

# MACHINE LEARNING EN PYTHON

**MARTELLI GINO – SENIS TAHITOÀ – VIEVILLE SÉBASTIEN**

# PLAN

INTRODUCTION

PRÉPARATION DES DONNÉES

FEATURES ENGINEERING

MODÉLISATION ET CLASSIFICATION

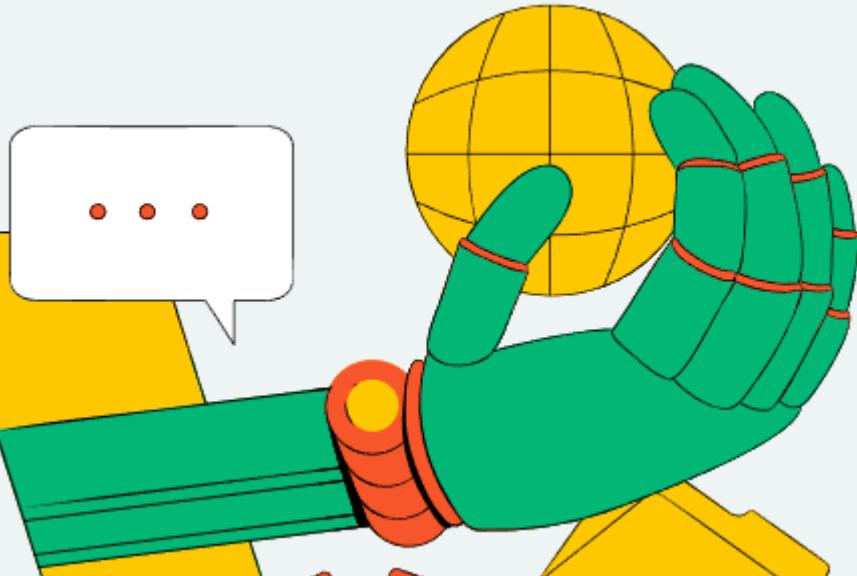
NEURAL NETWORK

CONCLUSION



# INTRODUCTION

LE MACHINE LEARNING PERMET À UN SYSTÈME D'APPRENDRE À PARTIR DES DONNÉES AFIN D'EFFECTUER DES PRÉDICTIONS SUR DES NOUVELLES DONNÉES.



OBJECTIF : LA MACHINE DOIT POUVOIR DÉTERMINER QUEL CHIFFRE EST SUR UNE IMAGE DE 8\*8 PIXELS QU'ELLE N'A JAMAIS VUS.

