



Applied QML

Lecture 9: Quantum Neural Networks

Christophe Pere

2024-03-14



Table des matières

NN classique

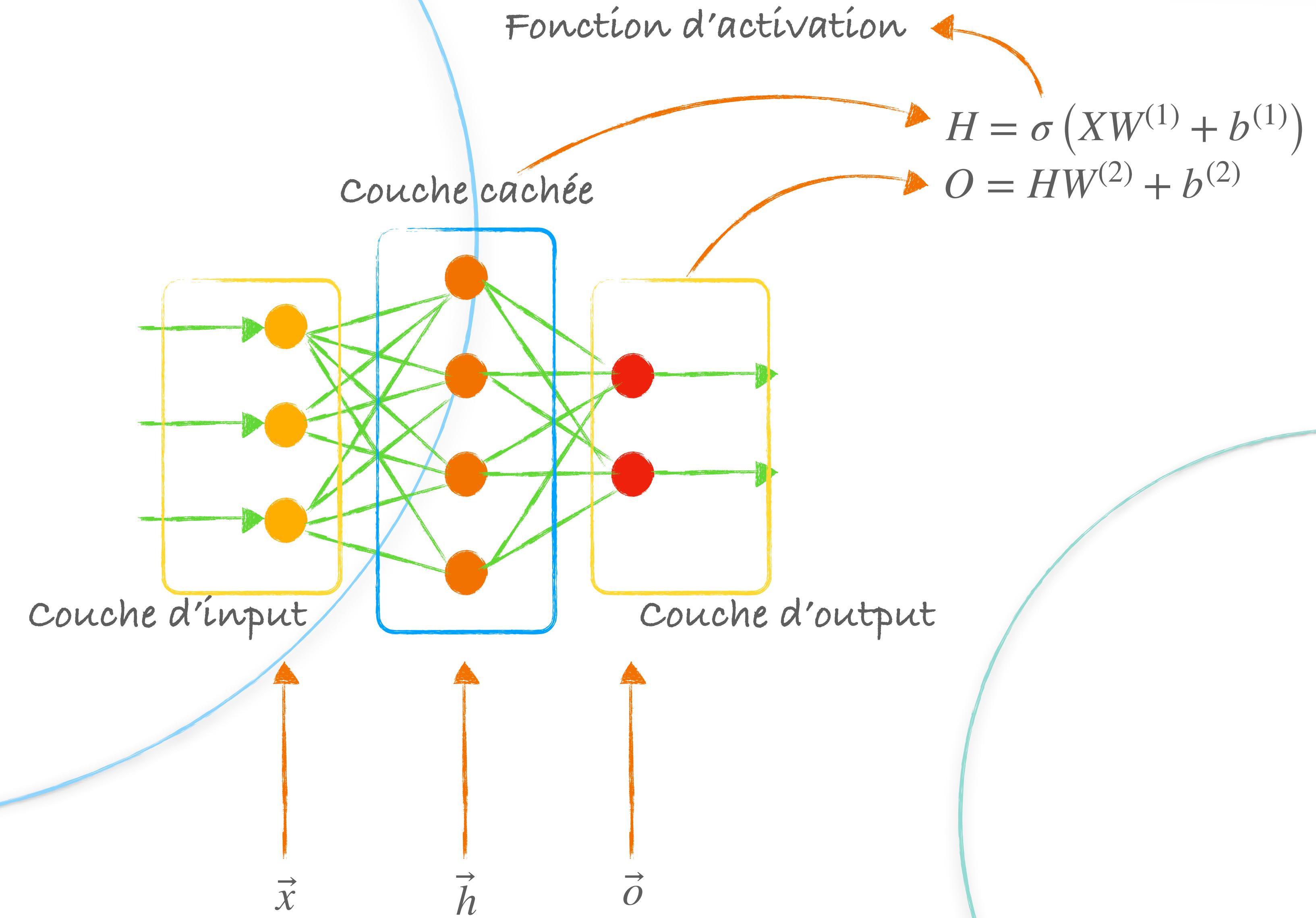
QNN

Quantum Generative Algo.

NN classique

Principes

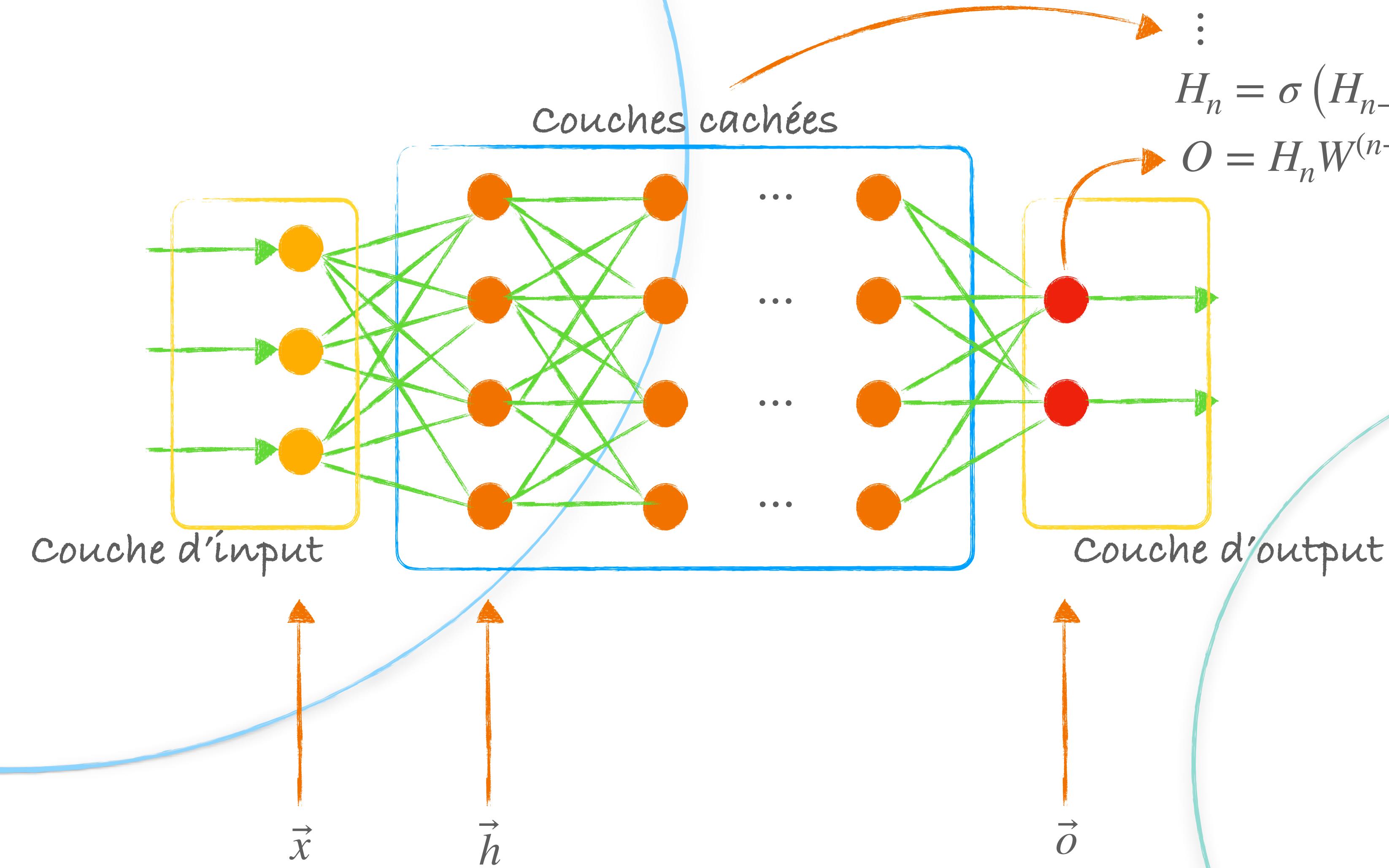
ANN



NN classique

Principes

DNN



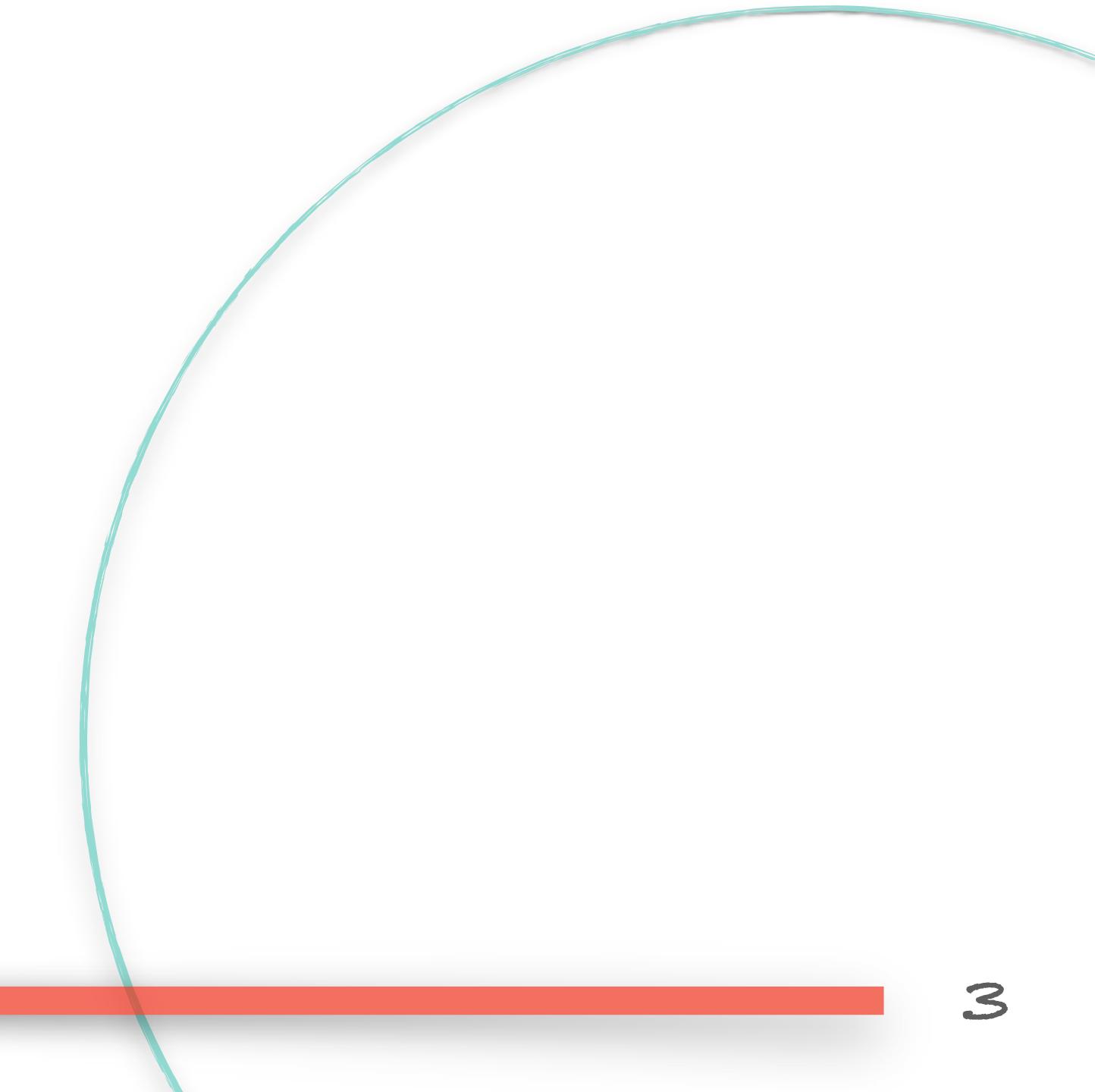
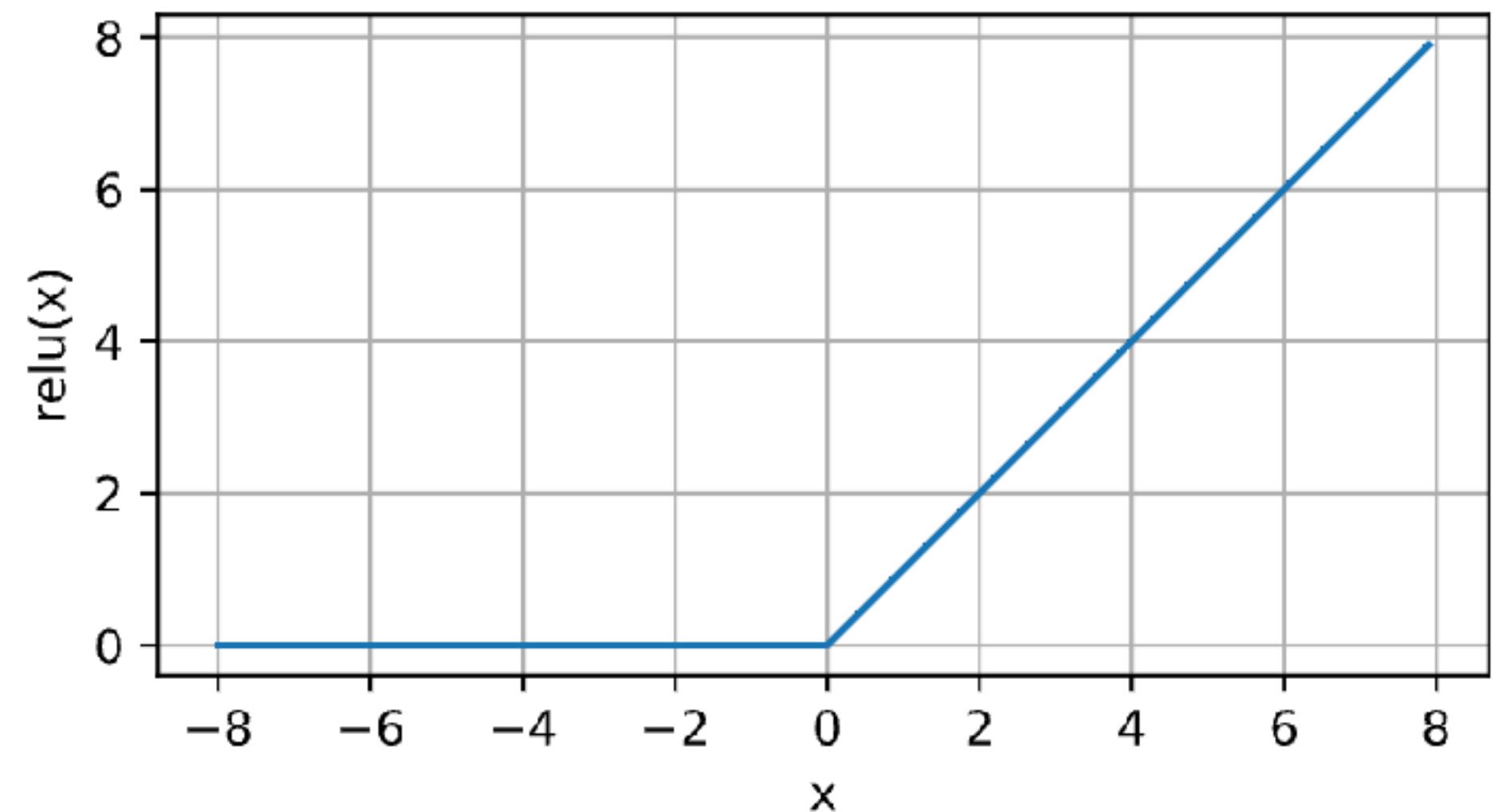
$$\begin{aligned} H_1 &= \sigma(XW^{(1)} + b^{(1)}) \\ H_2 &= \sigma(H_1 W^{(2)} + b^{(2)}) \\ &\vdots && \vdots \\ H_n &= \sigma(H_{n-1} W^{(n)} + b^{(n)}) \\ O &= H_n W^{(n+1)} + b^{(n+1)} \end{aligned}$$

NN classique

Fonctions d'activation

ReLU

$$\text{ReLU}(x) = \max(0, x)$$



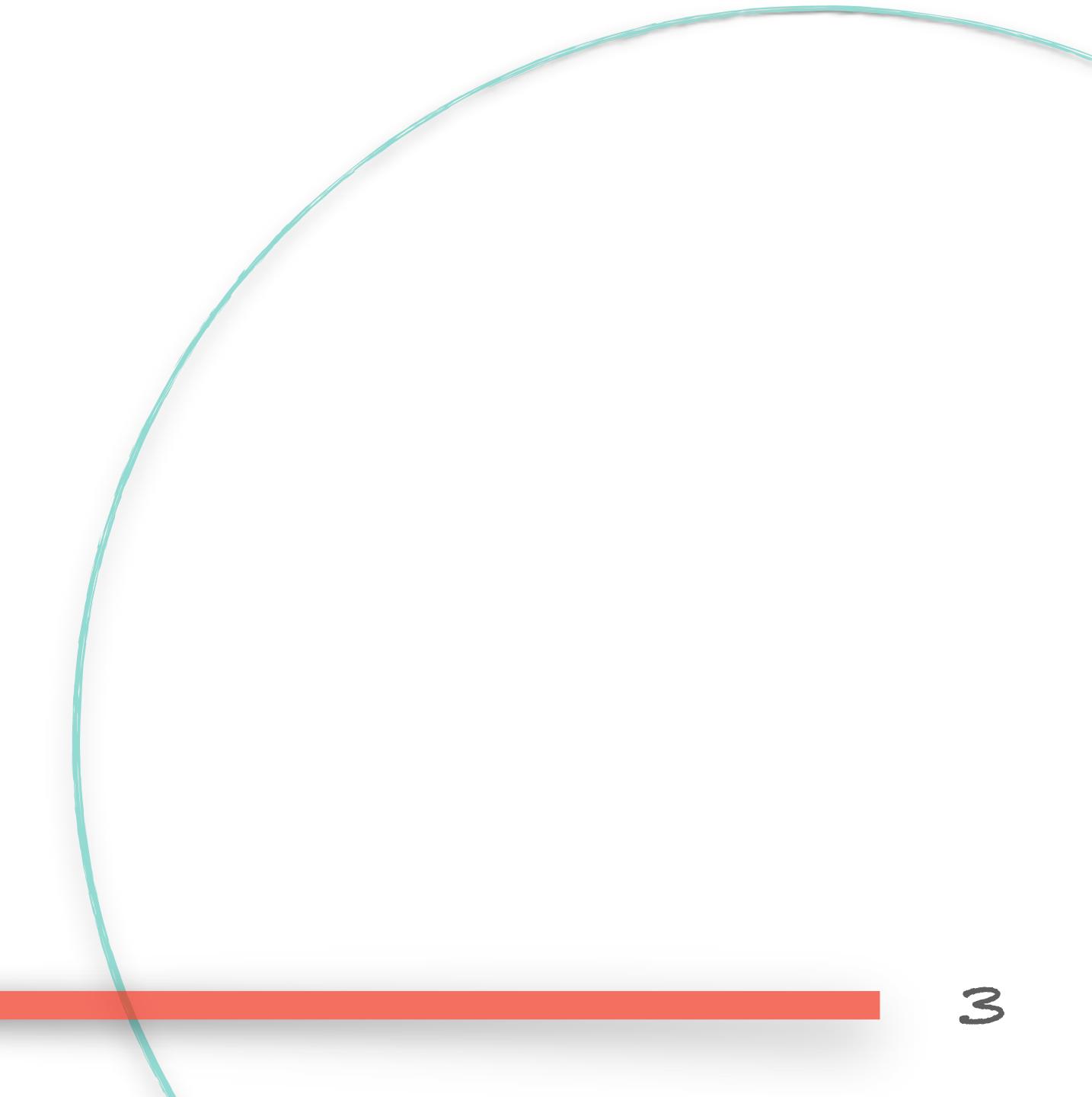
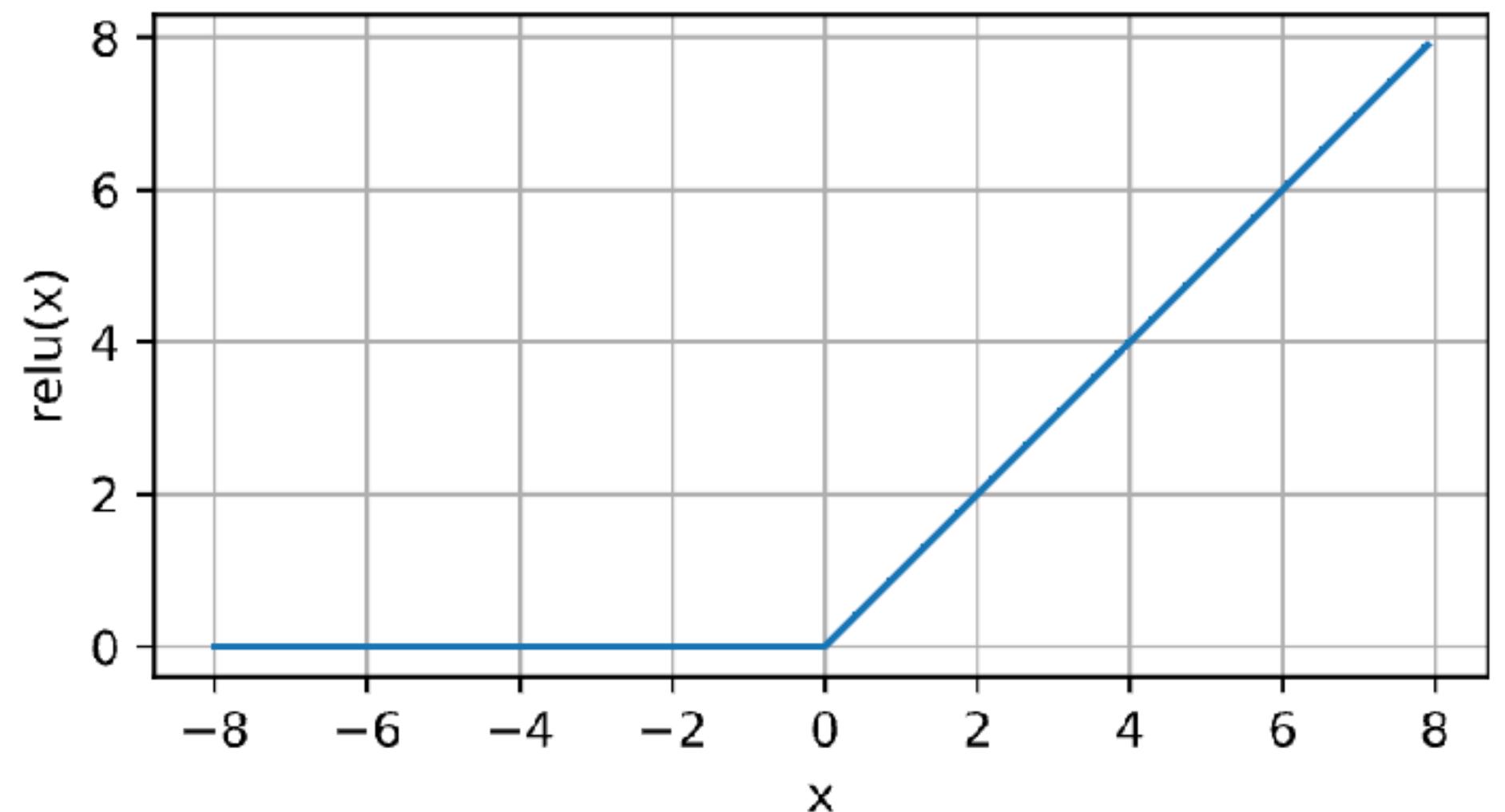
NN classique

Fonctions d'activation

ReLU

$$\text{ReLU}(x) = \max(0, x)$$

Adaptée?

