Le B.A.BA sur les Réseaux de Neurones

Cours « Recherche d'Information et Graphes de Données »

Nicolas Delestre

INSA ROUEN NORMANDIE

Aides de mes amis imaginaires

- Ce cours a été rédigé en LaTeX sous Emacs avec le mode Copilot
- J'ai très souvent échangé avec ChatGPT (version 5.2) et Gemini (version 3) pour présenter les concepts de ce cours plus clairement

INSA ROUEN NORMANDIE

Définition du *Machine Learning*

Qu'est-ce que le *Machine Learning* ?

Le Machine Learning (apprentissage automatique) est une branche de l'intelligence artificielle qui permet aux systèmes d'apprendre et de s'améliorer automatiquement à partir de données, sans être explicitement programmés pour chaque tâche spécifique

Trois grands types d'apprentissage

Apprentissage supervisé le modèle apprend à partir de données étiquetées
par exemple reconnaître des images de chats vs chiens

Apprentissage non supervisé le modèle découvre des structures dans des données non étiquetées, il regroupe des données similaires
par exemple regrouper des clients selon leurs comportements d'achat

Apprentissage par renforcement le modèle apprend à prendre des décisions en interagissant avec un environnement
par exemple apprendre à jouer à un jeu vidéo

Avantages et inconvénients du *Machine Learning*

Avantages

- Capacité à traiter des données très diverses (images, texte, son, vidéo, capteurs, etc.)
- Adaptabilité à des tâches variées
- Découverte de patterns complexes non évidents pour les humains

Inconvénients

- Besoin de grandes quantités de données d'entraînement
- Risque de surapprentissage (*overfitting*)
- Difficulté à interpréter les décisions du modèle (boîte noire)

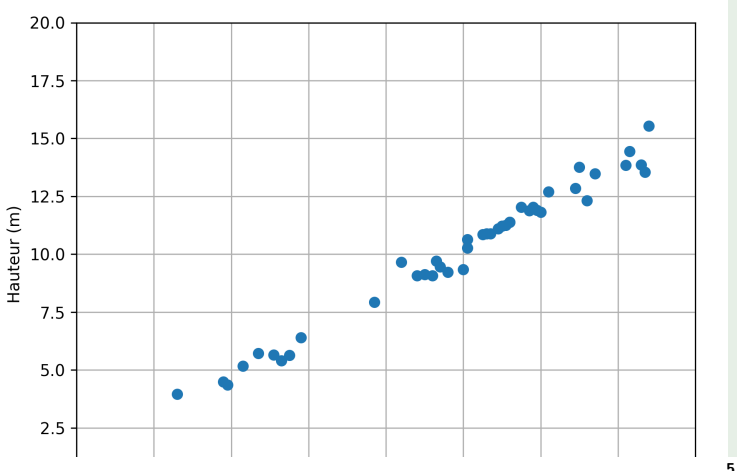
INSA ROUEN NORMANDIE

Principe de l'apprentissage supervisée à partir d'un exemple [a]

a. Science étonnante https://scienceetonnante.substack.com/p/chatgpt-va-t-il-chercher-ses-reponses

Connaître la hauteur d'un arbre à partir de son diamètre

	A	B
1	Diamètre (cm)	Hauteur (m)
2	101	10.4
3	129	12.9
4	26	3.6
5	130	13
6	101	10.4
7	58	6.5
8	84	8.9
9	115	11.7
10	38	4.7
11	143	14.2
12	51	5.9
13	142	14.1
14	134	13.4
15	100	10.3
16	132	13.2
17	120	12.1
18	92	9.6

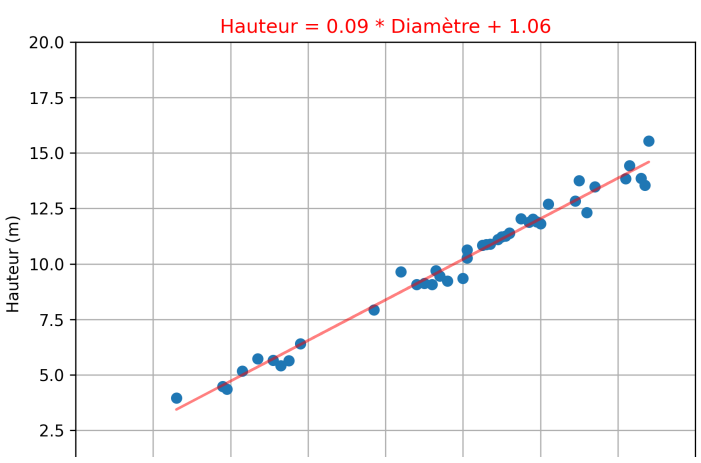


Principe de l'apprentissage supervisée à partir d'un exemple [a]

a. Science étonnante https://scienceetonnante.substack.com/p/chatgpt-va-t-il-chercher-ses-reponses

Connaître la hauteur d'un arbre à partir de son diamètre

	A	B
1	Diamètre (cm)	Hauteur (m)
2	101	10.4
3	129	12.9
4	26	3.6
5	130	13
6	101	10.4
7	58	6.5
8	84	8.9
9	115	11.7
10	38	4.7
11	143	14.2
12	51	5.9
13	142	14.1
14	134	13.4
15	100	10.3
16	132	13.2
17	120	12.1
18	92	9.6



