



Soutenance de Projet de fin d'étude

Soutenance de Projet de fin d'étude

Pour l'obtention du diplôme de mastère
professionnel en business Analytics and data
science (BADS)



Détection et classification automatisées des tumeurs cérébrales à l'aide de techniques d'apprentissage profond

Réalisé Par :

Dhaouadi Ibtihel

Devant le jury :

Président : Mme Bedoui Khaoula

Rapporteur : Mr Zouaoui Slim

Encadrant : Mr. Ghazouani Haythem

Année Universitaire : 2023 - 2024

7 Conclusion et perspectives

1 Introduction

6 Déploiement

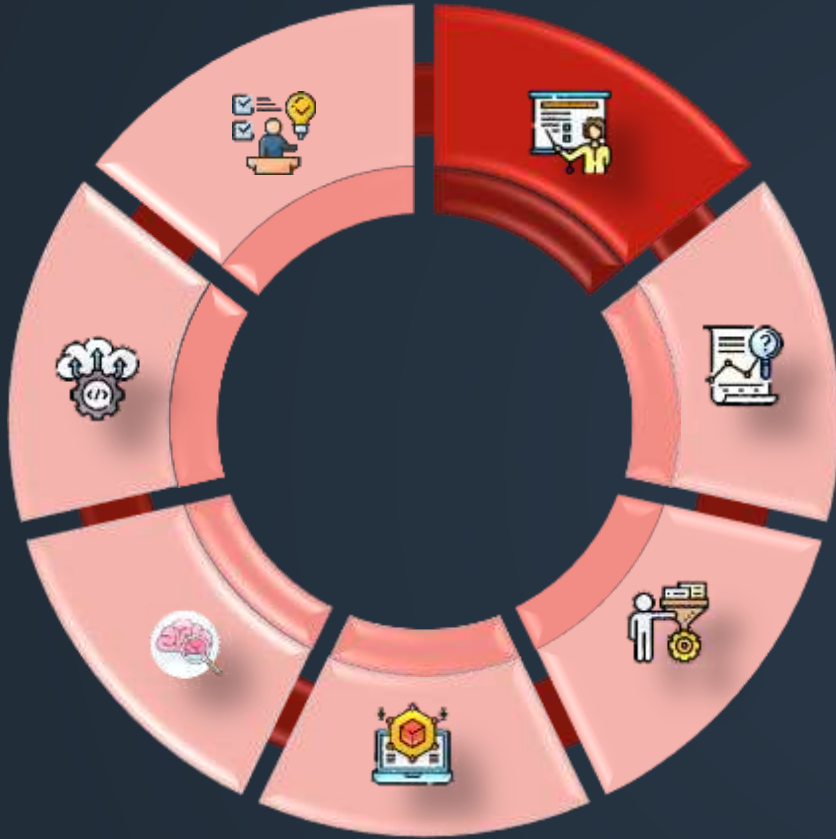
2 Compréhension des besoins

5 Segmentation : modélisation et évaluation

3 Compréhension et préparation des données

4 Classification : modélisation et évaluation





1

Introduction

- ➔ Contexte
- ➔ Objectif
- ➔ Méthodologie



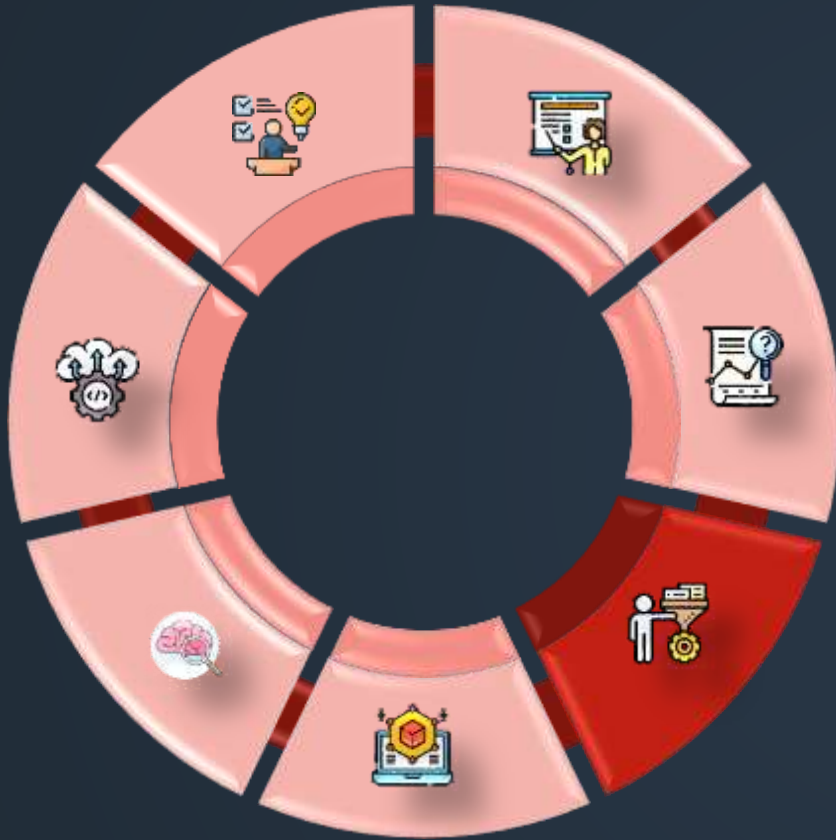


2

Compréhension des besoins

- ➔ les défis du diagnostic manuel
- ➔ Les avantages d'un système automatisé





3 Compréhension et préparation des données

- ➔ Description du jeu de données
- ➔ Les classes
- ➔ les étapes de préparation





4

Classification : modélisation et évaluation

- ➔ Présentation des modèles testés
- ➔ Evaluation des performances
- ➔ Comparaison et meilleur modèle





5

Segmentation : modélisation et évaluation

- ➔ Introduction au modèle U-Net
- ➔ Evaluation des performances
- ➔ Visualisation des résultats

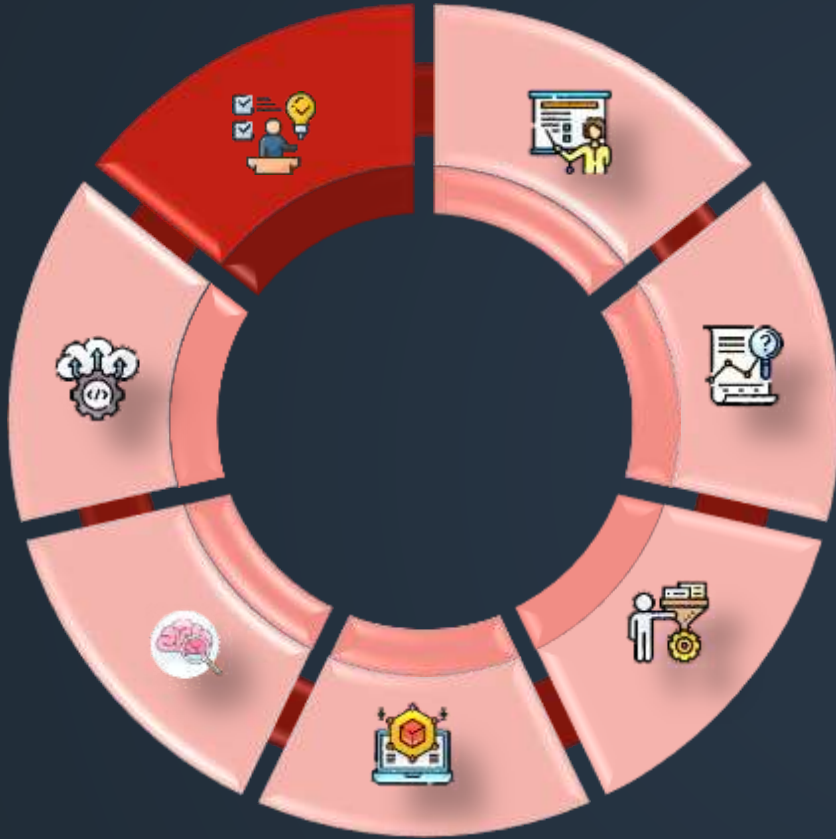




6

Déploiement





7

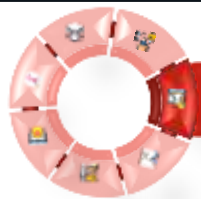
Conclusion et perspectives





1 Introduction





Introduction

contexte

objectif

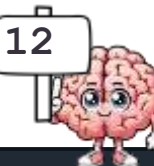
méthodologie

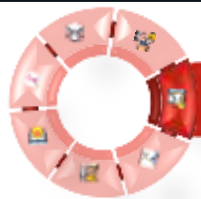
Les tumeurs cérébrales représentent un défi médical majeur en raison de leur diversité et de la complexité de leur diagnostic.

Face à l'augmentation des données médicales et au manque de ressources spécialisées,



l'intelligence artificielle offre une opportunité unique d'automatiser l'analyse des images IRM, permettant ainsi de fournir une aide précieuse, rapide et fiable aux professionnels de santé.





Introduction

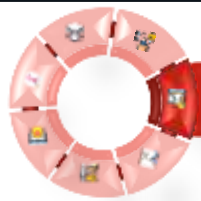
contexte

👉 objectif

méthodologie

L'objectif principal de ce projet est de développer un système robuste et fiable capable de **détecter**, **classifier** et **segmenter** les tumeurs cérébrales à partir d'images IRM en utilisant des réseaux neuronaux convolutifs (CNN) et des techniques de transfert d'apprentissage.





