Winter 2019

CSEN1022: Machine Learning

Discriminant Functions (2)

Seif Eldawlatly

Learning Classifier Parameters

$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

How to find $\mathbf{w}$ and w_0 ?

Least Squares

Fisher's Linear Discriminant

Perceptron

Fisher's Linear Discriminant

- Discriminant function performs dimensionality reduction

$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

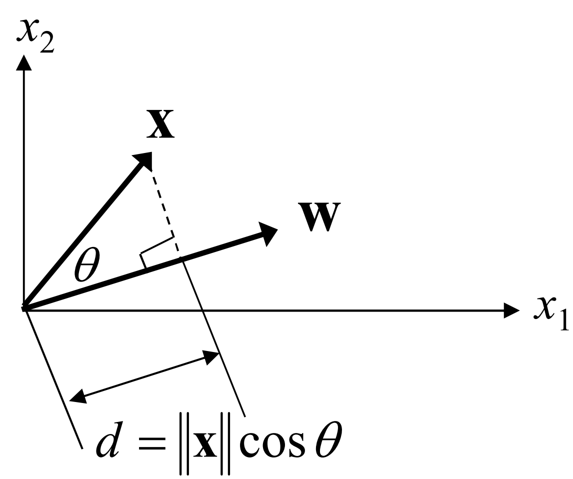
If $\mathbf{x}$ is n x 1, $\mathbf{w}$ must be n x 1 and so $y(\mathbf{x})$ is 1 x 1. Therefore, discriminant function reduces the dimensionality of the input data from n-dimensions to 1 dimension.

- Dimensionality reduction is achieved through the dot product of $\mathbf{w}$ and $\mathbf{x}$

$$\mathbf{w}^T \mathbf{x} = \mathbf{w} \cdot \mathbf{x}$$

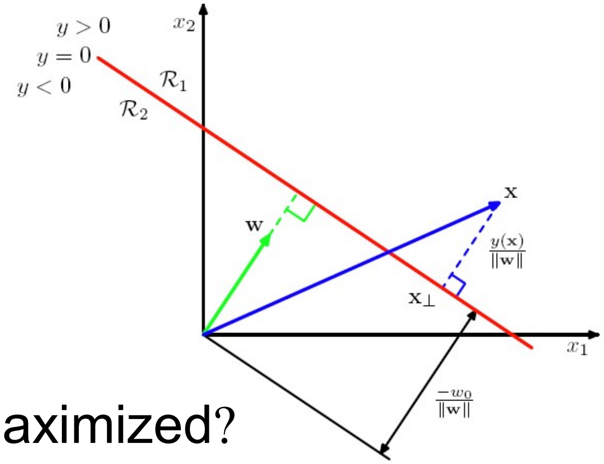
- The dot product of $\mathbf{w}$ and $\mathbf{x}$ is equivalent to projecting $\mathbf{x}$ on $\mathbf{w}$

$$\mathbf{w}^T \mathbf{x} = \mathbf{w} \cdot \mathbf{x} = \|\mathbf{w}\|.\|\mathbf{x}\| \cos\theta$$



Fisher's Linear Discriminant

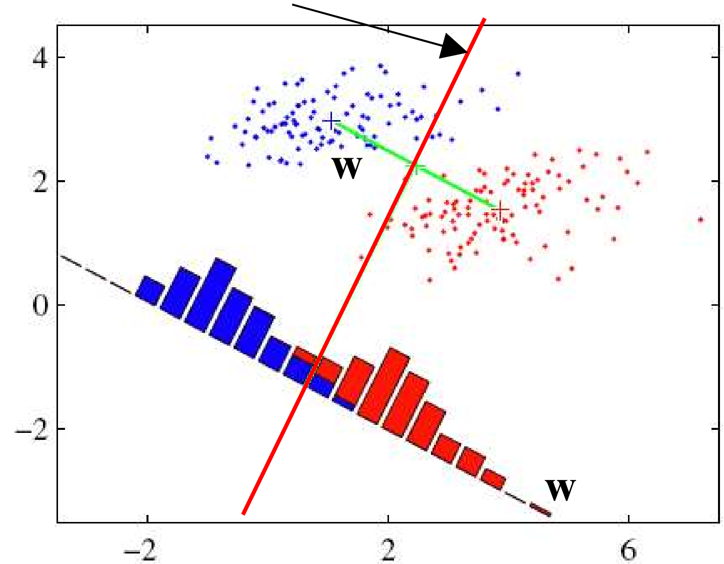
- Projected data might be less separable compared to original data



