TU WIEN Informatics

# Predicting Histologic Grade of Meningiomas using Deep Learning in Centralised and Federated Learning Settings

DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur**

im Rahmen des Studiums

**Data Science**

eingereicht von

**Inż. Lukasz Sobocinski**
Matrikelnummer 12123563

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Prof. Dr. Allan Hanbury
Mitwirkung: Univ.Prof. Dipl. Ing. Dr. Georg Langs
Dipl. Ing. Dr. Philipp Seeböck

Wien, 20. Februar 2025

Lukasz Sobocinski | Allan Hanbury

TU WIEN Informatics

# Predicting Histologic Grade of Meningiomas using Deep Learning in Centralised and Federated Learning Settings

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

**Diplom-Ingenieur**

in

**Data Science**

by

**Inż. Lukasz Sobocinski**
Registration Number 12123563

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof. Dr. Allan Hanbury
Assistance: Univ.Prof. Dipl. Ing. Dr. Georg Langs
Dipl. Ing. Dr. Philipp Seeböck

Vienna, February 20, 2025

| Lukasz Sobocinski | Allan Hanbury |
|---|---|

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

# Erklärung zur Verfassung der Arbeit

Inż. Lukasz Sobocinski

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.
Ich erkläre weiters, dass ich mich generativer KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Im Anhang „Übersicht verwendeter Hilfsmittel“ habe ich alle generativen KI-Tools gelistet, die verwendet wurden, und angegeben, wo und wie sie verwendet wurden. Für Textpassagen, die ohne substantielle Änderungen übernommen wurden, haben ich jeweils die von mir formulierten Eingaben (Prompts) und die verwendete IT- Anwendung mit ihrem Produktnamen und Versionsnummer/Datum angegeben.

Wien, 20. Februar 2025

Lukasz Sobocinski

# Acknowledgements

First of all, I would like to thank my motivated and supportive supervisors Georg Langs and Philipp Seeböck from the Medical University of Vienna. Your support and guidance throughout the practical part and your valuable feedback on my thesis were indispensable. I have learned a lot from you.

I would also like to thank Allan Hanbury from TU Wien for supervising the project and for the finalizing comments on my thesis.

Many thanks to the Computational Imaging Research Lab at the Medical University of Vienna for giving me the opportunity to work on this interesting project. It was a pleasure to join this motivated and helpful team.

Also, I am grateful to the Department of Biomedical Imaging and Image-guided Therapy at the AKH hospital in Vienna for providing the data and support. Thank you, Julia, Johaness and Karl-Heinz.

Finally, I would like to thank my caring girlfriend, my family and all my friends who accompanied me during my studies, through both the highs and the lows of this journey. Your support was irreplaceable. Thanks to you my studies were not only educational, but also fun.

# Kurzfassung

Das Meningiom ist der häufigste primäre intrakranielle Tumor. Die Prognose und Behandlung dieser Erkrankung hängen stark vom Schweregrad des Tumors ab, der durch den WHO-Grad definiert ist. Daher ist es von entscheidender Bedeutung, dass die Diagnose im frühestmöglichen Stadium gestellt wird. Deep Learning Modelle haben sich bei ähnlichen Aufgaben als sehr genau erwiesen und können daher zur nichtinvasiven Diagnostik des Schweregrades anhand von MRT-Bildern eingesetzt werden. Das Training dieser Modelle erfordert oft große Datenmengen, so dass es schwierig ist, genügend Proben zu sammeln Proben ohne interinstitutionelle Zusammenarbeit zu sammeln. Federated Learning ist eine neue Methode, die eine solche Zusammenarbeit erleichtert. Ihre Leistung wird jedoch oft durch statistische Heterogenität der Daten in den verschiedenen Einrichtungen beeinträchtigt.

Diese Arbeit untersucht die Machbarkeit des Einsatzes von Deep Learning und Federated Learning zur Bestimmung des WHO-Grades von Meningeomen anhand von MRT-Scans. Der Fokus liegt auf dem Vergleich von Modellen, die mir drei verschiedenen Architekturen des Federated Learning trainiert wurden und zentral trainierten Modellen. Es bewertet speziell die negativen Auswirkungen statistisch heterogener Daten über Institutionen hinweg auf Federated Learning und stellt eine neuartige Methode namens Federated Localized Ensemble vor, um diese zu verringern. Die Methode besteht darin, ein Ensemble von Modellen zu erstellen, die von jedem Kunden trainiert wurden, wobei jedes Modell entsprechend seiner Klassifizierungsgenauigkeit gewichtet wird.

Die Methoden werden anhand eines Datensatzes evaluiert, der aus 186 MRT-Aufnahmen des Gehirns von Patienten besteht, bei denen ein Meningiom diagnostiziert wurde. Da der Datensatz aus nur einer Einrichtung stammt, wird Federated-Learning-Umgebung durch die Integration spezifischer Bildgebungsmerkmale simuliert.