Deux types d'apprentissage supervisé

Classification

- Tâche : attribuer une étiquette à une entrée
- Exemple : reconnaissance d'images (chat vs chien)
- Modèles courants : k-ppv, arbres de décision, SVM, réseaux de neurones

Régression

- Tâche : prédire une valeur continue à partir d'une entrée
- Exemple : prédiction de la hauteur d'un arbre à partir de son diamètre
- Modèles courants : régression linéaire, régression polynomiale, réseaux de neurones

INSA ROUEN NORMANDIE

Paramètres et hyper-paramètres

Paramètres

- Variables internes du modèle ajustées durant l'entraînement
- Exemple : coefficients de la régression linéaire
- Objectif : minimiser l'erreur sur les données d'entraînement

Hyper-paramètres

- Paramètres externes définis avant l'entraînement
- Exemple : degrés de la régression polynomiale
- Objectif : optimiser la performance du modèle sur des données non vues

INSA ROUEN NORMANDIE

Utilisation des données

Données d'entraînement

- Ensemble de données utilisé pour ajuster les paramètres du modèle
- Doit être représentatif de la tâche à accomplir
- Exemple : images étiquetées pour la classification

Données de validation et de test

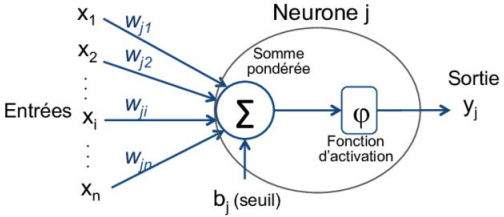
- Validation : ensemble de données utilisé pour choisir les meilleurs hyper-paramètres
- Test : ensemble de données utilisé pour évaluer la performance finale du modèle
- Importance : éviter le surapprentissage et garantir la généralisation

INSA ROUEN NORMANDIE

Réseaux de neurones artificiels

Inspiration biologique

- Modélisation simplifiée du fonctionnement des neurones biologiques
- Composé de couches de neurones artificiels interconnectés
- Chaque neurone applique une fonction d'activation sur une combinaison pondérée de ses entrées



Quelques fonctions d'activation φ courantes

- Identité : $\varphi(x) = x$
- Sigmoïde : $\varphi(x) = \frac{1}{1+e^{-x}}$
- ReLU : $\varphi(x) = \max(0, x)$
- Tanh : $\varphi(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

Apprentissage de w_{ji} et b_j d'un neurone j

Descente de gradient

- On pourrait penser à une régression linéaire, mais impossible dans la plupart des cas, à cause de la fonction d'activation (non linéaire)
- Méthode itérative de la descente de gradient :
  - On fixe des valeurs aléatoires aux w_{ji} et b_j
  - On présente les données d'apprentissage et on calcule la fonction de perte $\mathcal{L}$ (par exemple erreur quadratique moyenne) :

$$\mathcal{L}(w_{j1}, \ldots, w_{jn}, b_j) = \frac{1}{N} \sum_{i=1}^{N} (\overbrace{\varphi(\underbrace{b_j + \sum_{i=1}^{n} w_{ji} x_i}_{z_i})}^{\hat{y}_i} - y_i)^2$$

  - On ajuste les valeurs de w_{ji} et b_j en leur ajoutant la valeur de la dérivée partielle multipliée par un pas η :

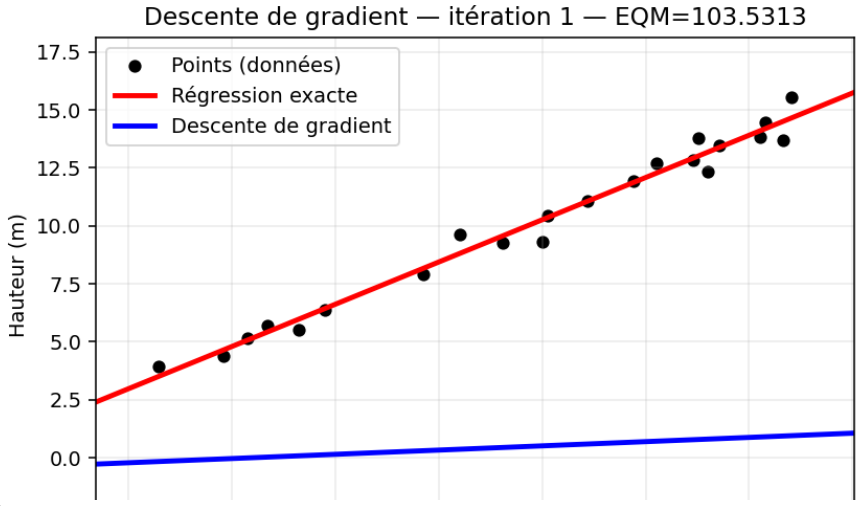
$$w_{ji} \leftarrow w_{ji} + \eta * \frac{\partial \mathcal{L}(w_{j1}, \ldots, w_{jn}, b_j)}{\partial w_{ji}}$$

Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes

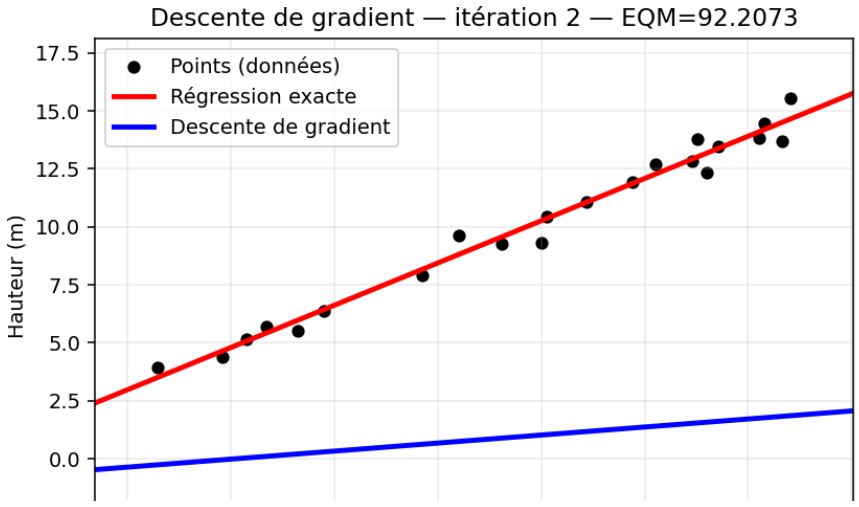
- $\hat{y} = ax + b$
- $\varphi = id$
- $\mathcal{L}(a, b) = \frac{1}{N} \sum_1^N (ax_i + b - y_i)^2$
- $\frac{\partial \mathcal{L}(a,b)}{\partial a} = \frac{2}{N} \sum_1^N x_i(ax_i + b - y_i)$
- $\frac{\partial \mathcal{L}(a,b)}{\partial b} = \frac{2}{N} \sum_1^N (ax_i + b - y_i)$

INSA ROUEN NORMANDIE

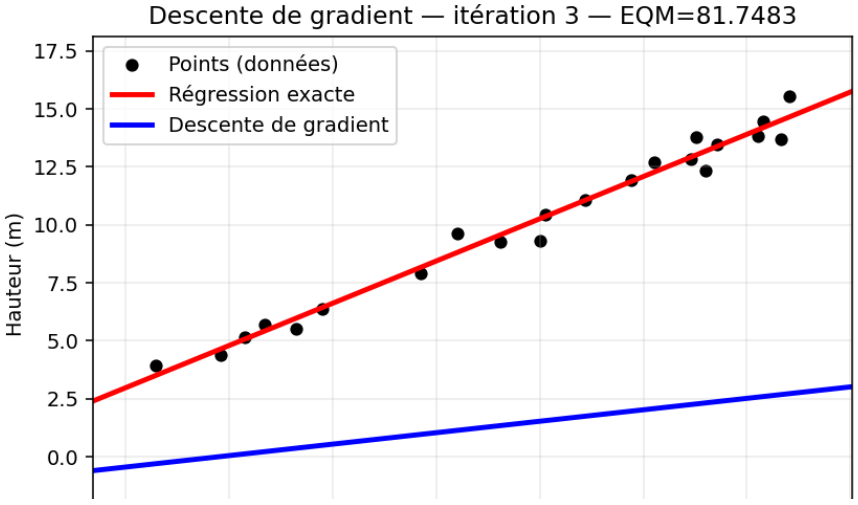
Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



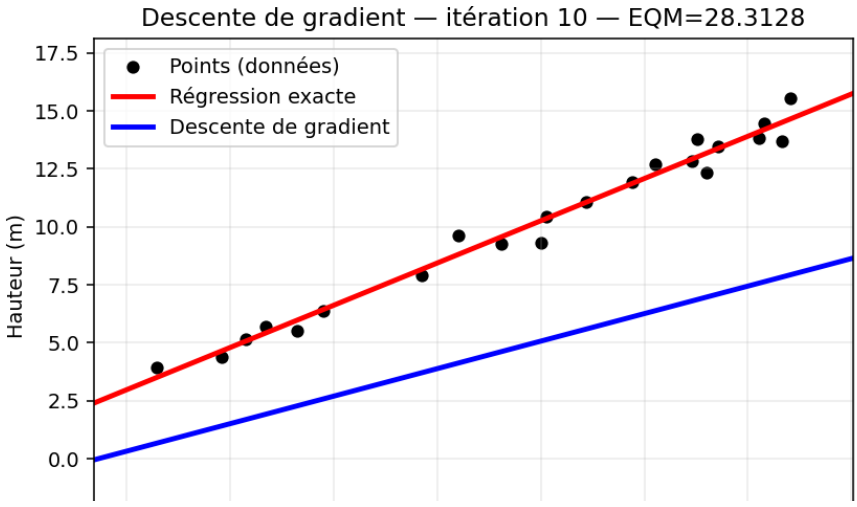
Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