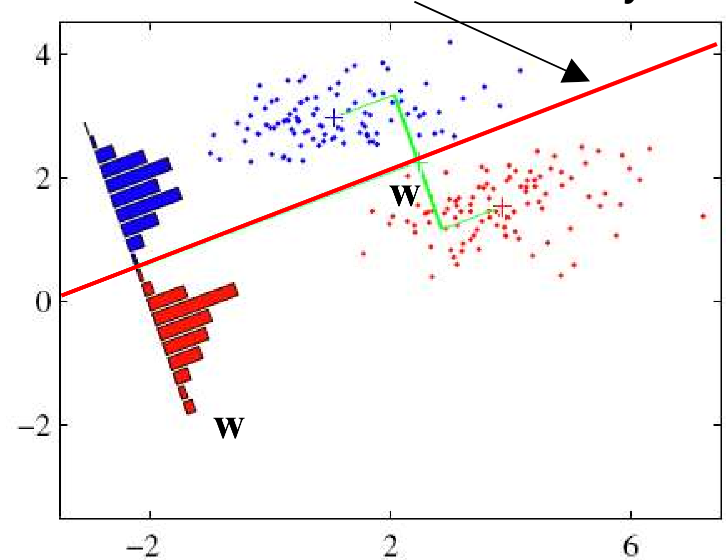
- Recall that the weights vector $\mathbf{w}$ is perpendicular to the decision boundary

- How to choose $\mathbf{w}$ and w_0 so that separation is maximized?

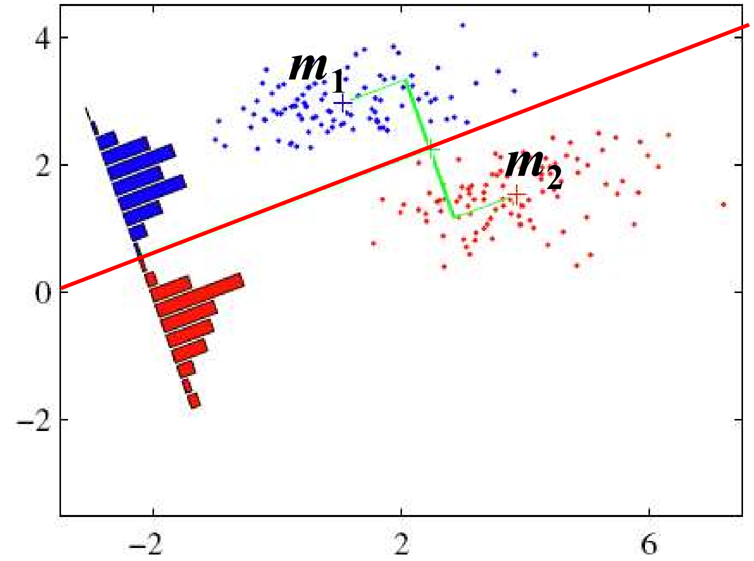
Not good decision boundary



Good decision boundary



Fisher's Linear Discriminant



- Class Means

$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \mathbf{x}_n, \qquad \mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \mathbf{x}_n$$

$$m_k = \mathbf{w}^{\mathrm{T}} \mathbf{m}_k$$

- Class Variance

$$s_k^2 = \sum_{n \in \mathcal{C}_k} (y_n - m_k)^2$$

- Goal:

  Maximize after-projection separation while minimizing the within-class variance

- Simplest measure of separation is the separation between the means

- Within-class variance can be approximated as the summation of the variances of both classes

Fisher's Linear Discriminant

- Fisher's criterion:
  Maximize separation while minimizing the within-class variance

$$J(\mathbf{w}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2} \longrightarrow J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}}$$

$$\mathbf{S}_{\mathrm{B}} = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^{\mathrm{T}}$$

$$\mathbf{S}_{\mathrm{W}} = \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^{\mathrm{T}} + \sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^{\mathrm{T}}$$