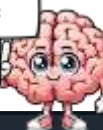
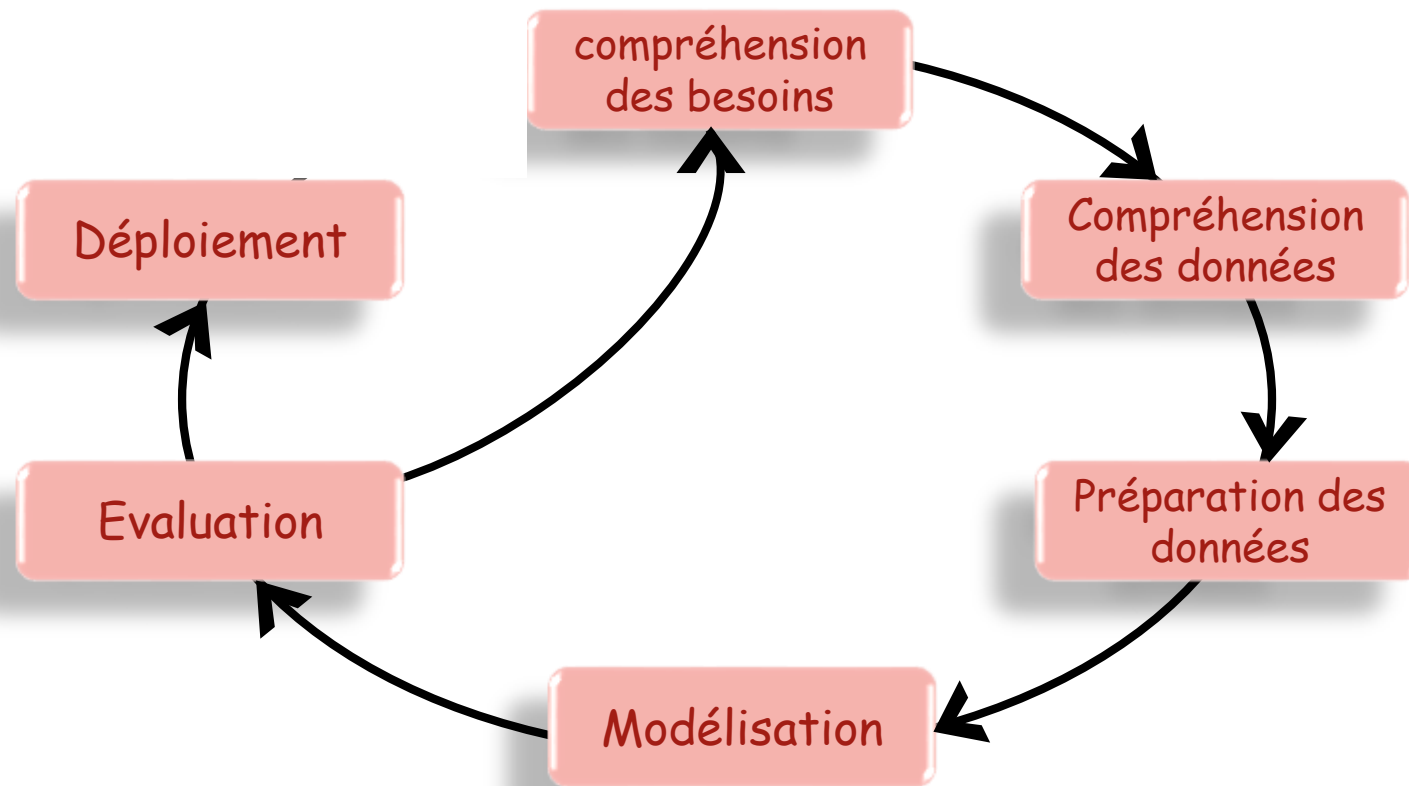
Introduction

contexte

objectif

 méthodologie

La méthodologie CRISP-DM (Cross-Industry Standard Process for Data Mining)





2 Compréhension des besoins





2 Compréhension des besoins

👉 les défis du diagnostic manuel

Les avantages d'un système automatisé

Les tumeurs cérébrales varient largement en forme, taille et caractéristiques,

le manque d'experts qualifiés ralentit le diagnostic.

1
Complexité
et
diversité

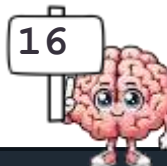
3
Ressources
limitées

2
Volume de
données

4
Variabilité
inter-
experts

le grand nombre d'images IRM à analyser est un défi pour les radiologues

Les diagnostics peuvent différer d'un radiologue à l'autre, en fonction de leur expertise ou de leur fatigue.



2 Compréhension des besoins

les défis du diagnostic manuel

Les avantages d'un système automatisé

Un système permettrait un diagnostic plus rapide et plus précis en réduisant les erreurs humaines.

Réduction
des erreurs

Gain de
temps

Un diagnostic plus rapide et précis permettrait de débiter le traitement des patients plus tôt

Accessibilité

Un système automatisé pourrait être déployé dans des régions éloignées ou en développement





3 Compréhension et préparation des données





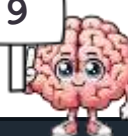
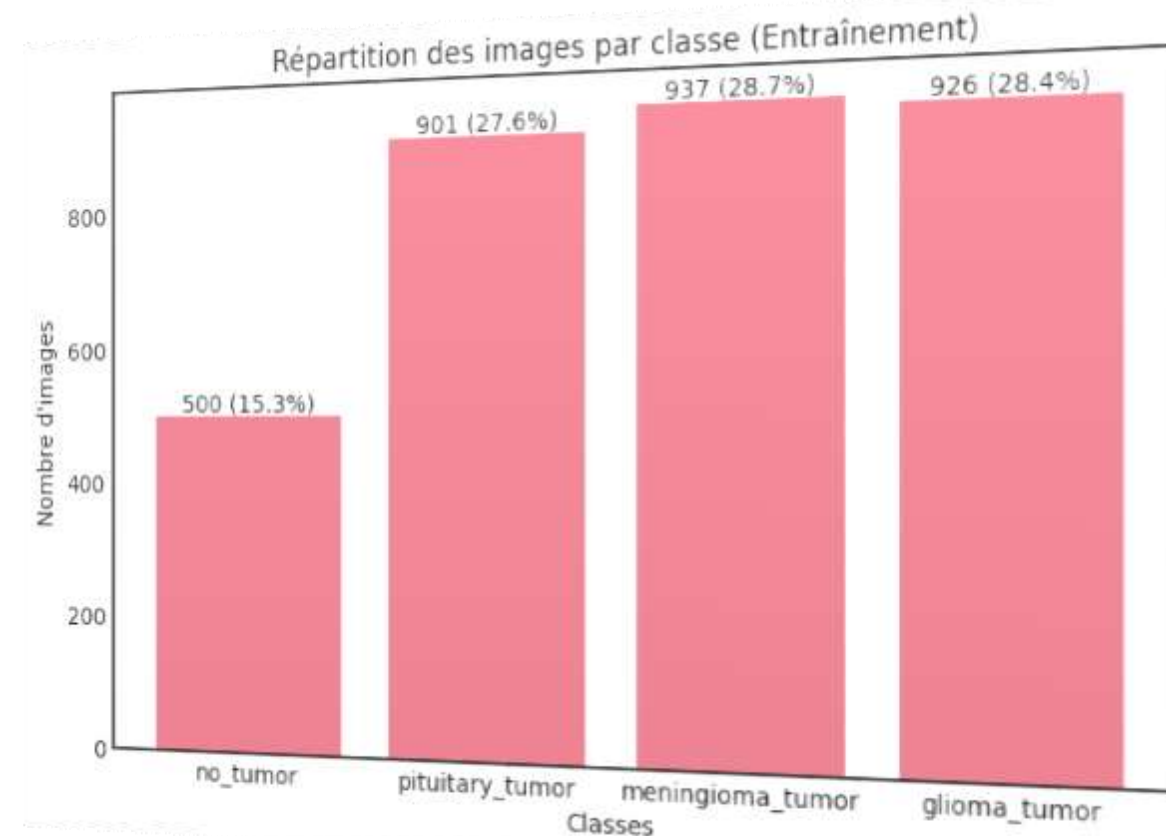
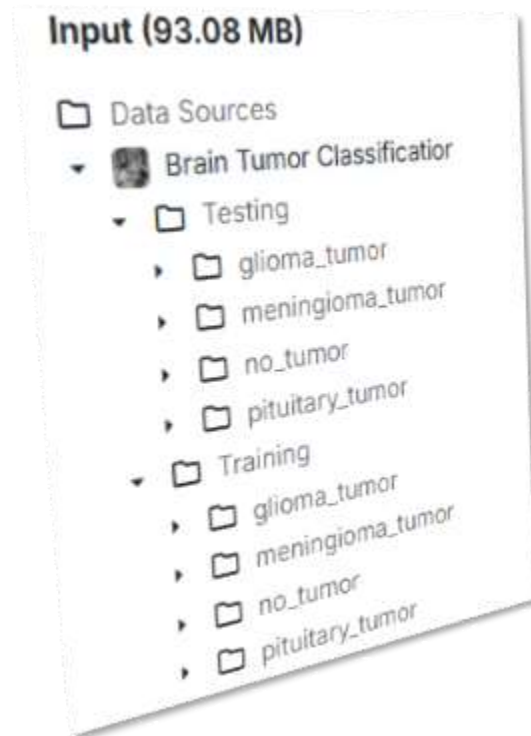
3 Compréhension et préparation des données

👉 Présentation du jeu de données

les classes

les étapes de préparation

kaggle





3 Compréhension et préparation des données

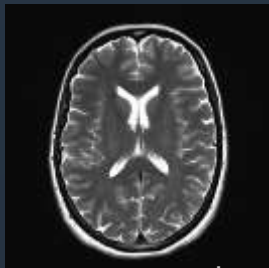
Présentation du jeu de données

les classes

les étapes de préparation

Classe : No-Tumor

les images IRM de patients ne présentant aucune tumeur visible



Axial



Sagittal



Coronal

plan
anatomique :



cerveau

Tronc
cérébral

Moelle
épinière

cervelle



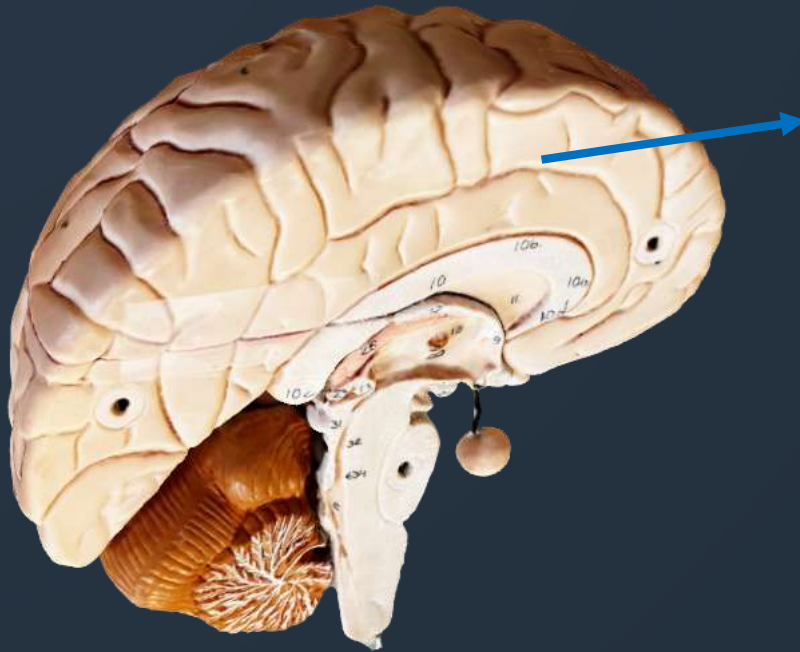


3 Compréhension et préparation des données

Présentation du jeu de données

les classes

les étapes de préparation



cerveau

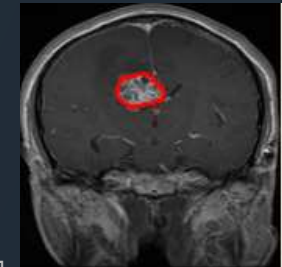
Classe : Glioma-Tumor



Axial



Sagittal



Coronal

Ce sont les tumeurs cérébrales les plus courantes, provenant des cellules gliales du cerveau