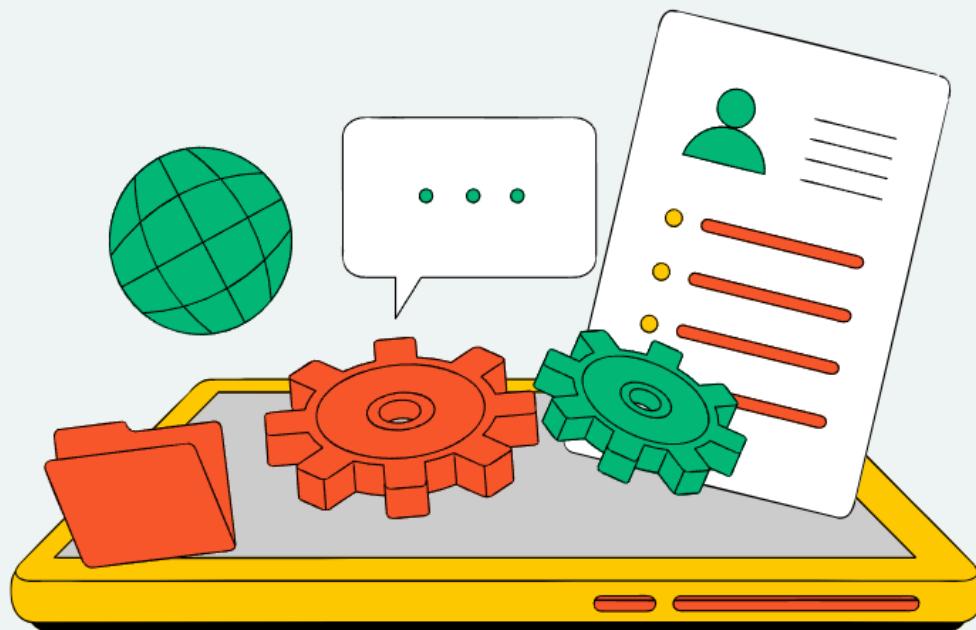


# MACHINE LEARNING

**APPRENTISSAGE PAR RENFORCEMENT**

**APPRENTISSAGE SUPERVISÉ**

**APPRENTISSAGE NON SUPERVISÉ**

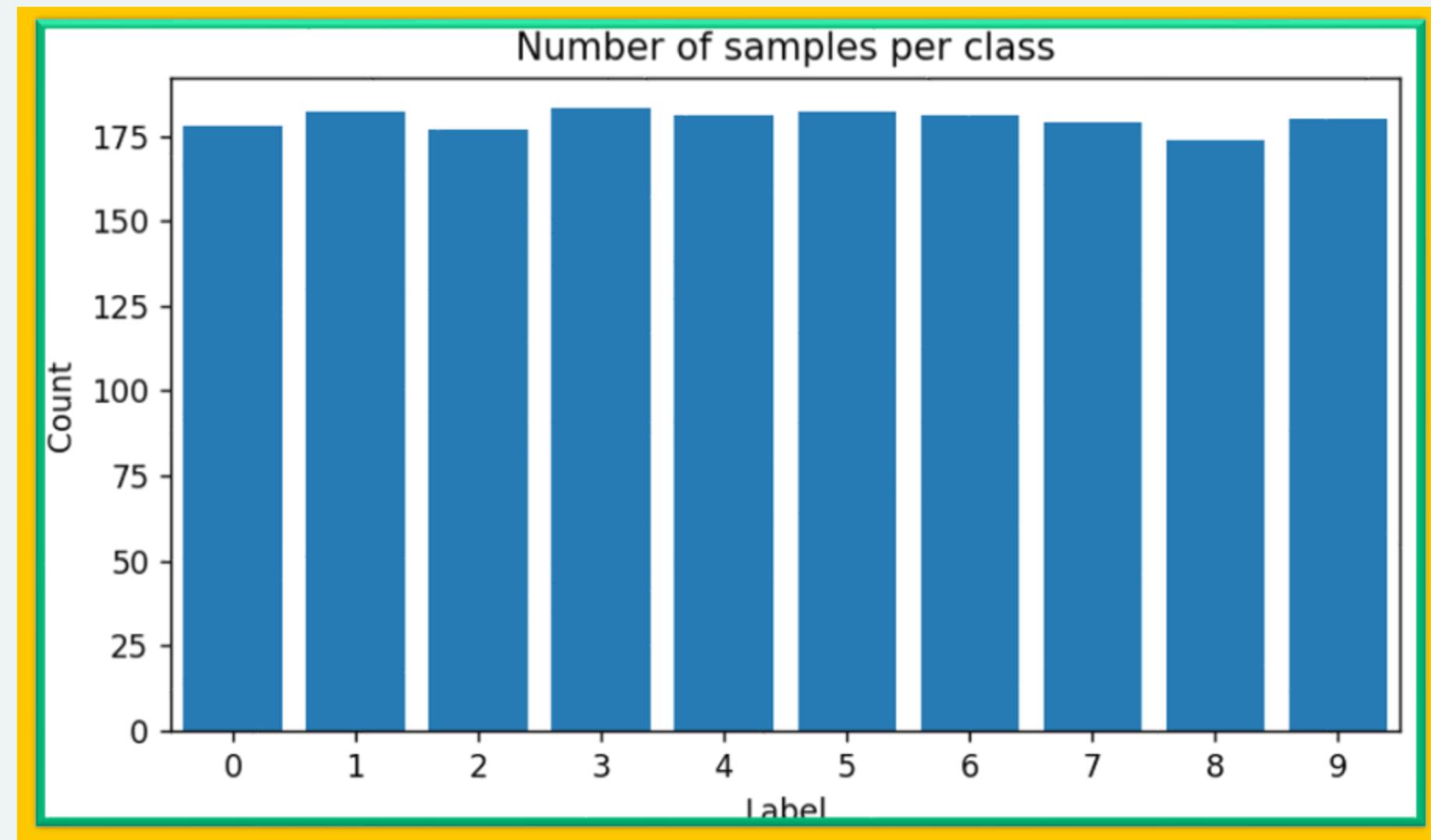


ON FOURNIT À LA MACHINE DES EXEMPLES AVEC LEUR RÉPONSES

LE MODÈLE APPREND À RECONNAÎTRE LES MOTIFS VISUELS ET À PRÉDIRE LA BONNE CLASSE

# PRÉPARATION DES DONNÉES

1797 EXEMPLES



# PRÉPARATION DES DONNÉES

## NORMALISATION

ON NORMALISE POUR ÉVITER LES DIFFÉRENCES DE DYNAMIQUE DANS LES VALEURS



VALEURS DES PIXELS ENTRE 0 ET 1



# FEATURES ENGINEERING

**FEATURES  
ENGINEERING**

ON CONSERVE SEULEMENT DES VALEURS PERTINENTES POUR SIMPLIFIER L'APPRENTISSAGE