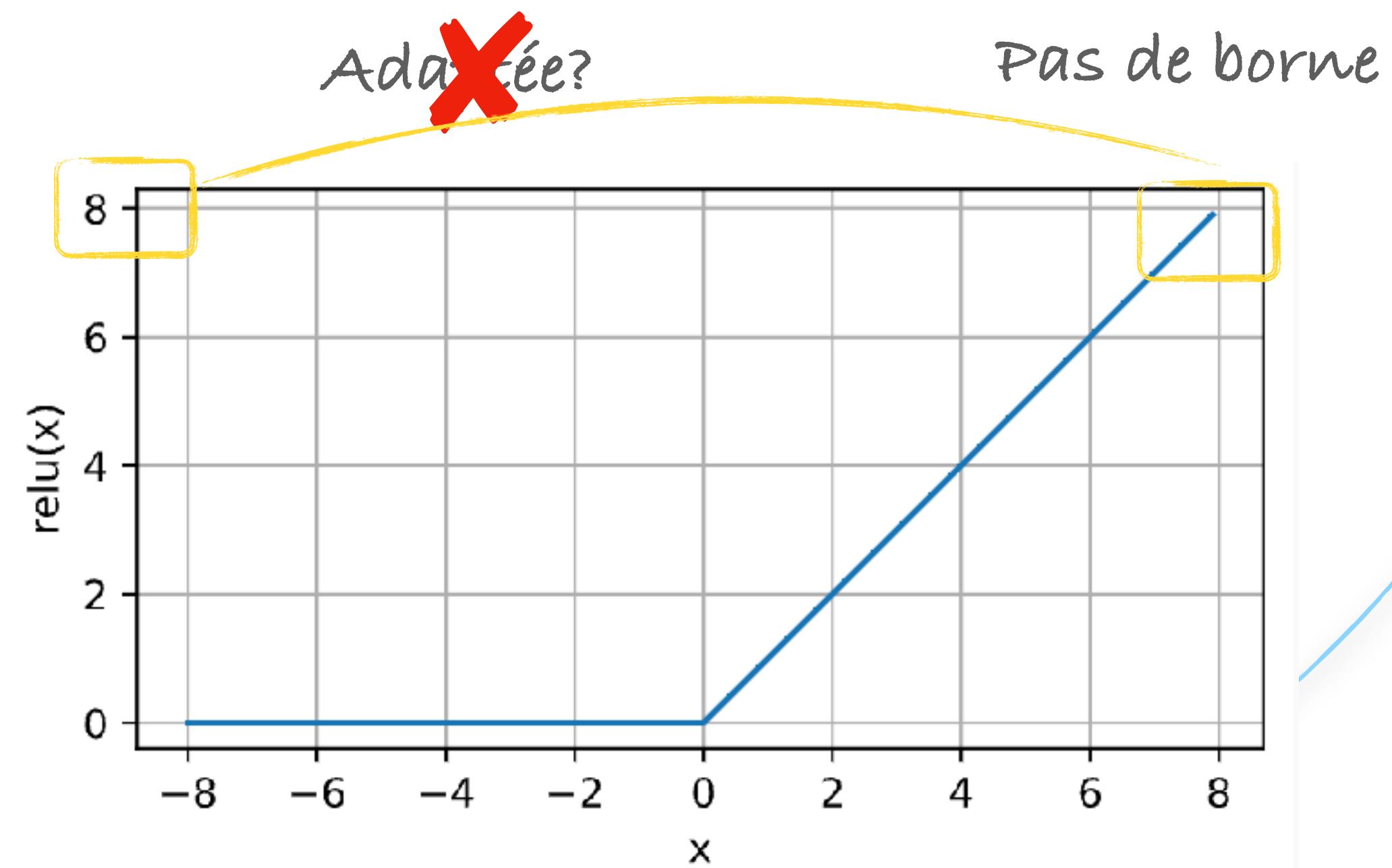


NN classique

Fonctions d'activation

ReLU

$$ReLU(x) = \max(0, x)$$



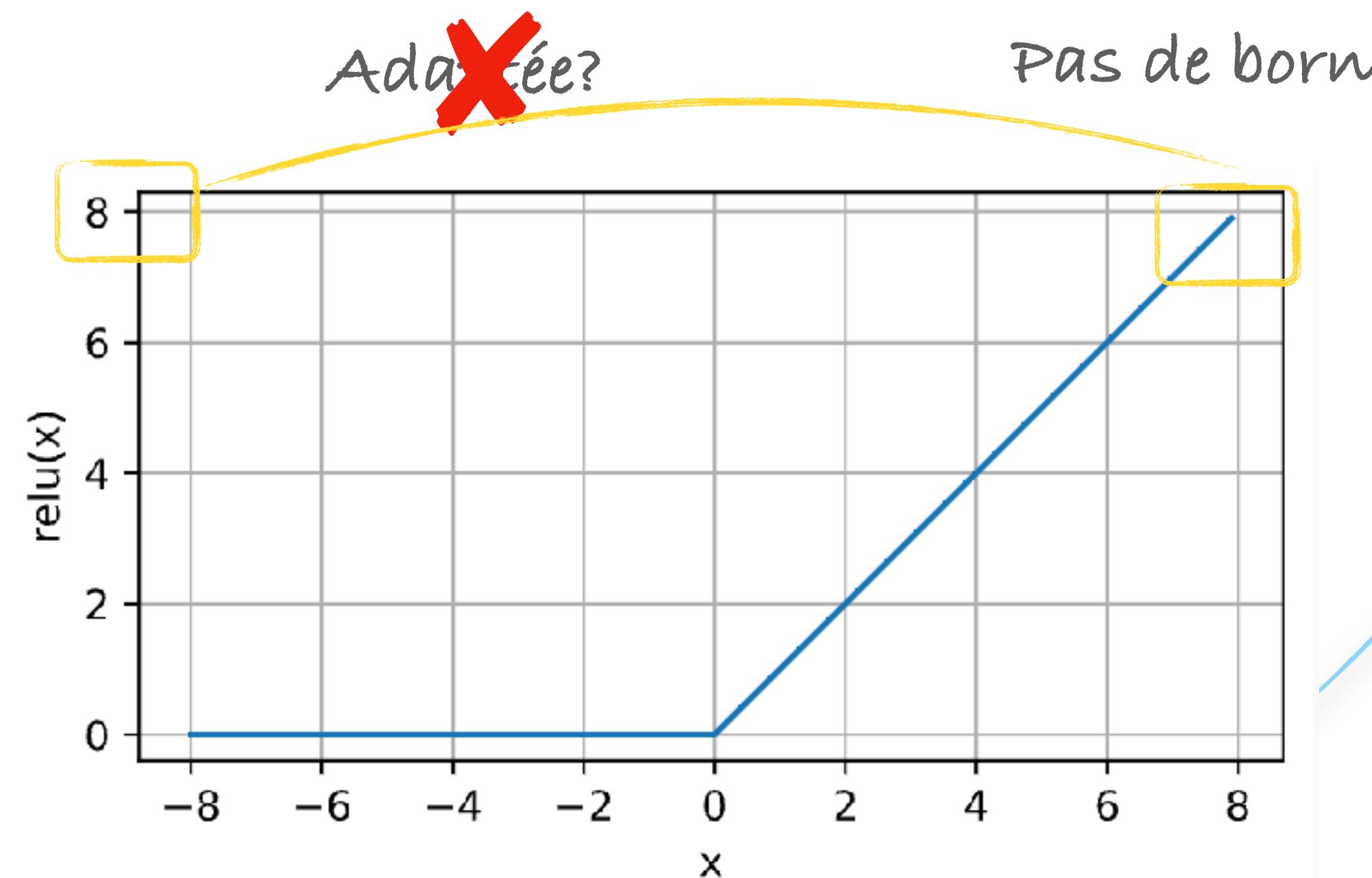
Pas de borne

NN classique

Fonctions d'activation

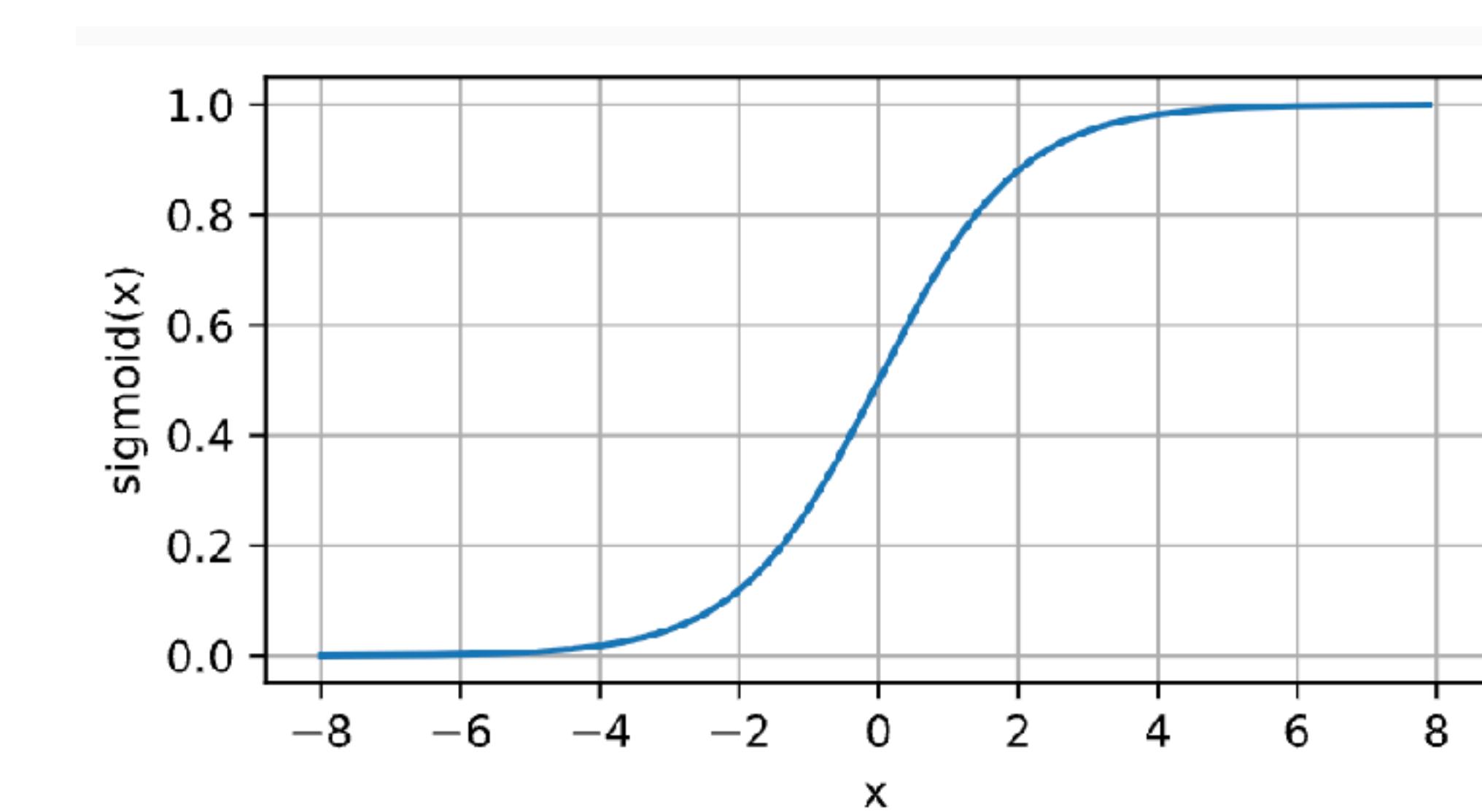
ReLU

$$ReLU(x) = \max(0, x)$$



Sigmoid

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}$$

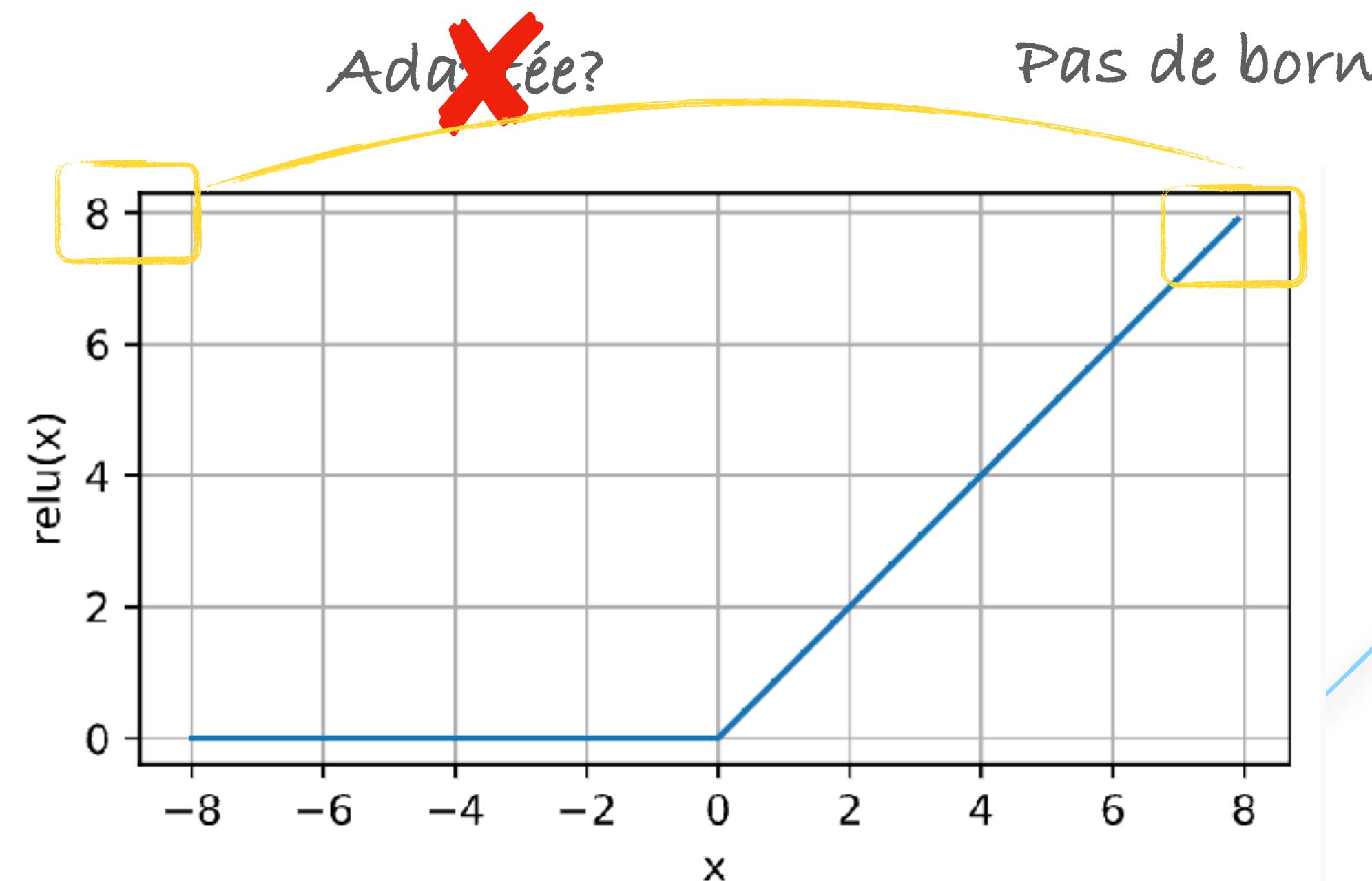


NN classique

Fonctions d'activation

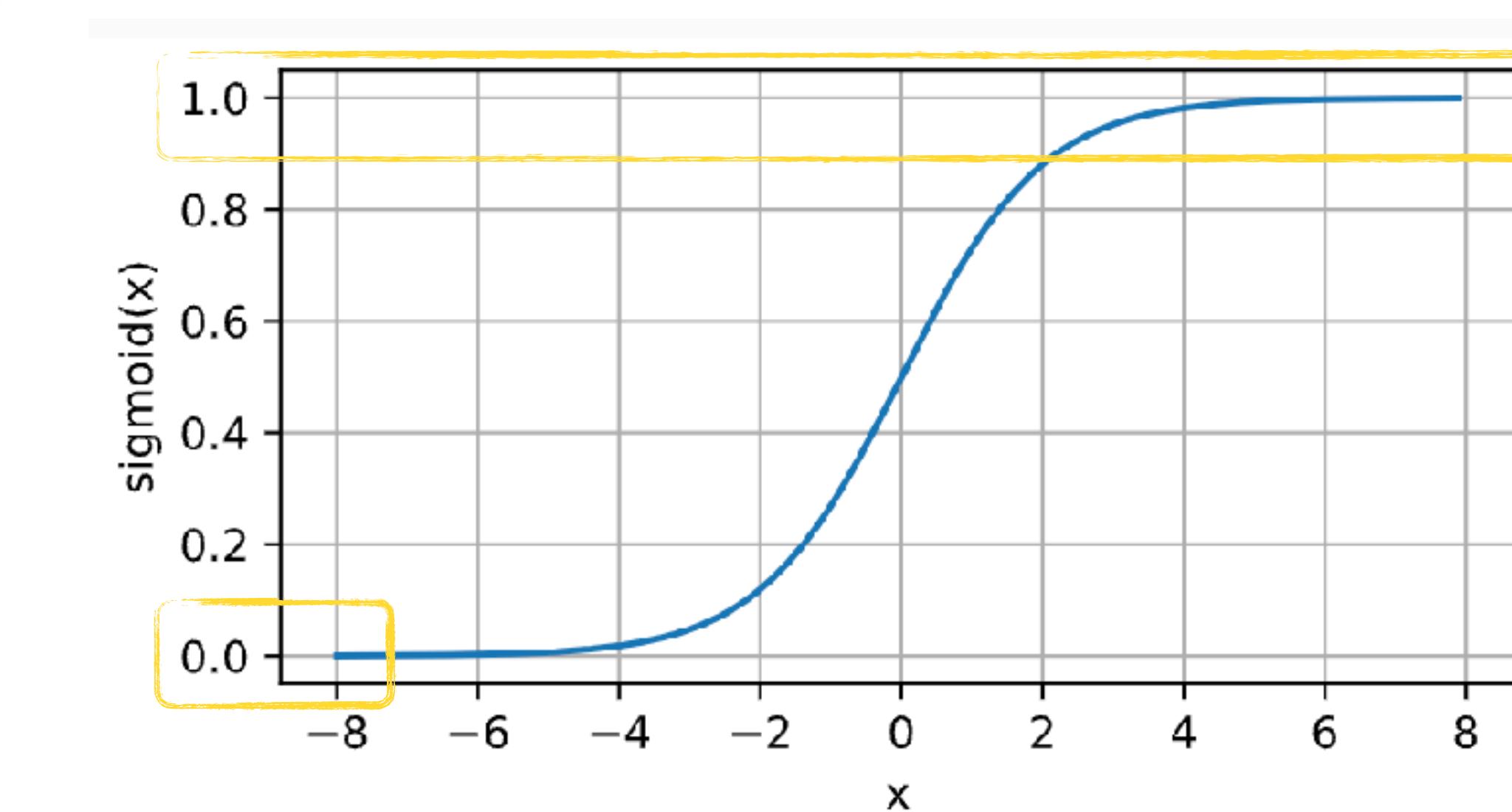
ReLU

$$ReLU(x) = \max(0, x)$$



sigmoid

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}$$



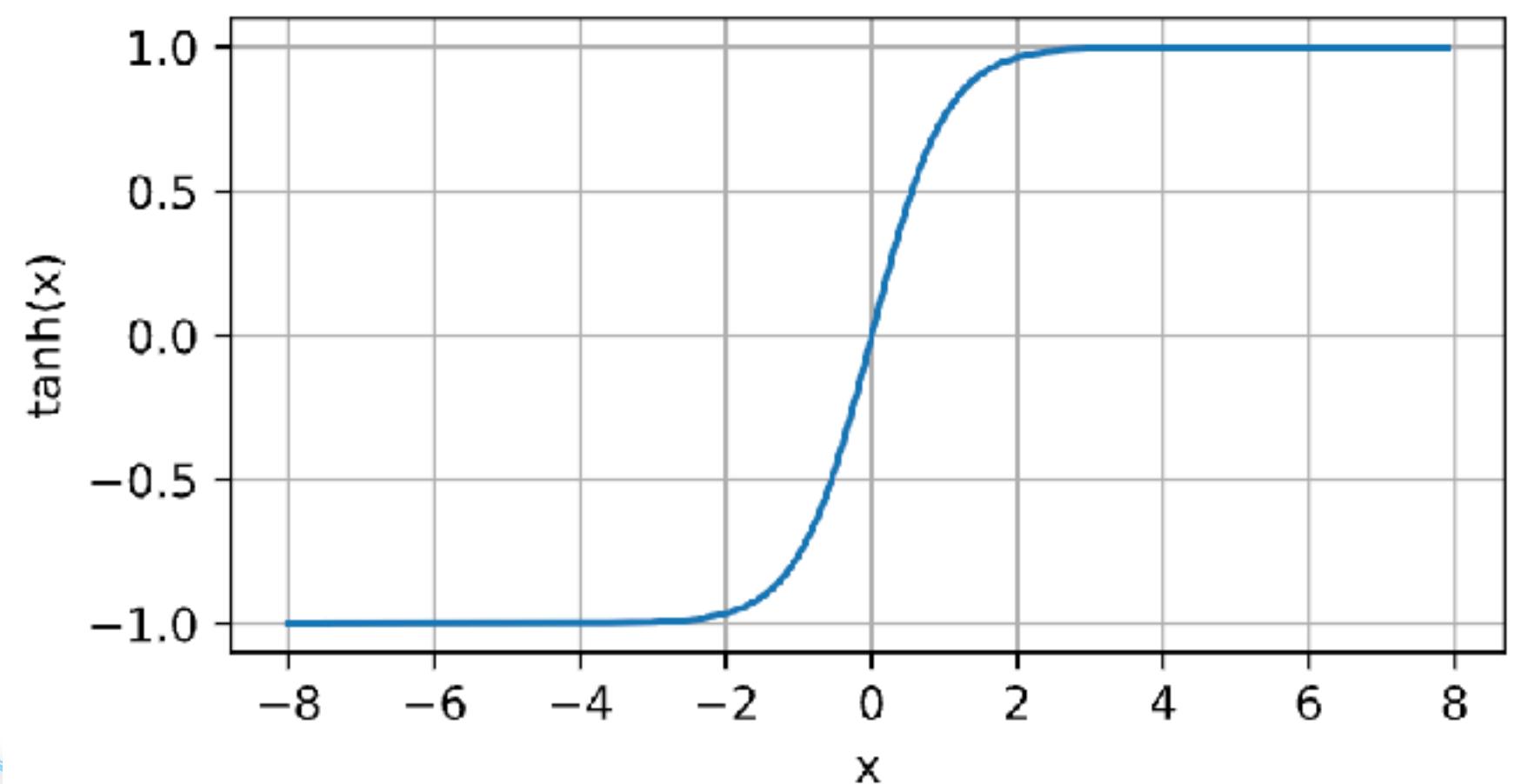
Bon input pour une couche quantique

NN classique

Fonctions d'activation

