



I N S E A







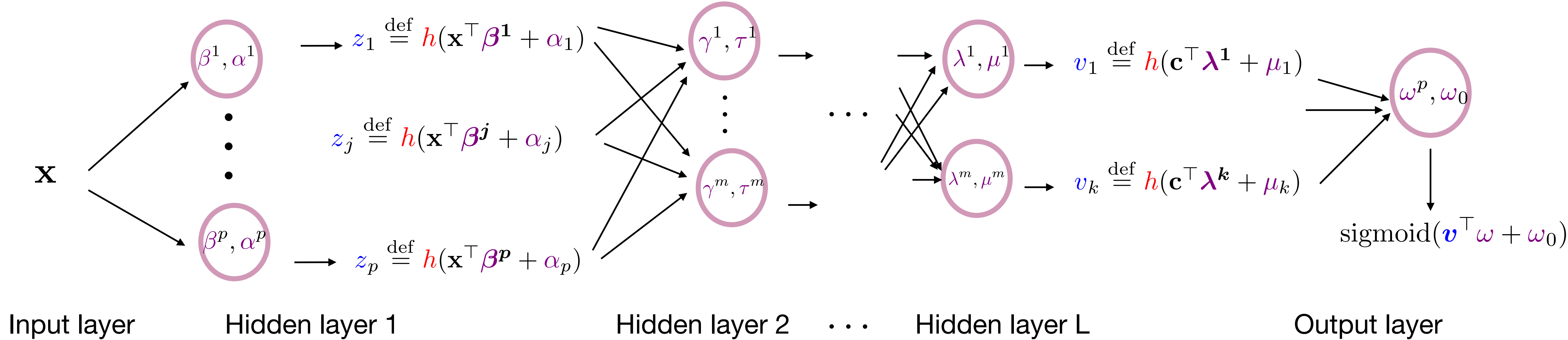
3

6

Machine learning clasique

Representation learning

On peut augmenter la complexité du modèle à l'infini...



On considère un problème de classification binaire avec le résultat ci-dessus optimisé avec une très bonne performance

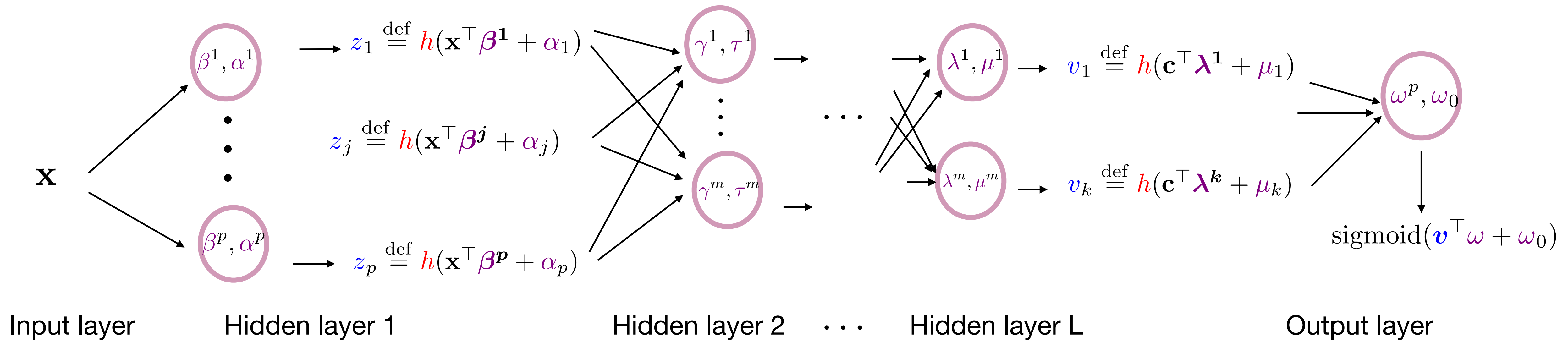
On définit la transformation des données en s'arrêtant à l'avant dernier layer: $g: \mathbf{x} \in \mathbb{R}^d \mapsto \mathbf{v} \in \mathbb{R}^k$

Apprendre à classer les $g(\mathbf{x}_i)$ est-il plus facile ou plus difficile que classer les \mathbf{x}_i ?

Plus facile car un neurone (ou petit) a suffisamment linéairement séparables

$g(\mathbf{x}_i)$ est une *embedding* ou une *représentation vectorielle* de \mathbf{x}_i

On peut augmenter la complexité du modèle à l'infini...



On considère un problème de classification binaire avec le réseau ci-dessus optimisé avec une très bonne performance.

On définit la transformation des données en s'arrêtant à l'avant dernier layer: $g : \mathbf{x} \in \mathbb{R}^d \mapsto \mathbf{v} \in \mathbb{R}^k$

Apprendre à classer les $g(\mathbf{x}_i)$ est-il plus facile ou plus difficile que classer les \mathbf{x}_i ?

Plus **facile**: car un seul neurone (output) a suffit pour les classer: ils sont **forcément** linéairement séparables

$g(\mathbf{x}_i)$ est un *embedding* ou une *représentation vectorielle* de \mathbf{x}_i

Chapitre 3. Applications et thématiques avancées

1. Modèles Bayésiens hiérarchiques (Assurance / Biostats)
2. Classical Machine learning: zero to hero
3. Bayesian Machine learning