- Solution: Take the derivative of $J(\mathbf{w})$ with respect to $\mathbf{w}$ and equate with 0

$$\frac{d}{d\mathbf{w}} J(\mathbf{w}) = \frac{\left(\mathbf{w}^T S_{\mathrm{W}} \mathbf{w}\right)\left(\frac{d}{d\mathbf{w}} \mathbf{w}^T S_{\mathrm{B}} \mathbf{w}\right) - \left(\mathbf{w}^T S_{\mathrm{B}} \mathbf{w}\right)\left(\frac{d}{d\mathbf{w}} \mathbf{w}^T S_{\mathrm{W}} \mathbf{w}\right)}{\left(\mathbf{w}^T S_{\mathrm{W}} \mathbf{w}\right)^2}$$

$$= \frac{\left(\mathbf{w}^T S_{\mathrm{W}} \mathbf{w}\right)\left(2 S_{\mathrm{B}} \mathbf{w}\right) - \left(\mathbf{w}^T S_{\mathrm{B}} \mathbf{w}\right)\left(2 S_{\mathrm{W}} \mathbf{w}\right)}{\left(\mathbf{w}^T S_{\mathrm{W}} \mathbf{w}\right)^2} = 0$$

$$\therefore \left(\mathbf{w}^T S_{\mathrm{W}} \mathbf{w}\right)\left(S_{\mathrm{B}} \mathbf{w}\right) = \left(\mathbf{w}^T S_{\mathrm{B}} \mathbf{w}\right)\left(S_{\mathrm{W}} \mathbf{w}\right)$$

Fisher's Linear Discriminant

- Divide both sides by $\mathbf{w}^T S_W \mathbf{w}$

$$\therefore S_B \mathbf{w} = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}} S_W \mathbf{w}$$

- Since $S_B \mathbf{w}$ is always in the direction of $(\mathbf{m}_2 - \mathbf{m}_1)$

$$S_B \mathbf{w} = (\mathbf{m}_2 - \mathbf{m}_1)\underbrace{(\mathbf{m}_2 - \mathbf{m}_1)^T}_{(1 \times 2)} \underbrace{\mathbf{w}}_{(2 \times 1)} = (\mathbf{m}_2 - \mathbf{m}_1)c$$

$$\therefore (\mathbf{m}_2 - \mathbf{m}_1)c = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}} S_W \mathbf{w}$$

$$S_W \mathbf{w} = \frac{\mathbf{w}^T S_W \mathbf{w}}{\mathbf{w}^T S_B \mathbf{w}} c(\mathbf{m}_2 - \mathbf{m}_1)$$

$$S_W \mathbf{w} \propto (\mathbf{m}_2 - \mathbf{m}_1)$$

$$\therefore \mathbf{w} \propto S_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1) \longrightarrow \textit{Fisher's Solution}$$

Learning Classifier Parameters

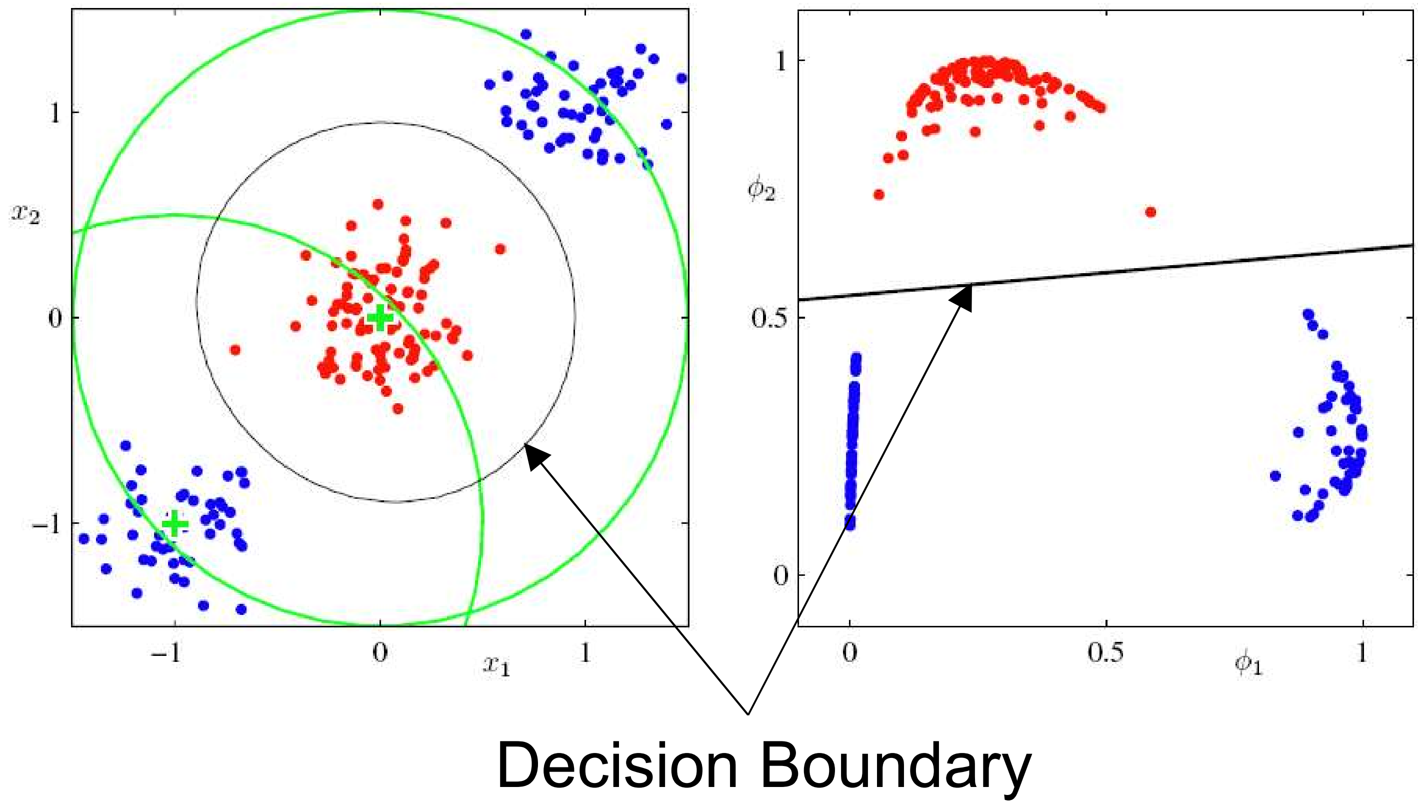
$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