Tanh

$$\tanh(x) = \frac{(1 - \exp(-2x))}{(1 + \exp(-2x))}$$

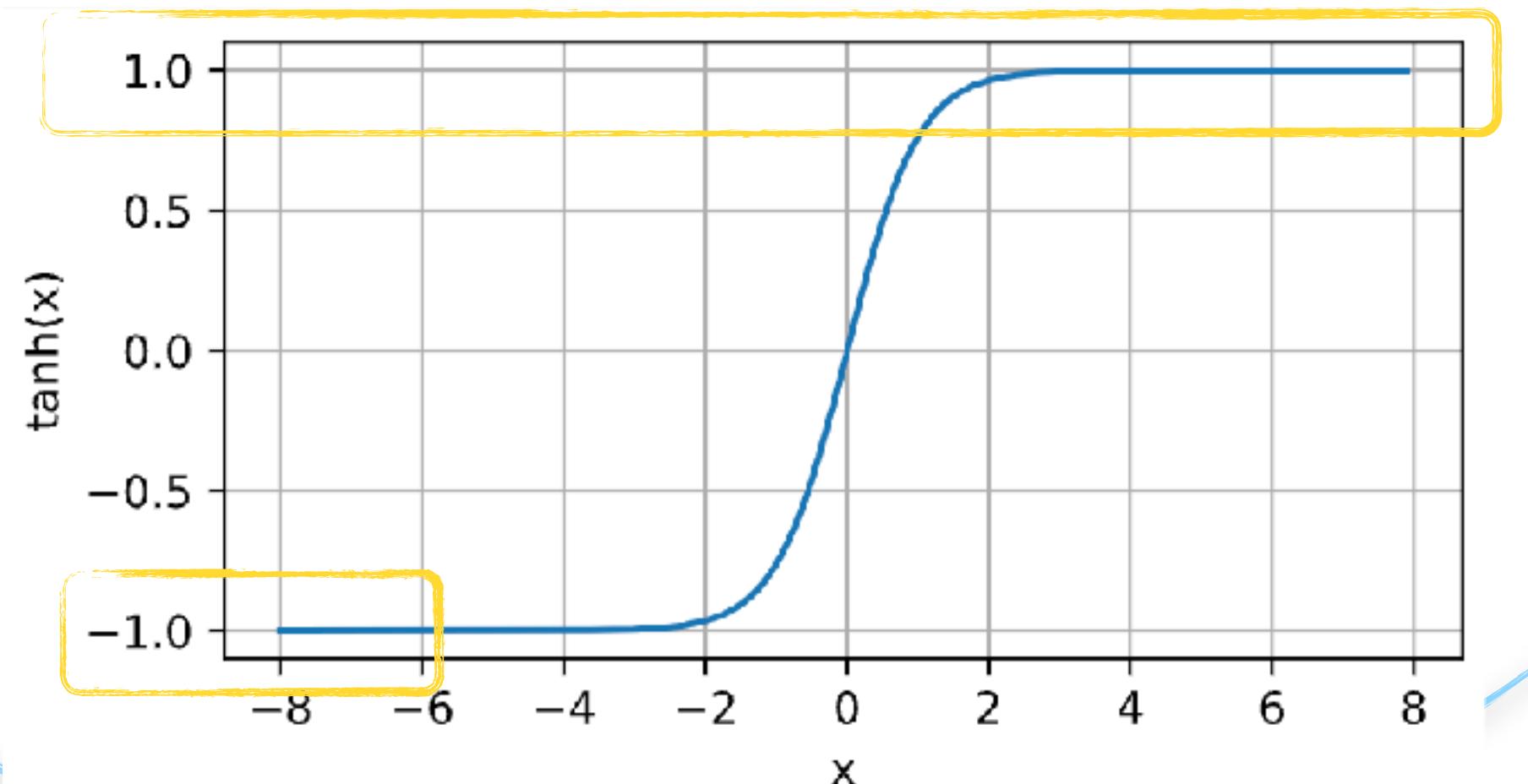


NN classique

Fonctions d'activation

Tanh

$$\tanh(x) = \frac{(1 - \exp(-2x))}{(1 + \exp(-2x))}$$



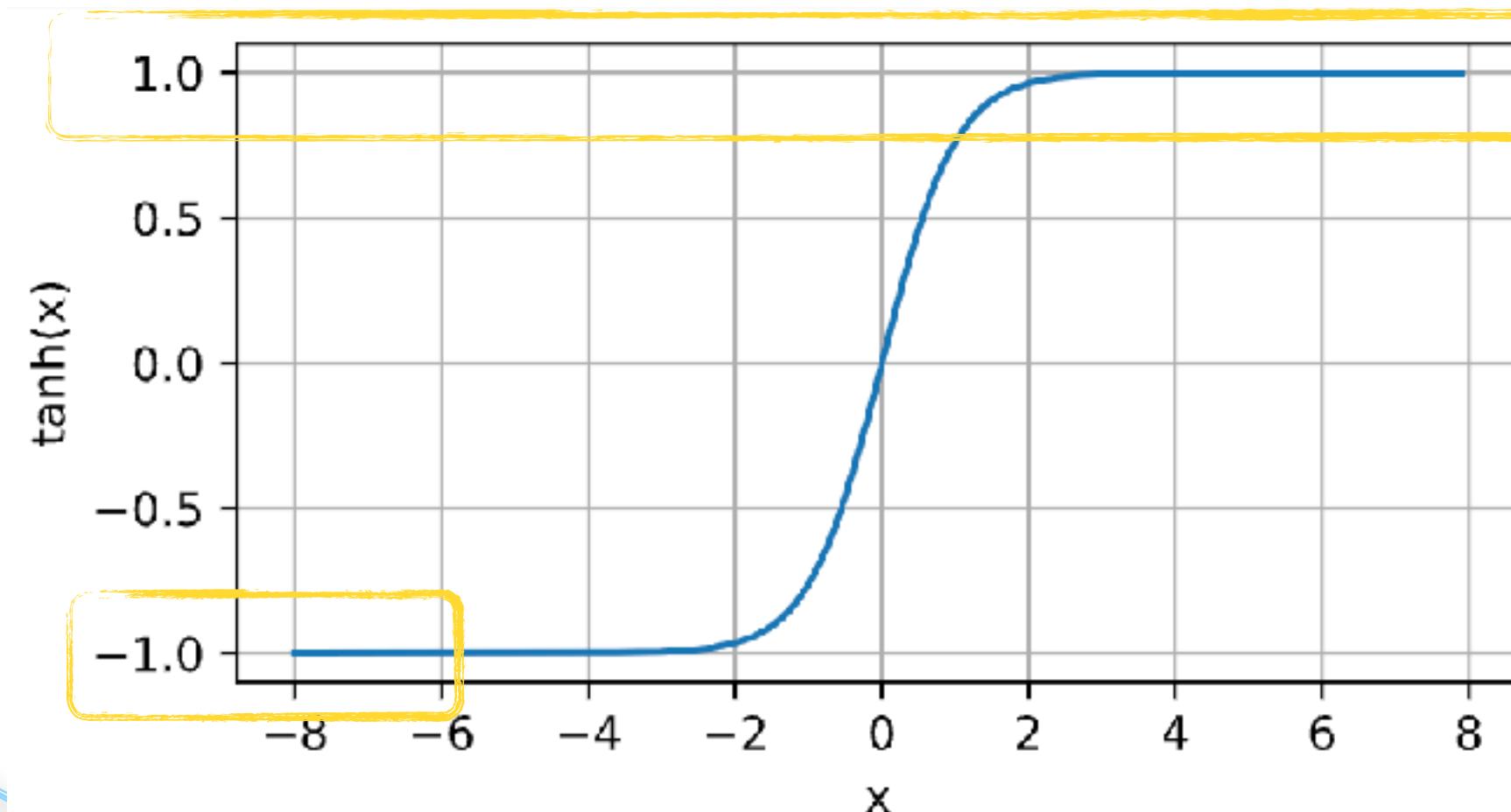
Bon input pour une couche quantique

NN classique

Fonctions d'activation

Tanh

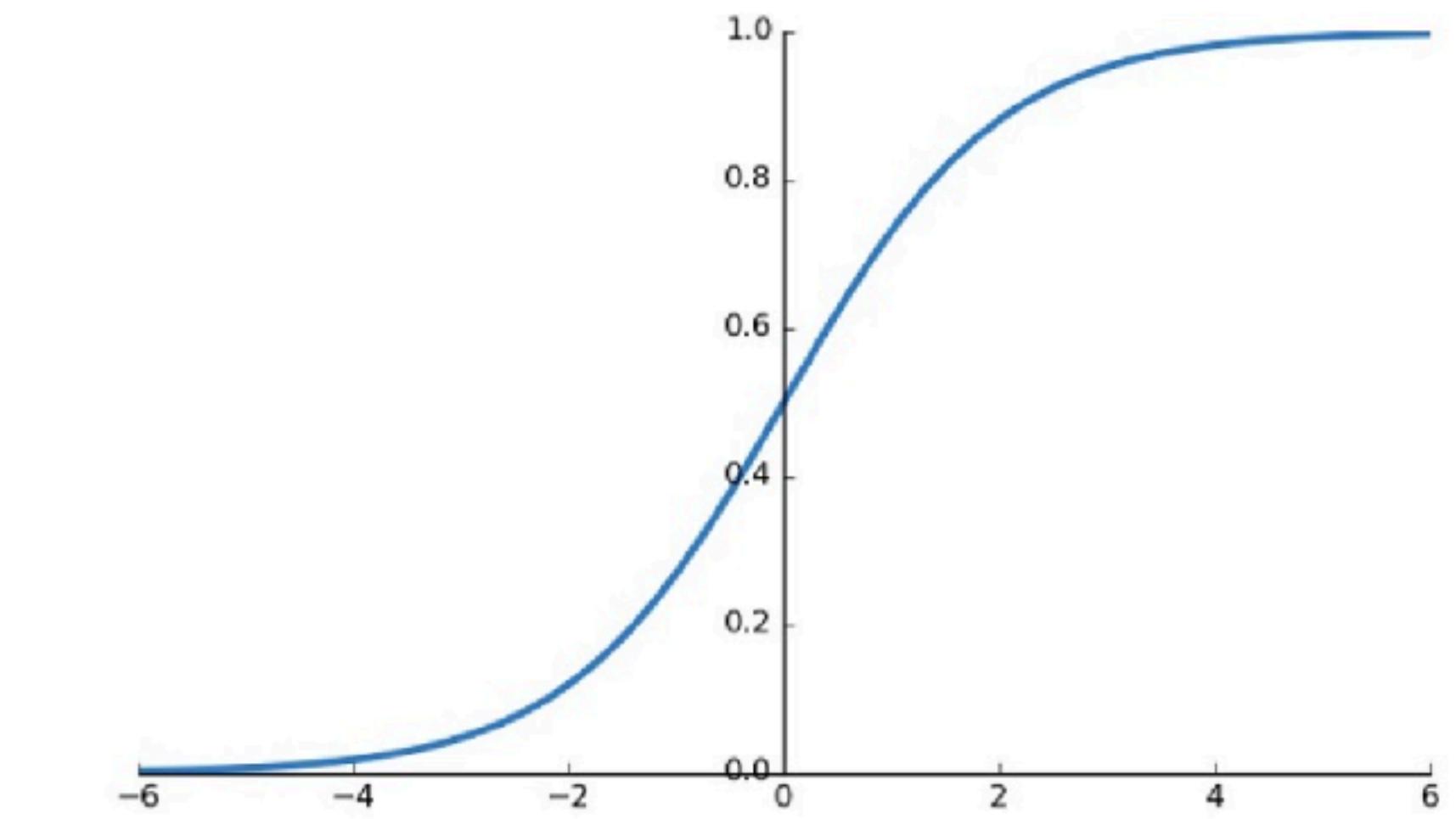
$$\tanh(x) = \frac{(1 - \exp(-2x))}{(1 + \exp(-2x))}$$



Bon input pour une couche quantique

Softmax

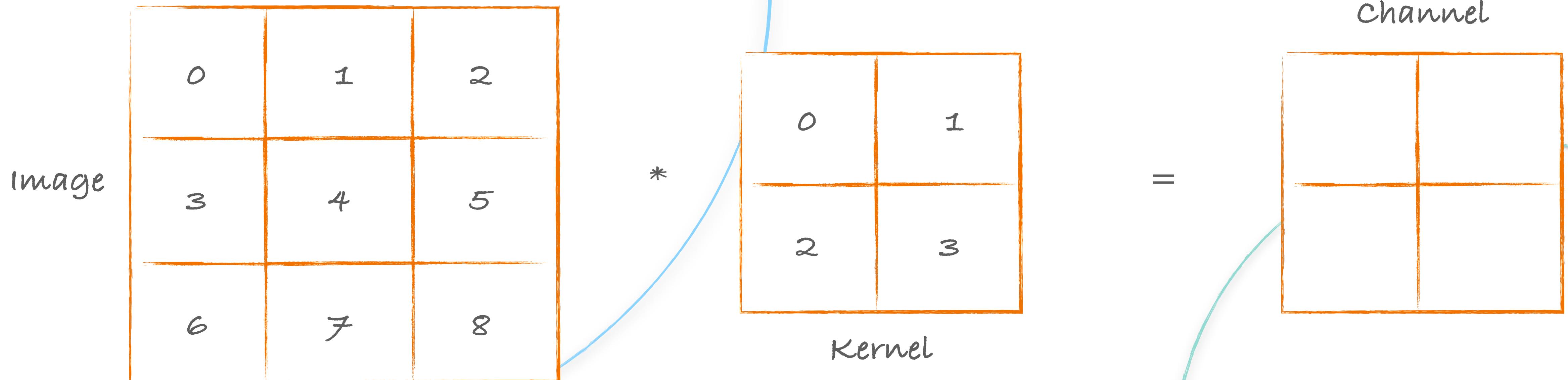
$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}}$$



Calcul la probabilité pour une classe. Généralement utilisé en classification multilabels.

