# The perceptron algorithm

Lecture 7

Frank Rosenblatt (1928 - 1971)

typically, $\varphi_0(\vec{x}) = 1$ (bias) is included

Two-class model; input $\vec{x} \Rightarrow$ feature vector $\vec{\varphi}(\vec{x})$

Consider $y(\vec{x}) = f(\vec{w}^T \cdot \vec{\varphi}(\vec{x}))$, where

$$f(a) = \begin{cases} +1, & a \geq 0 \\ -1, & a < 0 \end{cases}$$

Assume $t = \begin{cases} +1 & C_1 \\ -1 & C_2 \end{cases}$

Error function: perceptron criterion

Note that we seek $\vec{w}$ s.t. all patterns in $C_1$ have $\vec{w}^T \cdot \vec{\varphi}(\vec{x}_n) > 0$, and all patterns in $C_2$ have $\vec{w}^T \cdot \vec{\varphi}(\vec{x}_n) < 0$.

Then $\vec{w}^T \cdot \vec{\varphi}(\vec{x}_n)\, t_n > 0$ in both classes

Let's assign $\emptyset$ error to all correctly classified points (patterns), and $-\vec{w}^T \cdot \vec{\varphi}(\vec{x}_n)\, t_n$ to all missclassified points. Then

$$E(\vec{w}) = -\sum_{n \in \mathcal{N}} \vec{w}^T \cdot \vec{\varphi}(\vec{x}_n)\, t_n$$

$\mathcal{N}$: set of all misclassified points

needs to be minimized.

Note that $E(\vec{w}) > 0$ since $\sum_{n \in \mathcal{N}}$ sums over misclassified patterns only.

$E(\vec{w})$ is a piecewise linear : $\sim \vec{w}$ in regions of $\vec{w}$ space where $\vec{x}_n$ is misclassified, and $\emptyset$ otherwise.

contribution of $n$th term in $E(\vec{w})$ is

Now optimise $E(\vec{w})$ by gradient descent: cycle through $n$ and at each step, update

step number; learning rate

$$\vec{w}^{(\tau+1)} = \vec{w}^{(\tau)} - \eta \vec{\nabla}_n E(\vec{w}) =$$

$$= \vec{w}^{(\tau)} + \eta \vec{\varphi}(\vec{x}_n) t_n$$

Set $\eta = 1 \iff y(\vec{x})$ is insensitive to the choice of $\eta$.

Note that

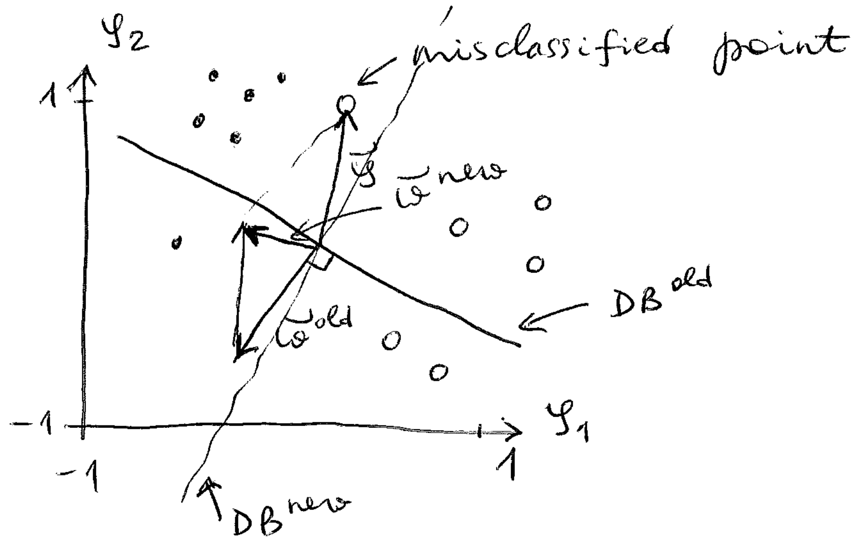
$$-\vec{w}^{(\tau+1)T} \underbrace{\vec{\varphi}_n}_{=\vec{\varphi}(\vec{x}_n)} t_n = -\vec{w}^{(\tau)T} \vec{\varphi}_n t_n - \underbrace{(\vec{\varphi}_n t_n)^T (\vec{\varphi}_n t_n)}_{>0} < -\vec{w}^{(\tau)T} \vec{\varphi}_n t_n.$$

Thus the error always decreases for a given term, but some other patterns may become misclassified $\Rightarrow E(\vec{w})$ is not guaranteed to be reduced at each step.