How to find $\mathbf{w}$ and w_0 ?

Least Squares   Fisher's Linear Discriminant   Perceptron

Perceptron

- First, let's deal with a nonlinear transformation of the data $\phi(\mathbf{x})$ (basis function)



Perceptron

- Define

$$y(\mathbf{x}) = f\left(\mathbf{w}^T \phi(\mathbf{x})\right) = \begin{cases} +1, & \mathbf{w}^T \phi(\mathbf{x}) \geq 0 \\ -1, & \mathbf{w}^T \phi(\mathbf{x}) < 0 \end{cases}$$ → Class C_1 / → Class C_2

$\phi(\mathbf{x})$: Feature vector (with a bias component $\phi_0(\mathbf{x}) = 1$)

$f(.)$: Activation function = $t \in \{-1, +1\}$

- Goal: Find $\mathbf{w}$ such that $\mathbf{w}^T \phi(\mathbf{x}_n) \geq 0$ if $\mathbf{x}_n \in C_1$ and $\mathbf{w}^T \phi(\mathbf{x}_n) < 0$ if $\mathbf{x}_n \in C_2$

Or $\mathbf{w}^T \phi(\mathbf{x}_n) t_n \geq 0$

Perceptron

- Perceptron Criterion
  - For correctly classified patterns, error = 0
  - For misclassified patterns, minimize the quantity $-\mathbf{w}^T\phi(\mathbf{x}_n)t_n$

Or minimize $E_P(\mathbf{w}) = -\sum_{n\in M}\mathbf{w}^T\phi(\mathbf{x}_n)t_n$ M: Misclassified patterns

If t_n = 1 and $\mathbf{w}^T\phi(\mathbf{x}_n)$ < 0, then $\mathbf{w}^T\phi(\mathbf{x}_n)t_n$ < 0
If t_n = -1 and $\mathbf{w}^T\phi(\mathbf{x}_n)$ > 0, then $\mathbf{w}^T\phi(\mathbf{x}_n)t_n$ < 0