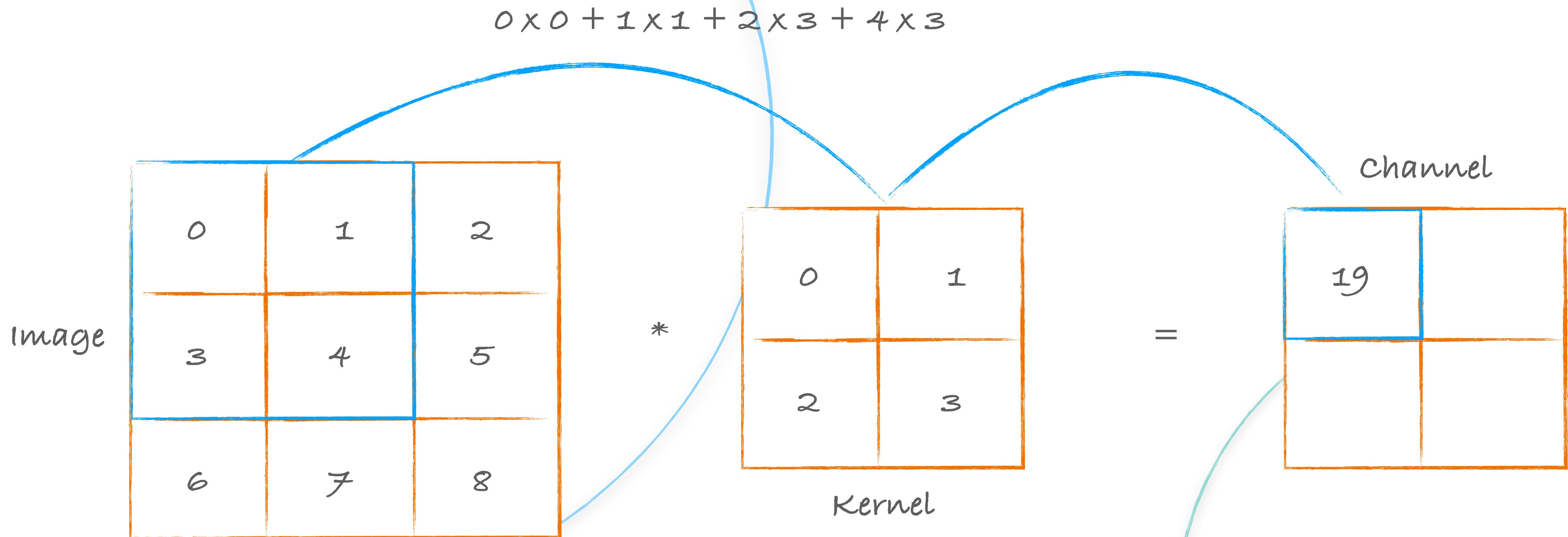
NN classique

CNN



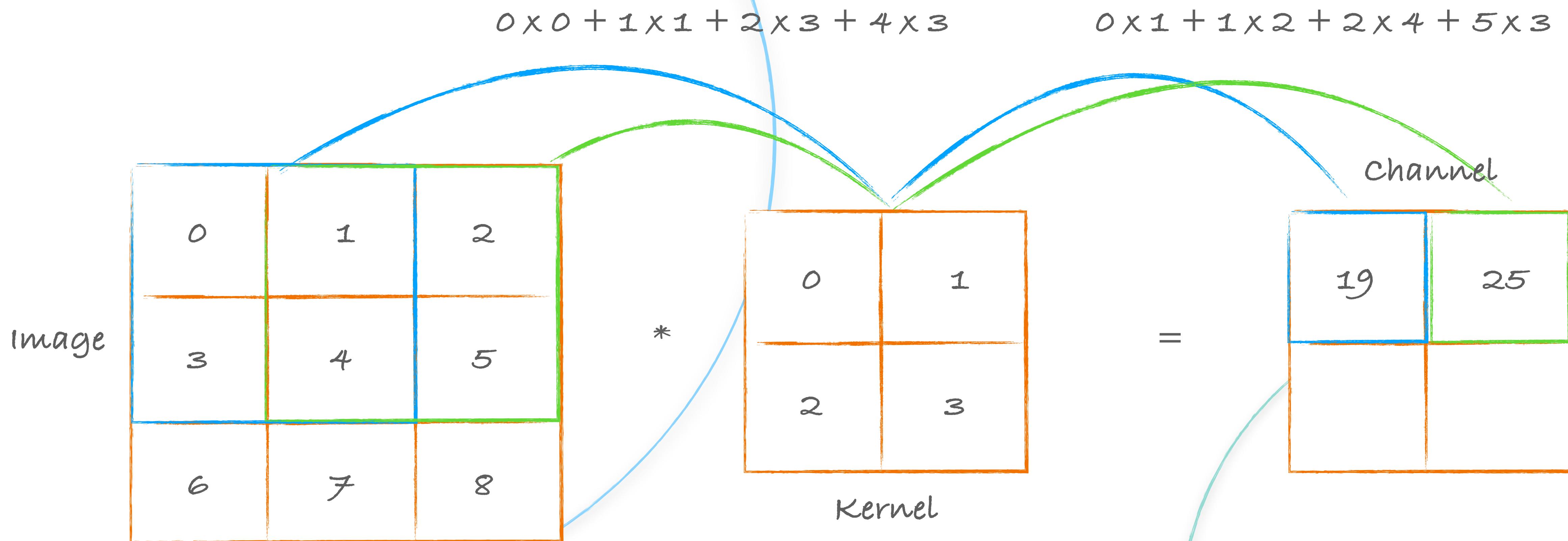
NN classique

CNN



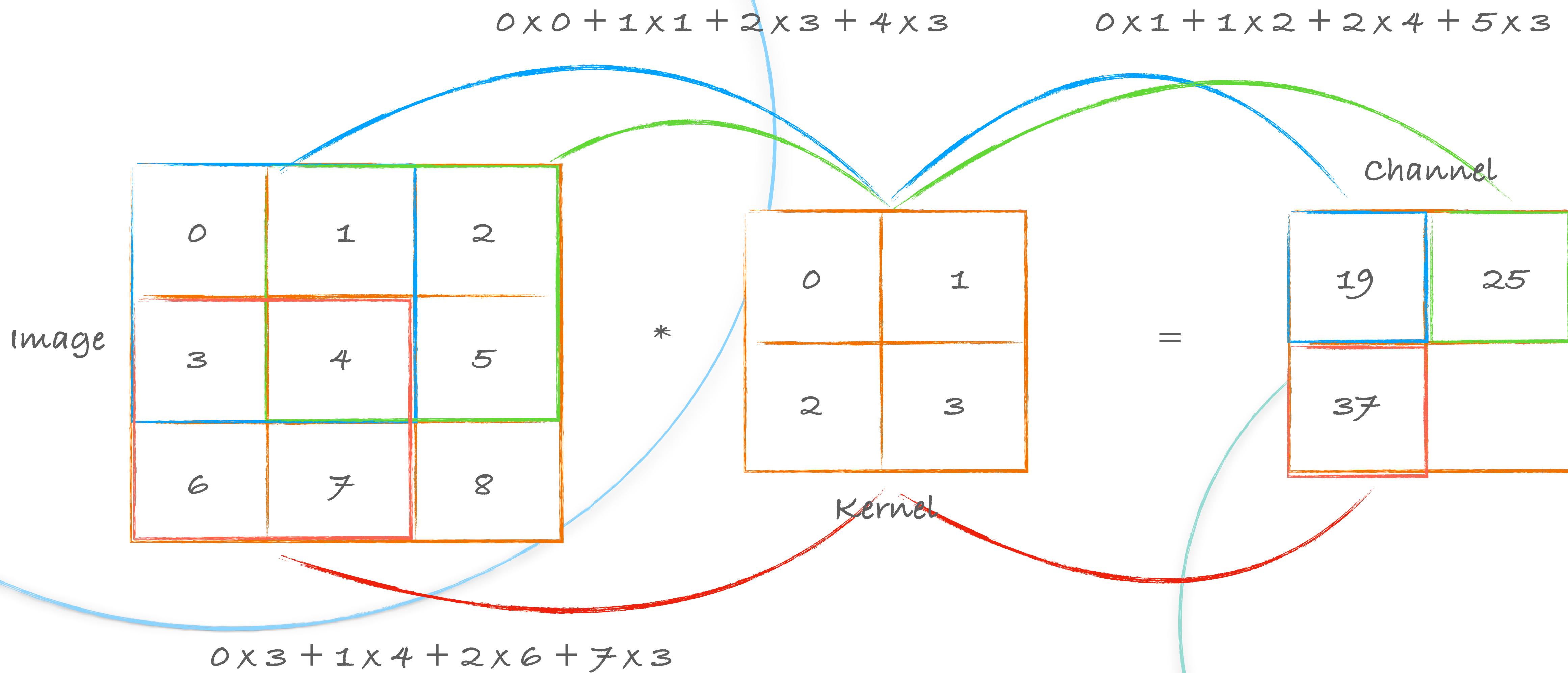
NN classique

CNN



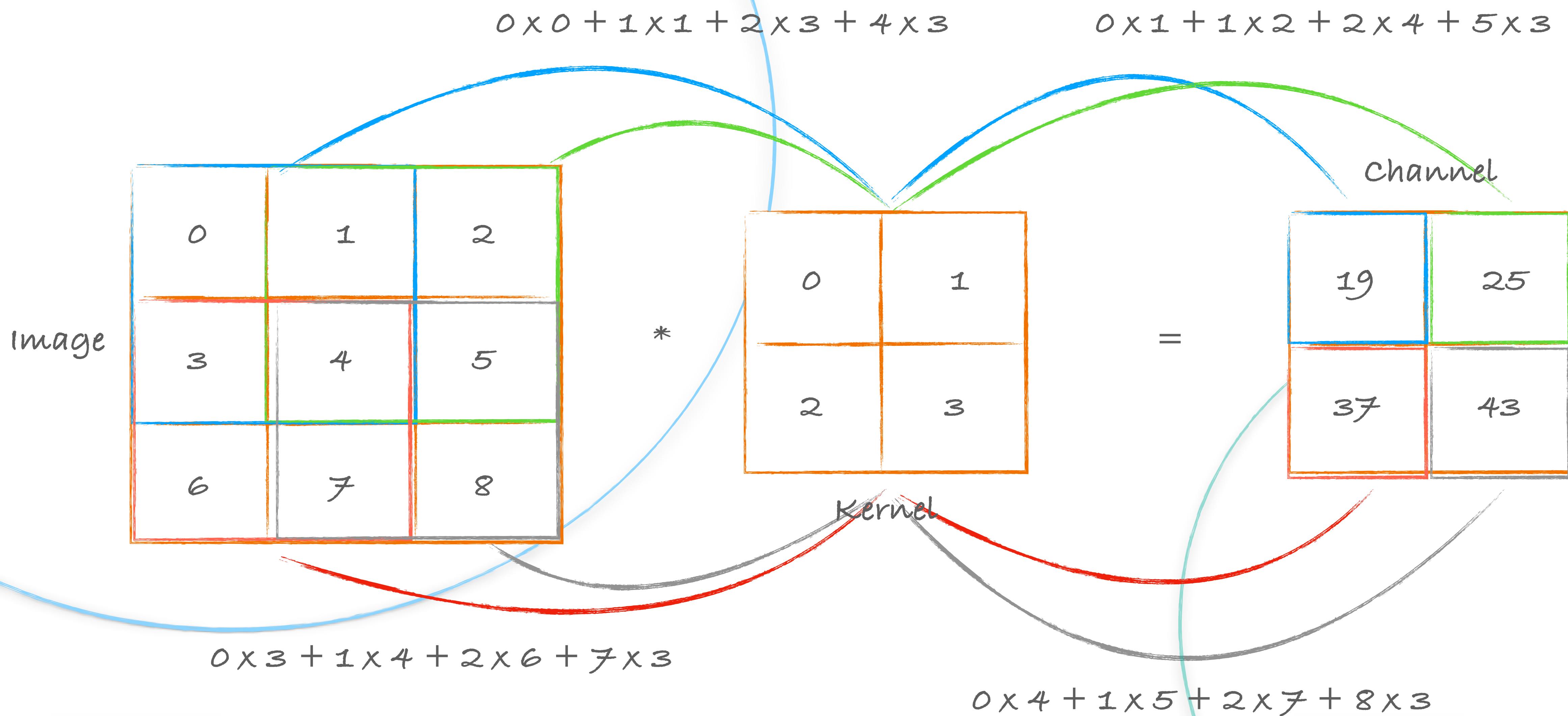
NN classique

CNN



NN classique

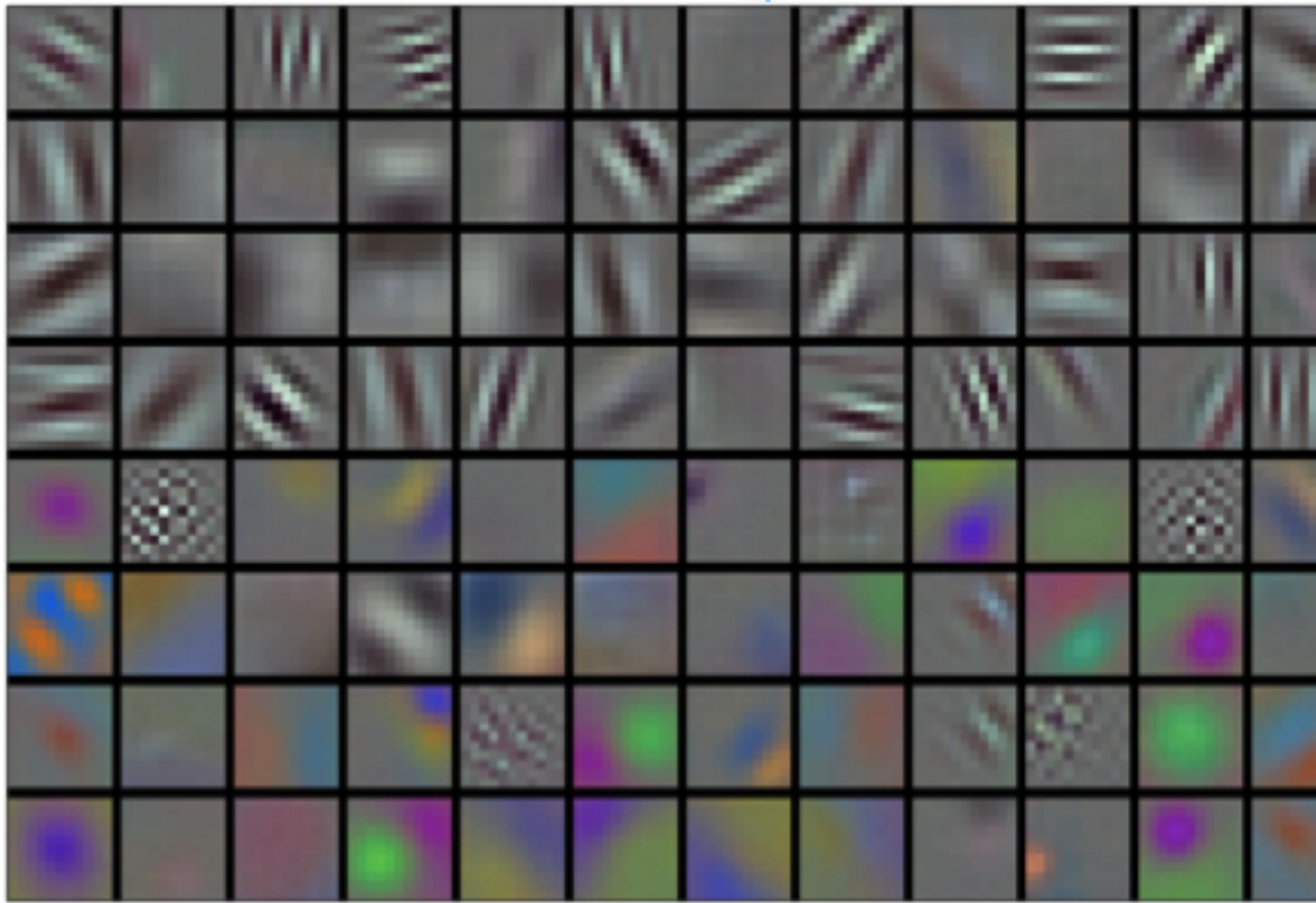
CNN



NN classique

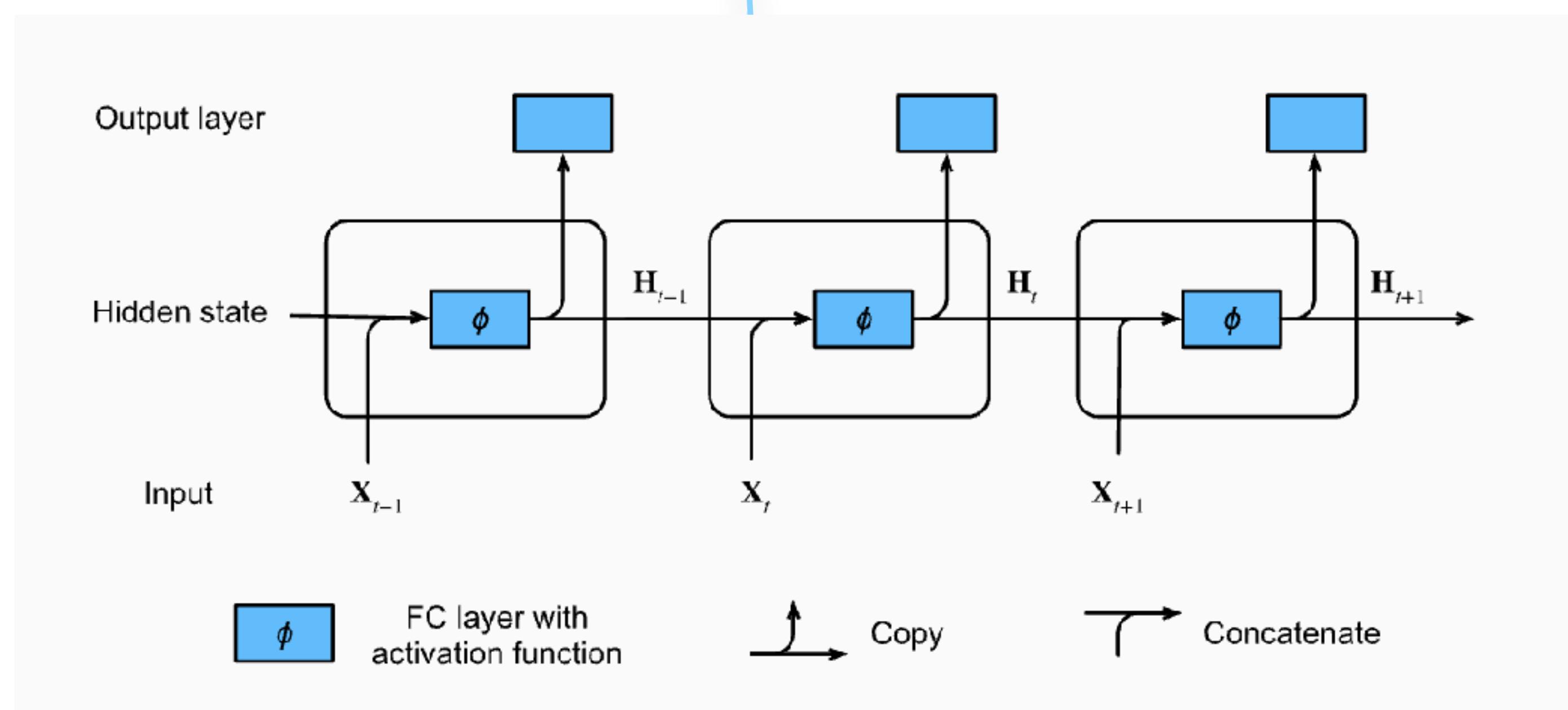
CNN

Kernel ou filtres



NN classique

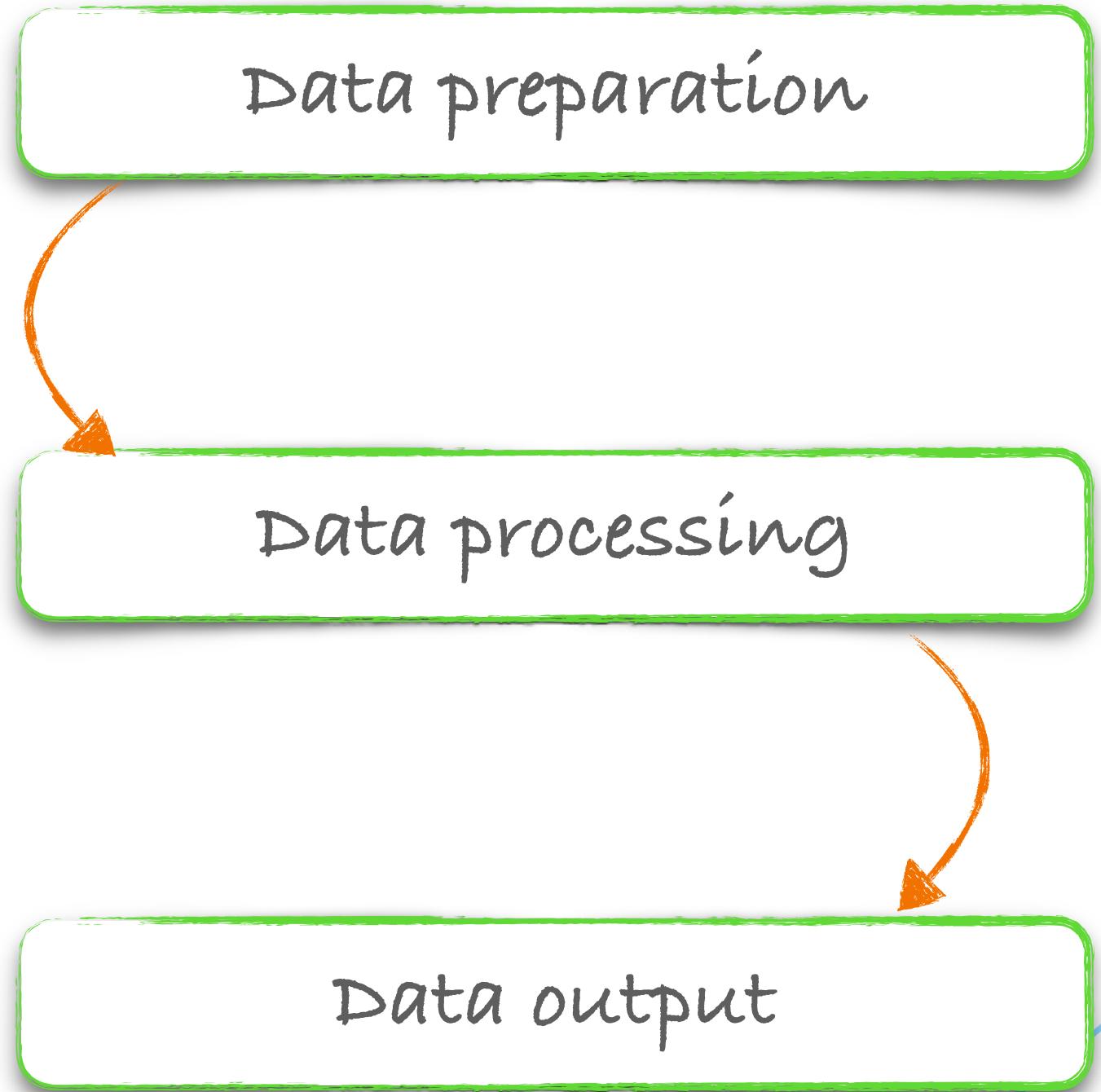
RNN



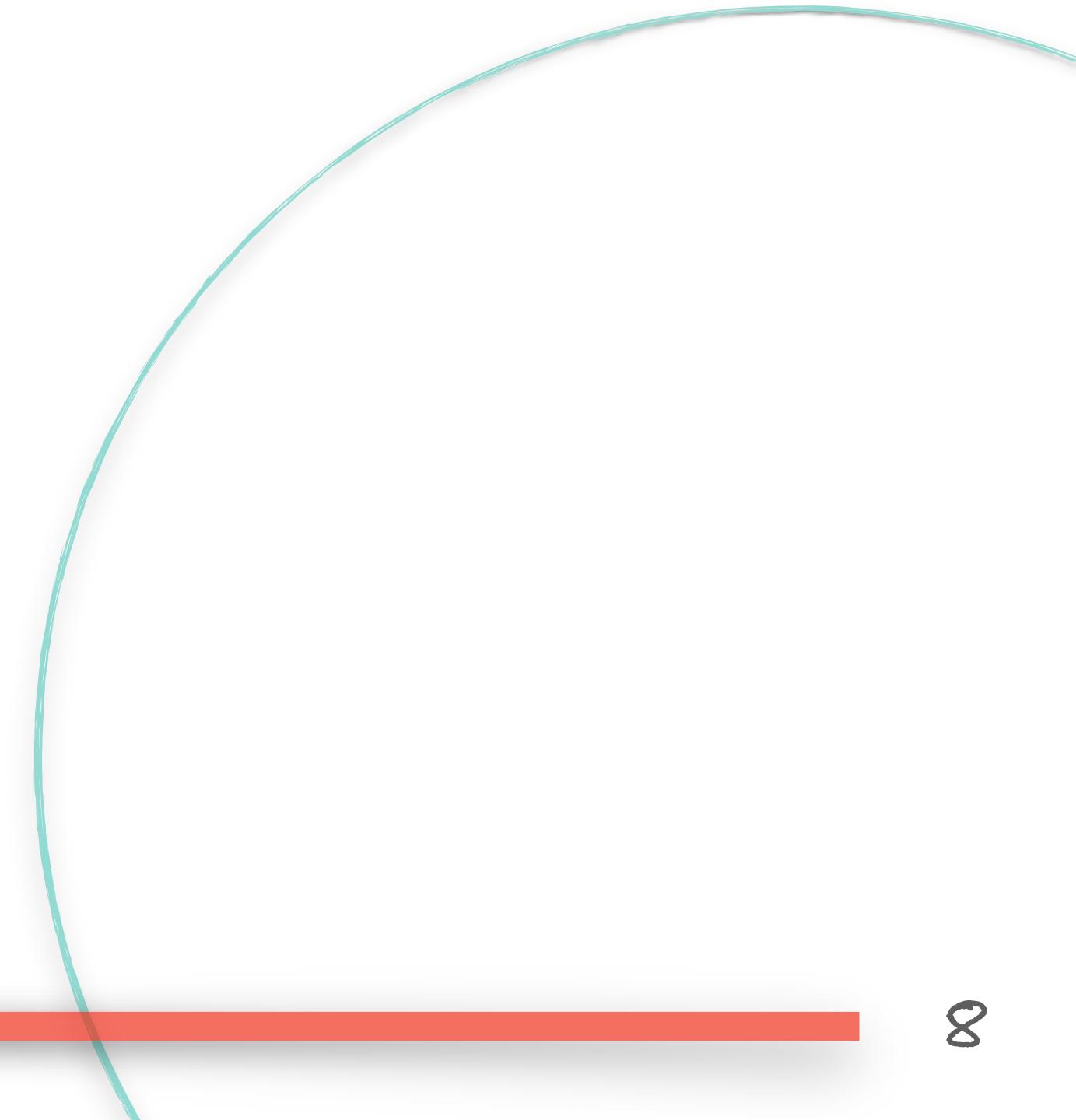
workflows

Introduction

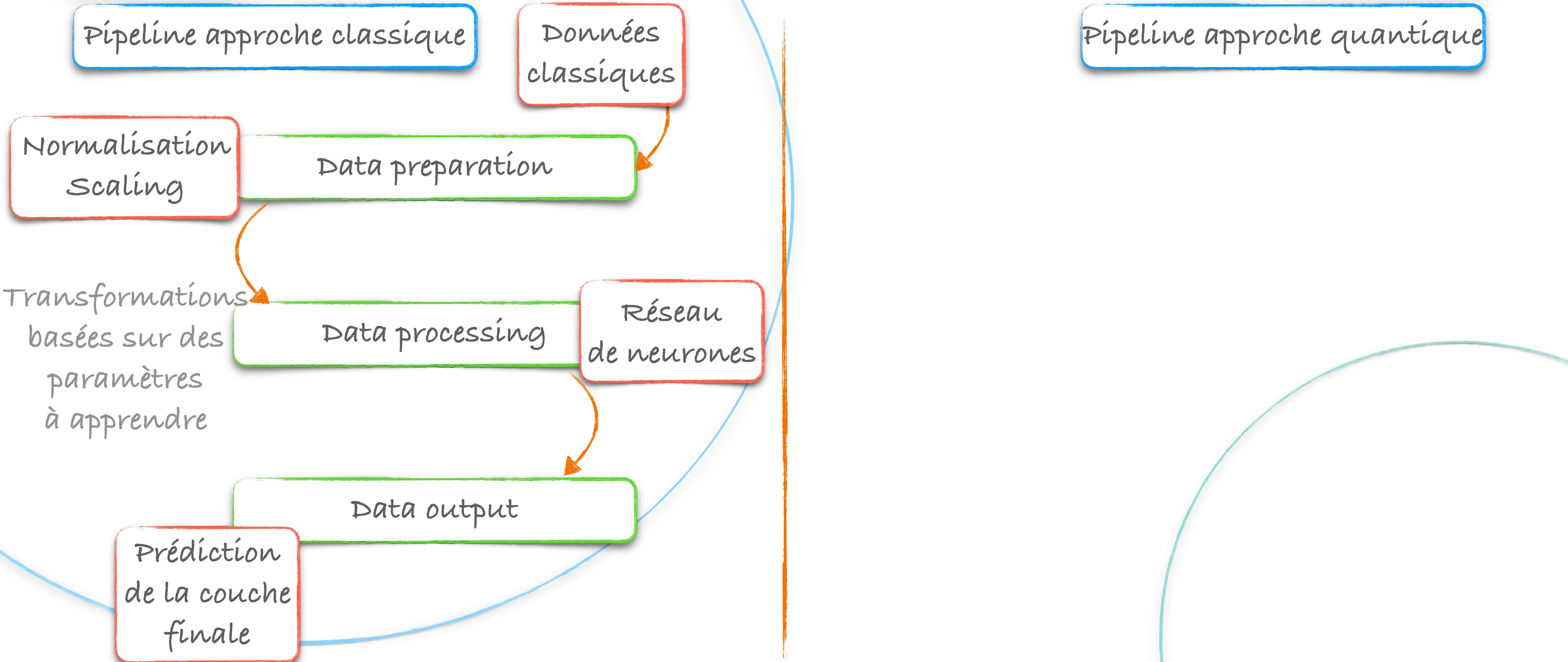
Pipeline approche classique



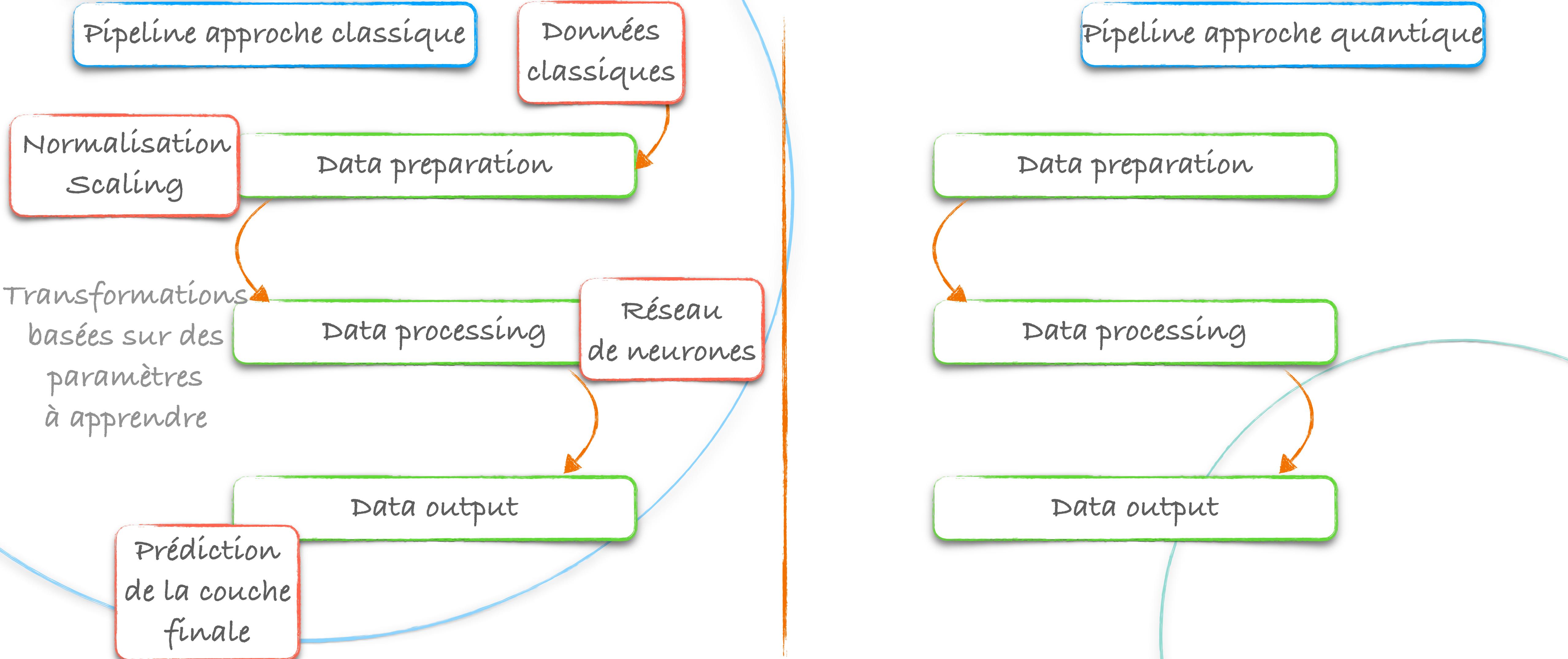
Pipeline approche quantique



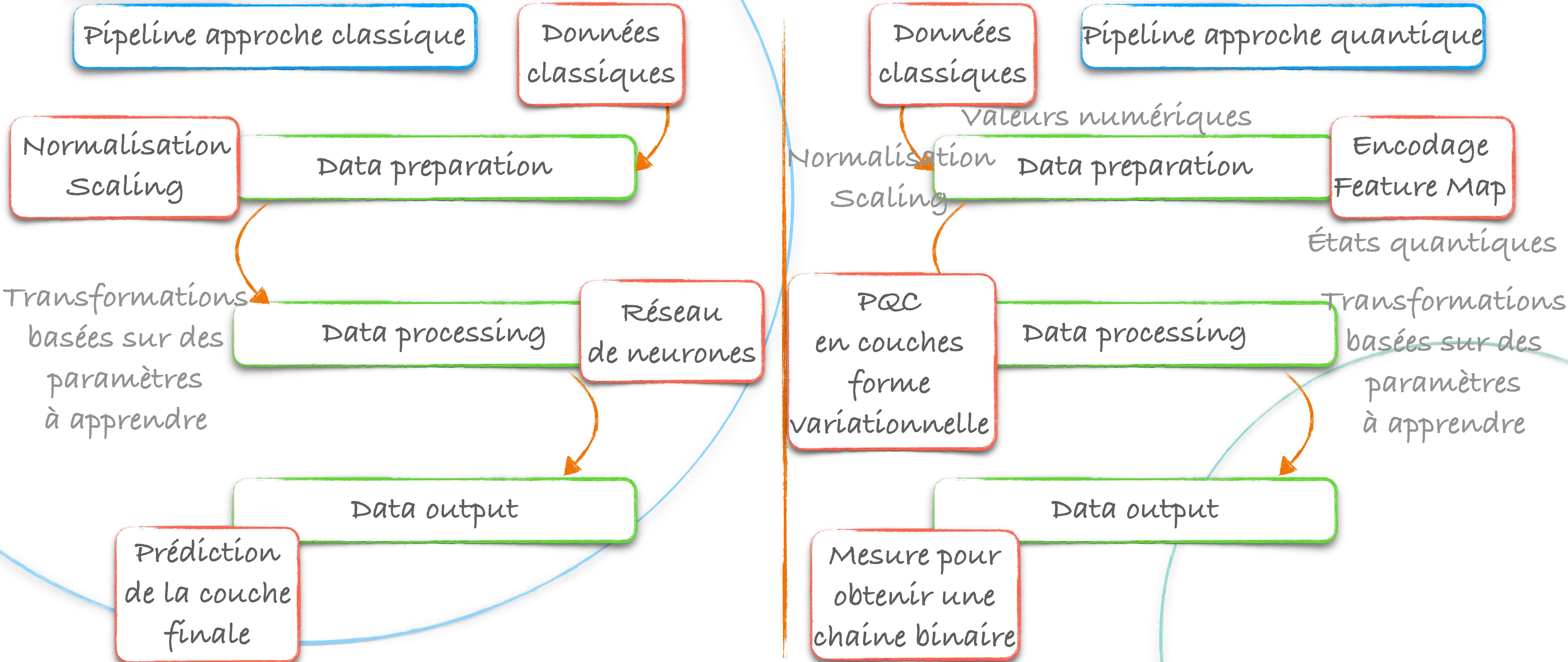
Introduction



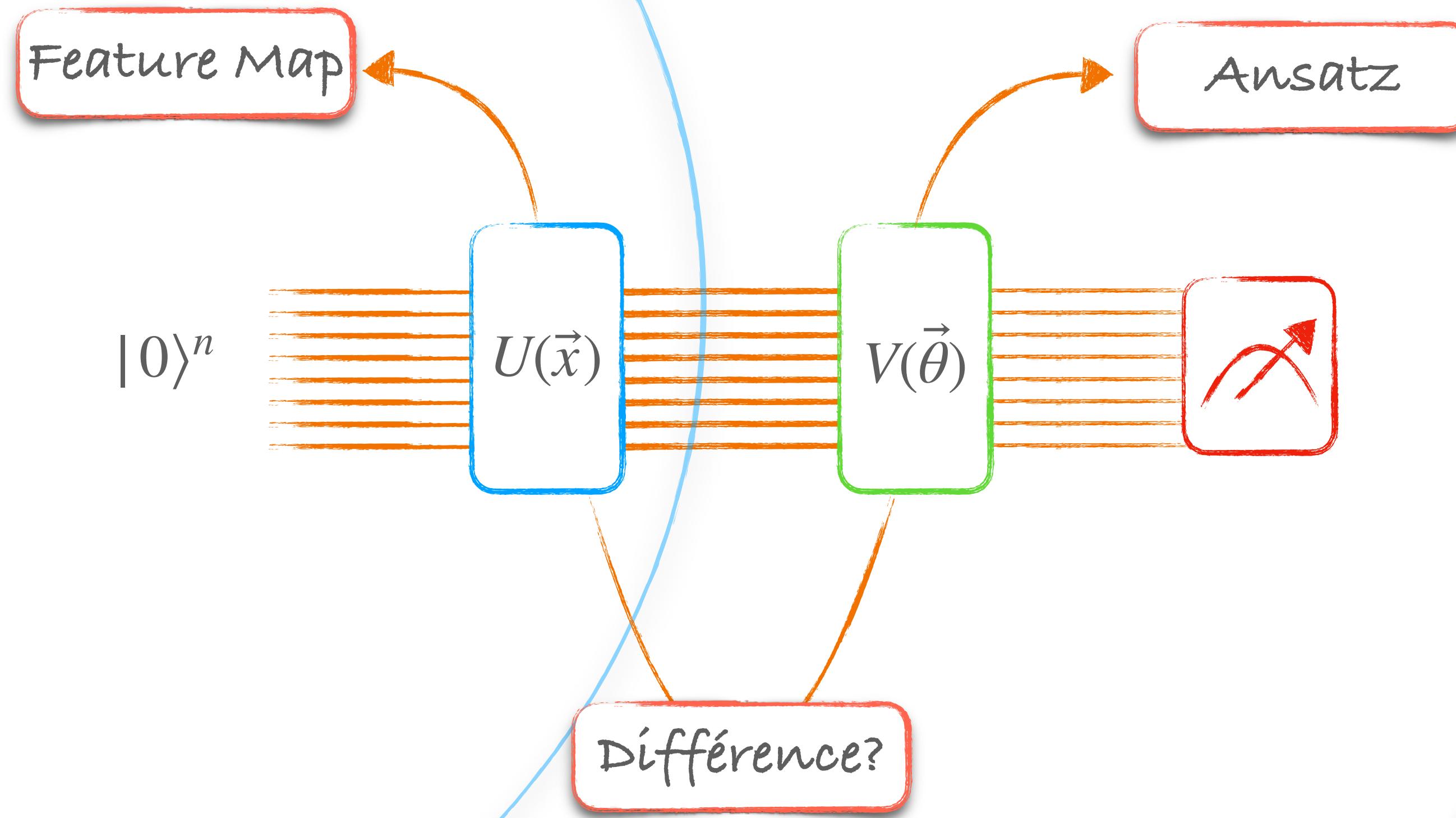
Introduction



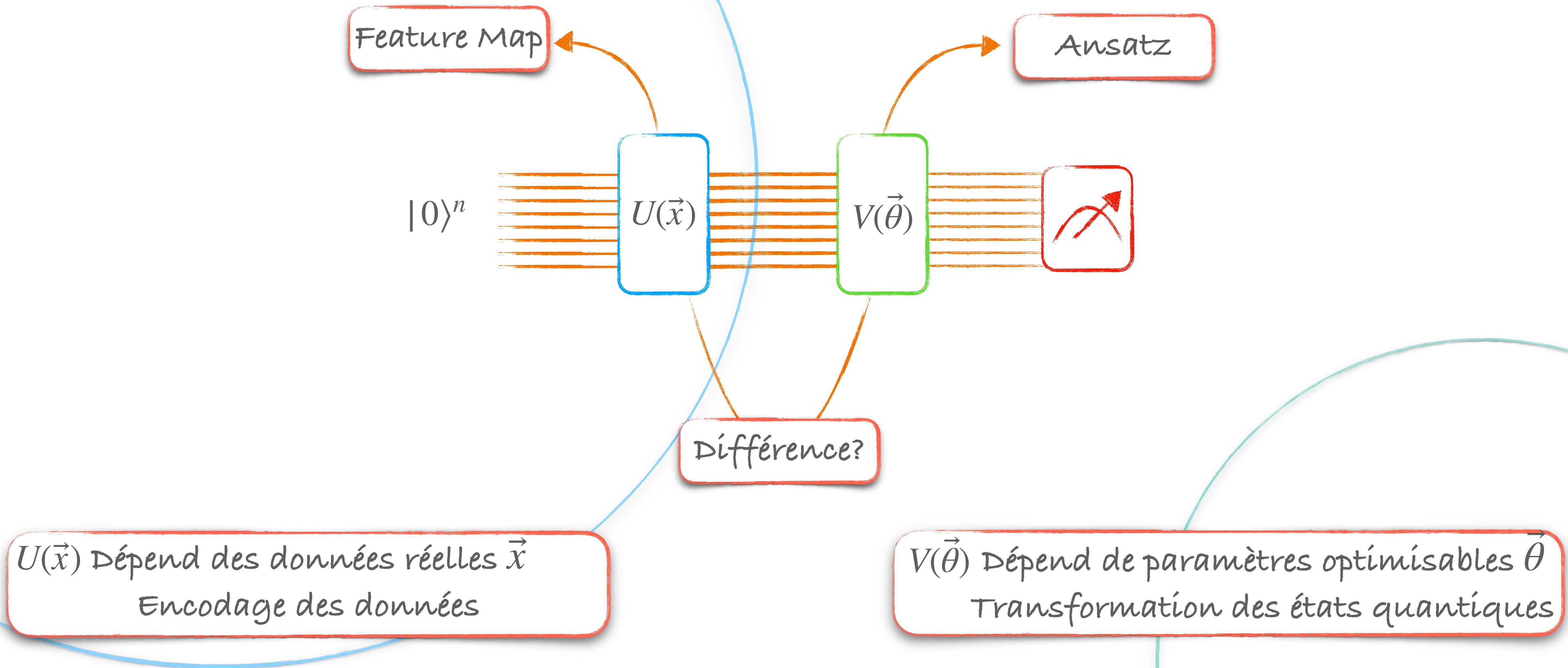
Introduction



Introduction

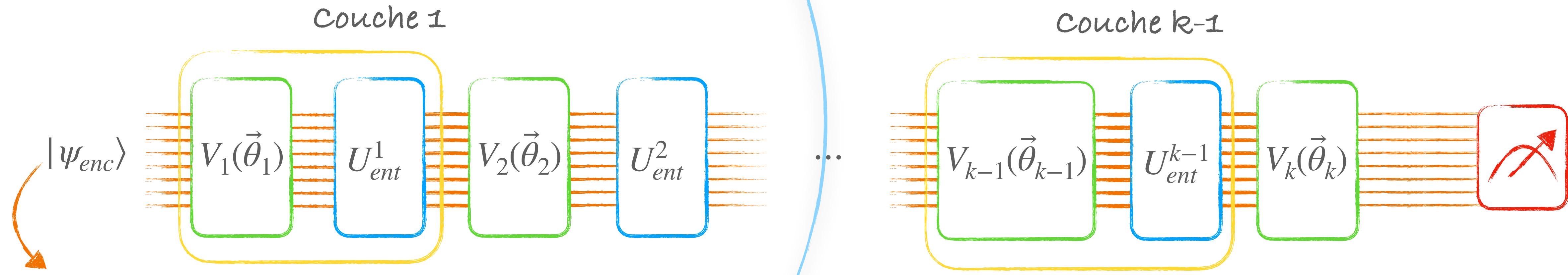


Introduction



Structure

Forme générale



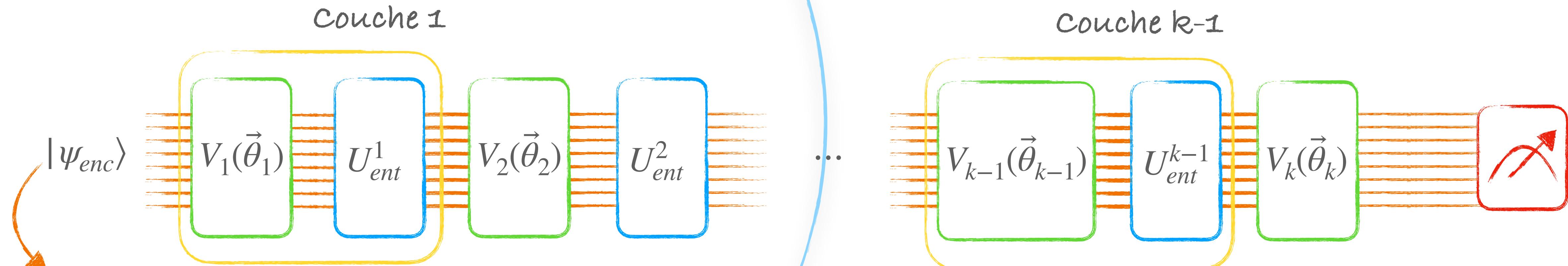
Feature Map

Paramètres $\vec{\theta}_1 \neq \vec{\theta}_2 \neq \dots \neq \vec{\theta}_k$

U_{ent} Intrication linéaire, circulaire ou complète

Structure

Forme générale



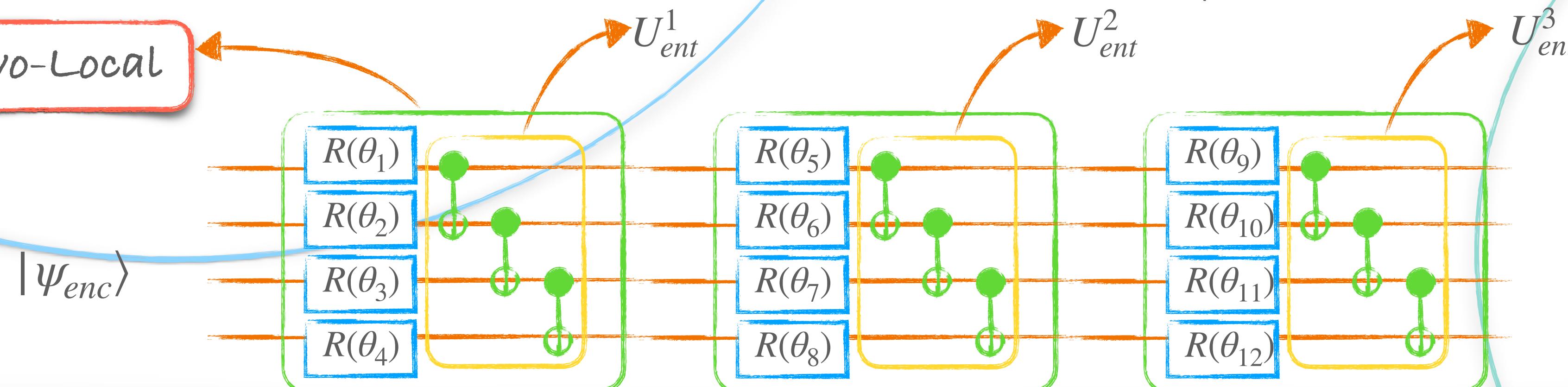
Feature Map

Paramètres $\vec{\theta}_1 \neq \vec{\theta}_2 \neq \dots \neq \vec{\theta}_k$

