



Réseaux de Neurones: Appliqués à des données réelles.

Cette présentation explore les réseaux de neurones, des modèles inspirés du cerveau humain, et leur application à des données réelles. Nous détaillerons le rôle crucial de la fonction sigmoïde dans leur fonctionnement et les défis rencontrés.

Introduction aux Réseaux de Neurones

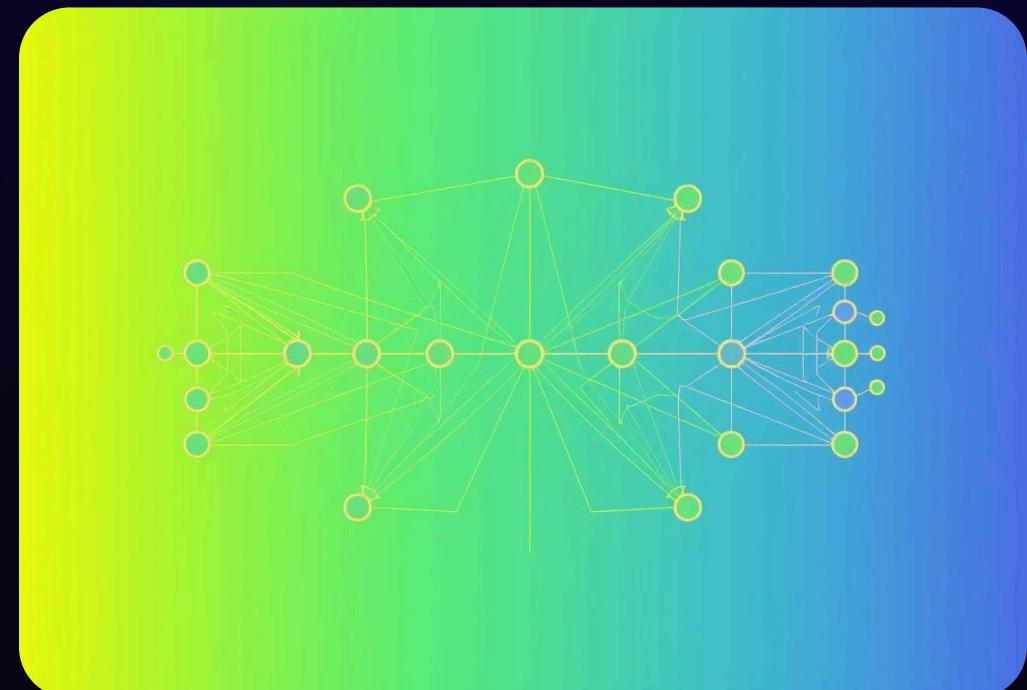


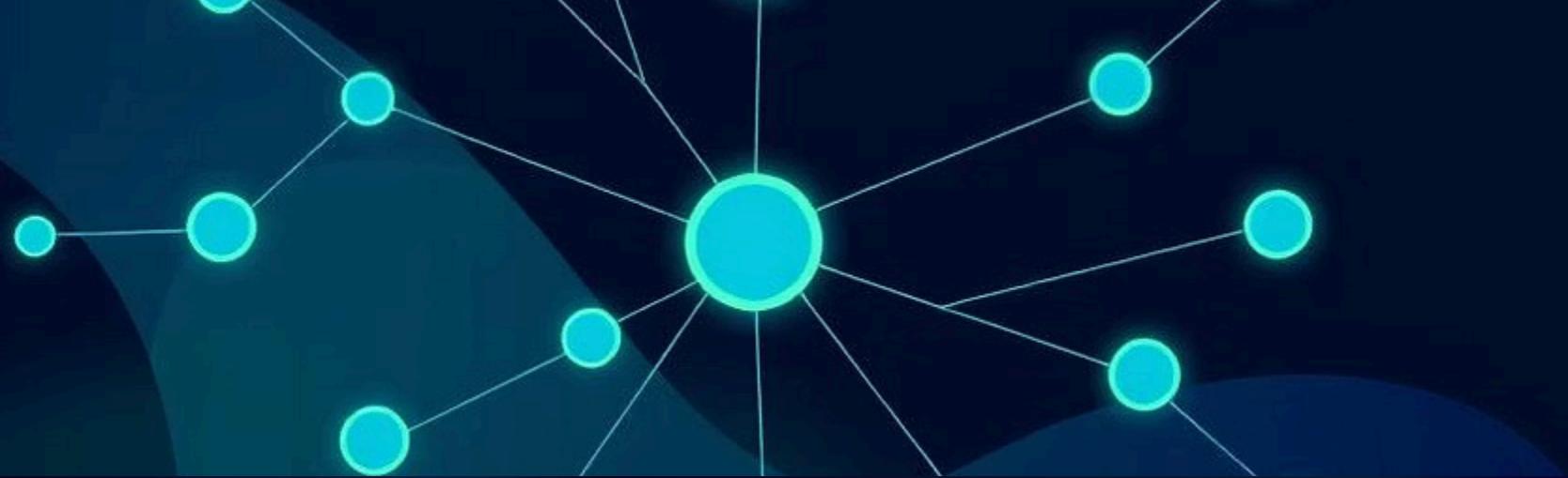
Pourquoi les Réseaux de Neurones ?