Die Ergebnisse zeigen, dass sowohl Deep-Learning- als auch Federated-Learning-Techniken das Potenzial haben, die nicht-invasive Klassifizierung von Meningeomen zu erleichtern. Sowohl die zentral trainierten als auch die mittels Federated Learning trainierten Modelle erreichten trotz der anspruchsvollen Aufgabe eine hohe Vorhersagegenauigkeit. Dies war auch im Szenario der statistischen Heterogenität der Fall, obwohl die Leistung des Federated Learning geringer war und zwischen den Methoden variierte. Das Federated Localized Ensemble übertraf die bestehenden Methoden nicht.

# Abstract

Meningioma is the most common primary intracranial tumour. The prognosis and treatment of this disease is highly dependent on the severity of the tumour, which is defined by its WHO grade. Thus, it is crucial that it is diagnosed at the earliest possible stage. Deep learning models have been shown to achieve high accuracy in similar tasks and may therefore be used to non-invasively assess this grade from MRI images. Training of these models often requires large amounts of data, making it difficult to collect enough samples without inter-institutional collaboration. Federated learning is an emerging method that facilitates such collaboration. However, its performance is often negatively affected when the data is statistically heterogeneous across institutions.

This thesis evaluates the feasibility of using deep learning and federated learning to assess the WHO grade of meningiomas from MRI scans. It focuses on a comparison between models trained using three different federated learning architectures and models trained centrally. It specifically evaluates the negative impact of statistically heterogeneous data across institutions on federated learning, and introduces a novel method termed Federated Localized Ensemble to reduce it. The method involves the creation of an ensemble of the models trained by each client, with each model weighted by its classification accuracy.

The methods are evaluated on a dataset consisting of 186 brain MRIs of patients diagnosed with meningioma. As the dataset comes from a single institution, the federated learning environment is simulated by introducing specific imaging characteristics.

The results show that both deep learning and federated learning techniques have the potential to facilitate non-invasive grading of meningiomas. Both centrally trained and federated learning trained models achieved predictive accuracy despite the challenging task. This was also the case in the statistical heterogeneity scenario, although the performance of federated learning was lower and varied between the methods. The Federated Localized Ensemble approach did not outperform the existing methods.

# Contents

**Kurzfassung** **ix**

**Abstract** **xi**

**Acronyms** **1**

**1 Introduction** **3**
1.1 Motivation . . . 3
1.2 Machine Learning in Medical Imaging . . . 4
1.3 The Role of Federated Learning . . . 5
1.4 Aims of the Work . . . 5
1.5 Structure of the Work . . . 6

**2 Background** **9**
2.1 Deep Learning . . . 9
2.2 Deep Learning in Cancer Imaging . . . 12
2.3 Federated Learning . . . 14
2.4 Main Challenges in Federated Learning . . . 20
2.5 Federated Learning in Medical Imaging . . . 23
2.6 Summary . . . 26

**3 Methodology** **29**
3.1 Input Data and Prediction Targets . . . 29
3.2 Image Preprocessing . . . 31
3.3 Deep Learning Model . . . 32
3.4 State-of-the-art Federated Learning Frameworks . . . 34
3.5 Federated Localized Ensemble Framework . . . 34

**4 Experimental Setup and Evaluation** **37**
4.1 Dataset . . . 37
4.2 Image Augmentation . . . 37
4.3 Federated Learning Scenario Simulation . . . 39
4.4 Evaluation of the Resulting Classifiers . . . 42
4.5 Experimental Setup . . . 43

**5 Results** **47**
5.1 Comparison of Centralized Training and Federated Learning in the IID Scenario . . . . . 47
5.2 Comparison of Federated Learning Frameworks in the Non-IID Scenario 49
5.3 Performance of Centrally Trained Classifiers . . . . . 52
5.4 Analysis of the Predictive Features . . . . . 53
5.5 Correlation of the Scores on the Test and Validation Sets . . . . . 54
5.6 Weights Assigned to the Models in the Federated Ensemble Method . 59
5.7 Insights from Hyperparameter Tuning . . . . . 59

**6 Discussion** **67**
6.1 Dataset . . . . . 67
6.2 Comparison of Centralized Training and Federated Learning in the IID scenario . . . . . 67
6.3 Federated Learning in the Non-IID Scenario . . . . . 68
6.4 Performance of the Centrally Trained Classifiers . . . . . 69