3 Compréhension et préparation des données

Présentation du jeu de données

les classes

les étapes de préparation

Classe : meningioma-Tumor

Ce type de tumeur se forme dans les méninges, qui sont les membranes protectrices recouvrant le cerveau et la moelle épinière



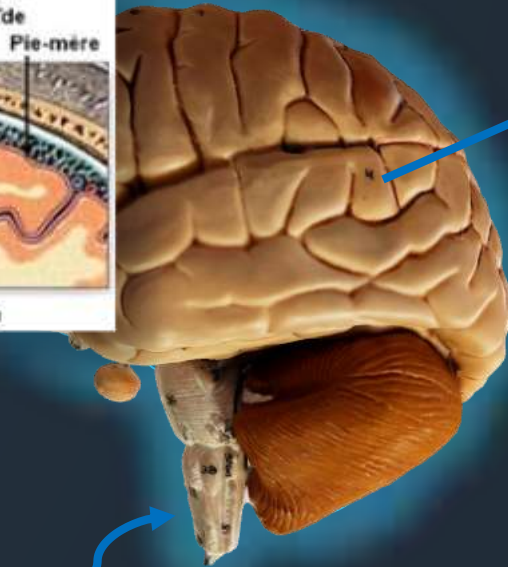
Axial



Sagittal



Coronal



cerveau

Moelle
épinière



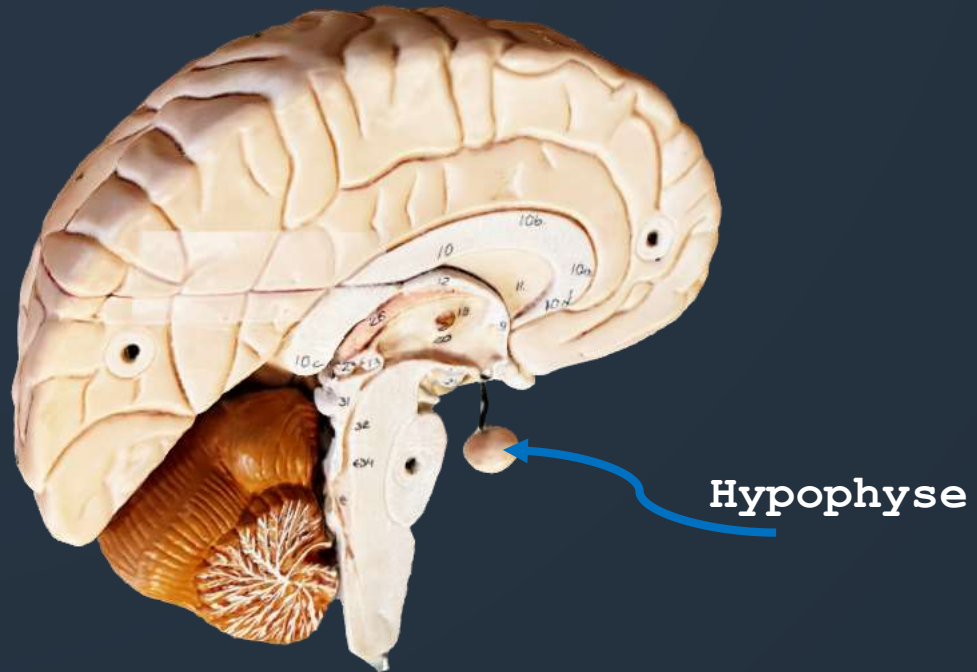


3 Compréhension et préparation des données

Présentation du jeu de données

les classes

les étapes de préparation



Classe : pituitary-Tumor

des images d'IRM des tumeurs de l'hypophyse



Axial



Sagittal



Coronal



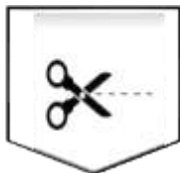


3 Compréhension et préparation des données

Présentation du jeu de données

les classes

➡ les étapes de préparation



Recadrage des images

Image Originale

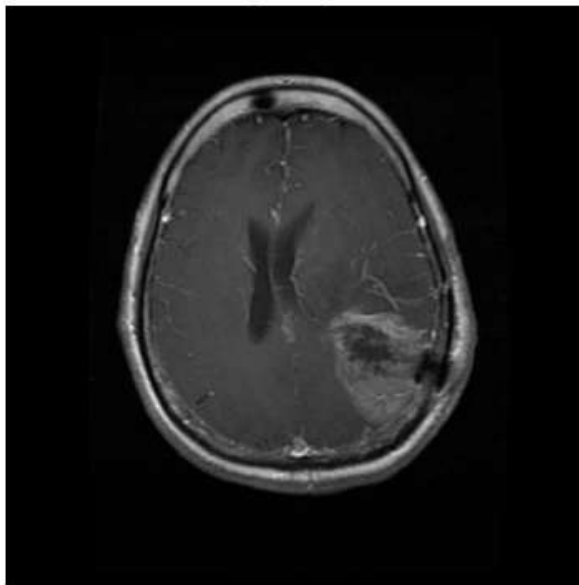
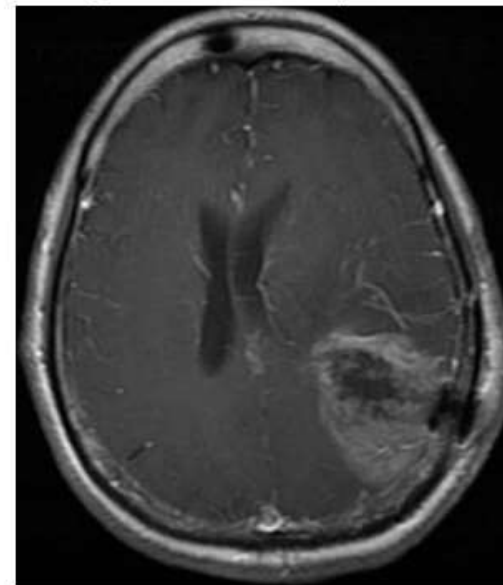


Image Recadrée (sans espaces vides)





3 Compréhension et préparation des données

Présentation du jeu de données

les classes

les étapes de préparation



Redimensionnement

Most frequent Train images shapes:

- (512, 512, 3): 2341
- (236, 236, 3): 79
- (225, 225, 3): 43

225x225 pixels





3 Compréhension et préparation des données

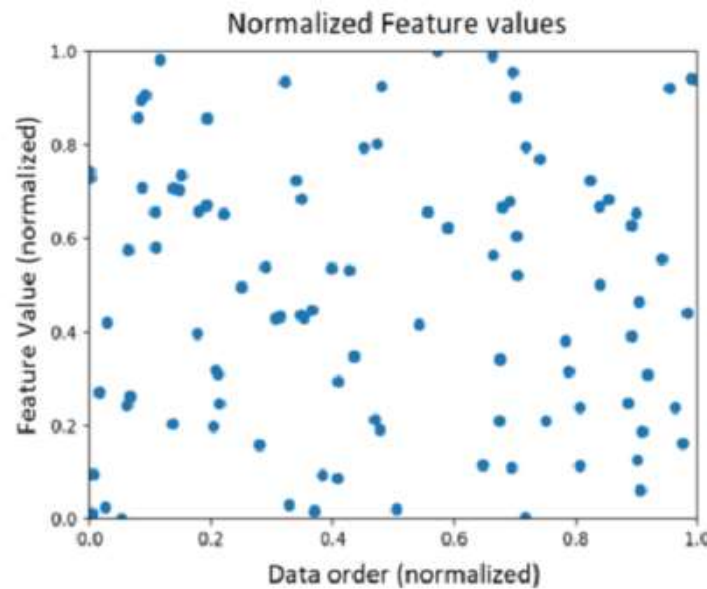
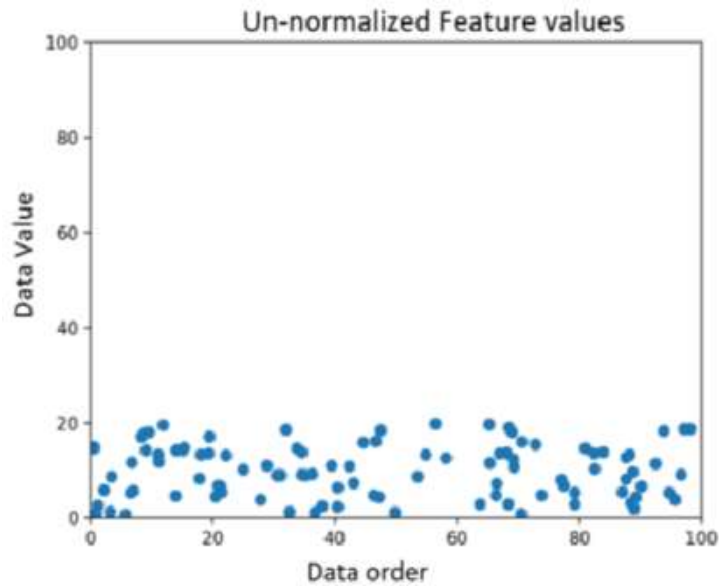
Présentation du jeu de données

les classes

➡ les étapes de préparation



Normalisation



La normalisation des images est importante pour s'assurer que les valeurs des pixels sont dans une gamme qui permet une meilleure convergence pendant l'entraînement du modèle.