U_{ent} intrication linéaire, circulaire ou complète

TWO-Local

Exemple

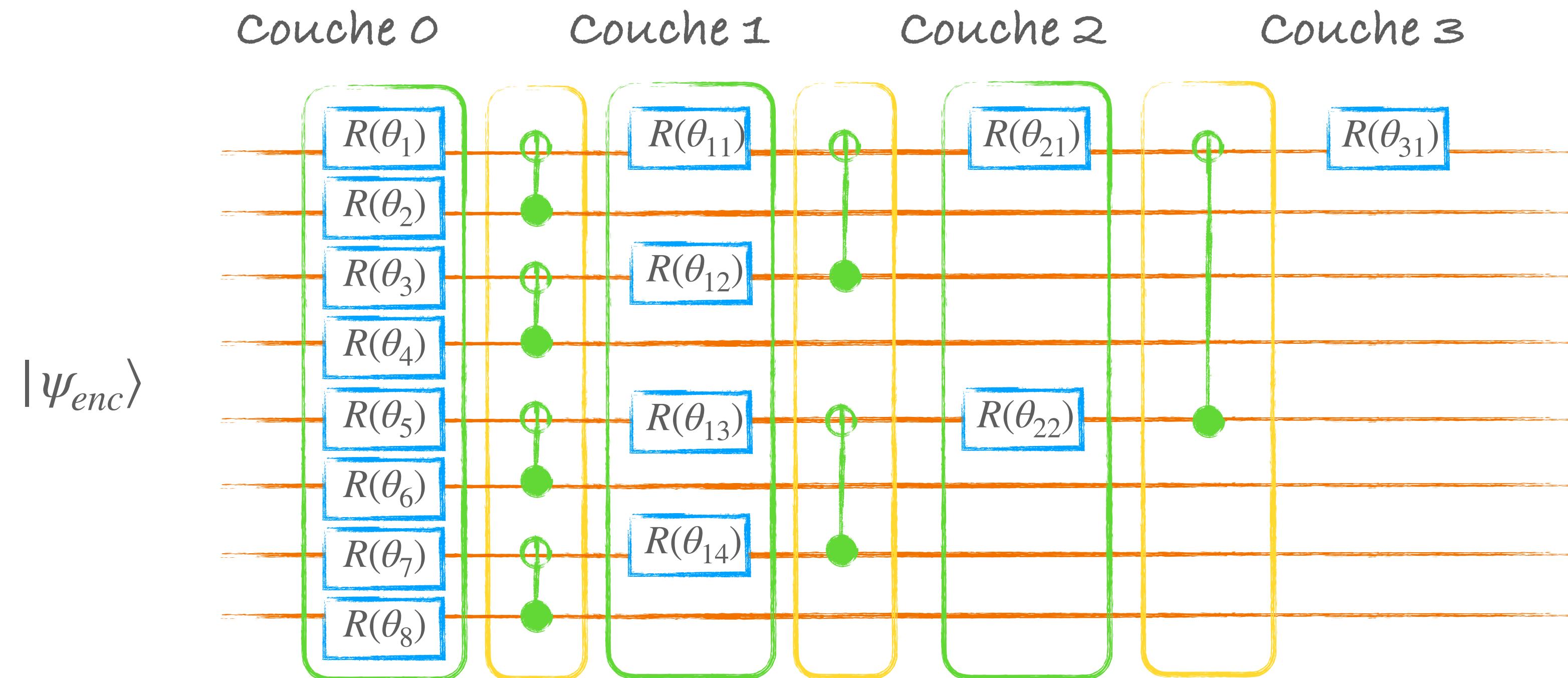


Structure

Tree Tensor

Pour $k+1$ layer il faut 2^k qubits

chaque couche aura la moitié du nombre de paramètres que la couche précédente

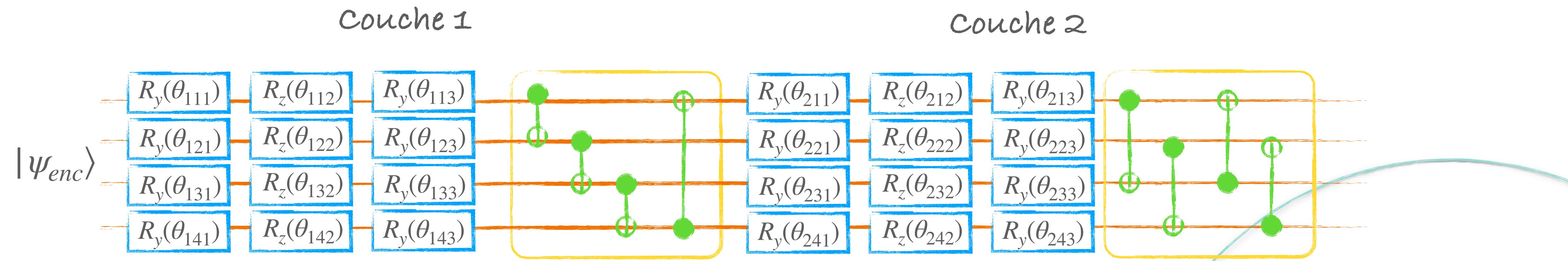


$8 = 2^3$ qubits pour $3+1$ couches

Application:
Classification
binaire

Structure

Strongly Entangled Layer



Entraînement

Algorithmes d'optimisation

Comment calculer le gradient avec un circuit quantique?

Approximation Numérique d'une dérivée...

Always works...

$$\frac{\partial f}{\partial x_j} = \frac{f(x_1, \dots, x_j + h, \dots, x_n) - f(x_1, \dots, x_n)}{h}$$

Pour un tout petit h et n input réels avec

$$f: \mathbb{R}^n \rightarrow \mathbb{R}$$

Problème: On va devoir exécuter un nombre infini de fois le QNN...

Entraînement

Algorithmes d'optimisation

Comment calculer le gradient avec un circuit quantique?

Approximation Numérique d'une dérivée...

Always works...

$$\frac{\partial f}{\partial x_j} = \frac{f(x_1, \dots, x_j + h, \dots, x_n) - f(x_1, \dots, x_n)}{h}$$

Pour un tout petit h et n input réels avec

$$f: \mathbb{R}^n \rightarrow \mathbb{R}$$

Problème: On va devoir exécuter un nombre infini de fois le QNN...