Ils modélisent des relations non linéaires complexes. Ils apprennent automatiquement des caractéristiques à partir des données. Ils sont le fondement du Deep Learning moderne.

Jalons Historiques

- **1957** : Invention du Perceptron pour la classification binaire.
- **Années 1980** : Avènement du Perceptron multicouche et de la rétropropagation.
- **Aujourd'hui** : Réseaux profonds traitant des milliards de paramètres.





Architecture d'un Réseau de Neurones Simple



Neurones

Unités fondamentales qui reçoivent et transforment les signaux.



Couches

Entrée (données brutes), Cachées (représentations complexes), Sortie (résultat final).



Poids et Biais

Paramètres modifiables essentiels à l'apprentissage du réseau.

Chaque neurone calcule $y = f(\sum_i w_i x_i + b)$, où f est la fonction d'activation, introduisant la non-linéarité.

Fonction d'activation : focus sur la sigmoïde

La fonction sigmoïde est utilisée pour transformer la sortie d'un neurone en une valeur entre 0 et 1.

C'est très utile quand on fait de la classification binaire (ex. malade / pas malade).

Formule :

$$\text{sigmoïde}(x) = 1 / (1 + e^{-x})$$

Elle permet de normaliser la sortie du neurone.

Limites et Évolution de la Sigmoïde

Saturation aux Extrémités

Lorsque x est très grand ou très petit, la dérivée $\sigma'(x)$ est proche de 0. Cela ralentit fortement l'apprentissage (problème du vanishing gradient).