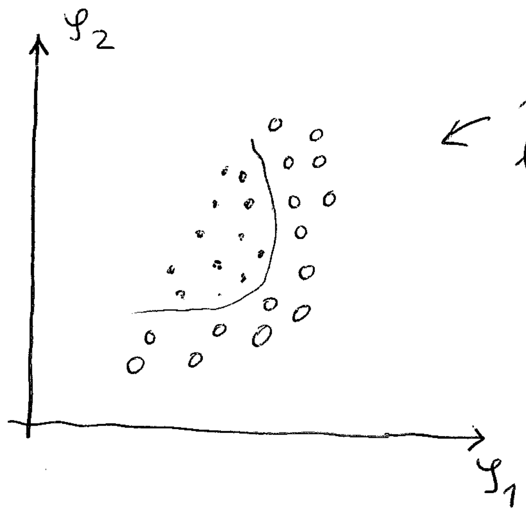
However, one can argue that the perceptron always finds a solution (but may need a lot of steps to do so) for linearly separable problems.

[The solution may not be unique]

Ex.




<u>Difficulties</u>: not probabilistic, does not generalize to $K>2$ classes easily.




lack of convergence on linearly non-separable datasets

## Probabilistic models

Class-conditional density : $p(\vec{x}|C_k)$
$k = 1, \dots, K$

Class prior: $p(C_k)$

Class posterior: $p(C_k|\vec{x})$

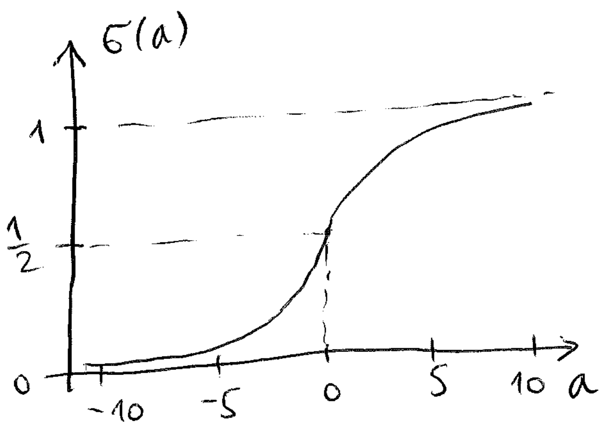
K=2 case:

$$p(C_1|\vec{x}) = \frac{p(\vec{x}|C_1)\,p(C_1)}{p(\vec{x}|C_1)\,p(C_1) + p(\vec{x}|C_2)\,p(C_2)} =$$

$$= \frac{1}{1 + \underbrace{\frac{p(\vec{x}|C_2)\,p(C_2)}{p(\vec{x}|C_1)\,p(C_1)}}_{= e^{-a}}} = \sigma(a)$$

sigmoid f'n

$$a = \log \frac{p(\vec{x}|C_1)\,p(C_1)}{p(\vec{x}|C_2)\,p(C_2)}.$$




Note that

$$\sigma(-a) = \frac{1}{1+e^{a}}$$

$$\parallel$$

$$1-\sigma(a) = 1 - \frac{1}{1+e^{-a}} = \frac{e^{-a}}{1+e^{-a}} = \frac{1}{e^{a}+1}.$$

also, $\underbrace{a = \log\left(\frac{\sigma}{1-\sigma}\right)}_{\text{logit f'n}}$.