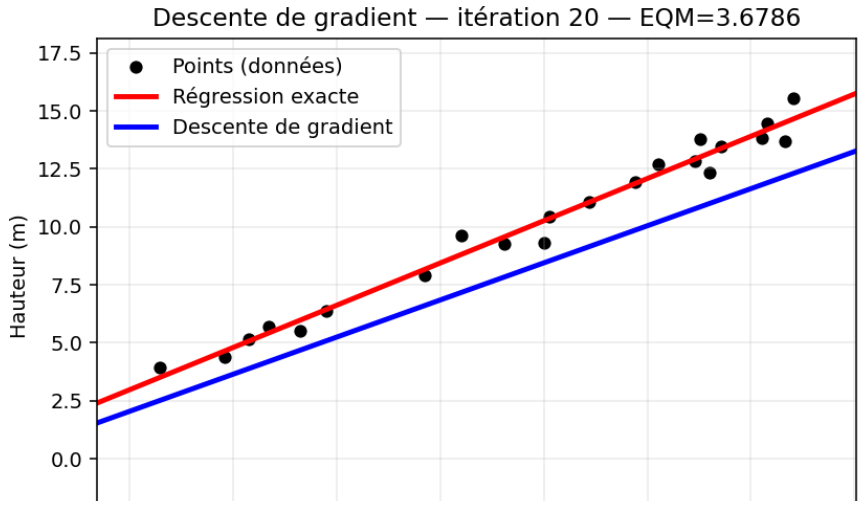
Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



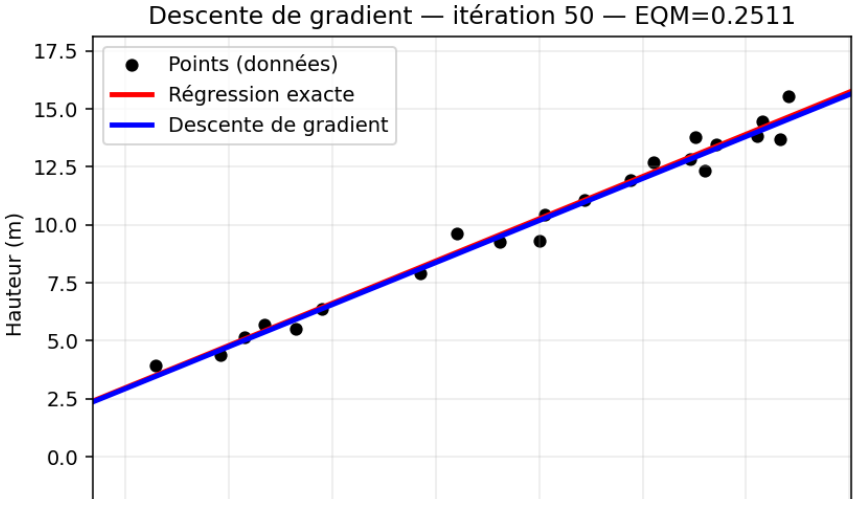
Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



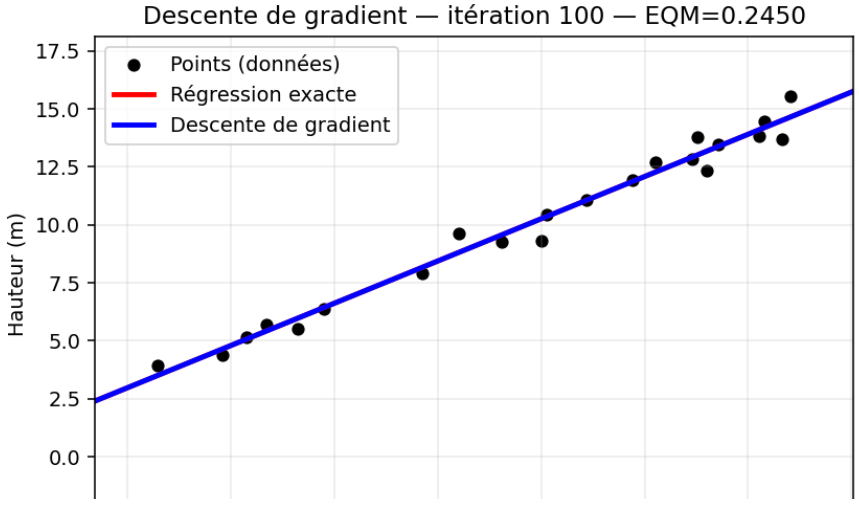
Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



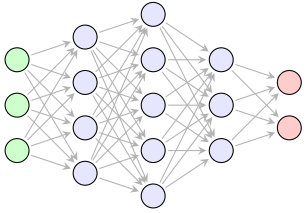
Exemple d'application de la descente de gradient avec une fonction d'activation *id* sur les données précédentes



Réseaux de neurones (MLP) 1 / 3

Architecture

- Composé de plusieurs couches de neurones
- La première couche est la couche d'entrée (les x_i), la dernière la couche de sortie (les $\hat{y}_i$), les autres sont des couches cachées $a_i^{(k)}$ (*i-ième* neurone de la *k-ième* couche)
- Les sorties des neurones d'une couche $k-1$ sont connectées aux entrées de tous les neurones de la couche k pondérées par des poids w_{ji}^k (poids entre le neurone i de la couche $k-1$ et le neurone j de la couche k)