3 Compréhension et préparation des données

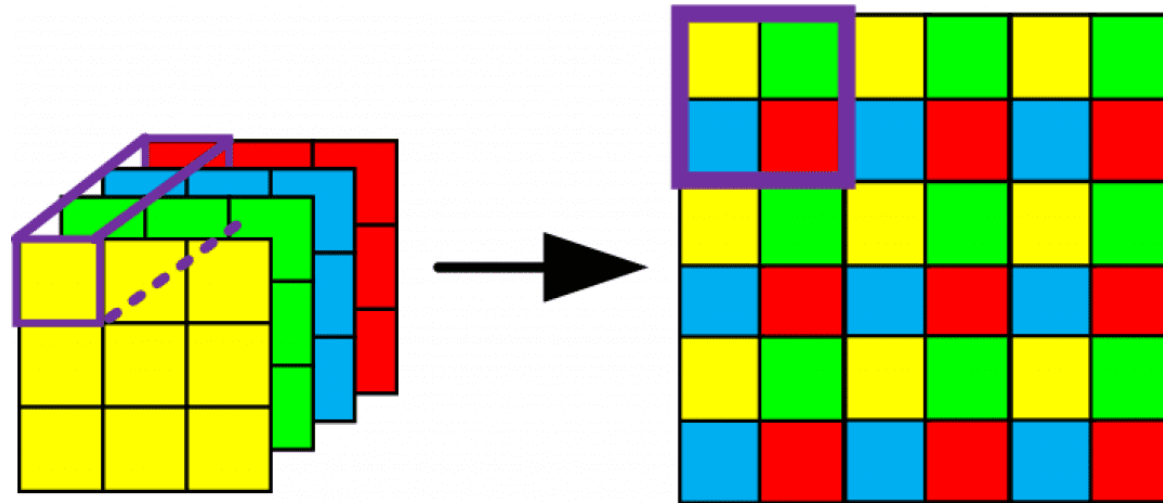
Présentation du jeu de données

les classes

les étapes de préparation



Mélanger les données





3 Compréhension et préparation des données

Présentation du jeu de données

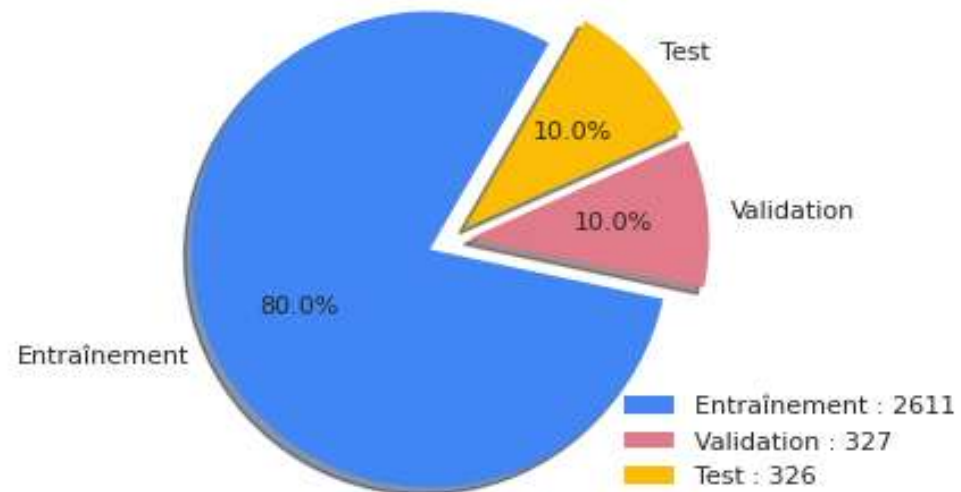
les classes

les étapes de préparation



répartition

Répartition des images d'entraînement et de test.





3 Compréhension et préparation des données

Présentation du jeu de données

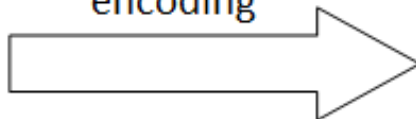
les classes  les étapes de préparation



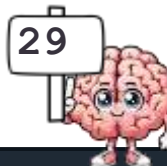
Encodage One-Hot

no_tumor
pituitary_tumor
meningioma_tumor
glioma_tumor

One-Hot
encoding



classe 1	classe 2	classe 3	classe 4
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1





3 Compréhension et préparation des données

Présentation du jeu de données

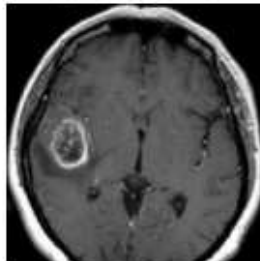
les classes ➡ les étapes de préparation



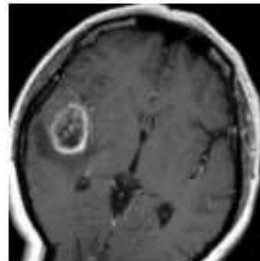
Augmentation des données

```
# Définir les paramètres d'augmentation des données
train_datagen = ImageDataGenerator(
    rotation_range=30,
    width_shift_range=0.,
    height_shift_range=0.,
    shear_range=0.2,
    zoom_range=0.05,
    horizontal_flip=True,
    vertical_flip=False,
    fill_mode='nearest'
)
```

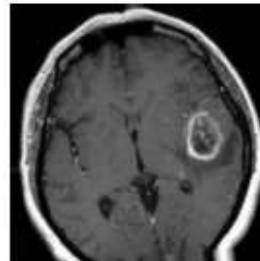
Originale
(glioma_tumor)



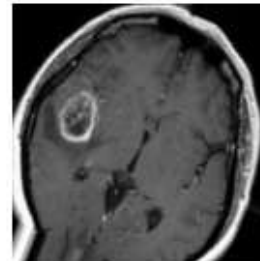
Augmentée 1



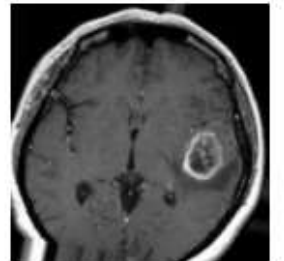
Augmentée 2



Augmentée 3



Augmentée 4





4

**Classification :
modélisation et évaluation**



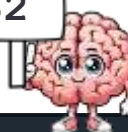
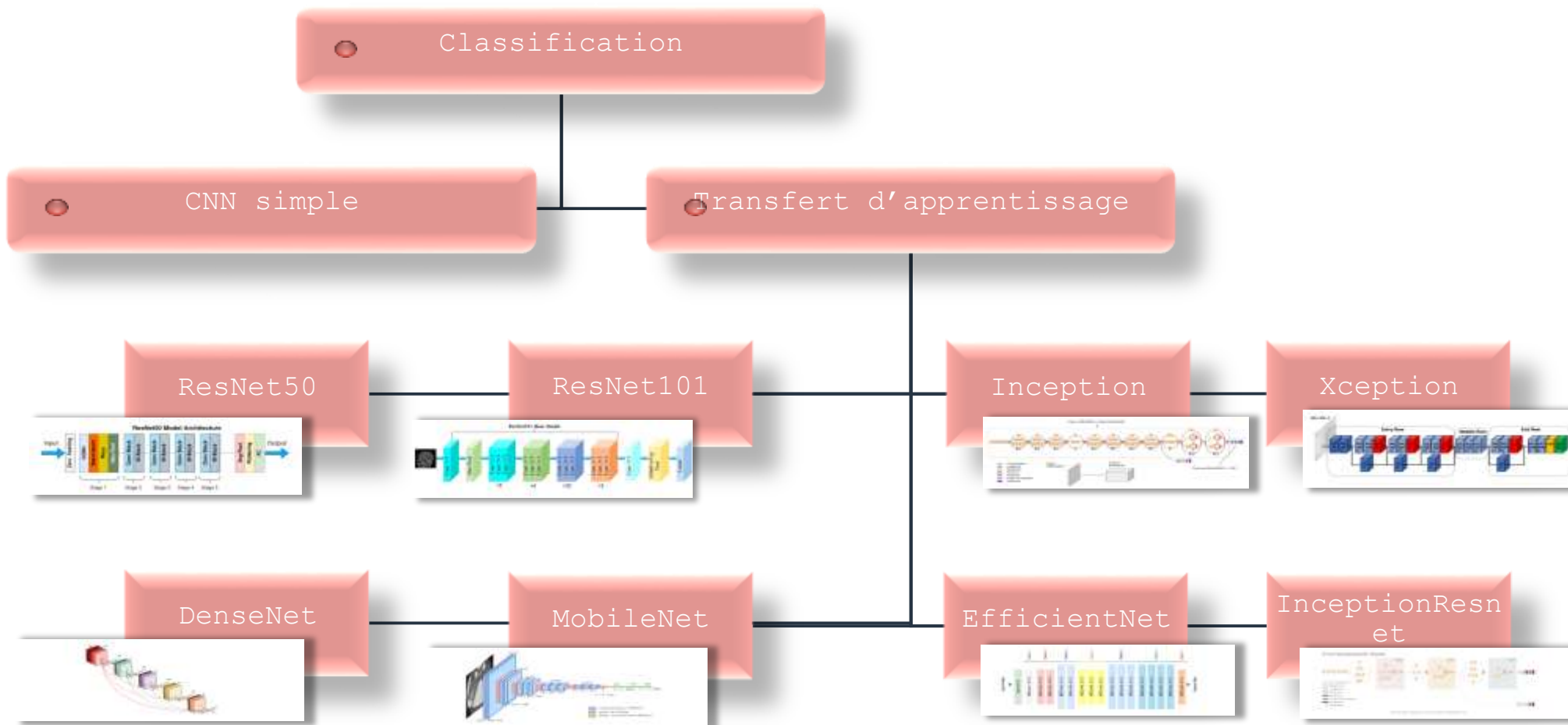


4 Classification : modélisation et évaluation

👉 Présentation des modèles testés

Evaluation des performances

Comparaison et meilleur modèle





4 Classification : modélisation et évaluation

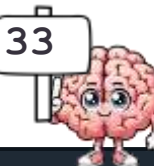
➡ Présentation des modèles testés

Evaluation des performances

Comparaison et meilleur modèle



Utilisation de Kaggle Notebook pour l'entraînement des modèles,
avec un accès aux **GPU** pour accélérer le calcul et
l'entraînement des réseaux de neurones et qui facilite
également l'accès aux **datasets** .