**7 Conclusion and Future Work** **71**

**Overview of Generative AI Tools Used** **73**

**Bibliography** **75**

# Acronyms

**CDS** Collaborative Data Sharing. 20, 21, 24–26, 71

**CNN** Convolutional Neural Networks. 9–13, 21, 32

**CV** Cross-validation. 43, 44

**DL** Deep Learning. 4–6, 9–14, 23–26, 34, 71, 72

**FedAvg** Federated Averaging. 5, 15–18, 20, 22, 24–26, 34, 44, 45, 48–50, 55, 64, 65, 67, 68

**FedProx** FedProx federated optimization algorithm. 17, 23, 26, 34, 45, 47–50, 55, 65, 68, 71

**FL** Federated Learning. 5–7, 9, 14–17, 19–26, 29, 34, 39, 43, 45, 47, 49–51, 64, 65, 67, 68, 71, 72

**FN** False Negatives. 48, 49, 53, 55

**FP** False Positives. 48, 49, 53, 55

**GAN** Generative adversarial network. 12, 22, 24

**Grad-CAM** Gradient-weighted Class Activation Map. 12, 53, 54, 56, 67

**IID** Independent and identically distributed. 6, 7, 26, 39, 44, 45, 49, 50, 59, 68

**KS test** Kolmogorov–Smirnov test. 21, 40

**KS test statistic** Kolmogorov–Smirnov test statistic. 21, 40

**ML** Machine Learning. 3–5, 9, 12, 13, 26, 67, 71

**Non-IID** Non-independent and identically distributed. 5–7, 9, 14, 15, 17, 20–23, 25, 26, 39, 40, 45, 47, 49, 50, 59, 68, 71

**ResNet** Residual neural network. 10, 13, 14, 31–34, 52, 59, 63, 64, 71

**RNN** Recurrent Neural Network. 12, 15

**ROI** Region of Interest. 12, 13

**SGD** Stochastic Gradient Descent. 23, 33

**TN** True Negatives. 53, 55

**TP** True Positives. 49, 52, 55

**WHO** World Health Organization. 3, 4, 6, 13, 26, 29, 37, 38, 53, 54, 57, 69, 71

CHAPTER 1

# Introduction

Meningioma is the most common primary intracranial tumour. It arises from the meninges in the brain and accounts for 13% to 36.6% of all primary tumours of the central nervous system [46]. Most meningiomas grow slowly for years without causing any symptoms, but in some cases they can cause severe risk to the patient. The severity of meningiomas is described by the World Health Organization (WHO) tumour grade, which can range from 1 (benign tumour) to 3 (the most aggressive form). The prognosis and treatment of the patient is highly dependent on this grade, so it is important that it is identified as early as possible. Most meningiomas are benign (in about 90% of cases) and rarely require surgery [62, 46]. However, surgery is the reference treatment for grade 2 and 3 meningiomas, as they are associated with a higher rate of recurrence and pose a much higher risk to the patient [46]. Table 1.1 shows the age-adjusted incident rate of meningiomas by WHO grade and sex in the USA (adapted from [30]).

## 1.1 Motivation

Currently, the grading is determined by pathological examination, which presents many challenges. These include the difficulty and risk of obtaining enough tissue for examination and the need to observe meningiomas for several years to monitor disease progression. Therefore, the idea of grade assessment from MRI scans is an interesting alternative or complement to pathological examination. However, this task is challenging even for trained clinicians [23]. Thus, there are several approaches to how technology can help clinicians make faster and more accurate diagnoses using medical imaging. One of these is Machine Learning (ML), where the information present in the historical data is used to infer predictions for unseen samples.

| WHO Grade | Male | Female |
|---|---|---|
| 1 | 3.68 | 8.56 |
| 2 | 0.26 | 0.30 |
| 3 | 0.08 | 0.09 |

Table 1.1: Table shows the age-adjusted incident rate of meningiomas by WHO grade and sex in the US based on the study described (adapted from [30])

## 1.2 Machine Learning in Medical Imaging

In the literature, several traditional ML models have been used to analyse medical images. They produce results of varying quality from task to task, with some yielding satisfactory outcomes. However, given the fact that the tasks in the medical domain are often challenging, these models often lack predictive power [56, 32]. In addition, traditional models are not designed for image processing by default. One of the reasons is that it is difficult for them to exploit the spatial relationships between the objects. There are feature engineering methods that can be applied to overcome these problems, such as Radiomics [59], but it is often difficult to fine-tune them for a specific task.

Another ML approach to the problem is Deep Learning (DL). It has been shown to perform well in a wide range of scenarios, including various applications in radiology [56, 32, 28]. Unlike traditional ML methods, it requires little to no feature engineering. In addition, the DL models are well suited to image analysis because they can encode spatial relationships well. In some cases, they perform as well as or better than trained radiologists [37, 36, 9]. For example, such models are already being used in commercial medical imaging products, such as the one described in [3], which assists the clinician during a colonoscopy procedure.

DL has proved effective in classifying and segmenting several types of brain tumours, including meningioma [27, 6, 2, 42]. Given the limitations of the current histopathological examination method described above, the development of a model that could accurately diagnose grading from MRI images would be valuable. As well as reducing the need for biopsies, this would make diagnosis cheaper, easier and they could be made more frequently [2, 42, 23]. Some studies explored this idea using both DL and traditional ML, but there is no definitive solution yet, as determining the grading of meningiomas from MRIs is a challenging task [2, 42, 23].