INSA ROUEN NORMANDIE

Réseaux de neurones (MLP) 2 / 3

Utilisation : feed forward

- On présente les données d'entrée x à la couche d'entrée
- Chaque neurone j d'une couche k calcule sa sortie $a_j^{(k)}$ en appliquant la fonction d'activation φ sur l'addition d'un biais $b_j^{(k)}$ avec la somme pondérée de ses entrées $\sum_i w_{ji}^{(k)} a_i^{(k-1)}$:

$$a_j^{(k)} = \varphi\left(b_j^{(k)} + \sum_i w_{ji}^{(k)} a_i^{(k-1)}\right)$$

- On obtient ainsi la sortie finale $\hat{y}$ du réseau
- Dans le cas d'une classification, très souvent $\hat{y}$ est une distribution de probabilité obtenue par l'utilisation de la fonction *softmax* :

$$\hat{y}_j = \frac{e^{a_j^{(K)}}}{\sum_i e^{a_i^{(K)}}}$$

Réseaux de neurones (MLP) 3 / 3

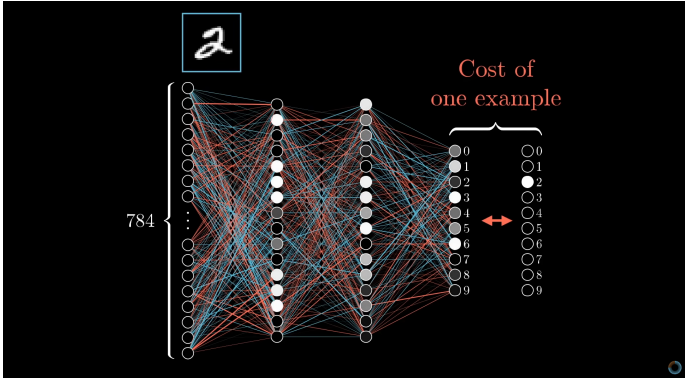
Implémentation fondée sur la multiplication de matrices

- Les couches sont représentées par des vecteurs $a^{(k)}$ de taille $d^{(k)}$
- Les poids des connexions entre deux couches $k-1$ et k sont représentés par des matrices $W^{(k)}$ de taille $d^{(k)} \times d^{(k-1)}$
- Les biais sont représentés par des vecteurs $b^{(k)}$
- La mise à jour des couches se fait par multiplication de matrices, addition de vecteurs et application de la fonction d'activation :

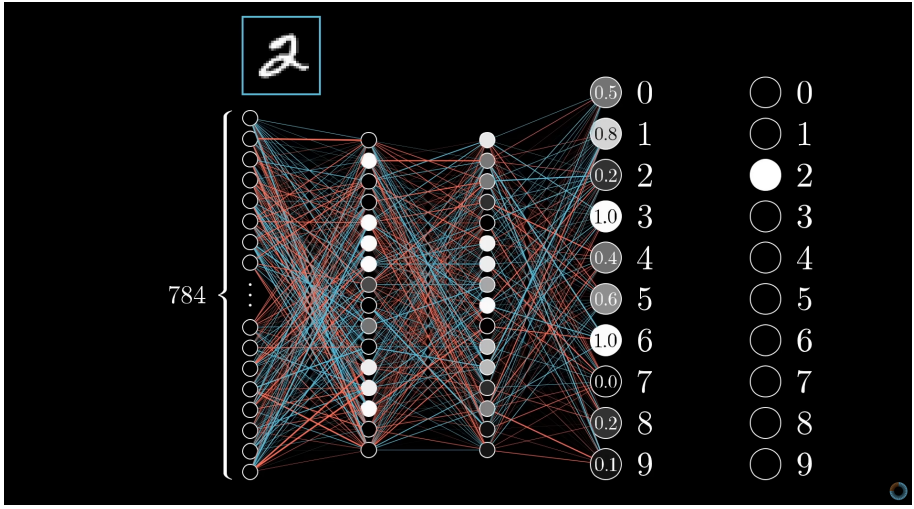
$$a^{(k)} = \varphi\left(b^{(k)} + W^{(k)} a^{(k-1)}\right)$$

