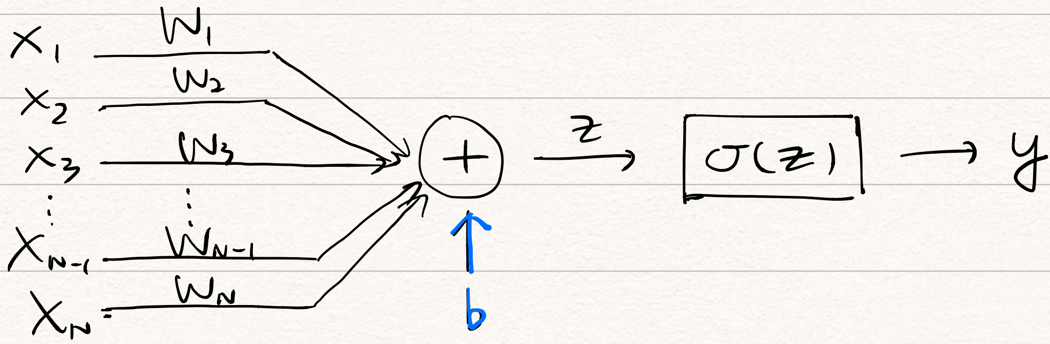
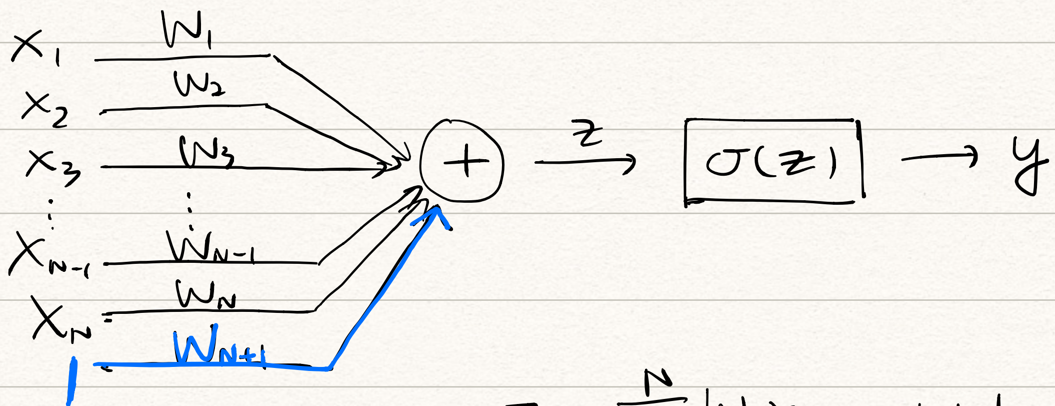
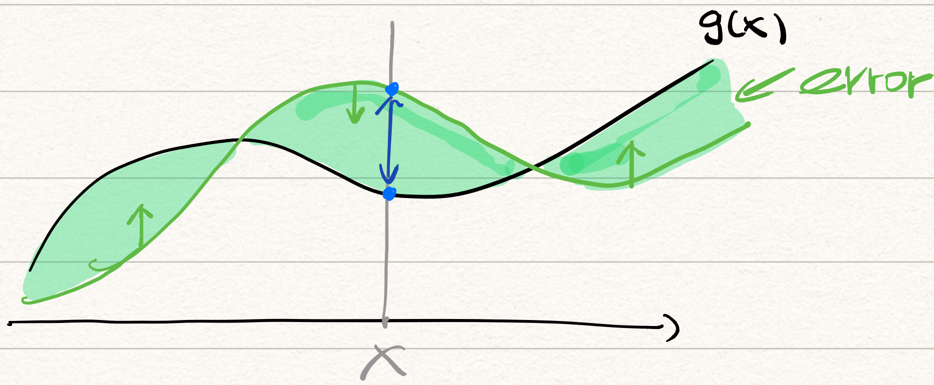
Preliminaries: The units in the network



$$z = \sum_{i}^{N} w_i x_i + b$$



$$z = \sum_{i}^{N} w_i x_i + 1 \cdot w_{N+1}$$



Divergence:

$$\hat{W} = \underset{W}{\operatorname{argmin}} \int_X \operatorname{div}(f(x;W), g(x))\, dx$$

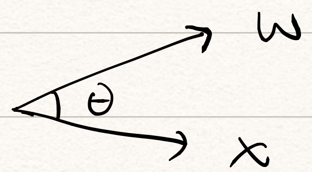
※ NOTICE: Learning a neural network $\Leftrightarrow$

determining the parameters of the network (weights and biases) required for it to model a desired function.