Définition automatique → simulateurs

Backpropagation ←

calcul du gradient parfait
chain rules

classical
Computer

Problème: Ne peut être utilisé que sur simulateurs

Entraînement

Algorithmes d'optimisation

Comment calculer le gradient avec un circuit quantique?

Parameter shift-rule utilisé sur les hardwares réels.

Comme l'approximation numérique on va faire varier les paramètres du circuits à optimiser.

Exemple: cas d'un circuit avec une porte de rotation $|0\rangle \xrightarrow{R_x(\theta)} \text{meter}$

On peut estime la valeur attendue $E(\theta)$ et les dérivées par:

$$\nabla_{\theta} E(\theta) = \frac{1}{2} \left(E\left(\theta + \frac{\pi}{2}\right) - E\left(\theta - \frac{\pi}{2}\right) \right)$$

Problème: N'est pas applicable dans tous les cas.

Doit être utilisé avec d'autres techniques

Entraînement

En pratique

- Faire des choix avisés - Trois composants principaux:

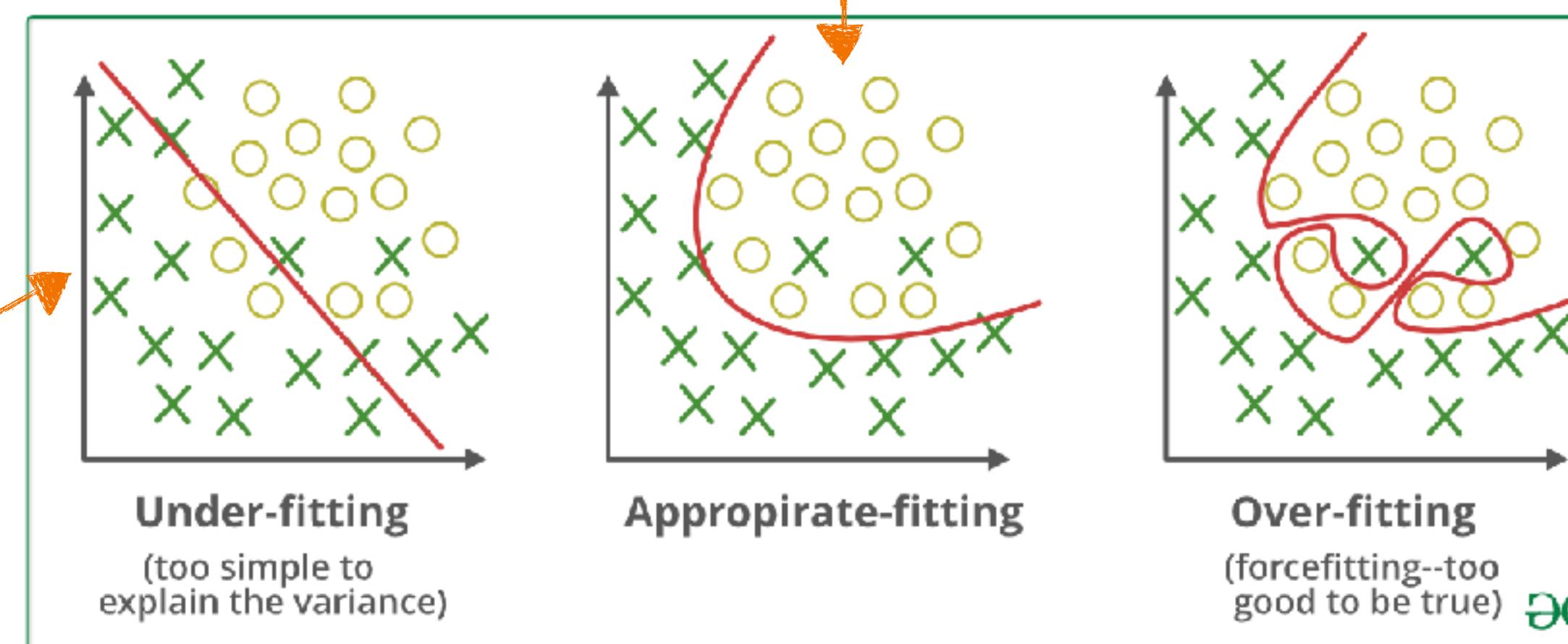
Feature Map
Ansatz
Opération de mesure

Dépend du problème
Le Barren Plateau est votre ennemi
Explorer la littérature pour des problèmes similaires

- La taille a son importance

Nombre de paramètres à optimiser dans l'ansatz

Si l'ansatz n'a pas assez de paramètres



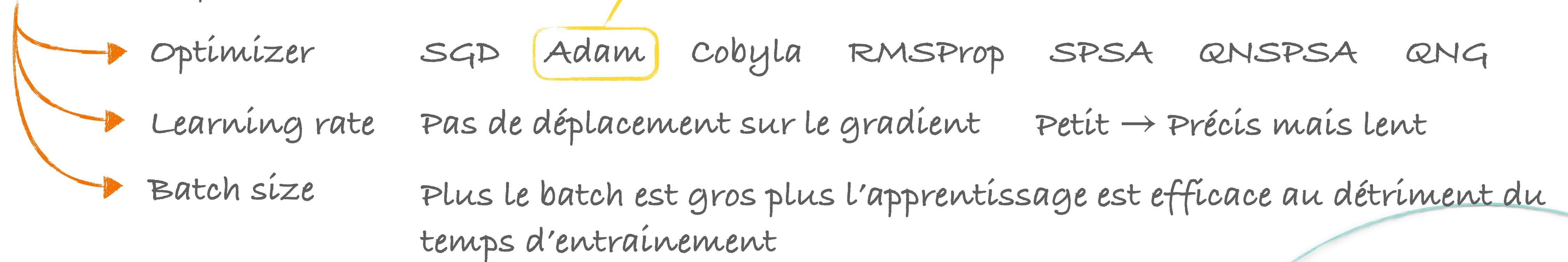
Si l'ansatz a le bon nombre de paramètres