INSA ROUEN NORMANDIE

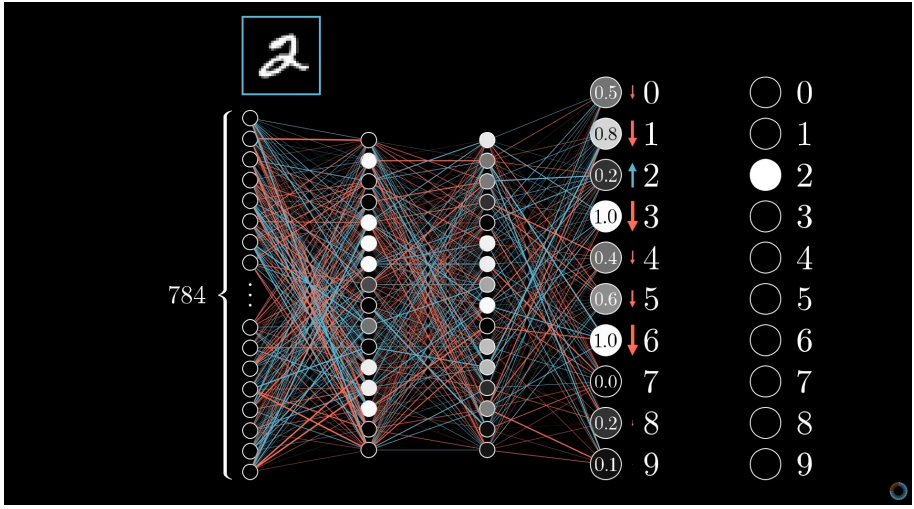
Apprentissage d'un MLP (rétro propagation) par l'exemple 1 / 10



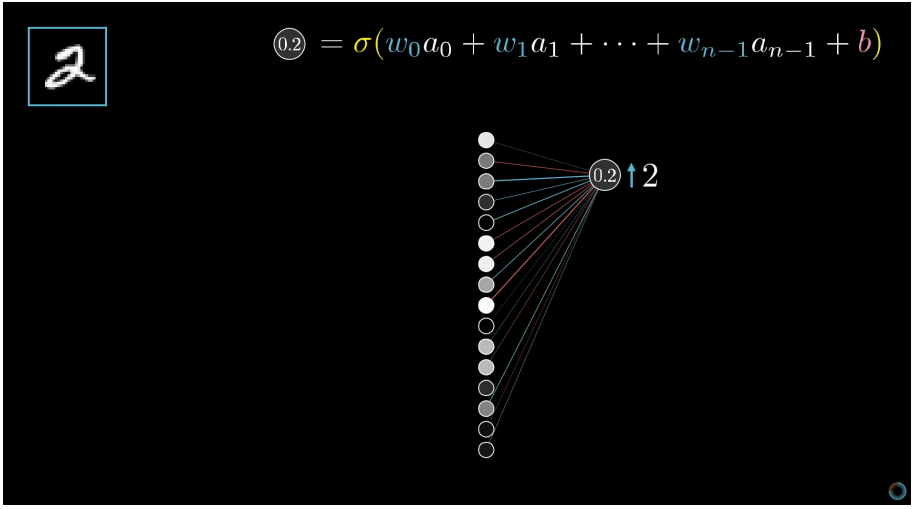
Apprentissage d'un MLP (rétro propagation) par l'exemple 2 / 10



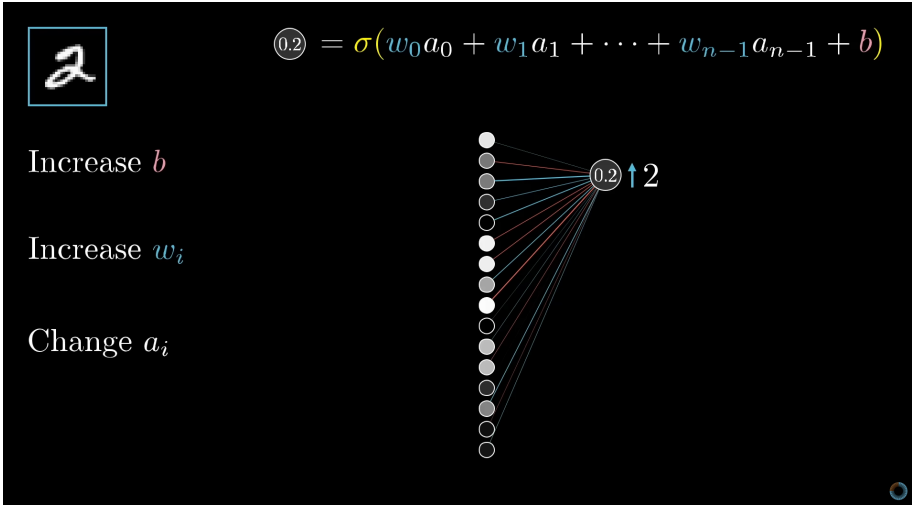
Apprentissage d'un MLP (rétro propagation) par l'exemple 3 / 10



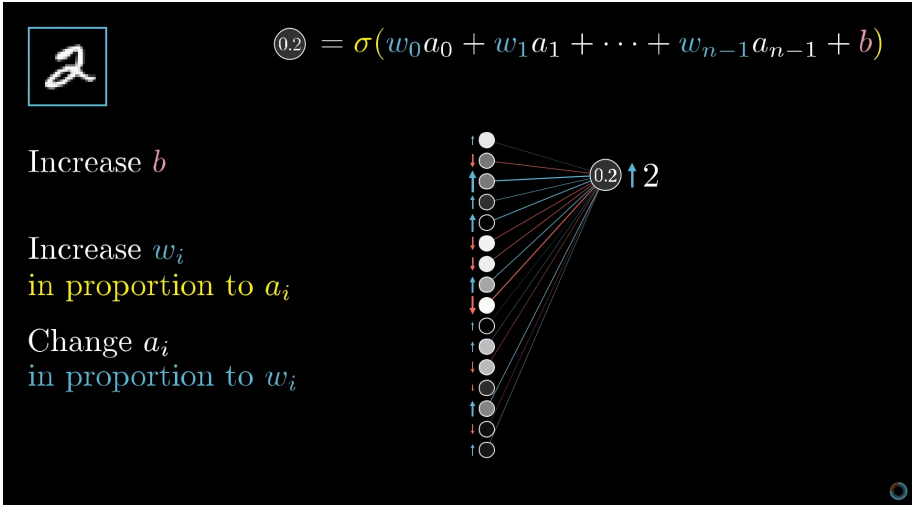
Apprentissage d'un MLP (rétro propagation) par l'exemple 4 / 10