One limitation of DL models is that they require a substantial amount of data to work [12]. There are many ways to overcome this problem, such as transfer learning [10]. These can help to some extent, but still require a dataset that is sufficiently representative of the population. In medicine, the population distribution is very broad and depends on many factors. In addition, the amount of annotated data is small because labelling requires highly trained practitioners and is expensive. As a result, data from a single institution is often insufficient to produce generalizable results. There are not many public datasets, so there is a need to share data between institutions. However, in the medical field,

most data is private. Therefore, data sharing requires a lot of coordination and official approvals, and in practice it is often impossible.

## 1.3 The Role of Federated Learning

To overcome this problem, [40] proposed a method to train a shared ML model using the data available on different sites without sharing the data itself, thus preserving the privacy. This method is called Federated Learning (FL). It is an interesting alternative to standard anonymisation because it does not require the data to be shared at all. Several experiments have shown that it has a high potential [72]. Apart from the privacy improvements, it showed a much lower communication cost compared to other methods such as the synchronised stochastic gradient descent approach [52].

Although FL achieves good results in many real-world applications, there are still many challenges and issues to work on. One of these is the challenge posed by statistically heterogeneous and Non-independent and identically distributed (Non-IID) data between institutions. This is often the case in the medical field, as not only does the population vary between sites, but the methods of data collection can also be different. Even though simple methods such as Federated Averaging (FedAvg) can deal with some degree of statistical heterogeneity between clients [40], the performance degradation is inevitable when working with heterogeneous datasets [71]. Improving ways to address these challenges has been identified as important in several studies [12, 72, 64, 66, 71].

Therefore, given the importance of meningioma grading and the limitations of current diagnostic methods, it would be beneficial to enable grading of meningiomas from MRI images. The DL models, which have shown good performance in similar tasks, could be an interesting solution. However, they usually require large annotated datasets to achieve satisfactory results. Data from a single institution is often not enough, especially for rare diseases such as meningioma. Data must therefore be shared between different medical institutions, which is often difficult due to privacy and technical issues. In addition, the fact that populations and data distributions are often highly heterogeneous across institutions needs to be addressed.

## 1.4 Aims of the Work

The thesis has two main objectives. The first is to develop a DL model capable of predicting the histological grade of meningiomas. The second aim is to compare models trained in a federated fashion, in particular, to evaluate a framework to mitigate the difficulties caused by heterogeneous data distribution between different sites participating in the training.

### 1.4.1 Research Questions

The following research questions were defined:

**RQ1: How accurately can a DL model predict the histological grading of meningiomas?** The main aim of this question is to investigate the accuracy of identifying the WHO grading of meningiomas from MRI scans in the Independent and identically distributed (IID) data scenario. The answer to this question indicates whether a DL model can produce clinically useful results on the given dataset.

**RQ2: How does the classification accuracy of state-of-the-art FL approaches compare to that of centralised training on the given dataset?** By design, models trained with FL achieve either the same or worse results than centrally trained models, as the training process is less optimal. It is therefore important to compare the quality of a federated model with that of a centrally trained model.

**RQ3: To what extent does Non-IID data across FL clients affect classification accuracy on the given dataset?** Statistical heterogeneity of data across clients can significantly affect the performance of FL [12, 72, 64, 66, 71]. The aim of this question is to understand the extent of the phenomenon on the given dataset and understand how different FL frameworks can address it.

**RQ4: Can a FL framework based on model ensembling reduce the adverse impact on classification accuracy when data is Non-IID between clients?** Since the Non-IID data between clients often hinders the performance of the existing FL frameworks, a novel FL framework was implemented and evaluated on the dataset.

### 1.4.2 Hypotheses

Based on the research questions, two hypotheses were formulated:

**Hypothesis 1** Statistical heterogeneity of data across clients significantly hinders the performance of FL on the given dataset.

**Hypothesis 2** Models trained using different FL frameworks have different prediction accuracies on the given dataset.

## 1.5 Structure of the Work

The structure of the thesis is as follows:

**Chapter 1** outlines the background information on meningioma, explains the motivation and introduces the problem statement. The second part of the chapter formulates the research questions and hypotheses.

**Chapter 2** introduces the scientific background and the state of the art in Deep Learning and Federated Learning. It provides an overview of related work in these areas with a focus on cancer imaging and meningioma.

**Chapter 3** describes the methodology used in the experiments. In the first part, it defines the prediction targets, introduces data preprocessing methods and describes the classifier used to make the predictions. The second part introduces the Federated Localized Ensemble FL framework.

**Chapter 4** explains the experimental setup and the evaluation methodology. It introduces the dataset, the image augmentation techniques, and describes how the comparison of the different FL frameworks was performed. Additionally, it details the methodology used to simulate the Non-IID data across the clients.

**Chapter 5** describes the results of the experiments. The first section focuses on comparing the results obtained using different FL frameworks, and the centrally trained classifier. The results are reported for both the IID and the Non-IID scenarios.

**Chapter 6** discusses the findings and results of the experiments, summarises the findings and answers the research questions and hypotheses.