IMAGES DE DIMENSIONS 64 À 24

**ACP**

20 DIMENSIONS

**ZONES**

3 DIMENSIONS

**CONTOURS**

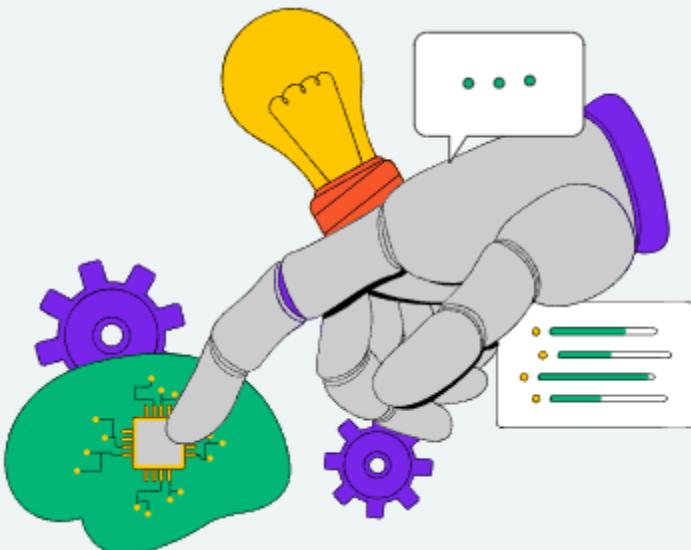
1 DIMENSION



# FEATURES ENGINEERING

## ANALYSE EN COMPOSANTES PRINCIPALES

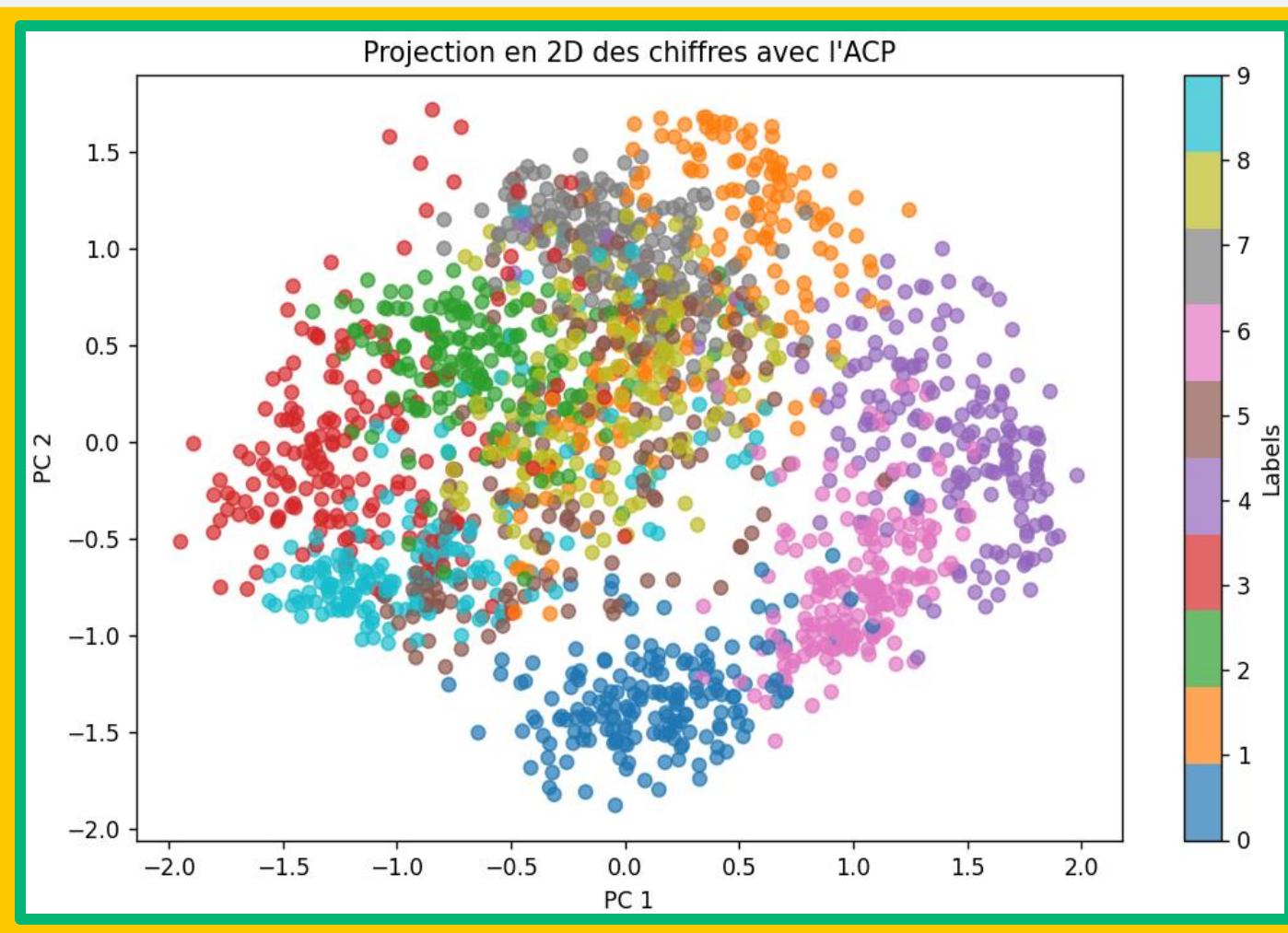
ON TRANSFORME LES VARIABLES INITIALES CORRÉLÉES  
EN UN NOUVEAU JEU DE VARIABLES NON CORRÉLÉES.



RÉDUCTION DE LA DIMENSION DES IMAGES TOUT  
EN CONSERVANT LE PLUS D'INFORMATIONS.

# FEATURES ENGINEERING

## ANALYSE EN COMPOSANTES PRINCIPALES



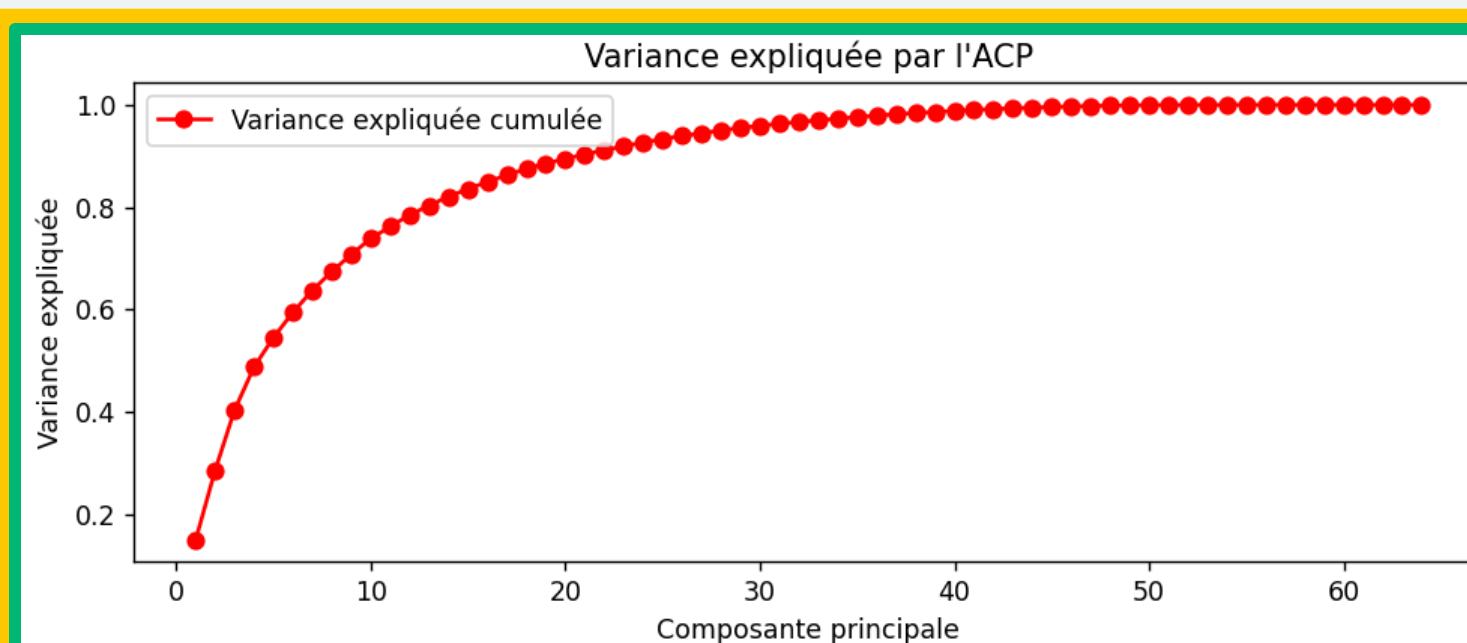
UTILISER SEULEMENT 2  
COMPOSANTES  
PRINCIPALES N'EST PAS  
SUFFISANT POUR PRÉDIRE  
CORRECTEMENT LE LABEL  
D'UNE IMAGE