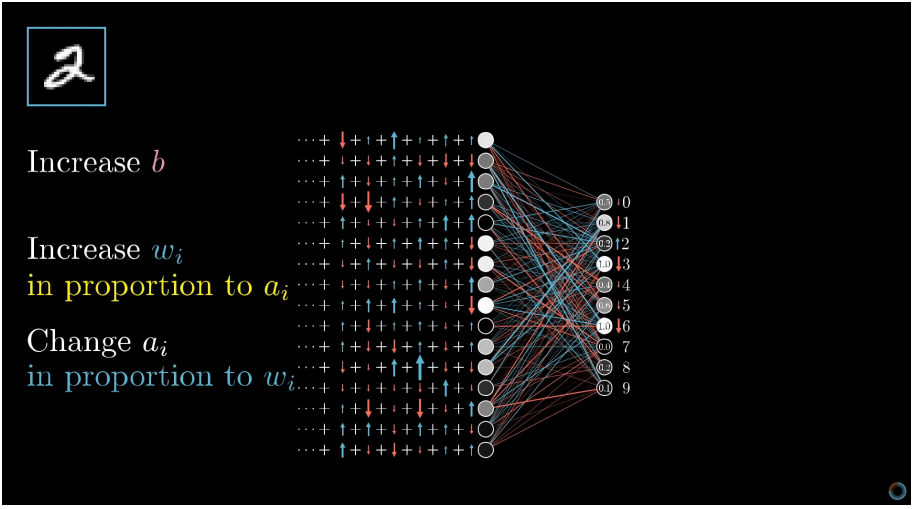
Apprentissage d'un MLP (rétro propagation) par l'exemple 5 / 10



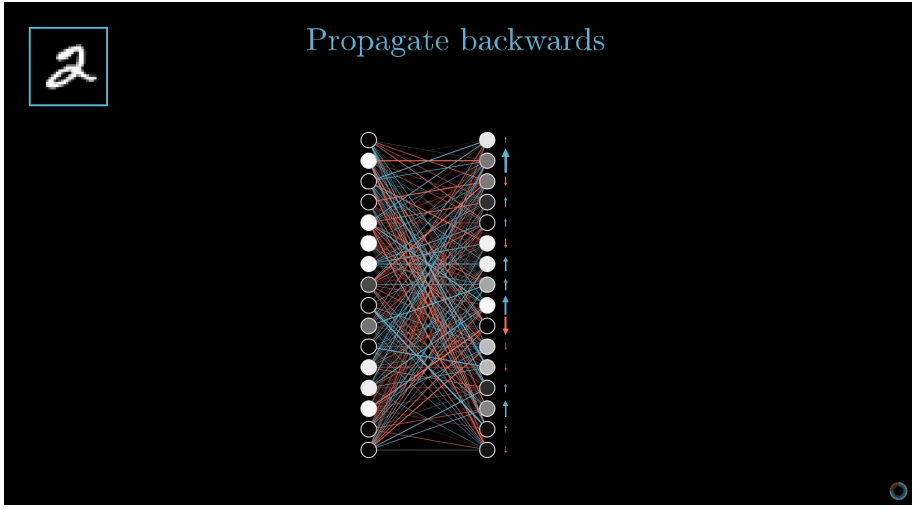
Apprentissage d'un MLP (rétro propagation) par l'exemple 6 / 10