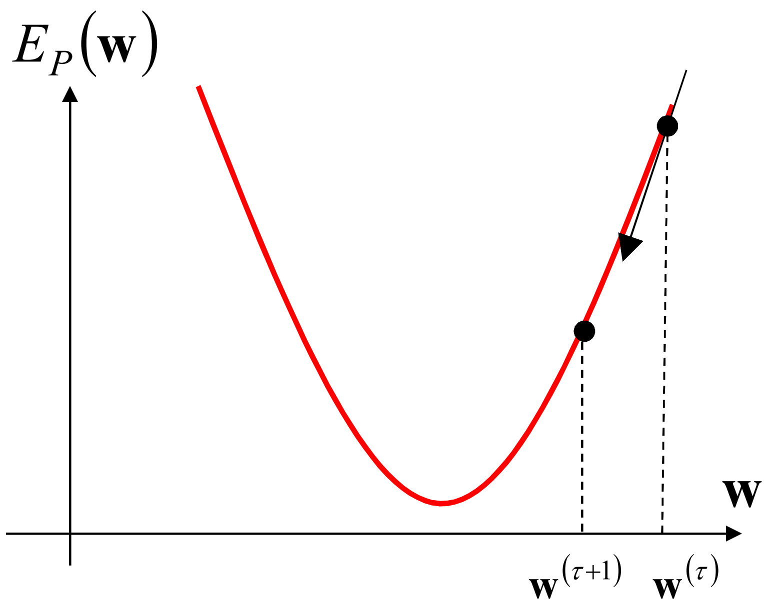
$\therefore E_P(\mathbf{w}) = -\sum_{n\in M}\mathbf{w}^T\phi(\mathbf{x}_n)t_n$ is always positive

Perceptron

- Using gradient descent we try to iteratively minimize

$$E_P(\mathbf{w}) = -\sum_{n \in M} \mathbf{w}^T \phi(\mathbf{x}_n) t_n$$

- Consider a 1-dimension $\mathbf{w}$:

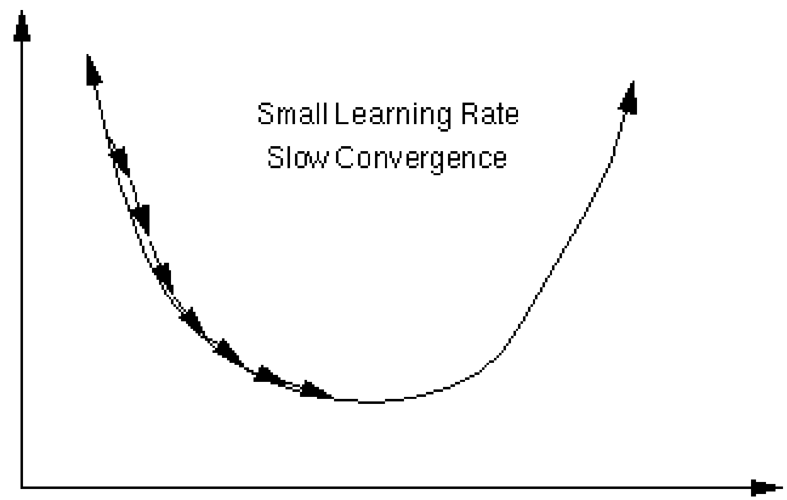


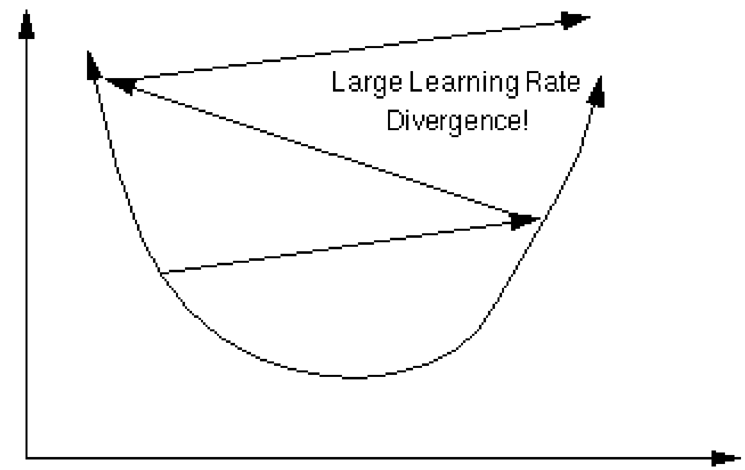
$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \frac{\partial E_P}{\partial \mathbf{w}^{(\tau)}} = \mathbf{w}^{(\tau)} + \eta \phi(\mathbf{x}_n) t_n$$