## K > 2 case:

$$p(C_k|\vec{x}) = \frac{p(\vec{x}|C_k)\,p(C_k)}{\sum_{j=1}^{K} p(x|C_j)\,p(C_j)} =$$

$$= \underbrace{\frac{e^{a_k}}{\sum_j e^{a_j}}}_{\text{softmax f'n :}} , \quad \text{if} \quad a_k = \log[p(\vec{x}|C_k)\,p(C_k)]$$

if $a_k \gg a_j$ , $\forall j \neq k$

$$\frac{e^{a_k}}{\sum_j e^{a_j}} \simeq 1 = p(C_k|\vec{x}) \Rightarrow p(C_j|\vec{x}) \simeq 0, \quad \forall j \neq k$$

---

Now assume that

same for all classes for now

$$p(\vec{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu}_k)^T \Sigma^{-1} (\vec{x}-\vec{\mu}_k)}$$

$D$ = # dim's in $\vec{x}$

K=2 case:

Then $$p(C_1|\vec{x}) = \sigma\Big(\underbrace{\log\frac{p(\vec{x}|C_1)}{p(\vec{x}|C_2)} + \log\frac{p(C_1)}{p(C_2)}}\Big)$$

$$\log\frac{p(C_1)}{p(C_2)} - \frac{1}{2}(\vec{x}-\vec{\mu}_1)^T \Sigma^{-1} (\vec{x}-\vec{\mu}_1) + \frac{1}{2}(\vec{x}-\vec{\mu}_2)^T \Sigma^{-1} (\vec{x}-\vec{\mu}_2) =$$

$$= \frac{1}{2}\vec{x}^T\Sigma^{-1}\vec{\mu}_1 + \frac{1}{2}\vec{\mu}_1^T\Sigma^{-1}\vec{x} - \frac{1}{2}\vec{x}^T\Sigma^{-1}\vec{\mu}_2 - \frac{1}{2}\vec{\mu}_2^T\Sigma^{-1}\vec{x} + \log\frac{p(C_1)}{p(C_2)} - \frac{1}{2}\vec{\mu}_1^T\Sigma^{-1}\vec{\mu}_1 + \frac{1}{2}\vec{\mu}_2^T\Sigma^{-1}\vec{\mu}_2 =$$

$$= \frac{1}{2}(\Sigma^{-1}\vec{\mu}_1)^T\vec{x} + \frac{1}{2}\underbrace{(\Sigma^{-1}\vec{\mu}_1)^T}_{\vec{\mu}_1^T(\Sigma^{-1})^T}\vec{x} - \frac{1}{2}(\Sigma^{-1}\vec{\mu}_2)^T\vec{x} - \frac{1}{2}(\Sigma^{-1}\vec{\mu}_2)^T\vec{x} + \ldots$$

$(\Sigma^{-1})^T = \Sigma^{-1}$

Defining
$$\begin{cases} \vec{w} = \Sigma^{-1}(\vec{\mu}_1 - \vec{\mu}_2), \\ w_0 = -\frac{1}{2}\vec{\mu}_1^T \Sigma^{-1}\vec{\mu}_1 + \frac{1}{2}\vec{\mu}_2^T \Sigma^{-1}\vec{\mu}_2 + \log\frac{p(C_1)}{p(C_2)}, \end{cases}$$

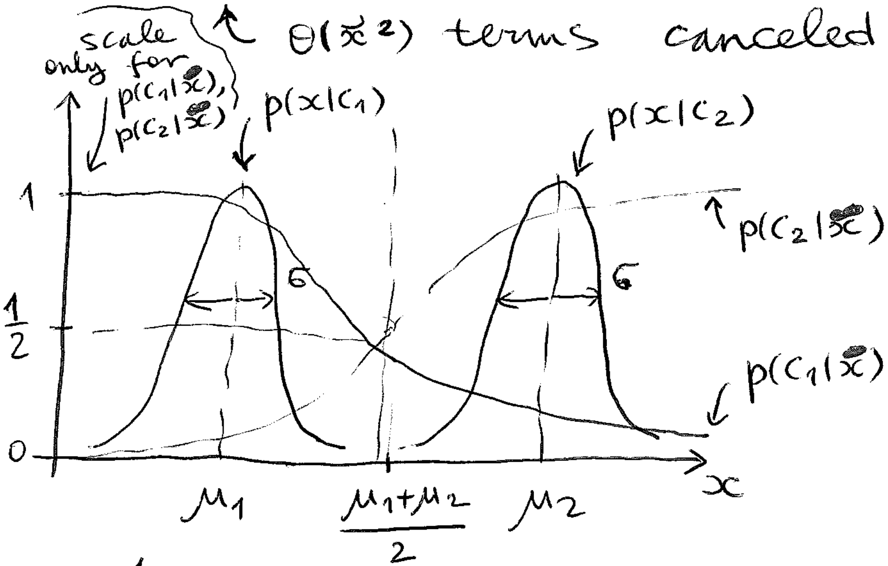
we obtain:

$$p(C_1|\vec{x}) = \sigma(\overbrace{\vec{w}^T\vec{x} + w_0}^{``a"}) \; ; \quad p(C_2|\vec{x}) = 1 - p(C_1|\vec{x})$$

$\uparrow$ $O(\vec{x}^2)$ terms canceled

D=1 ex.:

Assume $p(C_1) = p(C_2) = \frac{1}{2}$




Here, $\Sigma^{-1} \Rightarrow \frac{1}{\sigma^2}$ ;

$$\begin{cases} w = \frac{\mu_1 - \mu_2}{\sigma^2}, \\ w_0 = \frac{1}{2\sigma^2}(\mu_2^2 - \mu_1^2) \end{cases} \Rightarrow a = \frac{(\mu_1-\mu_2)x}{\sigma^2} + \frac{1}{2\sigma^2}(\mu_2^2 - \mu_1^2).$$

$a = 0$ when $x = \frac{\mu_1+\mu_2}{2}$,

s.t. $\sigma(a) = \frac{1}{2}$.

DB: surfaces at which

$$p(C_1|\vec{x}) = p(C_2|\vec{x})$$

$$\sigma(\vec{w}^T\vec{x} + w_0) = \sigma(-\vec{w}^T\vec{x} - w_0) \quad \leftarrow 1 - \sigma(a) = \sigma(-a)$$

Then $\vec{w}^T\vec{x} + w_0 = -\vec{w}^T\vec{x} - w_0$,

$\vec{w}^T\vec{x} + w_0 = 0 \Leftarrow$ linear eq'n for DB

In our 1D example,

$\underbrace{w \cdot x + w_0}_{a} = 0$ is satisfied by

$x = \frac{\mu_1 + \mu_2}{2} \Leftarrow$ DB

In fact, all contours of $p(C_1|\vec{x})$ & $p(C_2|\vec{x})$ are given by functions linear in $\vec{x}$.

$\boxed{K > 2 \text{ case:}}$

$$a_k(\vec{x}) = -\frac{D}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(\vec{x}-\vec{\mu}_k)^T \Sigma^{-1}(\vec{x}-\vec{\mu}_k) +$$

$$+ \log p(C_k) =$$

$$= -\frac{1}{2}\vec{x}^T\Sigma^{-1}\vec{x} + (\overbrace{\Sigma^{-1}\vec{\mu}_k}^{\vec{w}_k})^T\vec{x} \overbrace{- \frac{1}{2}\vec{\mu}_k^T \Sigma^{-1}\vec{\mu}_k - \frac{D}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| + \log p(C_k)}^{w_{0,k}} =$$

$$= -\frac{1}{2}\vec{x}^T\Sigma^{-1}\vec{x} + \vec{w}_k\vec{x} + w_{0,k}.$$

DB's are still linear however:

$p(C_k|\vec{x}) = p(C_j|\vec{x})$  $[j \neq k]$

$\Downarrow$

$e^{a_k} = e^{a_j}$, or $a_k = a_j$:

$$\vec{w}_k^T\vec{x} + w_{0,k} = \vec{w}_j^T\vec{x} + w_{0,j}.$$

$O(\vec{x}^2)$ terms cancel

linear eq'n in $\vec{x}$, gives $D-1$ dimensional decision surface (DB)

Finally, if $p(\vec{x}|C_k) = \mathcal{N}(\vec{x}|\vec{\mu}_k, \Sigma_k)$, the DB's at which the 2 largest posterior probs are equal, are given by quadratic f's of $\vec{x}$:

2D