Si l'ansatz a trop de paramètres

Entraînement

En pratique

• Paramètres d'optimisation

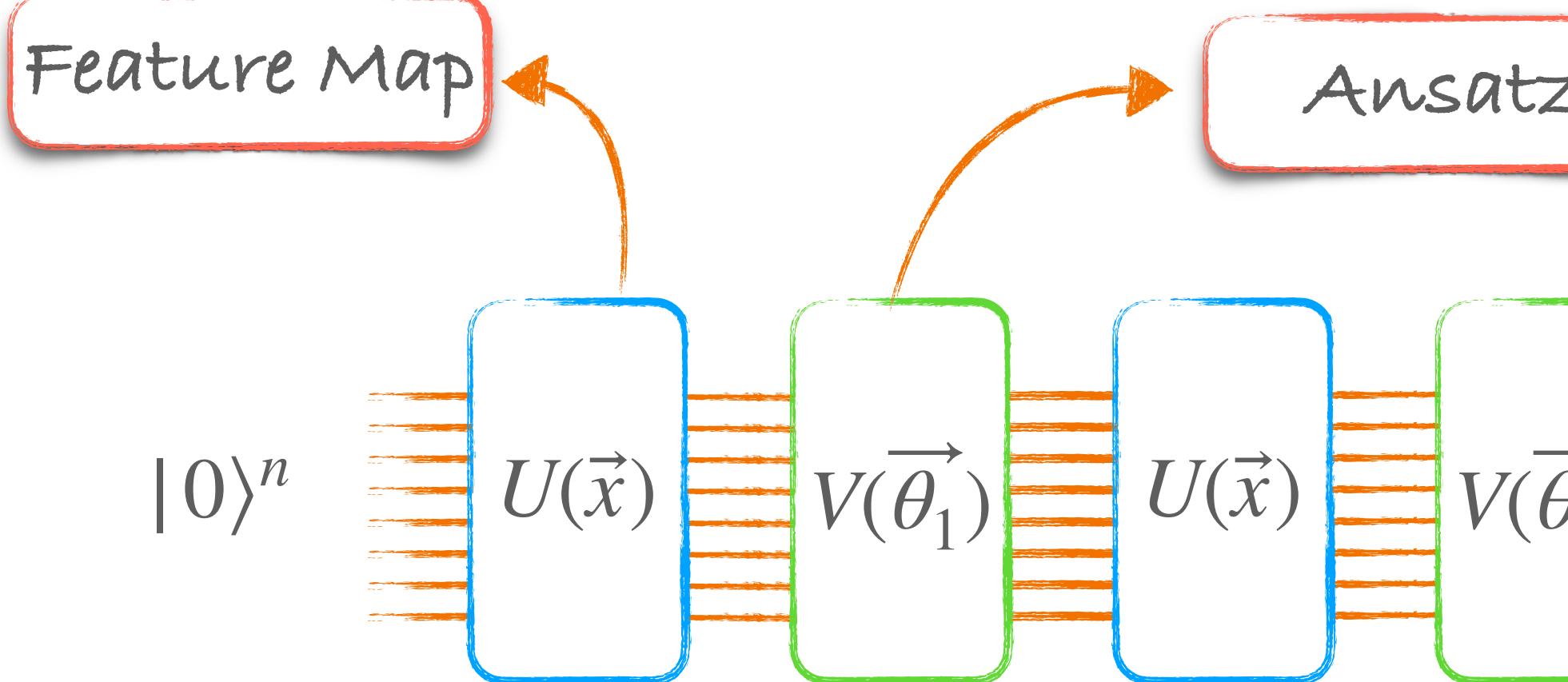


• Les données



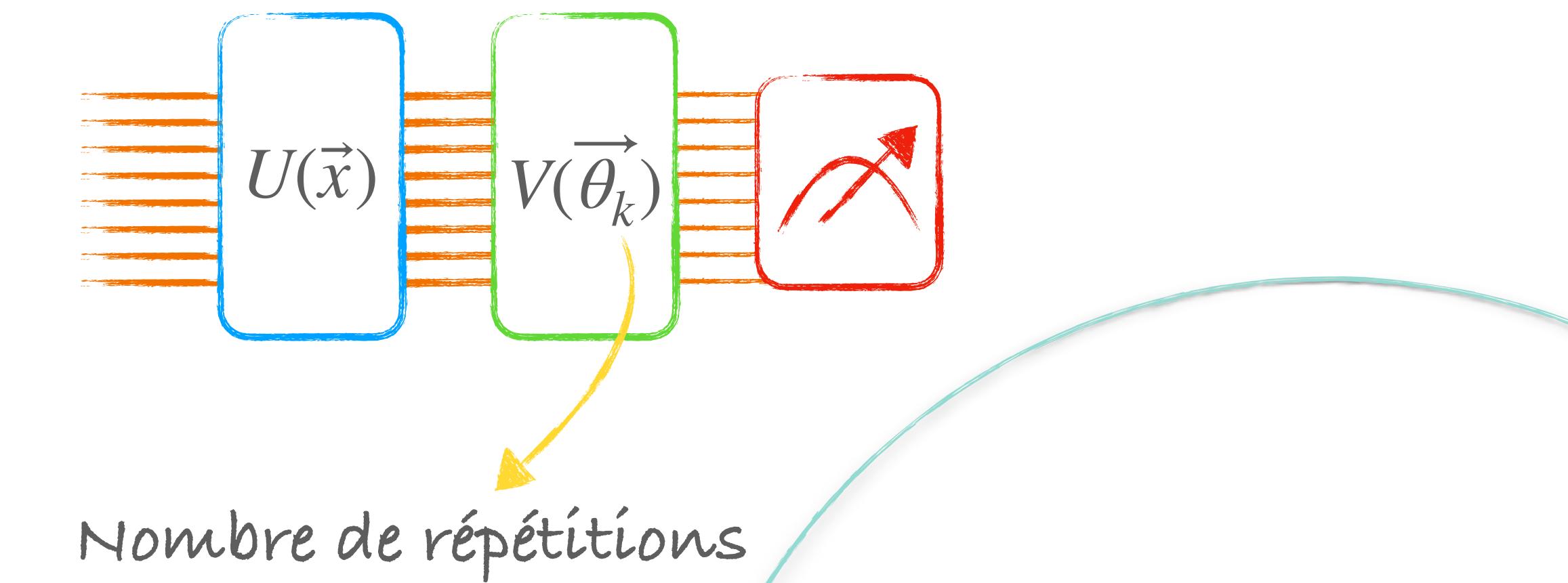
Entraînement

"Data reuploading"



Avec les vecteurs $\vec{\theta}_1, \vec{\theta}_2, \dots, \vec{\theta}_k$ indépendants

Les ansatz peuvent être différentes



Hybride



Hybride

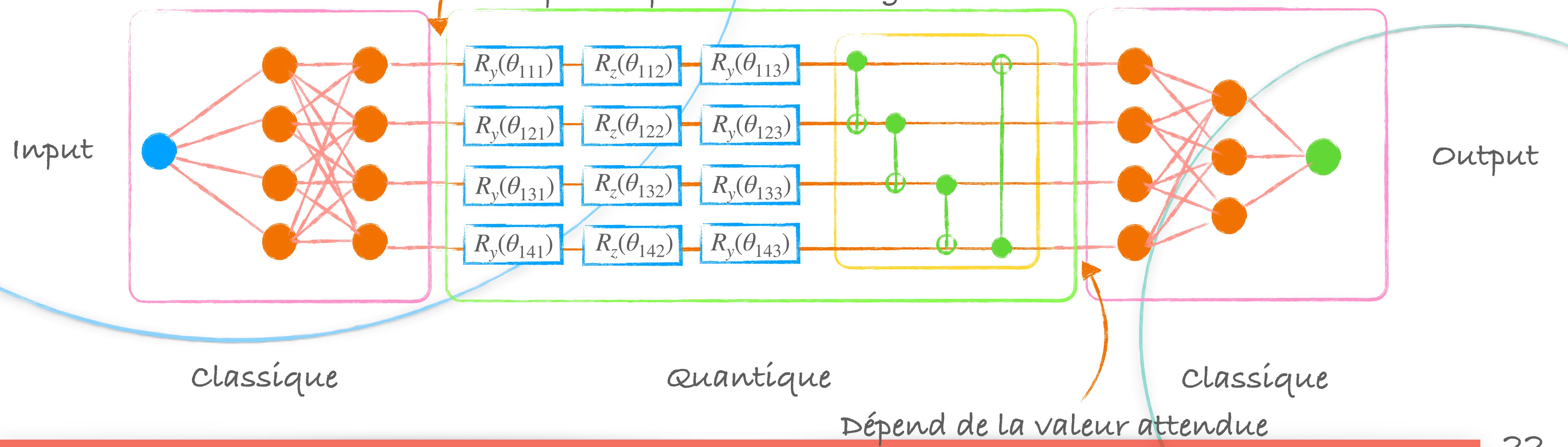
Définition

"QAOA, VQE, etc. nécessitent d'utiliser le calcul classique pour l'entraînement."

Architecture hybride veut dire que des parties du modèle sont conjointement classique et quantique. L'entraînement prend ce tout comme une unité.

Par exemple:

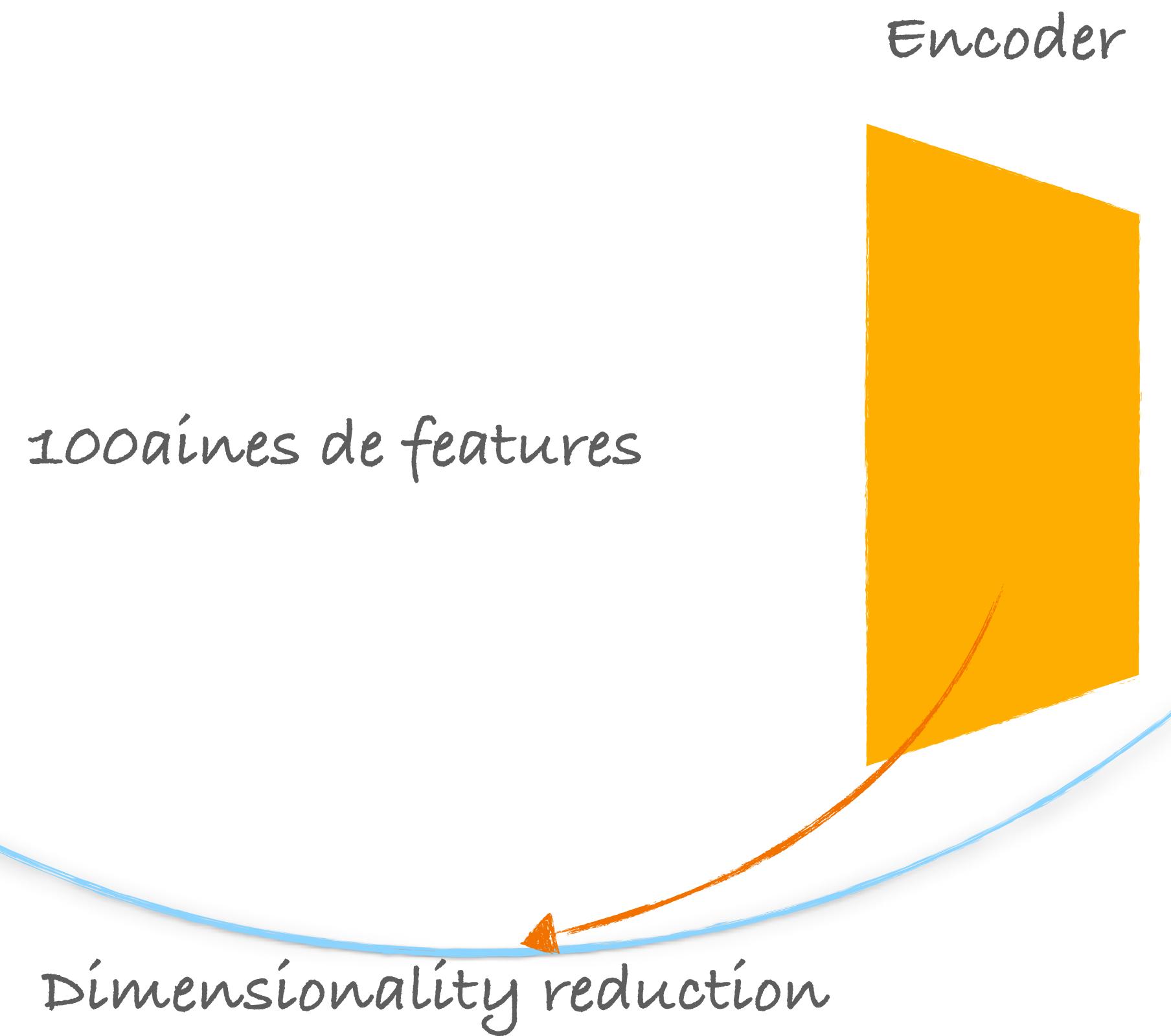
Peut contenir une feature map pour diminuer le nombre de qubits par amplitude encoding



Hybride

utilité?

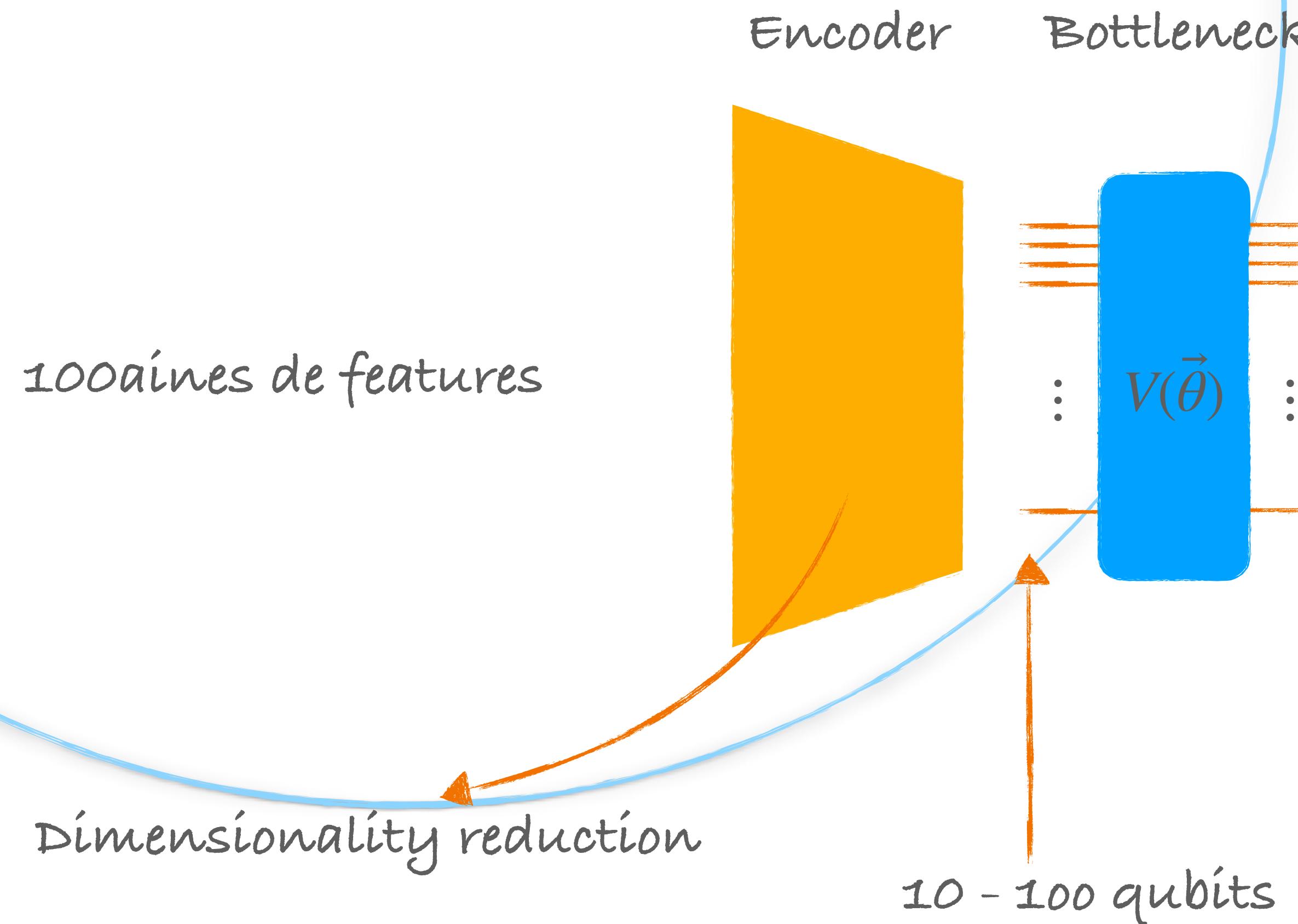
Actuellement, le nombre de qubits est limité.



Hybride

utilité?

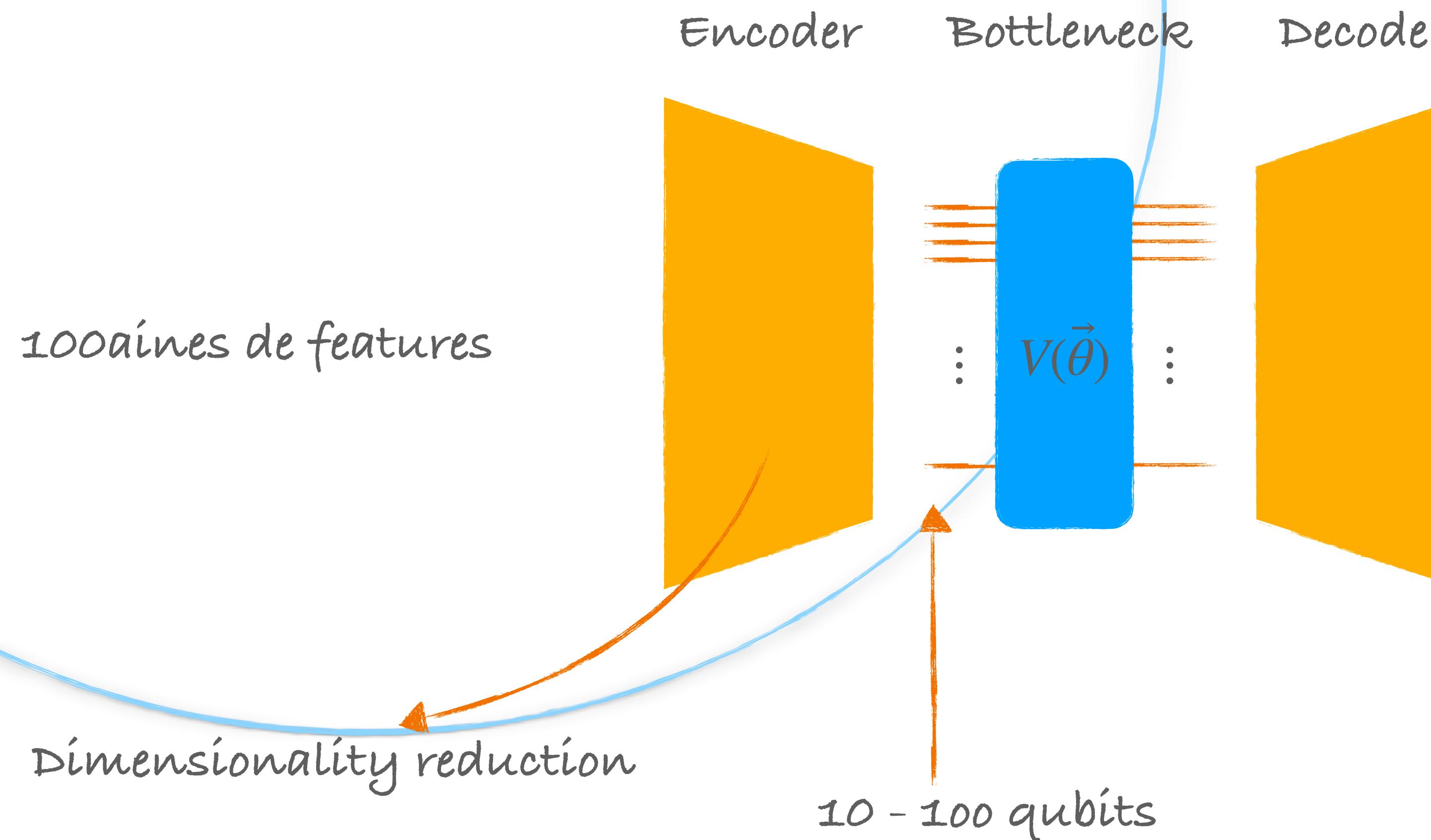
Actuellement, le nombre de qubits est limité.



Hybride

utilité?

Actuellement, le nombre de qubits est limité.

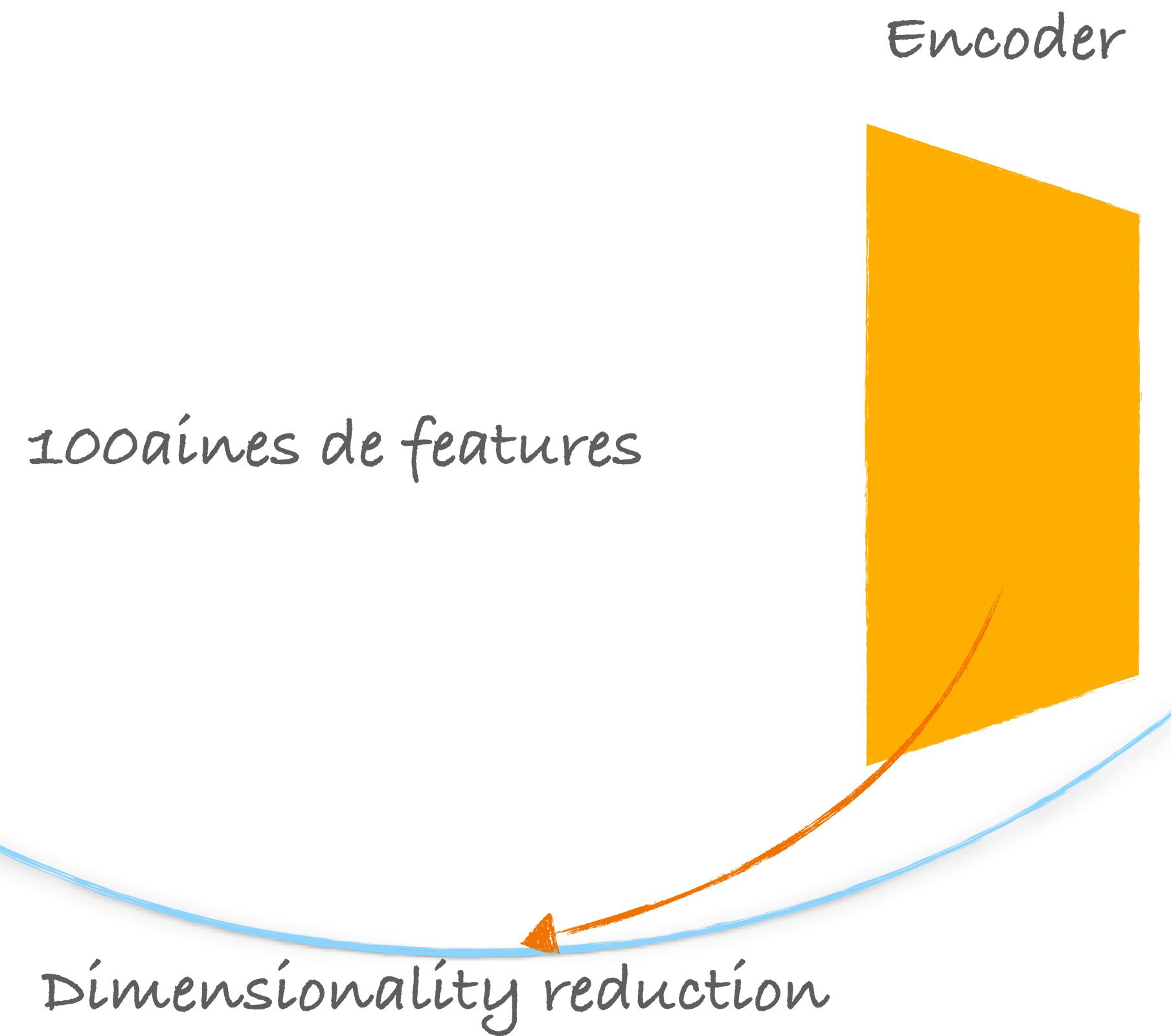


AutoEncoder quantique

Hybride

utilité?