4 Classification : modélisation et évaluation

👉 Présentation des modèles testés

Evaluation des performances

Comparaison et meilleur modèle

```
#le modèle CNN
CNN_model = Sequential()
CNN_model.add(Conv2D(16, (3,3), activation='relu', input_shape=shape))
CNN_model.add(MaxPooling2D(2,2))
CNN_model.add(Dropout(0.2))

CNN_model.add(Conv2D(32, (3,3), activation='relu'))
CNN_model.add(MaxPooling2D(2,2))
CNN_model.add(Dropout(0.2))

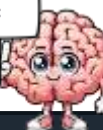
CNN_model.add(Conv2D(64, (3,3), activation='relu'))
CNN_model.add(MaxPooling2D(2,2))
CNN_model.add(Dropout(0.2))

CNN_model.add(Conv2D(128, (3,3), activation='relu'))
CNN_model.add(MaxPooling2D(2,2))
CNN_model.add(Dropout(0.2))

CNN_model.add(Conv2D(512, (3,3), activation='relu'))
CNN_model.add(MaxPooling2D(2,2))
CNN_model.add(Dropout(0.2))

CNN_model.add(Flatten())
CNN_model.add(Dense(512, activation='relu'))
CNN_model.add(Dropout(0.5))

CNN_model.add(Dense(len(classes), activation='softmax'))
```



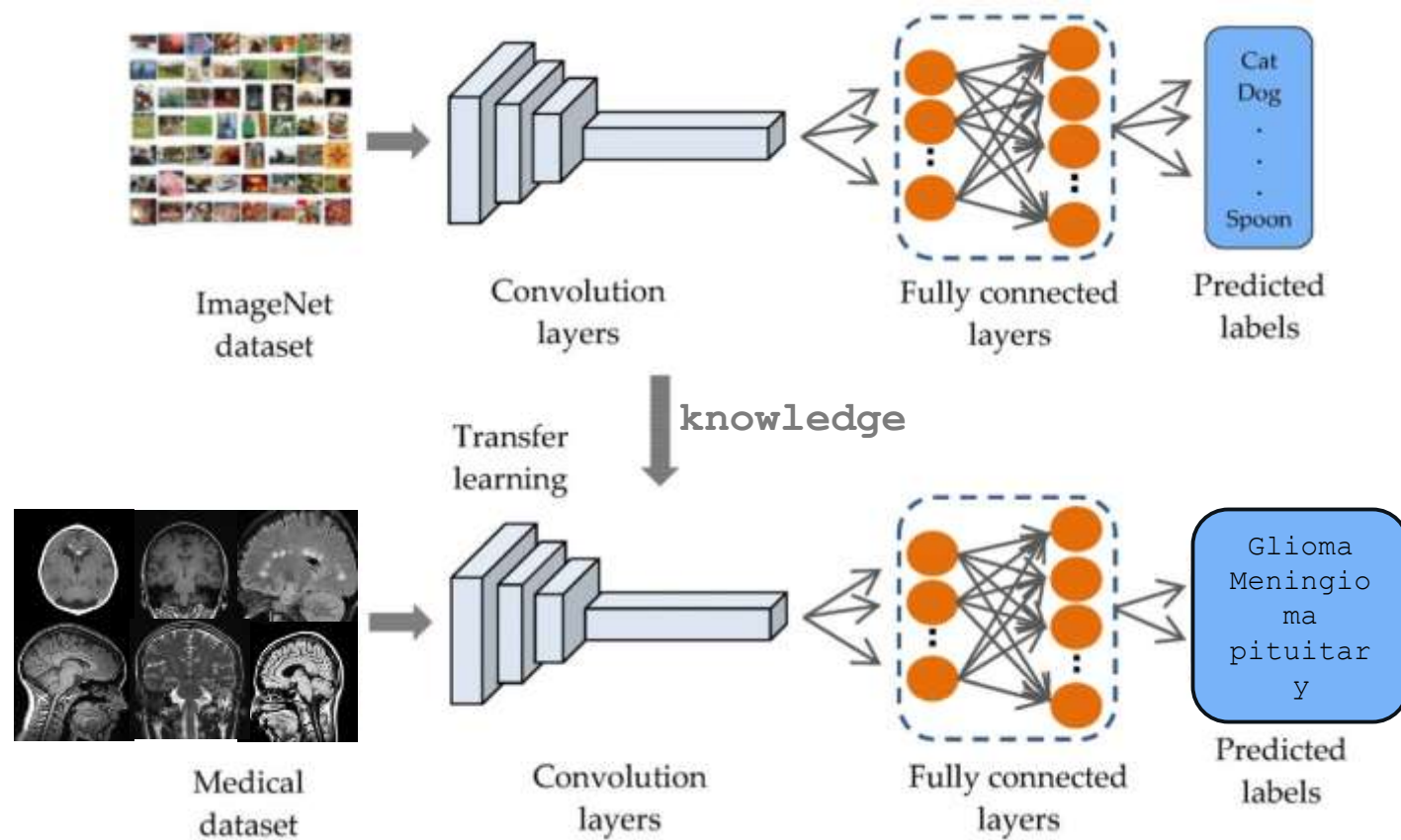


4 Classification : modélisation et évaluation

➡ Présentation des modèles testés

Evaluation des performances

Comparaison et meilleur modèle





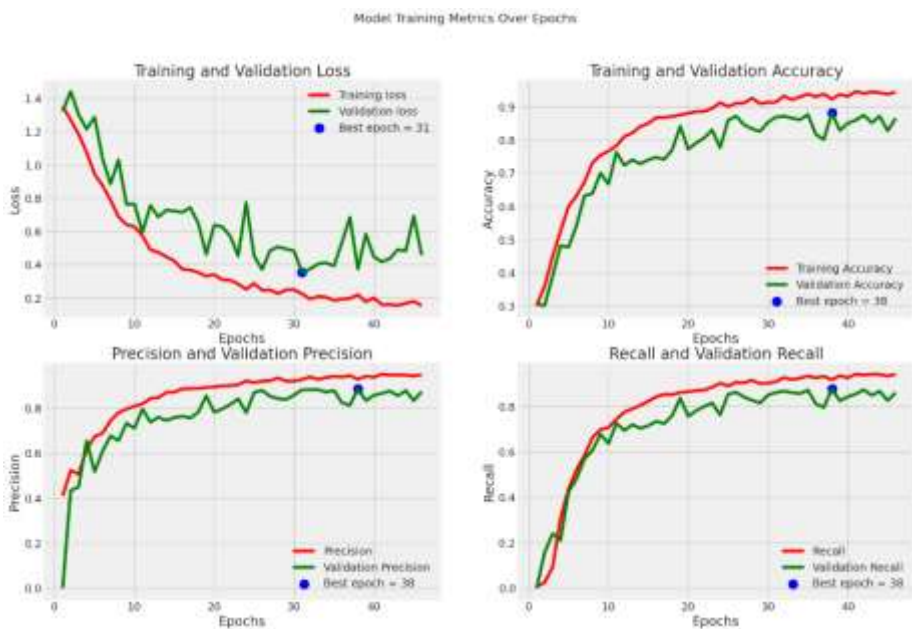
4 Classification : modélisation et évaluation

Présentation des modèles testés

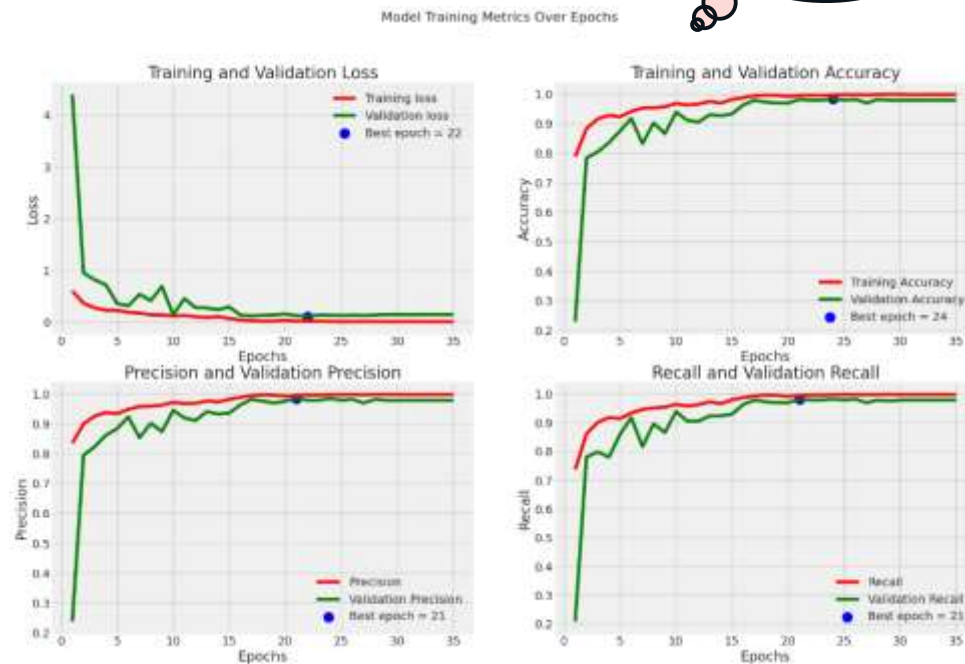
👉 Evaluation des performances

Comparaison et meilleur modèle

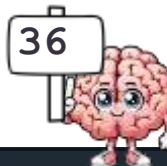
une convergence
plus rapide et
des courbes
plus stables



