + 2 = 9$$