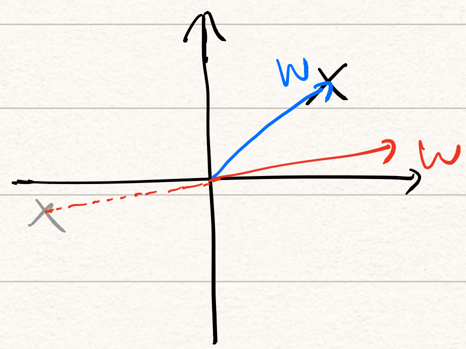
$W^T x = 0 \Rightarrow \underline{x}$ orthogonal $\underline{w}$

$y = \text{sign}(w^T x)$ "inner product"

$w^T x = |w|\,|x| \cos\theta$



$-\frac{\pi}{2} < \theta < \frac{\pi}{2}$



1. w is orthogonal to the plane
2. positive instances: $w \underset{负}{\overset{正}{\rightleftarrows}} x$

Perceptron learning algorithm:

① Given N training instances $(X_1, Y_1), (X_2, Y_2), \cdots$

* $Y_i = +1$ 或 -1

② Initialize W

③ Cycle through the training instances

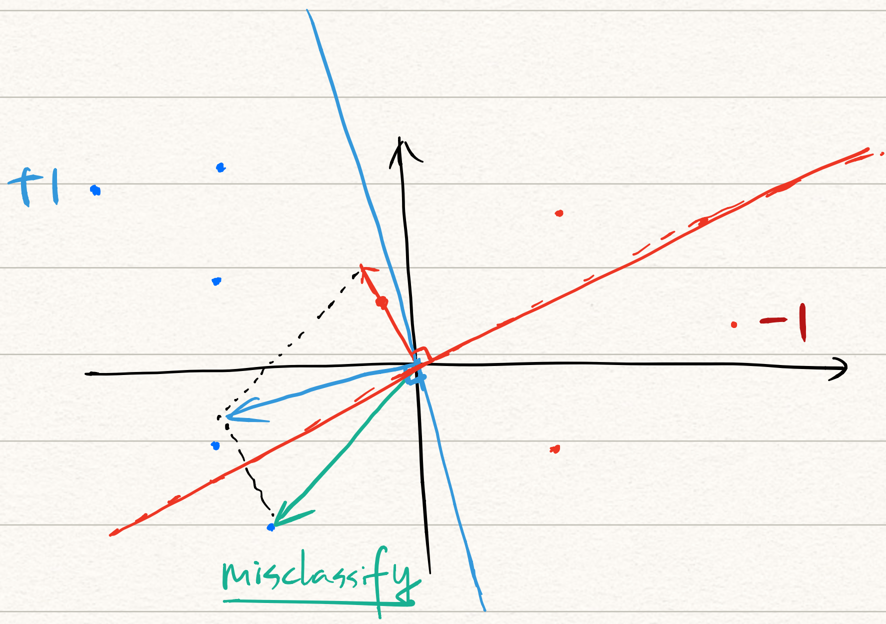
④ While more classification errors:

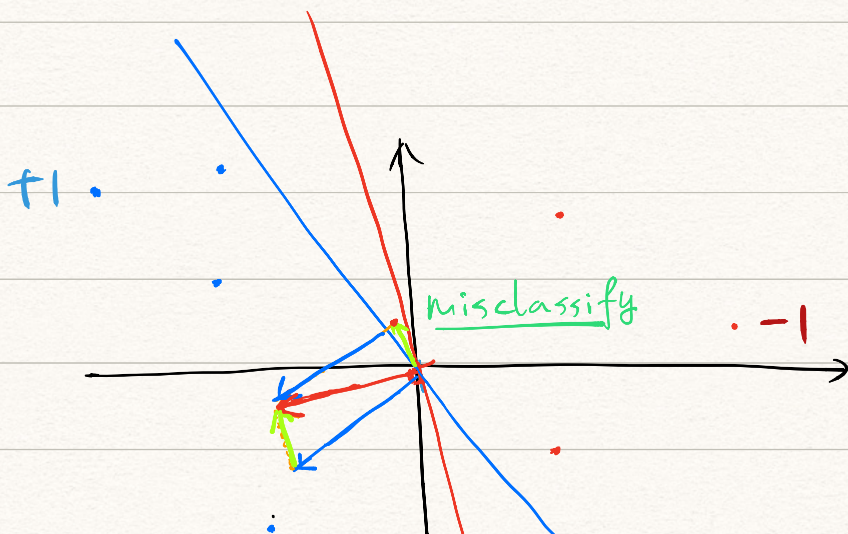
For $i = 1, 2, \ldots, N_{train}$

$$O(X_i) = \text{sign}(W^T X_i)$$

if $O(X_i) \neq Y_i$

$$W = W + Y_i X_i$$





1972 Paul Werbos MIT PhD "back propagation"

differentiable activation function:

$$\frac{dy}{dw_i} = \frac{dy}{dz} \cdot \frac{dz}{dw_i} = \sigma'(z)x_i$$

$$\frac{dy}{dx_i} = \frac{dy}{dz} \cdot \frac{dz}{dx_i} = \sigma'(z)w_i$$

input-output pairs: $(x_1, d_1), (x_2, d_2), \cdots, (x_N, d_N)$

$$\hat{W} = \underset{W}{\operatorname{argmin}} \int_x div(f(x;W), g(x))\,dx$$

$$\hat{W} = \underset{W}{\operatorname{argmin}} \int_X div(f(x;W), g(x))\,p(x)\,dx$$

$$= \underset{W}{\operatorname{argmin}} E\left[div(f(x;W), g(x))\right]$$

$$E[div(f(x;W), g(x))] \approx \frac{1}{N}\sum_{i=1}^{N} div(f(x_i, W), d_i)$$

Function Minimization

$$\begin{cases} Err(W) = \frac{1}{N} \sum_i \boxed{div(f(X_i; W), d_i)} \\ \hat{W} = \underset{W}{\operatorname{argmin}}\, Err(W) \end{cases}$$