CNN model



InceptionResnet model



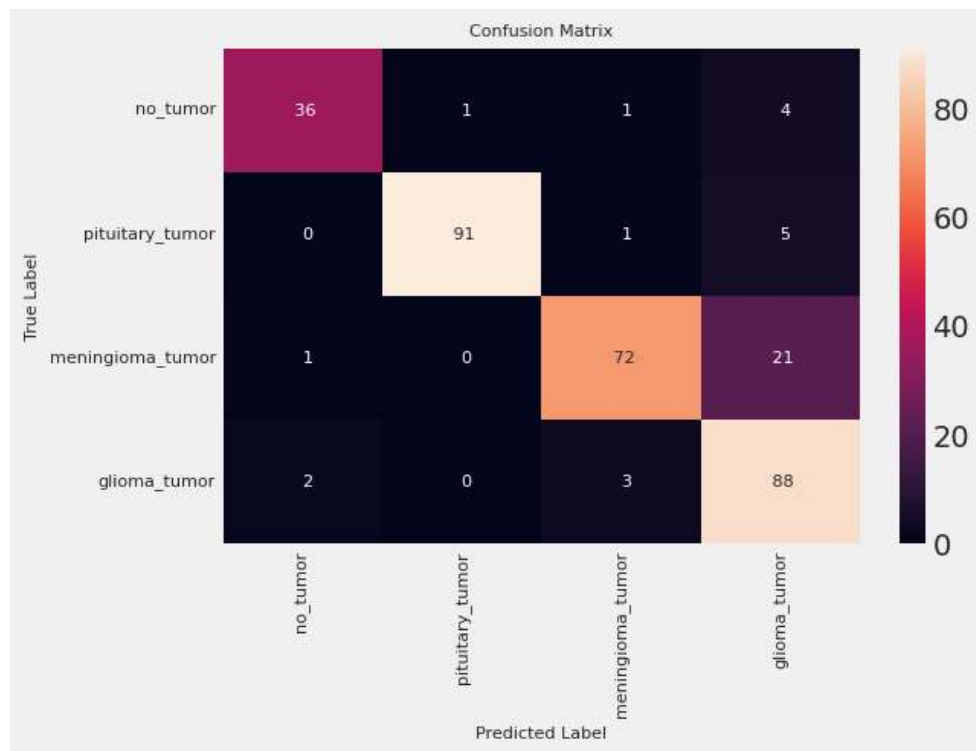


4 Classification : modélisation et évaluation

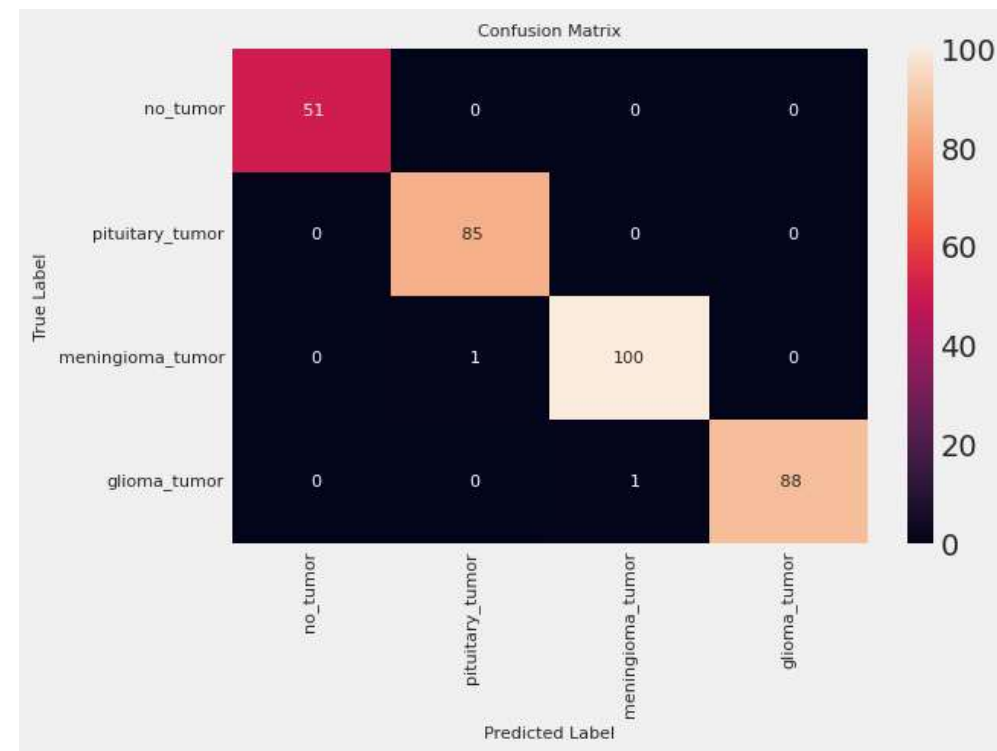
Présentation des modèles testés

👉 Evaluation des performances

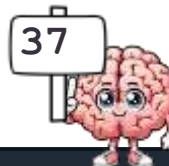
Comparaison et meilleur modèle



CNN model



InceptionResnet model





4

Classification : modélisation et évaluation

Présentation des modèles testés

👉 **Evaluation des performances**

Comparaison et meilleur modèle

Classification Report:

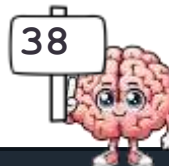
	precision	recall	f1-score	support
no_tumor	0.92	0.86	0.89	42
pituitary_tumor	0.99	0.94	0.96	97
meningioma_tumor	0.94	0.77	0.84	94
glioma_tumor	0.75	0.95	0.83	93
accuracy			0.88	326
macro avg	0.90	0.88	0.88	326
weighted avg	0.90	0.88	0.88	326

CNN model

Classification Report:

	precision	recall	f1-score	support
no_tumor	1.00	1.00	1.00	51
pituitary_tumor	0.99	1.00	0.99	85
meningioma_tumor	0.99	0.99	0.99	101
glioma_tumor	1.00	0.99	0.99	89
accuracy			0.99	326
macro avg	0.99	0.99	0.99	326
weighted avg	0.99	0.99	0.99	326

InceptionResnet model



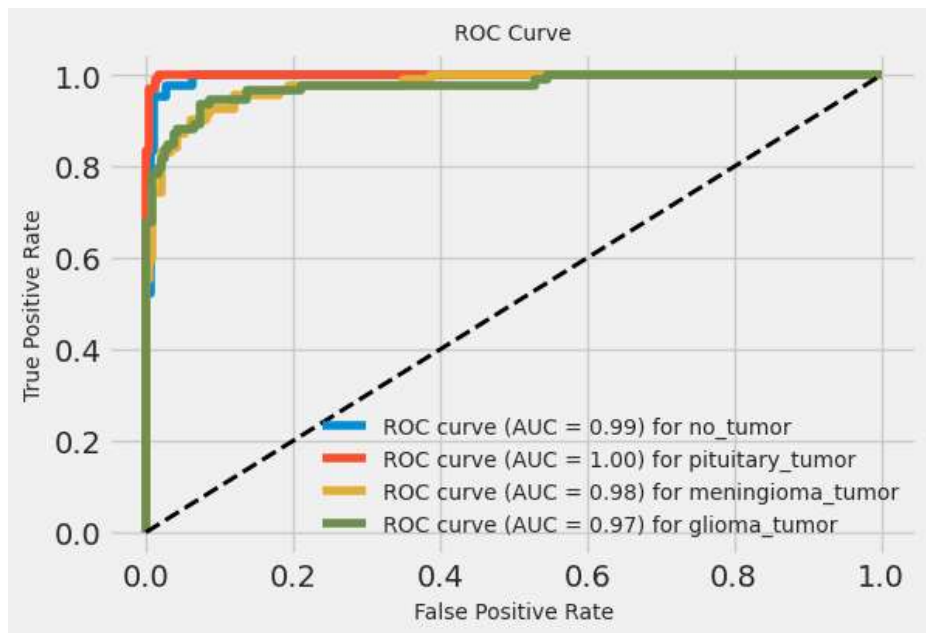


4 Classification : modélisation et évaluation

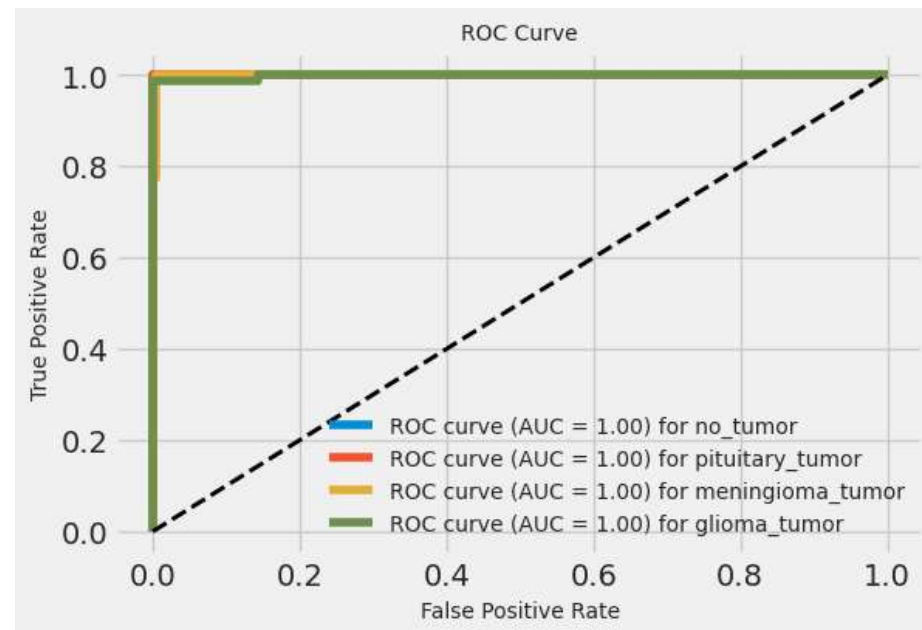
Présentation des modèles testés

👉 Evaluation des performances

Comparaison et meilleur modèle



CNN model



InceptionResnet model





4 Classification : modélisation et évaluation

Présentation des modèles testés

Evaluation des performances

➡ Comparaison et meilleur modèle

Nom du modèle	Entrainement				Test			
	Perte	Precision	F1-score	Rappel	Perte	Précision	F1-score	Rappel
Simple CNN	17.54%	93.70%	94.15%	93.08%	29.77%	88.04%	88.54%	87.73%
DenseNet121	0.99%	99.58%	99.58%	99.58%	5.96%	98.77%	98.77%	98.77%
InceptionResNetV2	0.42%	99.81%	99.81%	99.81%	4.08%	99.39%	99.39%	99.39%
MobileNetV1	0.49%	99.85%	99.85%	99.77%	9.26%	98.77%	98.77%	98.77%
EfficientNetB0	0.96%	99.62%	99.62%	99.62%	5.94%	98.47%	98.47%	98.47%
Xception	1.40%	99.39%	99.46%	99.39%	7.75%	98.16%	98.46%	98.16%
InceptionV3	2.21%	99.08%	99.08%	99.08%	7.72%	97.85%	97.85%	97.85%
ResNet50	3.39%	98.81%	98.81%	98.81%	5.76%	98.77%	98.77%	98.47%
ResNet101	7.09%	97.47%	97.62%	97.28%	17.72%	93.56%	93.54%	93.25%