# FEATURES ENGINEERING

## ANALYSE EN COMPOSANTES PRINCIPALES

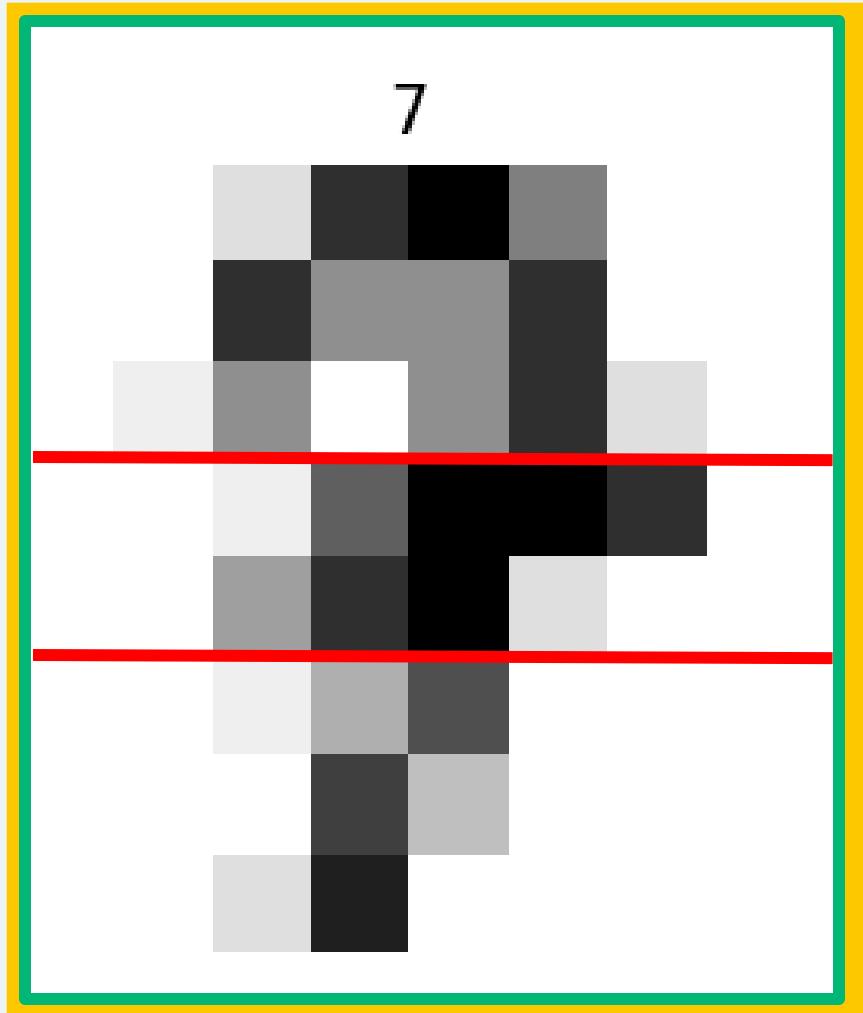


ON PREND 20 COMPOSANTES PRINCIPALES AFIN D'EXPLIQUER ENTRE 90 ET 95% DES VARIANCES AVEC UNE ERREUR DE 0,004.



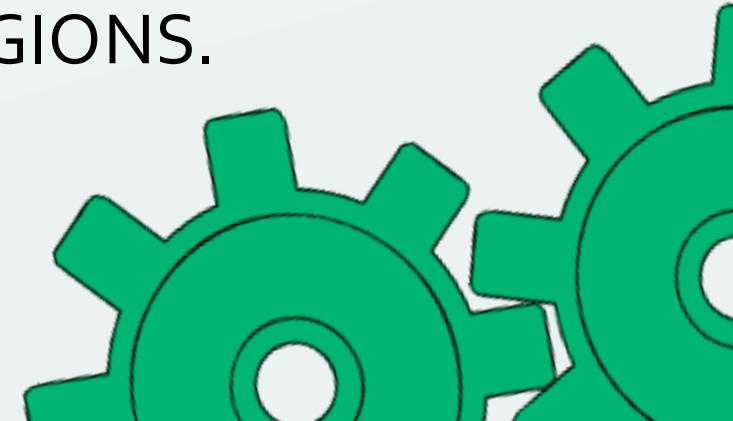
# FEATURES ENGINEERING

## PARTITIONNEMENT EN ZONES



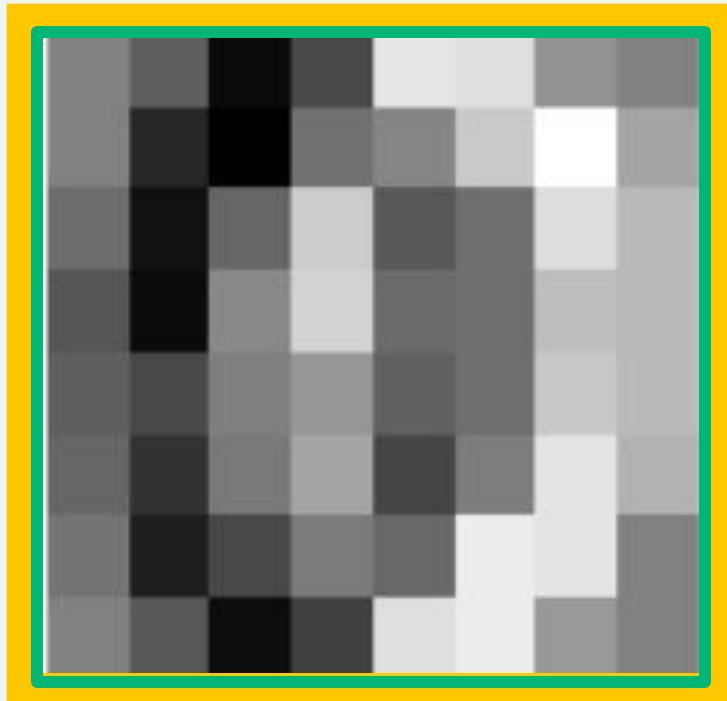
DIVISION DE L'IMAGE EN PLUSIEURS RÉGIONS POUR MIEUX ANALYSER LA STRUCTURE : RÉGION DU HAUT, DU MILIEU ET DU BAS.