**Chapter 7** formulates the conclusions of the thesis and indicates the future directions of work.

CHAPTER 2

# Background

This chapter presents the background and state of the art relevant to the thesis. The first section starts with an outline of the main DL concepts used in the thesis, focusing on classification using Convolutional Neural Networks (CNN)s, transfer learning, and methods of image preprocessing and augmentation. It also presents approaches to cope with limited data availability. Afterwards, it provides an overview of the DL methods used in cancer imaging. It includes examples of studies addressing the problem of meningioma grading, which may serve as an overview of the complexity of the task and the methods applied to it in the literature.

The second section introduces the concept of FL. It begins with an explanation of the most popular FL framework and the methods of model evaluation in the federated fashion. Then, an introduction to the problem of Non-IID data across institutions is outlined, together with a description of different types of Non-IID scenarios. Finally, several papers and frameworks that address this problem are presented.

The final section focuses on the use of FL in the medical domain. It is outlined why FL can significantly improve the quality of models, with several examples from the literature. Additionally, selected papers describing and addressing the problem with Non-IID data are presented.

## 2.1 Deep Learning

ML is a branch of computer science in which algorithms are trained to solve various problems or tasks by learning from historical data. Currently, most commercially available ML methods use the traditional "supervised" ML methods. There, the model is provided with the ground truth, such as the name of a disease, to learn an optimal decision boundary within a given feature space [35].

The DL models consist of multiple (usually more than three) layers of connected nodes called a neural network. It the main principle, they attempt to simulate the way how the human brain works. Each layer takes an input from the neurons in the previous layers, conducts a mathematical transformation (such as averaging, convolutions, summing), and passes it on to the neurons in the next layer. The different layers in a DL architecture aim to generate differt features and apply various mathematical operations to return a prediction.

### 2.1.1 Convolutional Neural Networks

While there are many DL model architectures, CNNs are the basis of the majority of the ones used for the computer vision tasks, including classification of medical images. The structure of CNN aims to mimic the connectivity patterns of neurons in the visual cortex of animals and humans[17], where individual neurons only respond to stimuli from a limited part of an image. In the context of the image processing, it means that CNNs can capture the spatial dependencies between the components and pixels in an image and place each of them in the context of its surrounding components.

The input image is provided a CNN as a matrix of pixel values. It then goes through three types of building blocks or layers: convolutional layers, pooling layers, and the classification layers. These layers are arranged in various ways to form specific CNN architectures, which are then trained by finding appropriate kernels, filters, and weights which try to learn a mapping between the input image and the ground truth labels provided in the training dataset.

In general, the predictive power of DL models comes from their depth and large number of layers. The training process aims to minimise the differences between the model's output and the given true labels through a process called gradient descent. However, in regular CNN models, adding more layers tends to result in significantly lower gradients in later layers than in earlier layers. This phenomenon is called the "vanishing gradient" problem, and it makes training deep models very difficult or sometimes impossible.

Residual neural network (ResNet) is a variant of the CNN architecture that aims to reduce this problem. In 2015, [21] proposed a novel model structure that reduces this problem, allowing much deeper networks to be developed. Their structure is based on the concept of residual blocks, which are blocks of layers where the input is added to the final output through a skip connection. An example of such a layer is shown in Figure 2.1. These skip connections allow the deeper layers to receive inputs in the earlier layers, thus avoiding the "vanishing gradient" problem.

### 2.1.2 Data Augmentation

Data augmentation involves artificially increasing the size and diversity of a dataset by applying transformations to existing data. This is an important technique for improving model generalisation and reducing overfitting. By introducing variation into the training

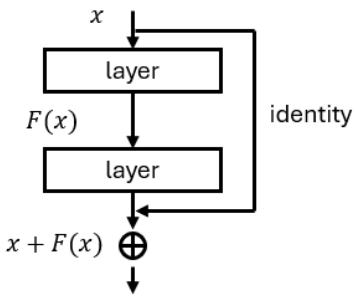


Figure 2.1: A residual block which skips two sequential layers.

data, augmented datasets simulate real-world variability and enable models to perform better on unseen examples [55].

In image processing, data augmentation techniques include any combination of affine transformations (rotations, flips, translations), cropping, occlusion, intensity adjustment, and noise addition [11]. For example, flipping an MRI image horizontally creates a new image that retains the same label, but adds variety to the dataset.

This increases artificially the size and variability of the dataset, which then increases the robustness of the model and reduces the need for large-scale data collection.

### 2.1.3 Transfer Learning

Transfer learning is a technique where a pre-trained model which was developed for one task is adapted instead to a different but related task. This approach leverages knowledge that a model has already gained in the previous training (usually over a larger dataset) and adds new domain-knowledge from a dataset which is usually much smaller. This makes it very beneficial in situation where the collection and appropriate labelling of data is challenging; for example, in the case of medical imaging.

For example, a CNN model trained on ImageNet, a huge dataset containing millions of images across different categories, can be fine-tuned for a specific use case such as Alzheimer's diagnosis/prognosis and tumour segmentation. Rather than training a model from scratch, which requires extensive computing resources and data, transfer learning enables faster model development and often achieves higher accuracy by building on the basic features learned during pre-training.