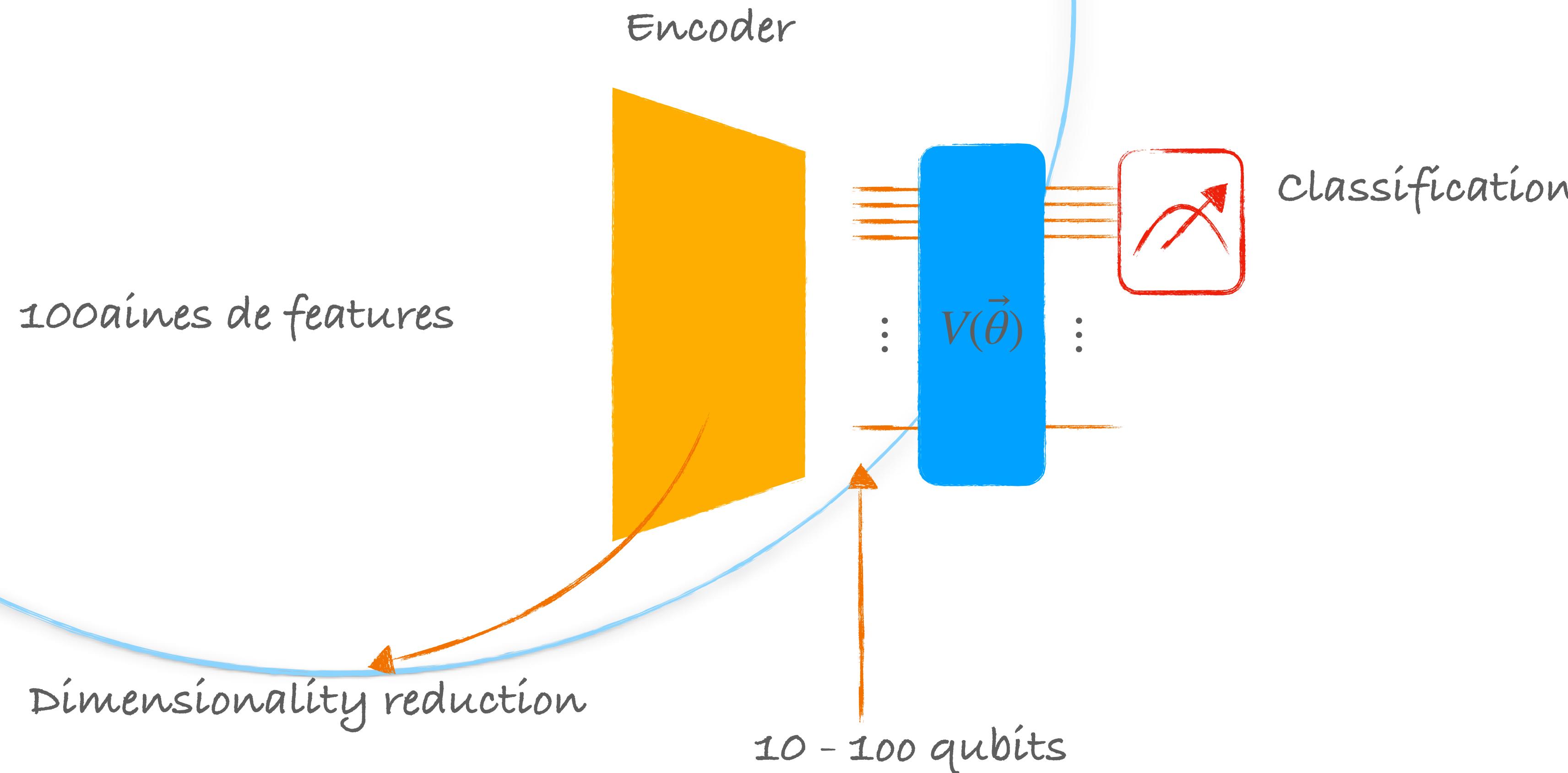
Actuellement, le nombre de qubits est limité.



Hybride

utilité?

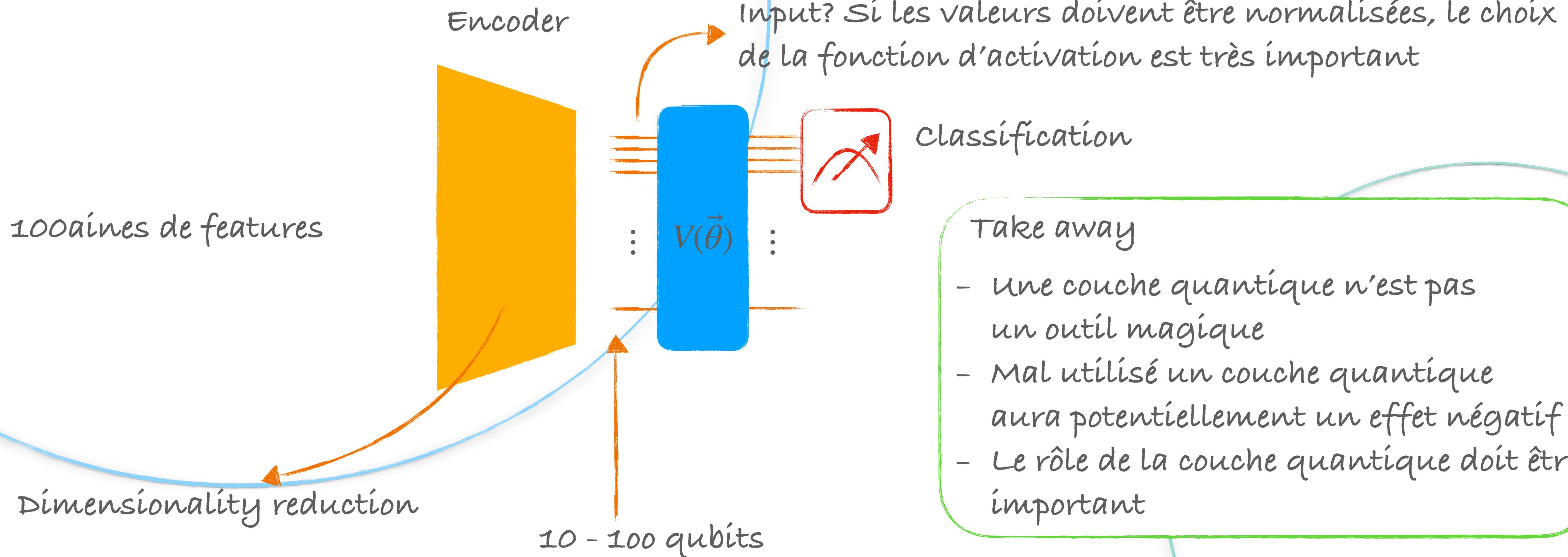
Actuellement, le nombre de qubits est limité.



Hybride

utilité?

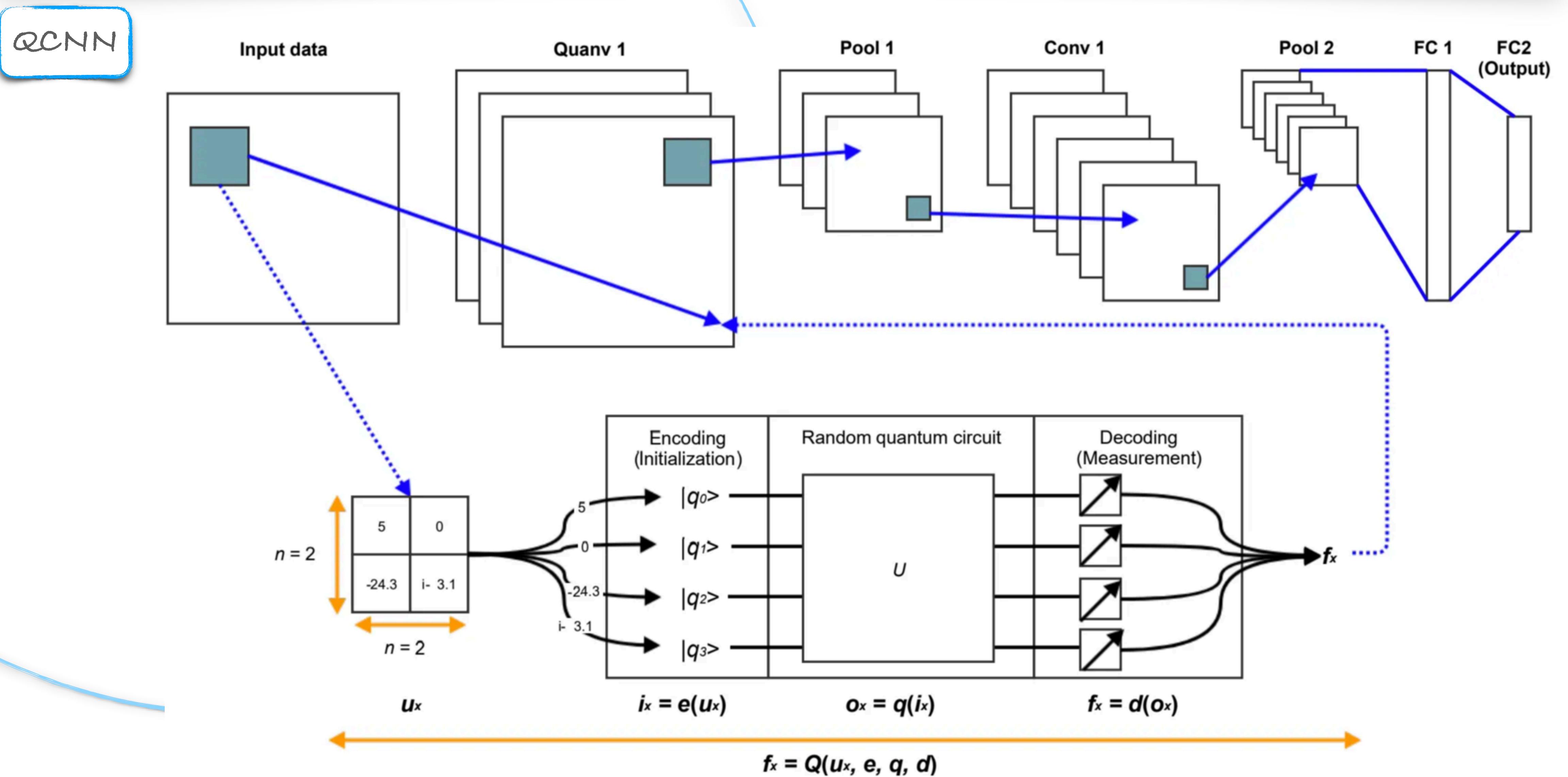
Actuellement, le nombre de qubits est limité.



Take away

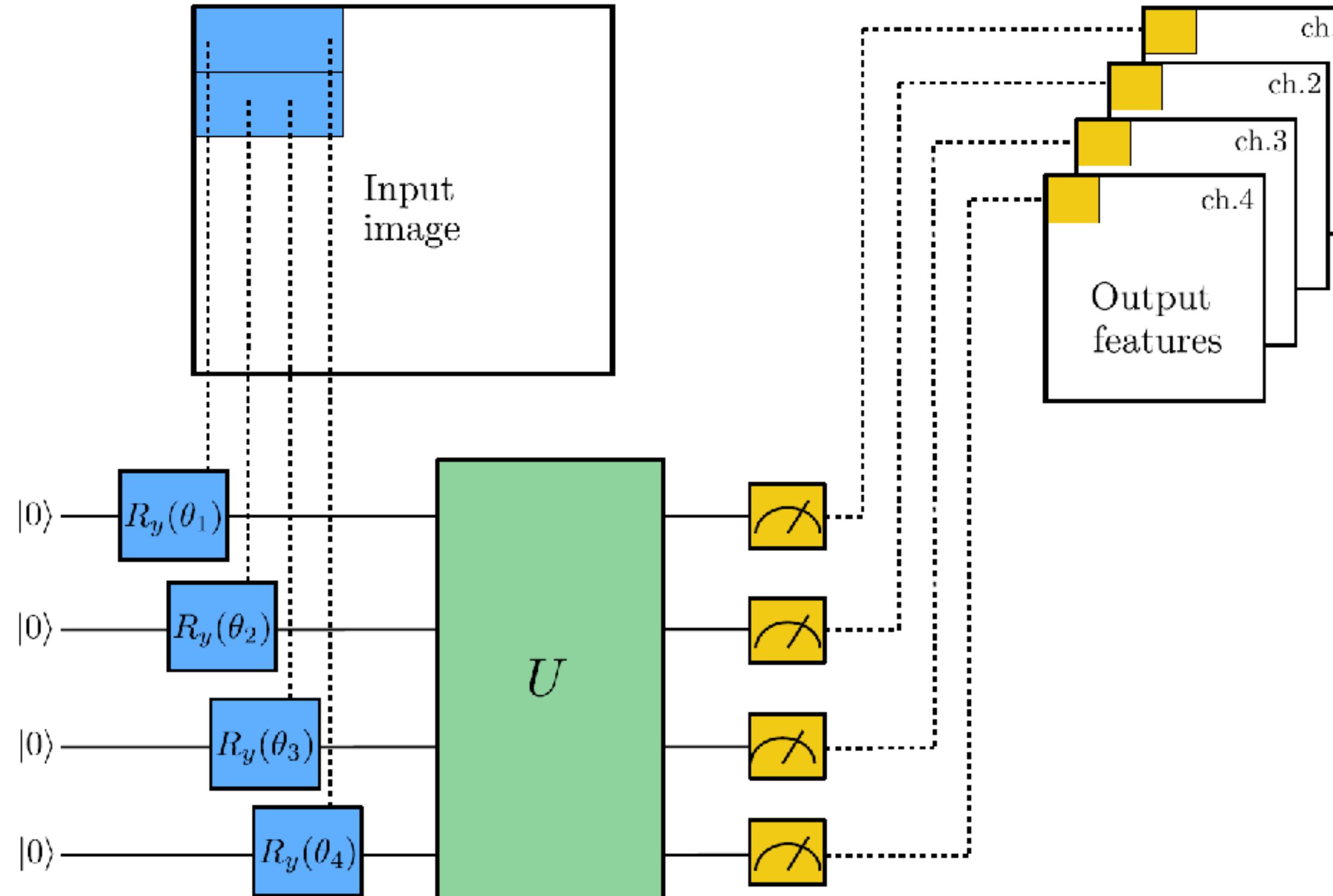
- Une couche quantique n'est pas un outil magique
- Mal utilisé un couche quantique aura potentiellement un effet négatif
- Le rôle de la couche quantique doit être important

Hybride



Hybride

QCNN

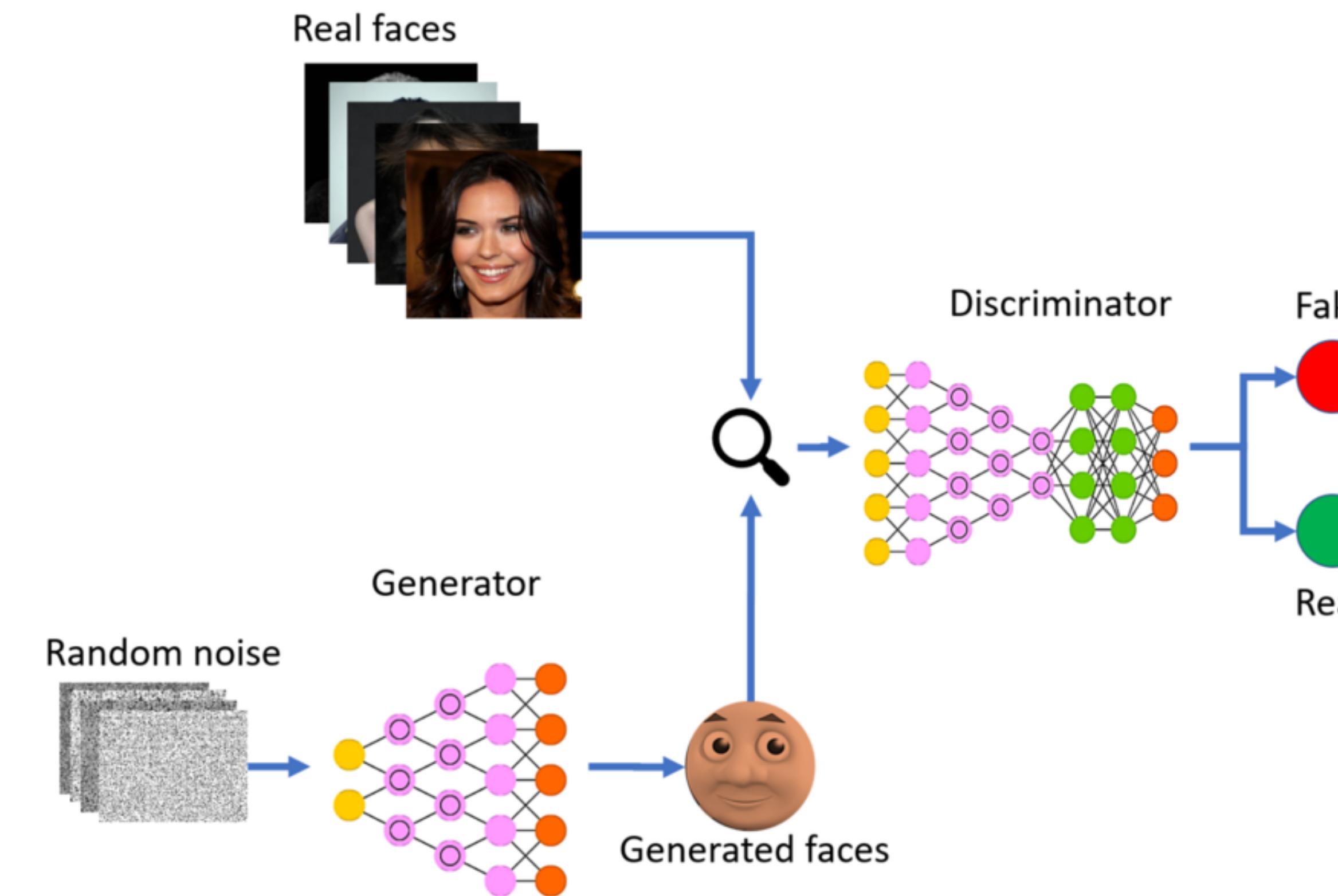


Démo!

Quantum Generative Models

Quantum Generative Models

QGAN



Quantum Generative Models

QGAN

cas d'usages

- Données quantiques
- Générateur quantique
- Discriminateur quantique

- États quantiques
- Pas besoin de feature map ou de mesure dans le processus

Quantum Generative Models

QGAN

cas d'usages

Données quantiques
Générateur quantique
Discriminateur quantique

- États quantiques
- Pas besoin de feature map ou de mesure dans le processus

Données quantiques
Générateur quantique
Discriminateur classique

- États quantiques
- Mesure dans le processus
- Mesure des données quantiques

Quantum Generative Models

QGAN

cas d'usages

- Données quantiques
- Générateur quantique
- Discriminateur quantique

- États quantiques
- Pas besoin de feature map ou de mesure dans le processus

- Données quantiques
- Générateur quantique
- Discriminateur classique

- États quantiques
- Mesure dans le processus
- Mesure des données quantiques

Le plus étudié

- Données classiques
- Générateur quantique ou classique
- Discriminateur quantique ou classique

- Feature Map pour encoder les données

Références

- [1] Zhang et al., 2023, Dive into Deep Learning, d2l.ai, chapter 5, 7, 9
- [2] Combarro E. F. & González-Castillo S., 2023, A Practical Guide to Quantum Machine Learning and Quantum Optimization, Chapters 10, 11, 12
- [3] Schuld M. & Petruccione F., 2021, Machine Learning with Quantum Computers, Chapter 7, 8
- [4] <https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/>
- [5] https://pennylane.ai/qml/demos/tutorial_quanvolution/
- [6] https://pennylane.ai/qml/demos/tutorial_data_reuploading_classifier/
- [7] https://pennylane.ai/qml/demos/tutorial_qgrnn/
- [8] https://pennylane.ai/qml/demos/tutorial_quantum_gans/
- [9] https://pennylane.ai/qml/demos/tutorial_QGAN/
- [10] https://pennylane.ai/qml/glossary/parameter_shift/