ON PREND LA MOYENNE D'INTENSITÉ DE CHACUNE DES RÉGIONS.



# FEATURES ENGINEERING

## DÉTECTION DES CONTOURS



SOBEL HORIZONTALE

FILTRE DE SOBEL POUR METTRE EN ÉVIDENCE LES STRUCTURES PRINCIPALES DE CHAQUE CHIFFRE.

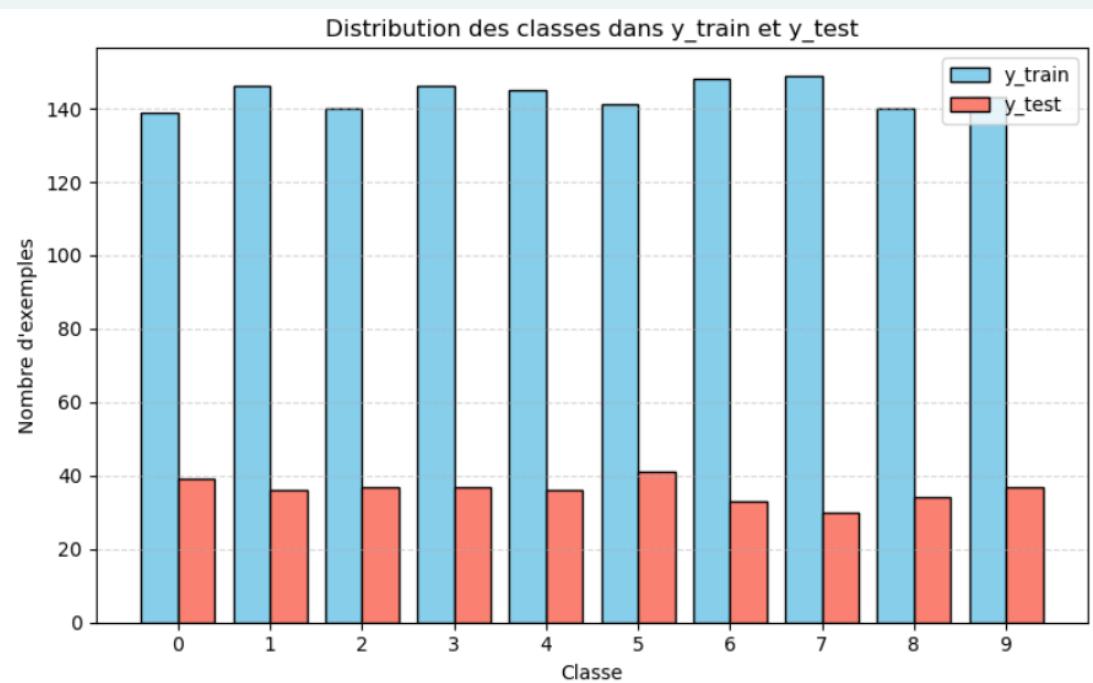
ON PREND LA MOYENNE DE LA VARIATION D'INTENSITÉ HORIZONTALE ET VERTICALE.



# MODÉLISATION ET CLASSIFICATION

## DIVISION DES DONNÉES

80/20 % de train/test



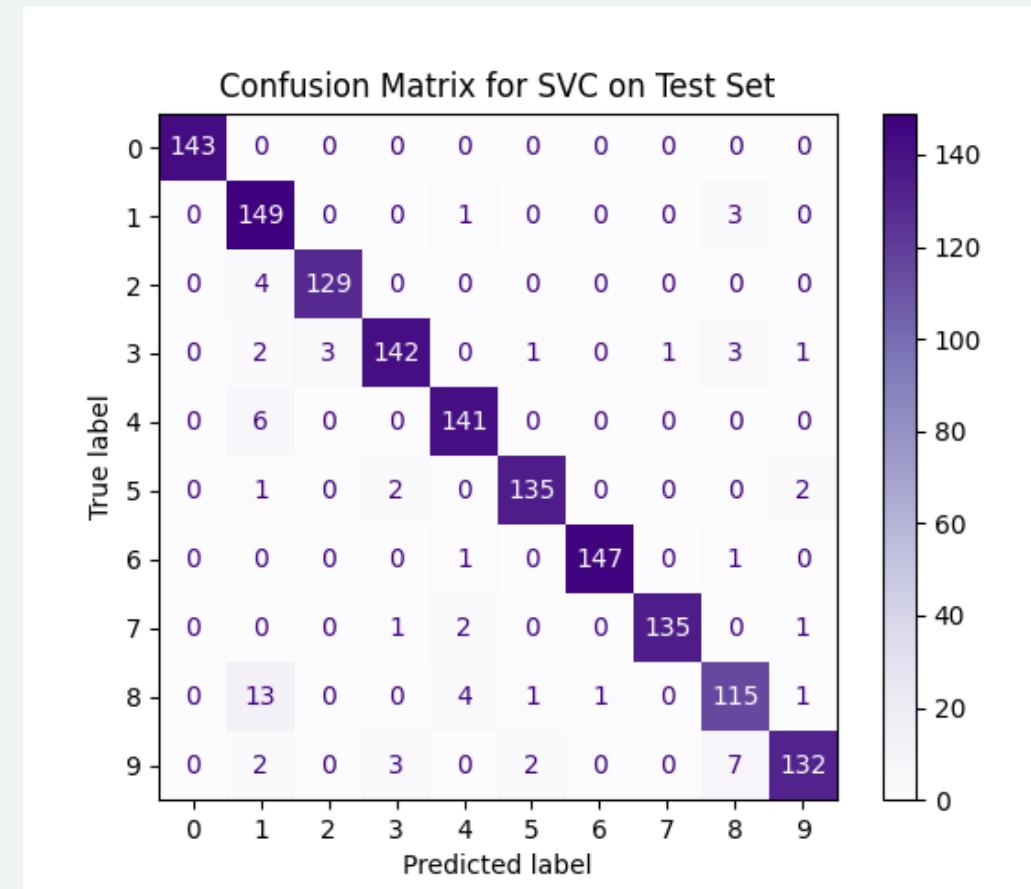
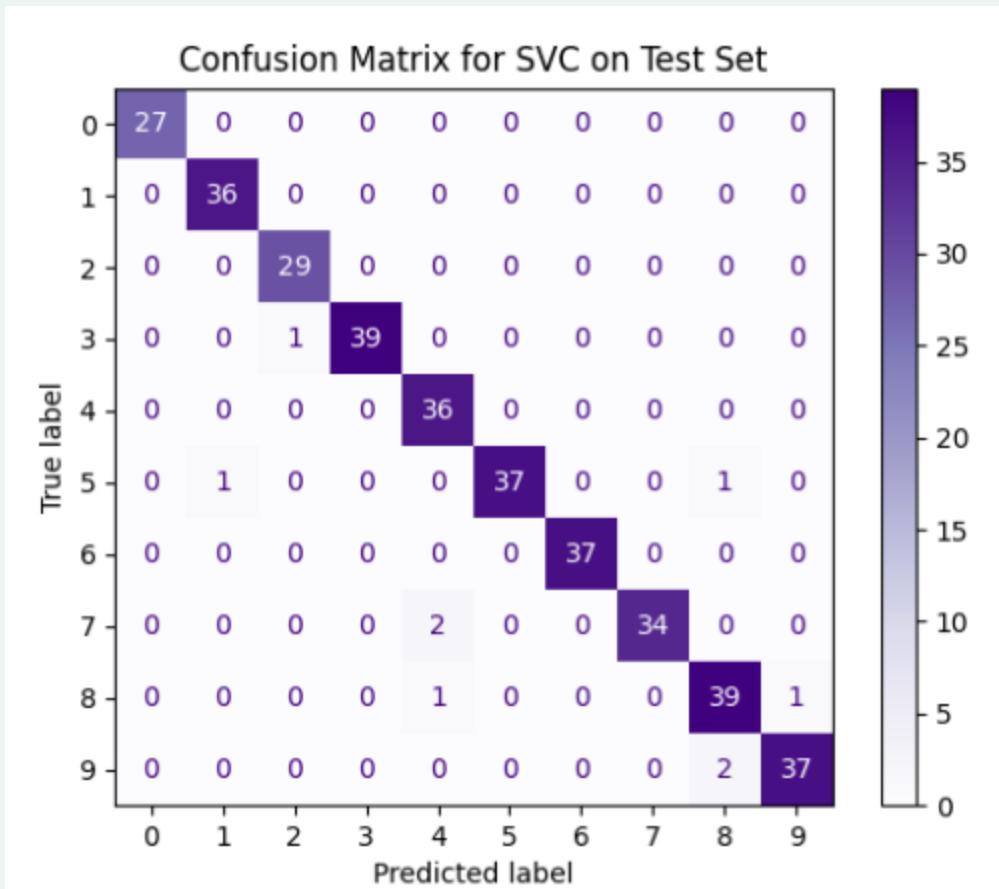
## MÉTHODE DE CLASSIFICATION