Apprentissage d'un MLP (rétro propagation) par l'exemple 7 / 10



Apprentissage d'un MLP (rétro propagation) par l'exemple 8 / 10

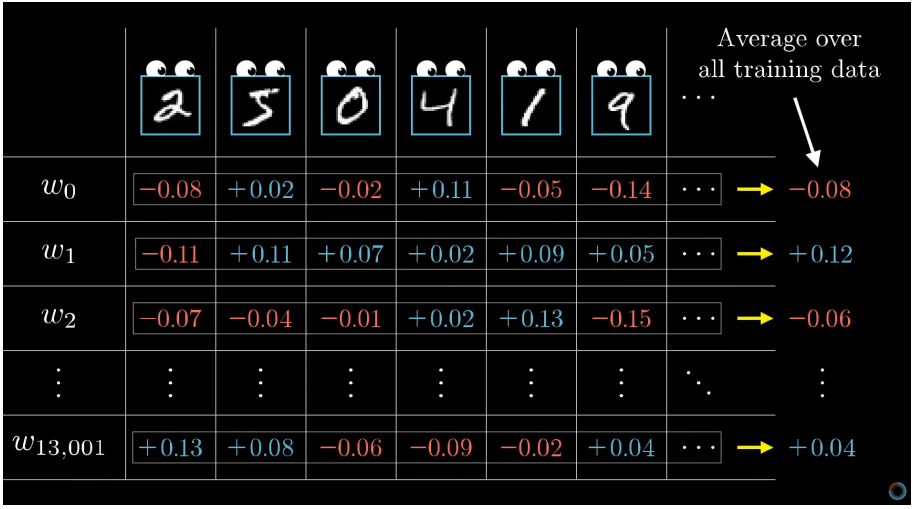


Apprentissage d'un MLP (rétro propagation) par l'exemple 9 / 10

							$\cdots$
w_0	-0.08	$+0.02$	-0.02	$+0.11$	-0.05	-0.14	$\cdots$
w_1	-0.11	$+0.11$	$+0.07$	$+0.02$	$+0.09$	$+0.05$	$\cdots$
w_2	-0.07	-0.04	-0.01	$+0.02$	$+0.13$	-0.15	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$
$w_{13,001}$	$+0.13$	$+0.08$	-0.06	-0.09	-0.02	$+0.04$	$\cdots$

INSA ROUEN NORMANDIE

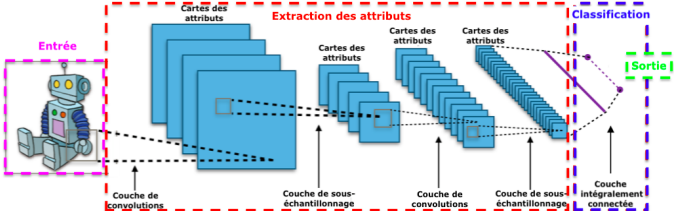
Apprentissage d'un MLP (rétro propagation) par l'exemple 10 / 10



Autres architectures principales de réseaux de neurones 1 / 3

Réseaux de neurones convolutifs (CNN)

- **Type de données en entrée :** un tenseur d'ordre 3 ($\mathbb{R}^{H \times L \times C}$) représentant des images, champs 2D ou 3D, données structurées spatialement
- **Caractéristiques :** couches de convolutions locales (application d'un même petit filtre sur toute l'image pour détecter des motifs) exploitant la structure spatiale des données
- **Usages typiques :** extraction automatique de caractéristiques
- **Exemples :** vision par ordinateur, imagerie médicale, analyse de champs de contraintes

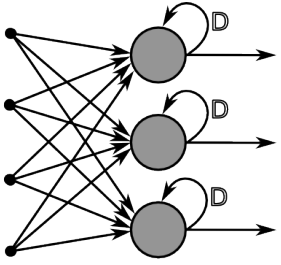


(Wikipédia)

Autres architectures principales de réseaux de neurones 2 / 3

Réseaux de neurones récurrents (RNN)

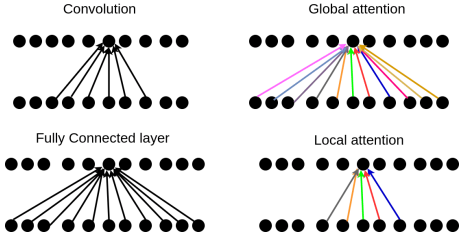
- **Type de données :** une suite de vecteurs de $\mathbb{R}^n$, n est la taille de la couche d'entrées. Les vecteurs sont présentés les uns après les autres, représentant des séquences et données temporelles
- **Caractéristique :** connexions récurrentes : l'état interne du réseau (issu du pas précédent) est réutilisé comme entrée lors du traitement de la donnée suivante
- **Usages typiques :** modélisation de dépendances temporelles
- **Exemples :** séries temporelles, signaux, systèmes dynamiques



Autres architectures principales de réseaux de neurones 3 / 3

Transformers

- **Type de données :** matrice ($\mathbb{R}^{k \times n}$, k longueur maximale des séquences et n la taille des données), représentant des séquences longues et complexes
- **Caractéristique :** lors de l'inférence (calcul du $\hat{y}$), l'influence entre les éléments de la séquence n'est pas fixée à l'avance : elle dépend à la fois des paramètres appris lors de l'apprentissage et des données x utilisées à l'inférence (mécanisme d'attention)
- **Usages typiques :** traitement efficace de longues dépendances
- **Exemples :** modèles de langage (LLM), traduction automatique, analyse de texte



https://theaisummer.com/attention/

Conclusion

- Le Machine Learning permet aux systèmes d'apprendre à partir de données
- L'apprentissage supervisé utilise des données étiquetées pour entraîner des modèles de classification ou de régression
- Les réseaux de neurones artificiels sont des modèles inspirés du cerveau humain, composés de couches de neurones interconnectés
- Différentes architectures de réseaux de neurones (*feed forward*, convolutifs, récurrents, transformers)
- Deux de ces types de réseaux de neurones (MLP et Transformers) sont très utilisés pour les modèles d'embedding de texte, les IA génératives de texte (LLM) et les RAG

INSA ROUEN NORMANDIE