Linear $\times$ Non-linear

$\mathbf{x} = (x_1, x_2, \ldots, x_d)$
$\mathbf{w} = (w_1, w_2, \ldots, w_d) \in \mathbb{R}^d, \; b \in \mathbb{R}$

Linear function:

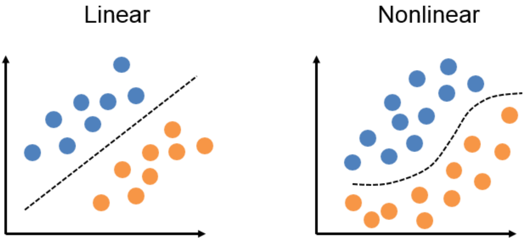
$$s = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d + b$$

Non-linear function – some examples:

$$s = w_1 x_1 + w_2 x_2 + w_3 x_1 x_2 + w_4 x_1^2 + w_5 x_2^2 + b$$
$$s = w_1 x_1^2 + w_2 x_2^2 + b$$

Any function $s : \mathbb{R}^d \rightarrow \mathbb{R}$ can be used for classification:

- $s < 0 \implies$ class := negative
- $s > 0 \implies$ class := positive
- $s = 0 \implies$ decision boundary

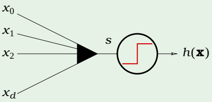


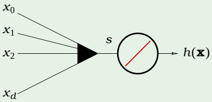
Fonte: https://jtsulliv.github.io/perceptron/

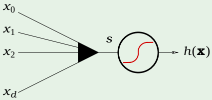
A third linear model

$$s = \sum_{i=0}^{d} w_i x_i$$

linear classification	linear regression	**logistic regression**
$h(\mathbf{x}) = \text{sign}(s)$	$h(\mathbf{x}) = s$	$h(\mathbf{x}) = \theta(s)$

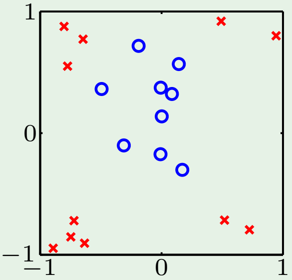




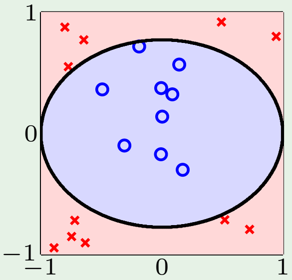


Linear is limited

Data:



Hypothesis:



Another example

Credit line is affected by 'years in residence'

but **not** in a linear way!

Nonlinear $[[x_i < 1]]$ and $[[x_i > 5]]$ are better.

Can we do that with linear models?

Linear in what?

Linear regression implements

$$\sum_{i=0}^{d} w_i \; x_i$$

Linear classification implements

$$\text{sign}\left(\sum_{i=0}^{d} w_i \; x_i\right)$$

Algorithms work because of **linearity in the weights**

$$f_2(x_1, x_2) = w_0 + w_1\, x_1^2 + w_2\, x_1\, x_2$$

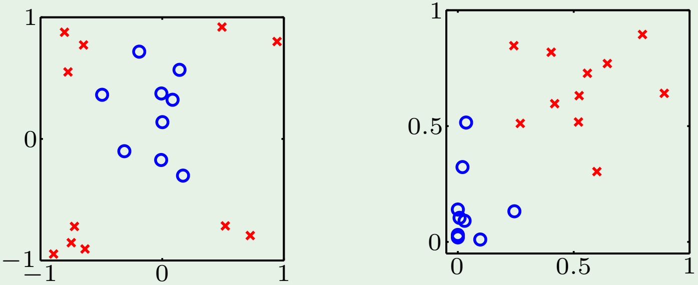
Non-linear with respect to x_i

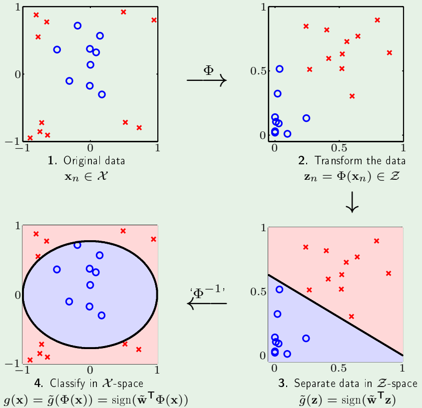
$$f_2(x_1, x_2) = w_0 + w_1\, x_1^2 + w_2\, x_1\, x_2$$

Linear with respect to w_i

Transform the data nonlinearly

$$(x_1, x_2) \xrightarrow{\Phi} (x_1^2, x_2^2)$$




Φ
1. Original data
$\mathbf{x}_n \in \mathcal{X}$
2. Transform the data
$\mathbf{z}_n = \Phi(\mathbf{x}_n) \in \mathcal{Z}$
'Φ^{-1}'
3. Separate data in $\mathcal{Z}$-space
$\tilde{g}(\mathbf{z}) = \text{sign}(\tilde{\mathbf{w}}^{\mathsf{T}}\mathbf{z})$
4. Classify in $\mathcal{X}$-space
$g(\mathbf{x}) = \tilde{g}(\Phi(\mathbf{x})) = \text{sign}(\tilde{\mathbf{w}}^{\mathsf{T}}\Phi(\mathbf{x}))$

Nonlinear transforms

$$\mathbf{x} = (x_0, x_1, \cdots, x_d) \quad \xrightarrow{\Phi} \quad \mathbf{z} = (z_0, z_1, \cdots\cdots\cdots\cdots, z_{\tilde{d}})$$

Each $z_i = \phi_i(\mathbf{x})$ $\qquad$ $\mathbf{z} = \Phi(\mathbf{x})$

Example: $\mathbf{z} = (1, x_1, x_2, x_1 x_2, x_1^2, x_2^2)$

Final hypothesis $g(\mathbf{x})$ in $\mathcal{X}$ space:

$\mathrm{sign}\left(\tilde{\mathbf{w}}^{\mathsf{T}}\Phi(\mathbf{x})\right)$ $\qquad$ or $\qquad$ $\tilde{\mathbf{w}}^{\mathsf{T}}\Phi(\mathbf{x})$

The price we pay

$$\mathbf{x} = (x_0, x_1, \cdots, x_d) \xrightarrow{\Phi} \mathbf{z} = (z_0, z_1, \cdots\cdots\cdots\cdots, z_{\tilde{d}})$$

$$\downarrow \qquad\qquad\qquad \downarrow$$

$$\mathbf{w} \qquad\qquad\qquad \tilde{\mathbf{w}}$$

$$d_{\text{VC}} = d + 1 \qquad\qquad d_{\text{VC}} \leq \tilde{d} + 1$$

- Linear models are simple but have limited ability to discriminate classes

- When non-linear transformation is applied on the data, it may become a poweful tool

  SVMs explore this fact in combination with kernel transfromations

- There are many non-linear algorithms

  Neural networks, decision trees, etc