**Support vector classification (SVC) :**  
Le SVC consiste à tracer le meilleur hyperplan possible pour séparer toutes les classes.

Accuracy of the SVC on the test set: 0.98333333333333  
Accuracy of the SVC on the train set: 0.9979123173277662

# MODÉLISATION ET CLASSIFICATION

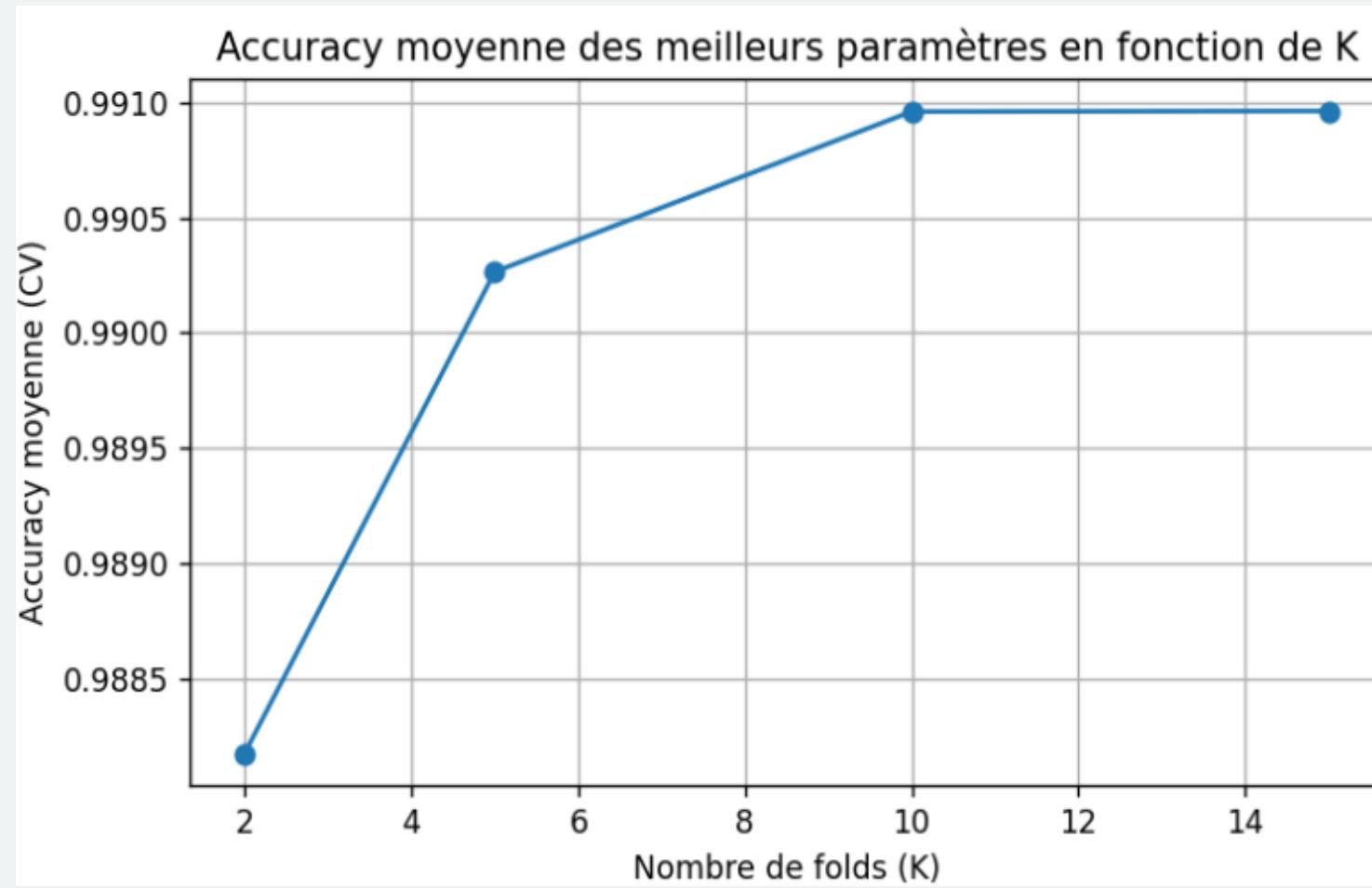
## CONFUSION MATRIX



# MODÉLISATION ET CLASSIFICATION

## GRID SEARCH

Nous utilisons la fonction “GridSearchCV” pour trouver les paramètres optimaux (nombre de composantes PCA, type de kernel, valeur de C, méthode de scaling).



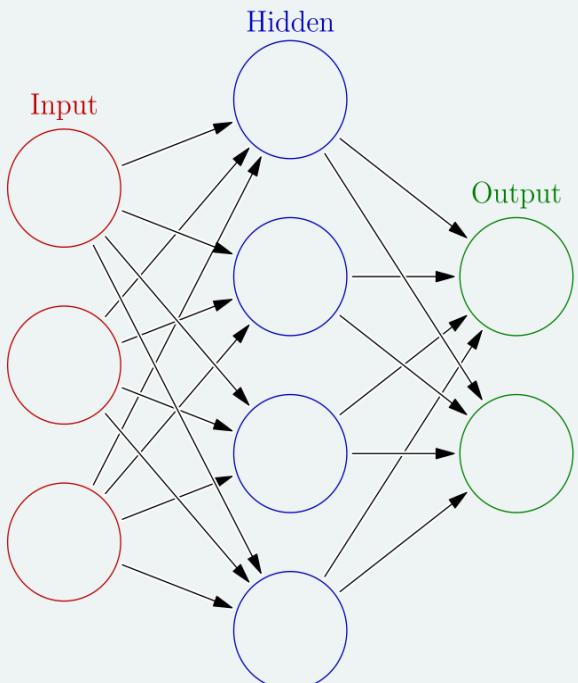
# MODÉLISATION ET CLASSIFICATION

## OVO AND OVR METHODS

Critère	OvO (One vs One)	OvR (One vs Rest)
Score de test	0.969	0.965
Nombre de classifieurs	45	10
Temps d'entraînement (s)	0.762	0.529