where η is the learning rate parameter

Perceptron

- Choice of η



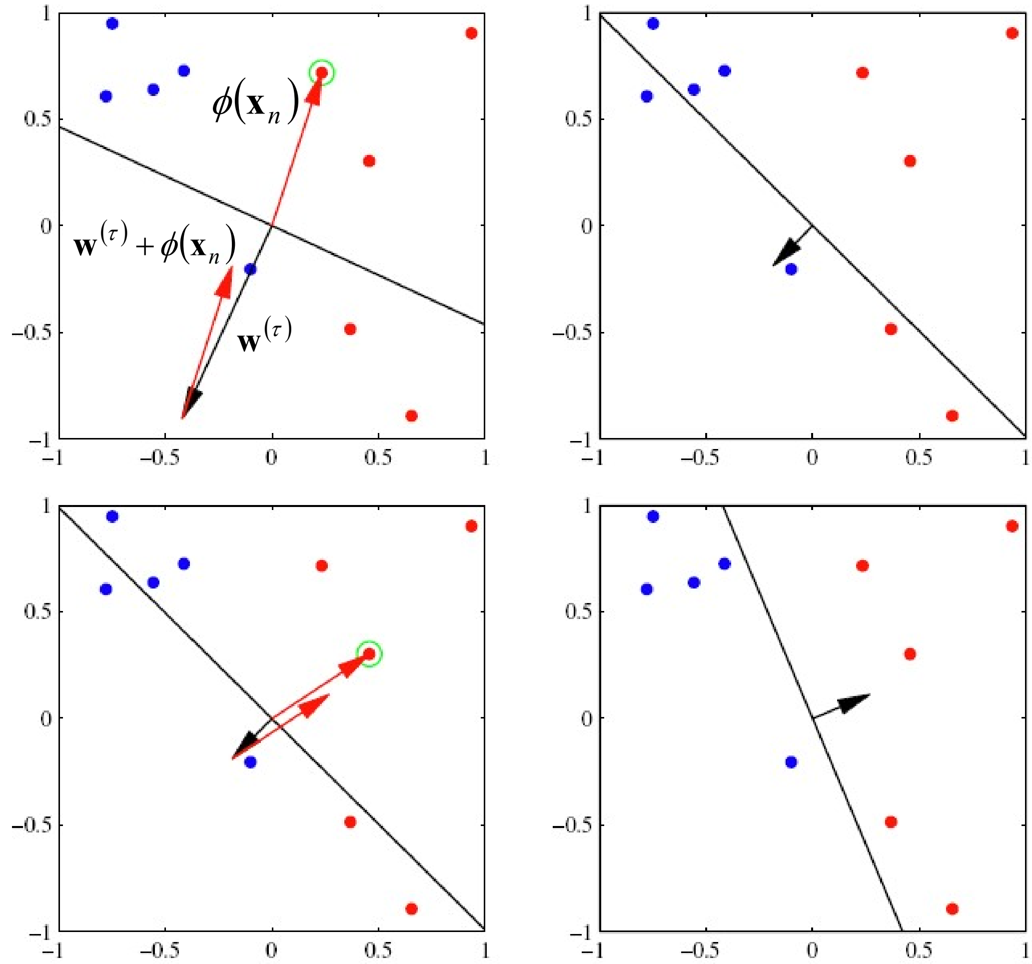


Perceptron

- Example

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \eta\phi(\mathbf{x}_n)t_n$$

Assume $\eta = 1$ and t_n for red class = +1



Perceptron

- Perceptron algorithm always converges

$$\because \mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \phi(\mathbf{x}_n)t_n \quad \text{for} \quad \eta = 1$$

Multiply both sides by $-\phi(\mathbf{x}_n)t_n$

$$-\mathbf{w}^{(\tau+1)T}\phi(\mathbf{x}_n)t_n = -\mathbf{w}^{(\tau)T}\phi(\mathbf{x}_n)t_n - \left(\phi(\mathbf{x}_n)t_n\right)^T\phi(\mathbf{x}_n)t_n$$

$$\because -\mathbf{w}^{(\tau)T}\phi(\mathbf{x}_n)t_n > 0 \text{ and } \left(\phi(\mathbf{x}_n)t_n\right)^T\phi(\mathbf{x}_n)t_n > 0$$

True for any miss-classified point

True since it's equivalent to squaring

$$\therefore -\mathbf{w}^{(\tau+1)T}\phi(\mathbf{x}_n)t_n < -\mathbf{w}^{(\tau)T}\phi(\mathbf{x}_n)t_n$$

Error at iteration $\tau+1$

Error at iteration τ

Since the error is always decreasing, then the algorithm is converging