Sortie Toujours Positive

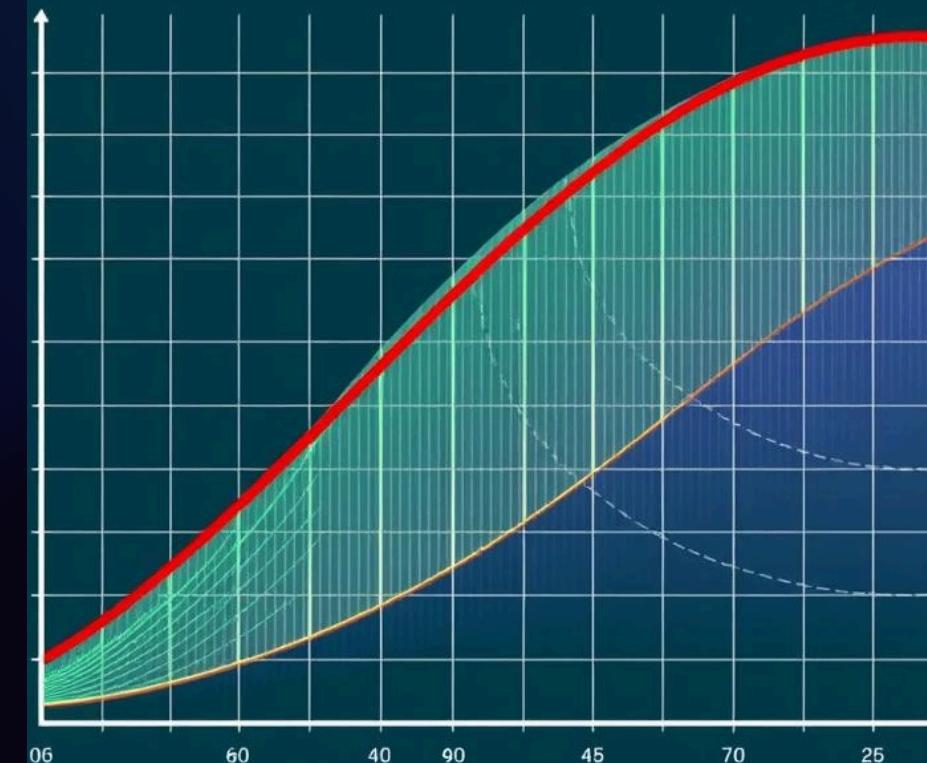
La sortie positive peut entraver la convergence du réseau, rendant l'optimisation moins efficace.

Utilisation Actuelle

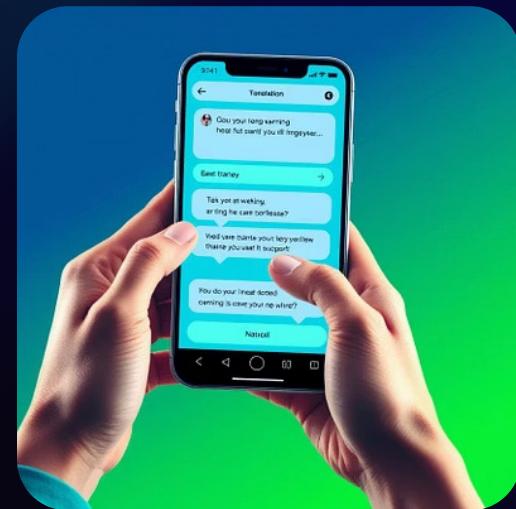
Principalement utilisée en couche de sortie pour la classification binaire. D'autres fonctions comme ReLU dominent dans les couches cachées.

Varaincent Garature Grrdlem

Gradien trr mlets ad nereal cropet coity



Applications Réelles des Réseaux de Neurones



Les réseaux de neurones transforment de nombreux secteurs. Ils extraient des caractéristiques pertinentes, même de données non structurées. Leur flexibilité permet une adaptation à des domaines variés.

Entraînement d'un Réseau de Neurones



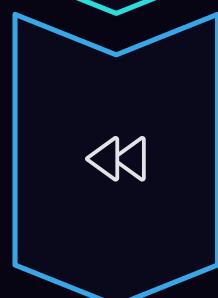
Propagation Avant

Calcul des sorties du réseau à partir des entrées initiales.



Fonction de Coût

Mesure de l'erreur globale entre la prédiction et le résultat attendu.