Impact	Plus précis pour des petits datasets mais prend plus de temps car il fait beaucoup de classes.	Meilleur en général pour les grands datasets, il prend également moins de temps à se faire car il calcule moins de classe, cependant on perd en précision.

# NEURAL NETWORK

**MODÈLE D'APPRENTISSAGE COMPOSÉ DE COUCHES DE NEURONES ARTIFICIELS QUI TRANSFORMENT PROGRESSIVEMENT LES DONNÉES D'ENTRÉE POUR PRODUIRE UNE PRÉDICTION.**



64 INPUTS : CHAQUE PIXELS DE L'IMAGE

1 COUCHE DE 32 NEURONES

10 OUTPUTS : LA PROBABILITÉ POUR CHAQUE CLASSE

# NEURAL NETWORK

Couche d'entrée et couche cachée : fonction d'activation  
**RELU**

Couche de sortie : fonction d'activation **SOFTMAX**

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

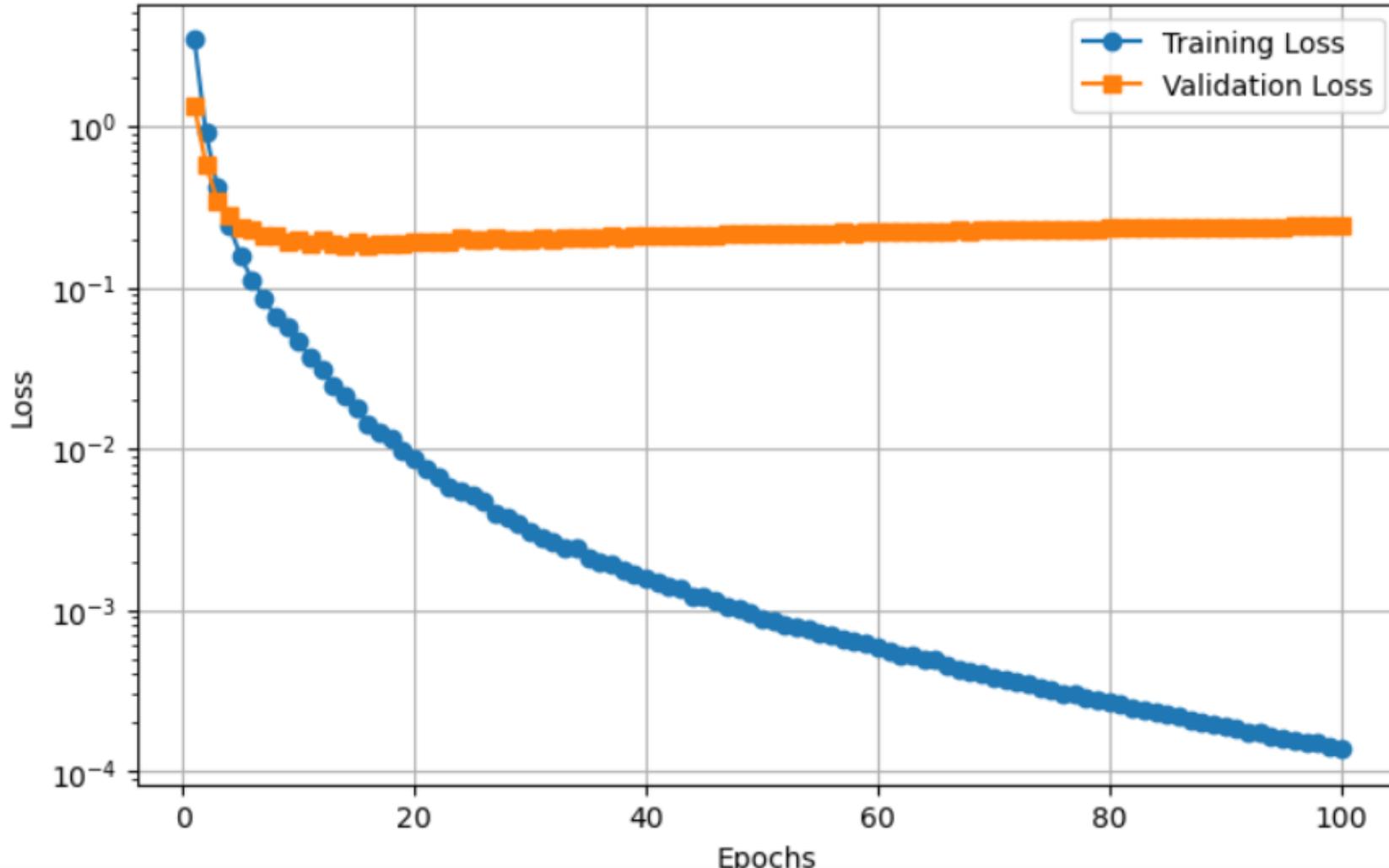
$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^k e^{z_j}}$$