This approach is growing in popularity and suggests that fine-tuning all parameters, rather than just the final layers (which is a more traditional approach), has shown significant improvements in accuracy [58, 10].

### 2.1.4 Gradient-weighted Class Activation Map

Historically, DL networks in general and CNN in particular have provided significant performance boosts in image classification tasks compared to traditional machine learning

methods. However, explainability remains a major challenge with DL networks due to the black-box nature of these algorithms, where an output is generated by not necessarily clear neuron activations in some sort of structure. Gradient-weighted Class Activation Map (Grad-CAM) is a technique that visualises areas considered important by a CNN[50].

The method calculates the gradient of the final model output with respect to the final feature map or layer in a CNN and then takes the global average and multiplies this by the feature map. It is then reasonable to assume that higher values or gradients would be areas that the model considered important contributors to the final model score. This can then be displayed and interpreted as heat maps over the input image.

In [14], the researchers showed that the global average pooling at the end of the Grad-CAM process occasionally tends to highlight irrelevant locations that the model did not actually pay attention to. As an alternative, they proposed HiResCAM, which multiplies the gradients *element-wise* with the feature map instead of the global average gradient. It is shown that this better reflects the model's computations and the actual Region of Interest (ROI)s.

## 2.2 Deep Learning in Cancer Imaging

The DL models are also used in many fields of medicine, and they prove to achieve satisfying results across various tasks and workloads both in academia [32, 56, 28, 69] and industry [3]. They often outperform traditional ML models and require less or no task-specific feature extraction [28, 69]. Their flexibility allows them to achieve high performance in various tasks such as classification, segmentation, image registration, image reconstruction or object detection, which are tasks often encountered in the medical field [3, 69, 56].

Their high performance is especially noticeable in the imaging tasks, where ML can provide a non-invasive and cheap way of diagnosis. Techniques such as CNNs, Recurrent Neural Network (RNN)s and Generative adversarial network (GAN)s allow processing large amounts of data and understanding the spatial relationships and patters to provide clinically useful results. As a result, DL is often used in radiology, where it not only assists clinicians and speeds up diagnosis, but sometimes achieves higher diagnostic accuracy than trained radiologists [2]. Such assistance is particularly valuable in cancer imaging, where the accuracy of diagnosis is critical and difficult even for trained clinicians.

[27] presents a study aimed at classifying gliomas, one of the most common brain tumours. Similar to the problem in the thesis, the paper classifies the grading of the tumour from MRIs. It uses a CNN model called AlexNet [29]. The model achieves a relatively high level of accuracy, even though neither augmentation nor transfer learning was used. However, the researchers mention that the use of a pre-trained model would probably be beneficial.

Also in [6], the researchers report good results for glioma classification. In the study, the model is set up to predict the isocitrate dehydrogenase (IDH) status of gliomas from

MRI images with manually marked ROIs. Instead of using the entire 3D volume, the three central slices of each axis were extracted and used as input to a ResNet model. The researchers utilized augmentation to improve the training process and achieve reasonable prediction accuracy.

### 2.2.1 Computational Meningioma WHO Grading Classification

Treatment and disease progression of meningiomas are highly dependent on tumour grading and need to be monitored over time. However, the grading is currently only obtainable by an invasive histopathological examination. It is therefore an area of considerable interest. Nevertheless, grading from imaging data alone is a difficult task [42, 39, 23].

The task of automatically grading meningiomas from MRI images has been addressed in the literature. Several retrospective studies using ML for this task are summarised in [42] and [39]. Researchers approach the problem using both traditional ML and DL techniques. Traditional methods often use radiomics [59] and other feature extraction methods to convert images into a tabular format and make a prediction. One such study [9], reported a classification accuracy similar to that of a trained clinician. However, the accuracy of the model was still relatively low.

Although DL has been shown to have high performance for brain tumour classification, a relatively small number of studies have applied it to meningioma grading. The existing DL papers use a variety of different CNN models, such as InceptionV3 [57] or AlexNet [29], to approach the task. They also use different MRI modalities, often more than one at a time. However, most papers train and evaluate the methods on rather small datasets, as larger public datasets (for example the Brats dataset [41]) rarely include the WHO grading of meningiomas. The datasets are also heavily skewed towards grade 1, due to the relatively low prevalence of grades 2 and 3. Therefore, papers sometimes transform the task into a two-class classification problem. [42, 39]

The papers report different results, with accuracies ranging from 55% to 95% [42, 39]. For example, in [68], a CNN model achieved an accuracy of 81.5% on a relatively large dataset of 5088 samples. However, most studies evaluated the methods on much smaller datasets with a high label bias. Therefore, the reported results should often be taken with a grain of salt.

### 2.2.2 Approaches to Cope with Limited Training Data

Training of DL models usually requires a lot of data. In the medical field, it is often a challenge to collect sufficiently large datasets. This is because data collection is often expensive and time-consuming, and labelling the data requires skilled professionals. In addition, medical data is mostly private, and it is difficult to share datasets between institutions [10]. Thus, methods for reducing the amount of data required for training are needed.