Rétropropagation

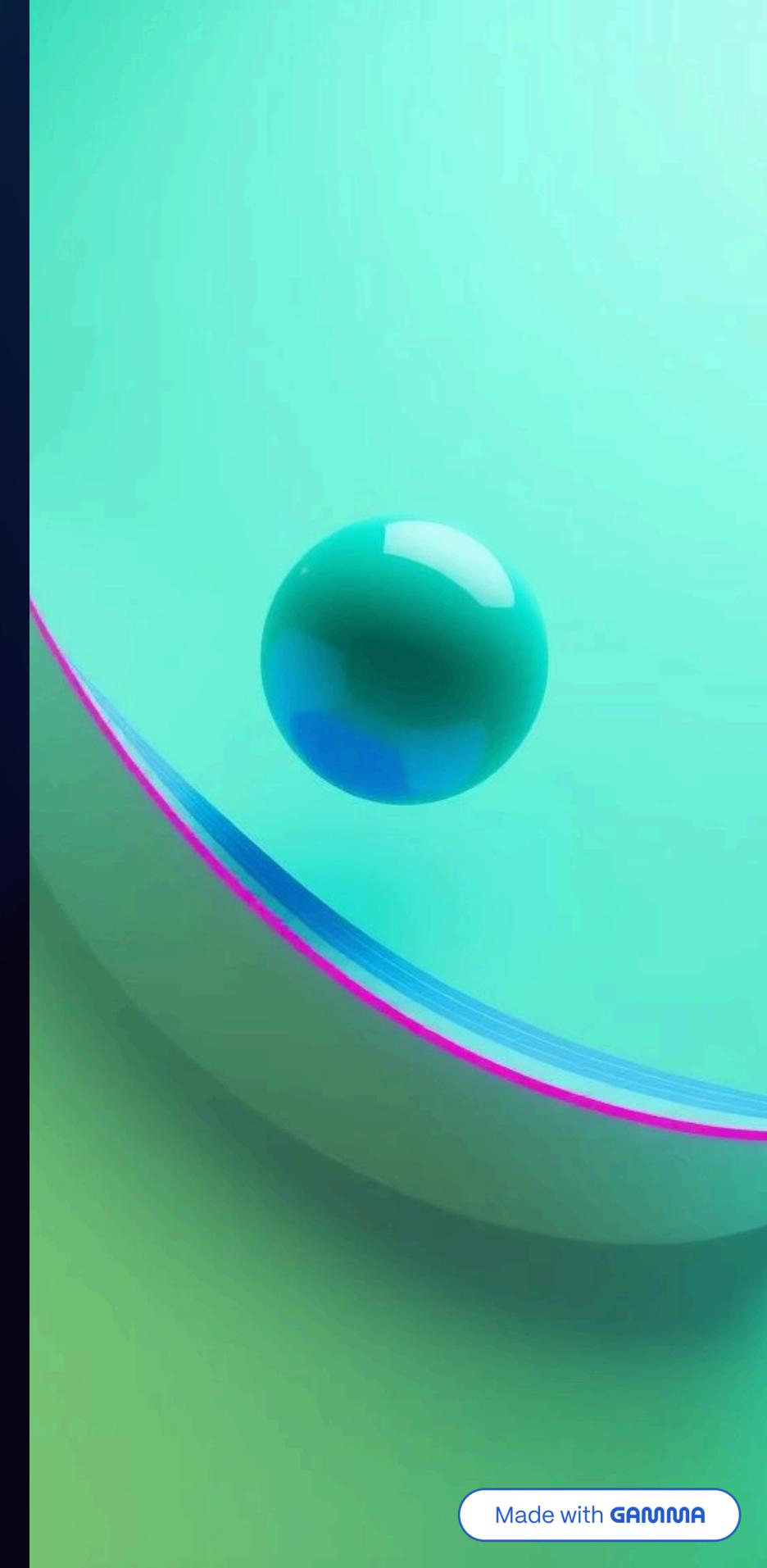
Calcul des gradients (utilisant la dérivée de la sigmoïde) pour ajuster les paramètres.



Descente de Gradient

Mise à jour progressive des poids et des biais pour minimiser l'erreur.

La différentiabilité de la fonction sigmoïde est cruciale pour la rétropropagation. Des dérivées proches de zéro peuvent cependant ralentir ou bloquer l'apprentissage.





Conclusion : Points Clés à Retenir

1 Modélisation Inspirée du Cerveau

Les réseaux de neurones offrent une capacité de modélisation puissante.

2 Fonction Sigmoïde

Une activation classique, simple et utile pour la classification binaire. Elle a des limites, mais reste fondamentale pour la compréhension.

3 Entraînement Crucial

L'entraînement par rétropropagation ajuste les paramètres via les dérivées de la sigmoïde.

4 Applications Multiples

Ces concepts sont à la base de nombreuses innovations en santé, finance, et vision par ordinateur.

Démonstration Pratique : Python & TensorFlow