5

**Segmentation :
modélisation et évaluation**





5

Segmentation : modélisation et évaluation

👉 Dataset et préparation

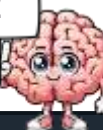
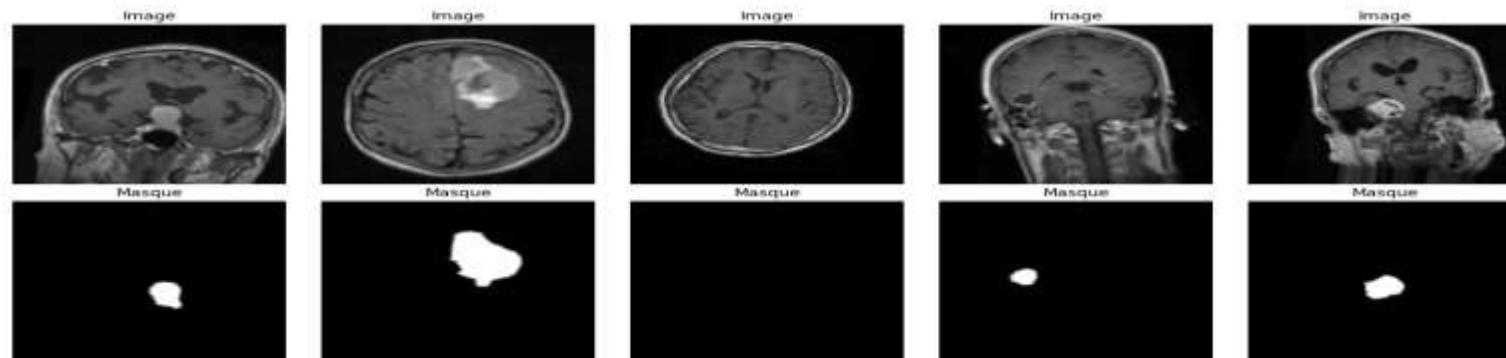
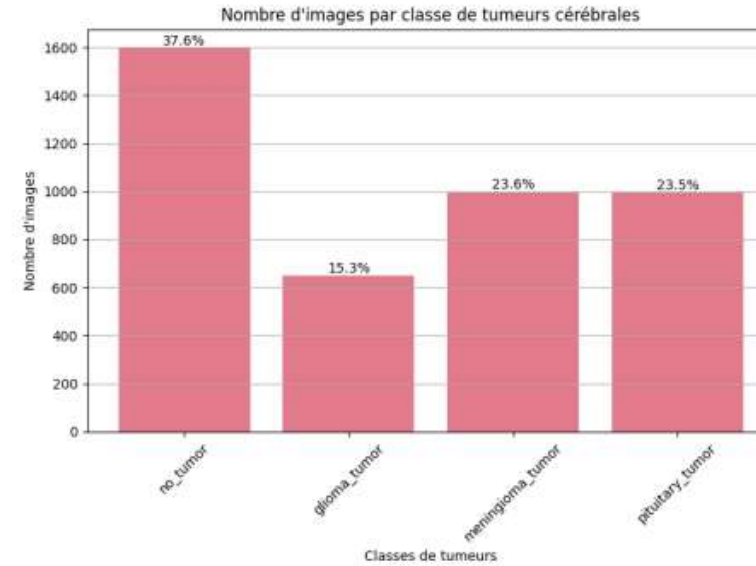
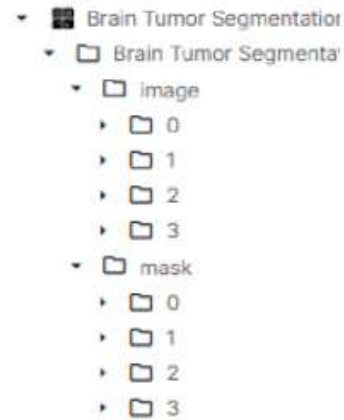
modèle U-Net

Evaluation des performances

Visualisation des résultats

Total : 4237
images

- Image
 - 0 (No Tumor)
 - 1 (Glioma)
 - 2 (Meningioma)
 - 3 (Pituitary)
- Mask
 - 0 (No Tumor)
 - 1 (Glioma)
 - 2 (Meningioma)
 - 3 (Pituitary)





5

Segmentation : modélisation et évaluation

👉 Dataset et préparation

modèle U-Net

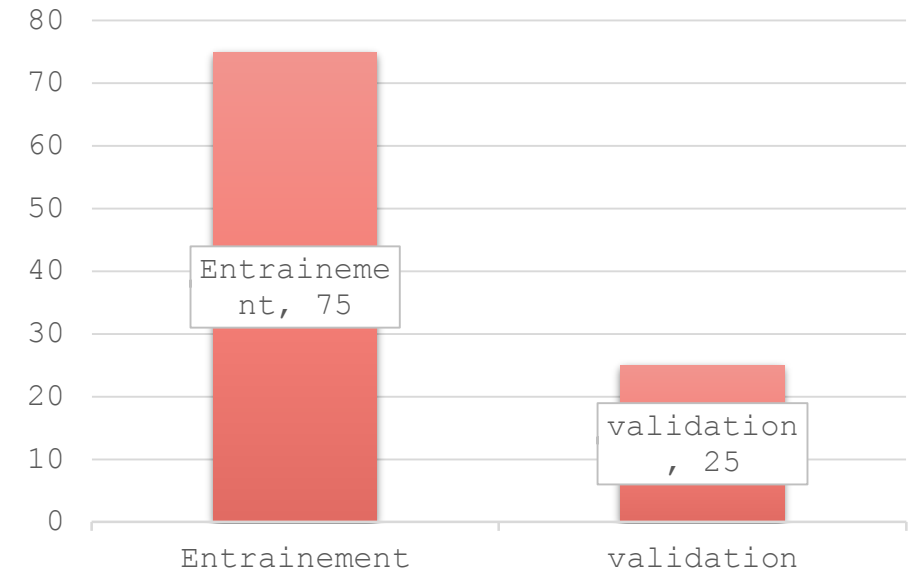
Evaluation des performances

Visualisation des résultats



répartition des données

```
# Diviser les données en ensembles d'entraînement (75%) et de validation (25%)
images_train, images_val, masks_train, masks_val, labels_train, labels_val = train_test_split(
    images, masks, labels, test_size=0.3, random_state=42, stratify=labels, shuffle=True
)
```





5

Segmentation : modélisation et évaluation



Dataset et préparation

modèle U-Net

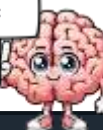
Evaluation des performances

Visualisation des résultats



Augmentation des données

```
def train_generator(images, masks, batch_size, seed=42):  
    image_datagen = ImageDataGenerator(  
        rotation_range=20,  
        width_shift_range=0.1,  
        height_shift_range=0.1,  
        shear_range=0.1,  
        zoom_range=0.1,  
        horizontal_flip=True,  
        fill_mode='nearest'  
    )  
  
    mask_datagen = ImageDataGenerator(  
        rotation_range=20,  
        width_shift_range=0.1,  
        height_shift_range=0.1,  
        shear_range=0.1,  
        zoom_range=0.1,  
        horizontal_flip=True,  
        fill_mode='nearest'  
    )
```





5

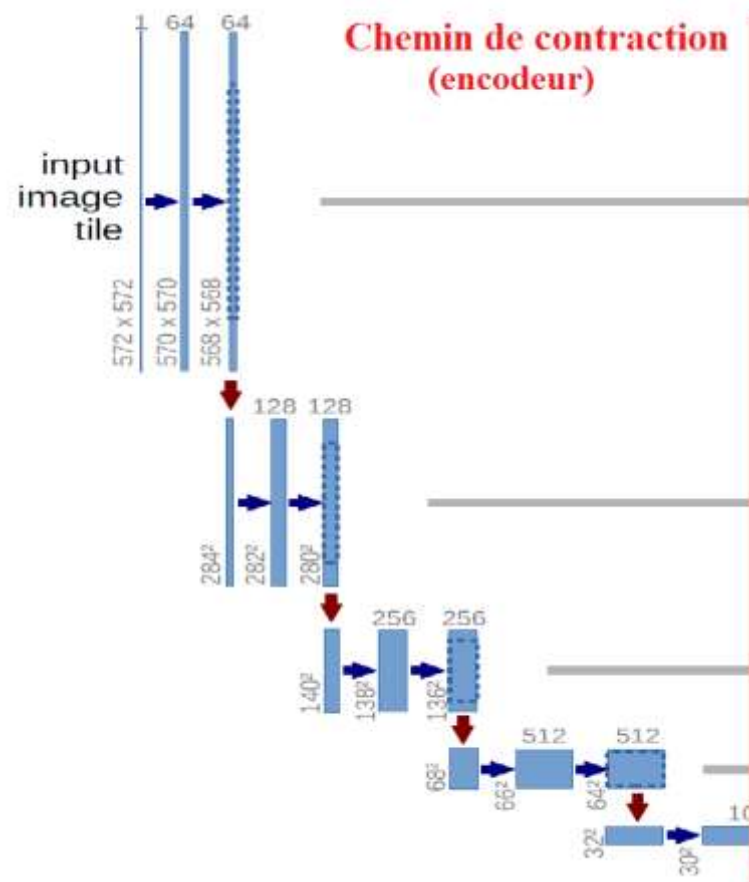
Segmentation : modélisation et évaluation

Dataset et préparation

👉 **modèle U-Net**

Evaluation des performances

Visualisation des résultats





5

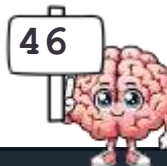
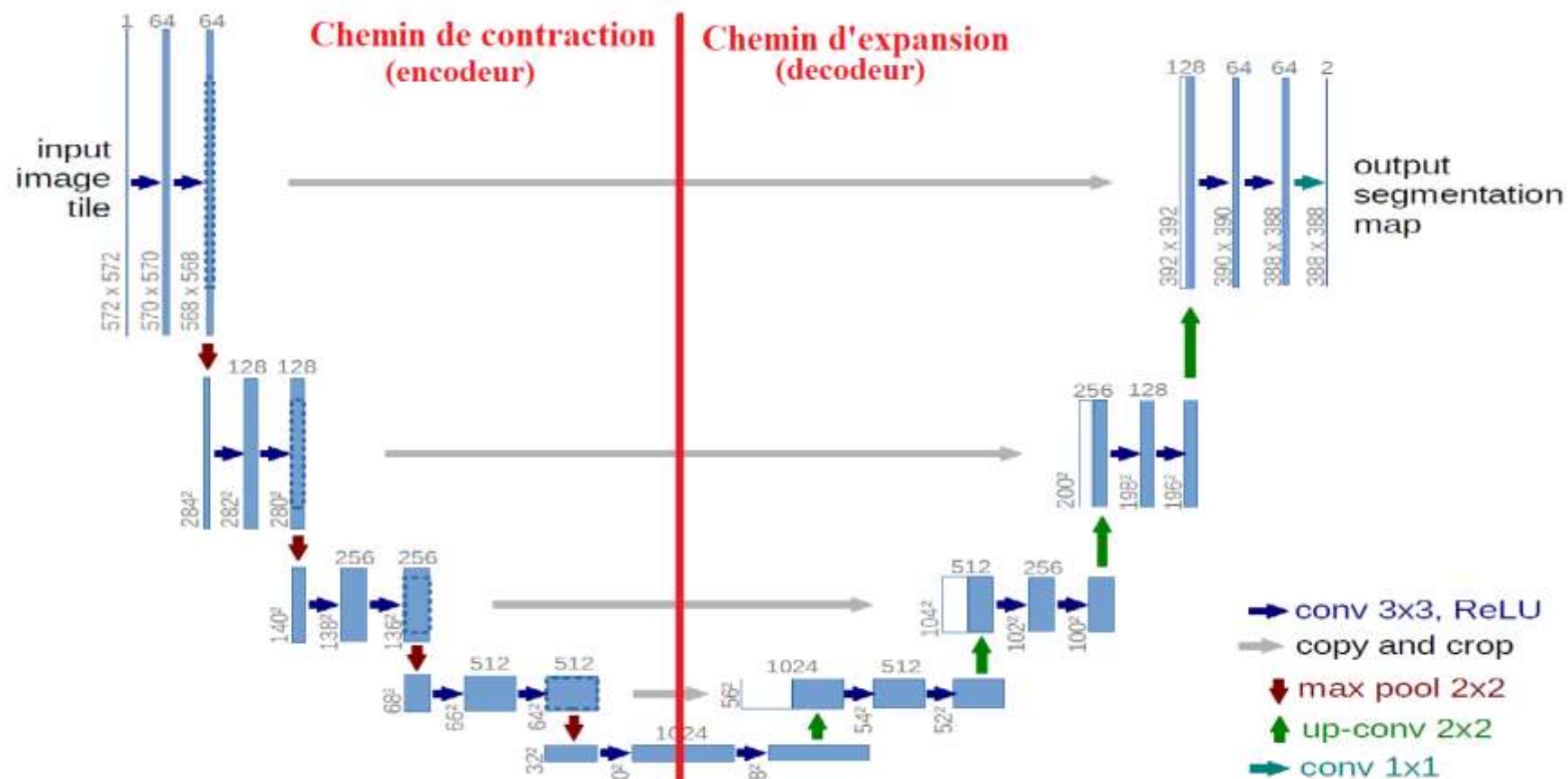
Segmentation : modélisation et évaluation