# NEURAL NETWORK

Epoch 100/100

45/45 ————— 0s 3ms/step - accuracy: 1.0000 - loss: 1.2849e-04 - val\_accuracy: 0.9556 - val\_loss: 0.2408

Loss vs Epochs



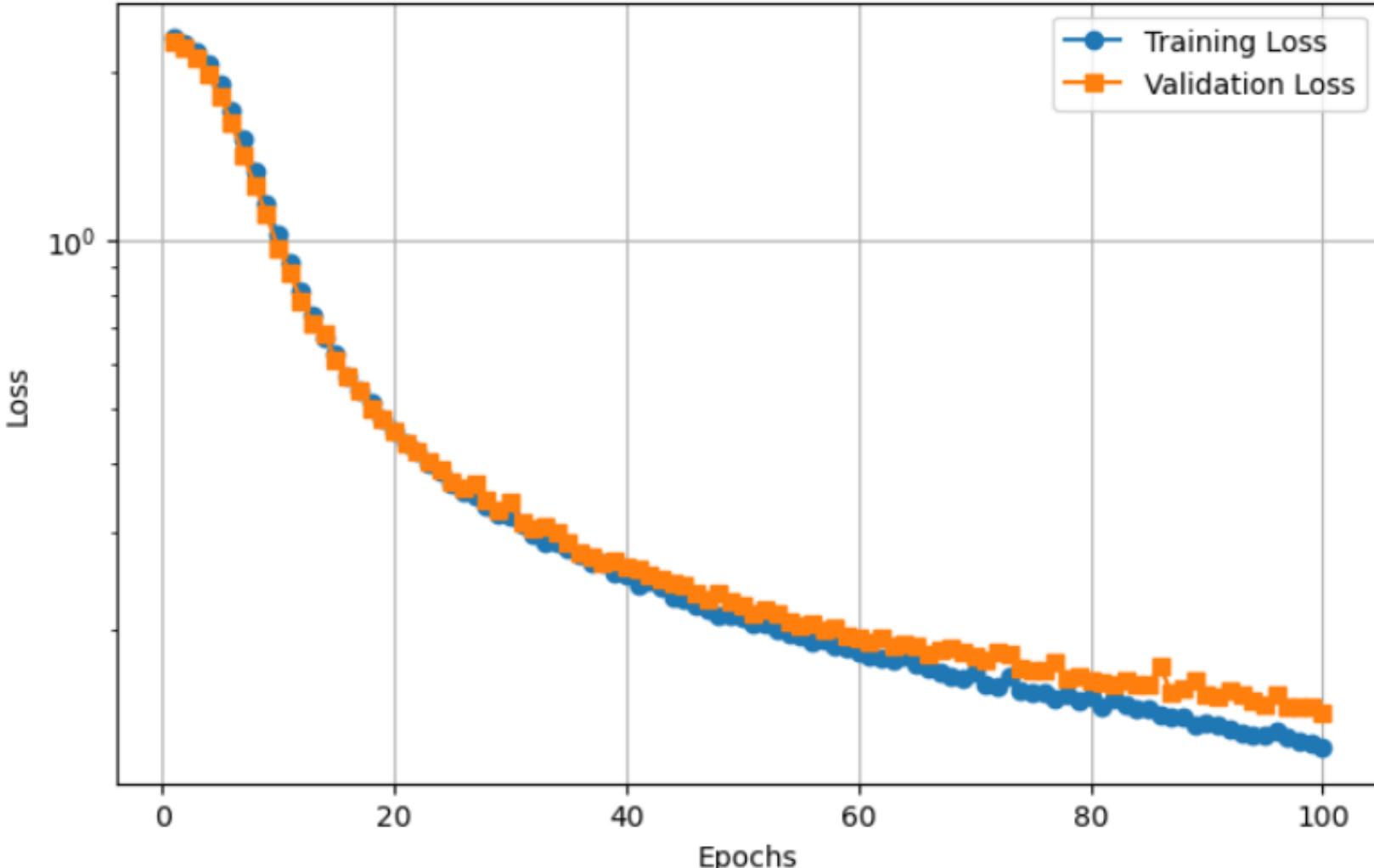
# NEURAL NETWORK

Epoch 100/100

45/45

0s 4ms/step - accuracy: 0.9634 - loss: 0.1290 - val\_accuracy: 0.9583 - val\_loss: 0.1428

Loss vs Epochs



# CONCLUSION



**LE ML EST UN OUTIL PUISSANT  
POUR RÉSOUDRE DES  
PROBLÈMES COMPLEXES À  
PARTIR DE DONNÉES**

**LA QUALITÉ D'UN MODÈLE  
DÉPEND FORTEMENT DE LA  
PRÉPARATION DES DONNÉES ET  
DES CARACTÉRISTIQUES.**