





I N S E A







Machine learning classique

Intro to neural nets

En pratique, comment évaluer les fonctions pour des données complexes? Pourquoi à votre avis?



Idée:

$$z_1 \stackrel{\text{def}}{=} h(\mathbf{x}^\top \boldsymbol{\beta}^1 + \alpha_1)$$

$$\vdots$$

$$z_p \stackrel{\text{def}}{=} h(\mathbf{x}^\top \boldsymbol{\beta}^p + \alpha_p)$$

Linéarités

$$z_1 \stackrel{\text{def}}{=} \mathbf{x}^\top \boldsymbol{\beta}^1 + \alpha_1$$

$\vdots$

$$z_p \stackrel{\text{def}}{=} \mathbf{x}^\top \boldsymbol{\beta}^p + \alpha_p$$

non-linéarité

$$\max(z_1, \dots, z_p) \longrightarrow \text{sigmoid}$$

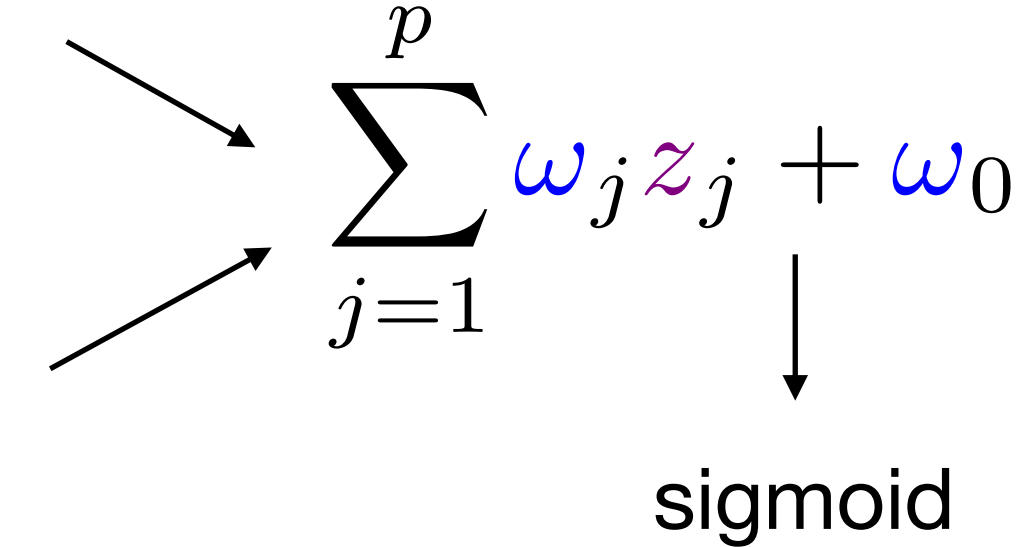


1. On n'utilise qu'une seule non-linéarité

2. Elle est fixée par la fonction ~~max~~: on ne l'apprend pas

1. Appliquer plusieurs non-linéarités *h* plus tôt

2. Combiner les  $z_j$  linéairement avec  $w_j$  à optimiser



The diagram illustrates the combination of features  $z_j$  and weights  $w_j$  into a weighted sum, followed by a sigmoid activation. Two arrows point from the text 'à optimiser' to the summation formula. A vertical arrow points from the formula to the word 'sigmoid'.

$$\sum_{j=1}^p \omega_j z_j + \omega_0$$

sigmoid



Il faut donc utiliser plusieurs **fonctions** linéaires complexes

Linéarités

$$\begin{array}{c}
 z_1 \stackrel{\text{def}}{=} \mathbf{x}^\top \boldsymbol{\beta}^1 + \alpha_1 \\
 \vdots \\
 z_p \stackrel{\text{def}}{=} \mathbf{x}^\top \boldsymbol{\beta}^p + \alpha_p
 \end{array}
 \begin{array}{c}
 \nearrow \\
 \nearrow \\
 \nearrow
 \end{array}
 \begin{array}{c}
 \text{non-linéarité} \\
 \text{max}(z_1, \dots, z_p) \longrightarrow \text{sigmoid}
 \end{array}$$

En pratique, ce modèle ne fonctionne pas pour ces données complexes. Pourquoi à votre avis ?

1. On n'utilise qu'une seule non-linéarité
2. Elle est fixée par la fonction **max**: on ne l'apprend pas

Il faudrait donc: utiliser plusieurs **non-linéarités simples** + les combiner pour apprendre des fonctions non-linéaires complexes

Idée:

1. Appliquer plusieurs non-linéarités  $h$  plus tôt
2. Combiner les  $z_j$  linéairement avec  $w_j$  à optimiser

$$\begin{array}{c}
 z_1 \stackrel{\text{def}}{=} h(\mathbf{x}^\top \boldsymbol{\beta}^1 + \alpha_1) \\
 \vdots \\
 z_p \stackrel{\text{def}}{=} h(\mathbf{x}^\top \boldsymbol{\beta}^p + \alpha_p)
 \end{array}
 \begin{array}{c}
 \nearrow \\
 \nearrow
 \end{array}
 \sum_{j=1}^p \omega_j z_j + \omega_0
 \downarrow
 \text{sigmoid}$$



$$\begin{array}{l} z_1 \stackrel{\text{def}}{=} h(\mathbf{x}^\top \boldsymbol{\beta}^1 + \alpha_1) \\ \vdots \\ z_p \stackrel{\text{def}}{=} h(\mathbf{x}^\top \boldsymbol{\beta}^p + \alpha_p) \end{array} \begin{array}{l} \searrow \\ \nearrow \end{array} \text{sigmoid}\left(\sum_{j=1}^p \omega_j z_j + \omega_0\right)$$