Dataset et préparation

👉 modèle U-Net

Evaluation des performances

Visualisation des résultats





5

Segmentation : modélisation et évaluation

Dataset et préparation

modèle U-Net

👉 **Evaluation des performances**

Visualisation des résultats

Les paramètres utilisés pour
l'entraînement :

Nombre d'époque : 100

taille de lot : 32

Générateur d'augmentation

callbacks :

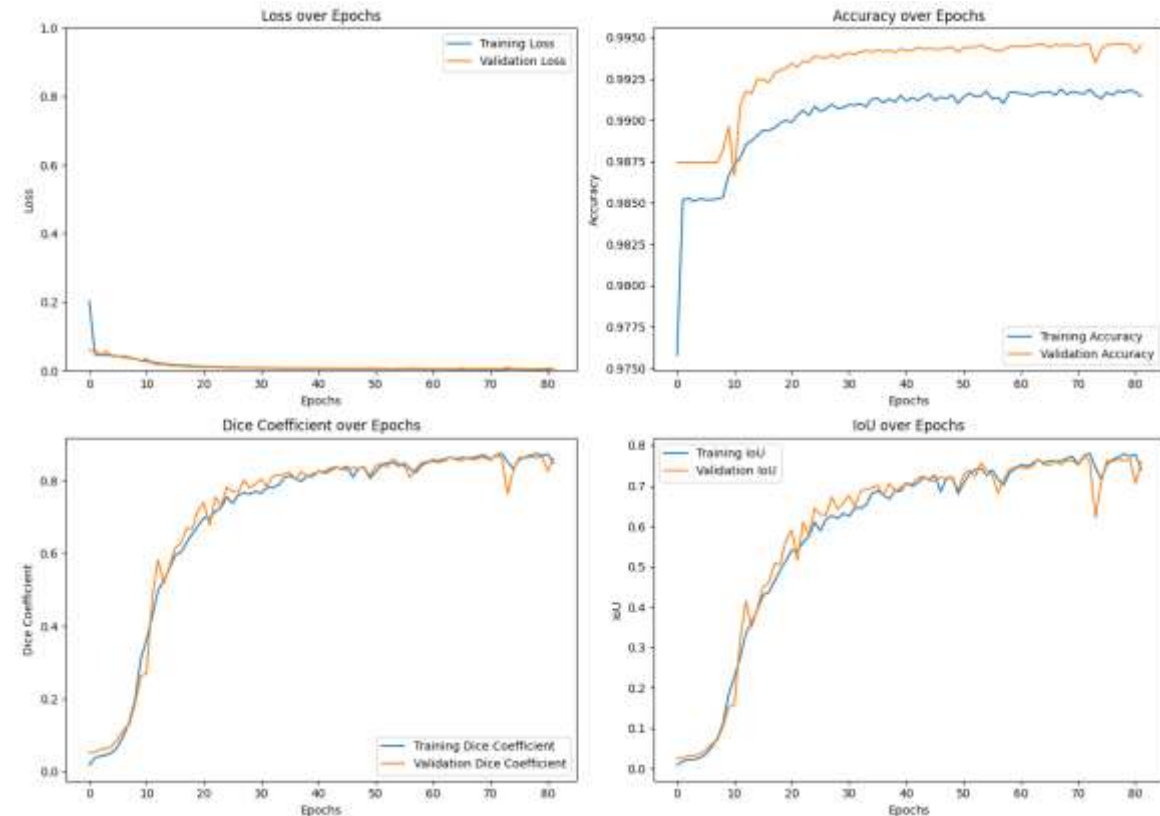
ModelCheckpoint,
EarlyStopping

Validation Loss: 0.0054

Validation Accuracy: 0.9946

Validation Dice Coefficient: 0.8619

Validation IoU Coefficient: 0.7584





5

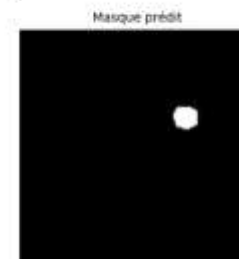
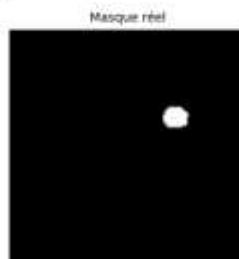
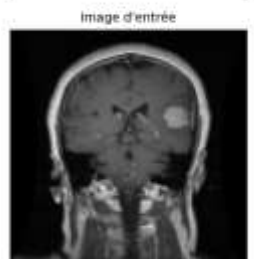
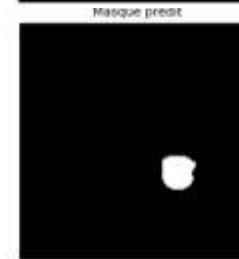
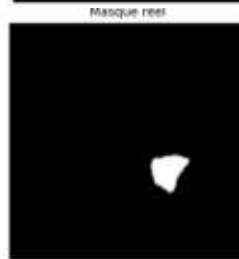
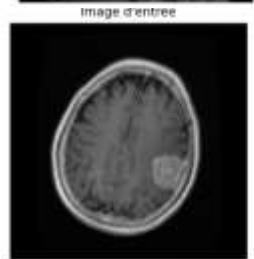
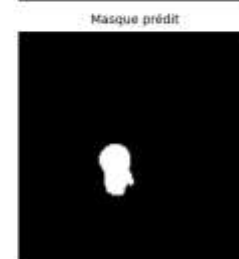
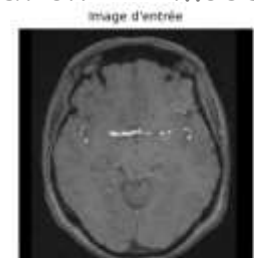
Segmentation : modélisation et évaluation

Dataset et préparation

modèle U-Net

Evaluation des performances

➡ Visualisation des résultats





6

Déploiement





6

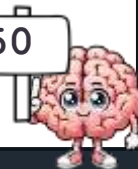
Déploiement

Tkinter

Windows | Frames | Widgets



python





6

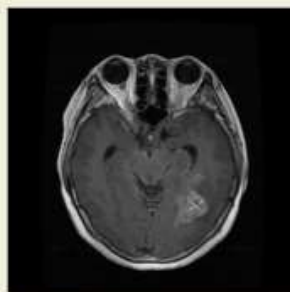
Déploiement

Brain tumor Detection and classification

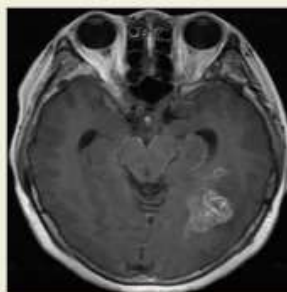
Upload Image

Detection

Reset



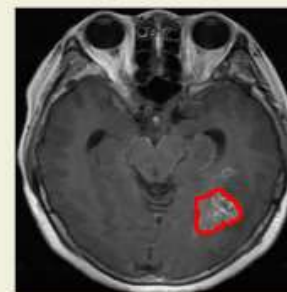
Original Image



preprocessed image



Mask Image



Segmented image

Tumor Type :

glioma_tumor

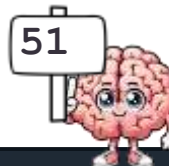
Elapsed Time : 0.44 seconds

Classification Trust : 99.98%

Tumor Size : 825.0 pixels

↓ Save Result

Project by Dhaouadi Ibtihef





7 Conclusion et perspectives



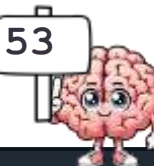


7 Conclusion et perspectives

Conclusion

les perspectives d'amélioration

Ce projet a démontré le potentiel des techniques d'apprentissage profond, notamment les réseaux neuronaux convolutifs et le transfert d'apprentissage, pour améliorer la classification et la segmentation des tumeurs cérébrales à partir d'images IRM.





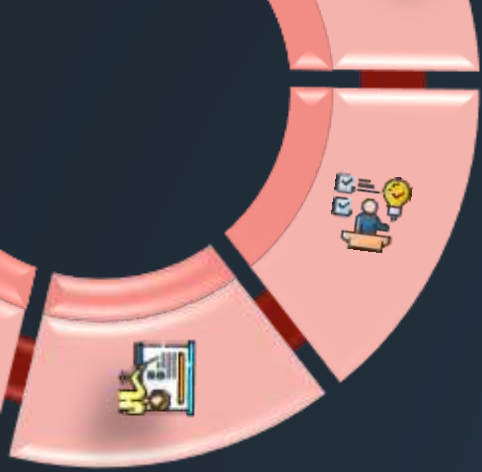
7 Conclusion et perspectives

Conclusion

➡ les perspectives d'amélioration

À l'avenir, il serait pertinent d'élargir l'ensemble de données pour inclure une plus grande diversité de cas et d'étendre le système à d'autres types de tumeurs cérébrales ou à des pathologies cérébrales variées.





Merci pour votre attention