One of these, which is widely used in the medical field, is transfer learning. Its main principles were outlined in the section 2.1.3. As mentioned, there are a number of publicly available pretrained models. Most of them have been trained on large public datasets containing a huge variety of real-world images. However, medical images have a very specific structure and properties that are often not well represented in the large benchmark datasets such as ImageNet [13]. Therefore, pretrained models tuned specifically for use in the medical domain can facilitate transfer learning and improve results [10]. An interesting pretrained model called MedicalNet has been proposed in [10] and implemented in Pytorch[1]. The researchers propose multiple 3D ResNet models that have been trained on data containing different modalities, target organs and pathologies. It is shown that using MedicalNet for transfer learning can enhance the model performance.

Another method of improving results with limited data is data augmentation. It is also used in other fields, but it is important to tailor it to the specific task. There are many augmentation methods in the literature that are optimised for medical imaging. Many of them are described in [11].

Often the size of a dataset available at one institution is too small to train an accurate model, even using the techniques described above. It is therefore necessary to share data between institutions. However, this may not be possible due to privacy and technical issues. Many papers propose to use FL to overcome this problem. An overview of the main principles and publications on this method can be found in the next sections.

## 2.3 Federated Learning

The term "Federated Learning" was first introduced in [40]. The paper proposed a method to train a shared DL model using the data available on different client sites without sharing the data itself, thus preserving the data privacy. It is an interesting alternative to standard anonymization, as it removes the need to share the data completely. Experiments showed that his method has a high potential, even in the case of unbalanced and Non-IID data across the clients. Also, apart from the privacy improvements, it showed a much lower communication cost compared to other methods like the synchronized stochastic gradient descent approach [52].

The section begins by introducing examples of the use of FL in industry. Then, it describes two FL frameworks that will be used in the experiments. Afterwards, the existing methods for model evaluation are outlined. Lastly, the problems that arise when data is Non-IID across clients are presented, together with an overview of existing approaches to tackle this problem.

[1] https://github.com/Tencent/MedicalNet

### 2.3.1 Federated Learning in the Industry

Federated learning has already been used in the commercial settings. One of its first applications was made by Google and described in [20]. An RNN language model was trained there using FL. It allowed using the data stored on edge devices without the need to share it with the server, thus increasing the privacy of the users. In the paper, centralized training using stochastic gradient descent is compared to the FL approach using the FedAvg algorithm. Model trained using FL achieved a better performance than a model trained just on the data available on the server side, as more data could be used to train the model. This was the case even though the data was highly Non-IID distributed.

Another example from the industry is a white paper that focused on applying FL in the commercial setting [38]. In the work, an end-to-end FL framework is proposed. Except for the FL algorithms themselves, it implements various technical aspects of the process, like communication protocols between the participating parties, coordination of the learning process and handling data heterogeneity.

There are multiple approaches to perform FL. Two of these, implemented and evaluated in the thesis, are described in the next sections.

### 2.3.2 Federated Averaging

There are several different approaches on how to perform FL. Amongst those, FedAvg is the most well-known framework, and it often serves as a baseline for evaluation of the novel methods [72]. Even though the idea is relatively simple, it yields satisfying results in many settings [72, 12, 52].

Server is the main player in the framework. It orchestrates the training process and aggregates the results of the clients participating in training. A simplified graph visualizing the training process can be seen in 2.2. The training starts by initializing a global model on the server side. Then, the training itself starts. It is an iterative process consisting of 4 steps, which are repeated for a predefined number of rounds:

1. Server sends the global model parameters to the clients participating in the training. Depending on the training settings, it uses all the clients or just a subset of them.
2. Clients fine-tune the global model over a predefined number of epochs $n_{\text{local epochs}}$ using the locally available data without sharing it.
3. Clients send to the server the weight vector updates obtained by fine-tuning the global model.
4. Server aggregates the weight updates from the clients. In this framework, a simple weighted average is used. The update from each site is weighted by the number of samples that the site has. It can be expressed by the following formula:

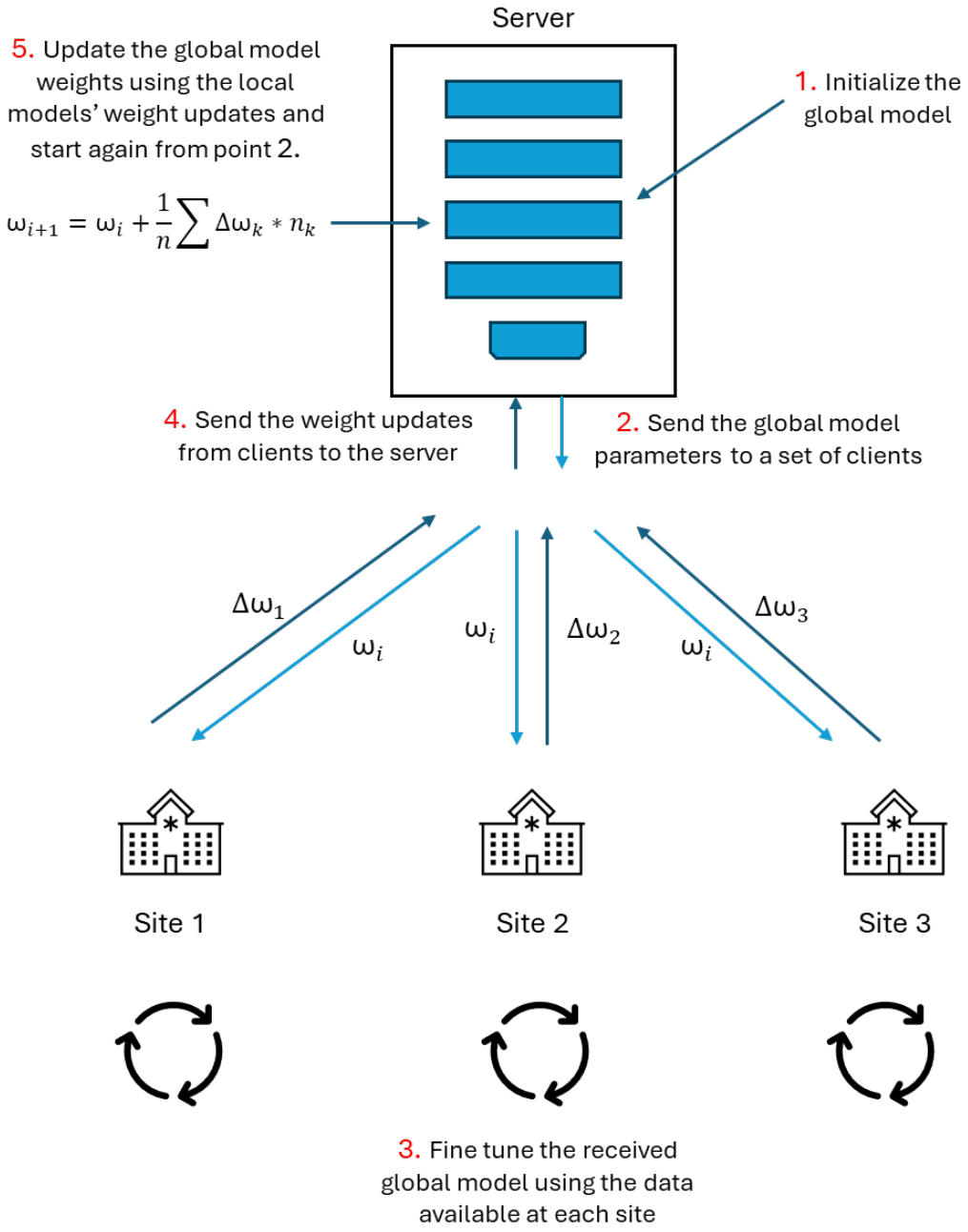


Figure 2.2: A simplified schema depicting the training steps of the most common FL approach called FedAvg.

$$w_{i+1} = w_i + \frac{1}{n}\sum_{k=1}^{K} n_k \cdot \Delta w_k$$

Where:

- $w_i$ are weights of the global model at a time point $i$.
- $K$ is the number of clients participating in a round.
- $\Delta w_k$ is a weight update coming from a fine-tuned model on the client $k$.
- $n$ is the total number of samples.
- $n_k$ is the number of samples available on a client $k$.

Then, after the weights of the global model are updated, the next round starts.

Those steps are repeated for a predefined number of rounds $n_{rounds}$. Then, after the last round, the best global model is returned and can be used by all the clients to make inferences. [40, 72, 20, 61, 67]

Apart from the training itself, the methodology for selecting the best model generated during training has to be defined. In the regular centralized training, this choice is made by selecting the model that performed best on a dedicated validation set. A similar approach is adapted to work in the FL setting. A portion of the data at each institution is designated as its validation set. After each round of training, the server sends the global model to each client. The clients then evaluates this global model on their local validation sets and reported the classification performance to the server. A simplified diagram of this procedure is shown in Figure 2.3. At the end of training, the model with the best prediction performance is selected as the final global model. The whole procedure was implemented using the Flower library[2].

### 2.3.3 FedProx Framework

Another framework, called FedProx federated optimization algorithm (FedProx), was also implemented and evaluated in the thesis. It aims to reduce the impact of Non-IID data across clients, which often occurs in the medical field. The framework has been implemented using the guidelines described in the paper that introduced this method [33]. However, only the part of the methodology that proposes a modification of the loss function has been implemented.

The framework is very similar to FedAvg described in the section 3.4. The training procedure is the same, the only difference is the loss function used to train the models on the client side. The loss function is extended by the proximal term, which measures the deviation of the local model from the global model. The extended loss function can be formulated as

$$loss_i = loss_i^m + \frac{\mu}{2}||\omega_i - \omega_g||^2$$

where,

[2] https://flower.ai/

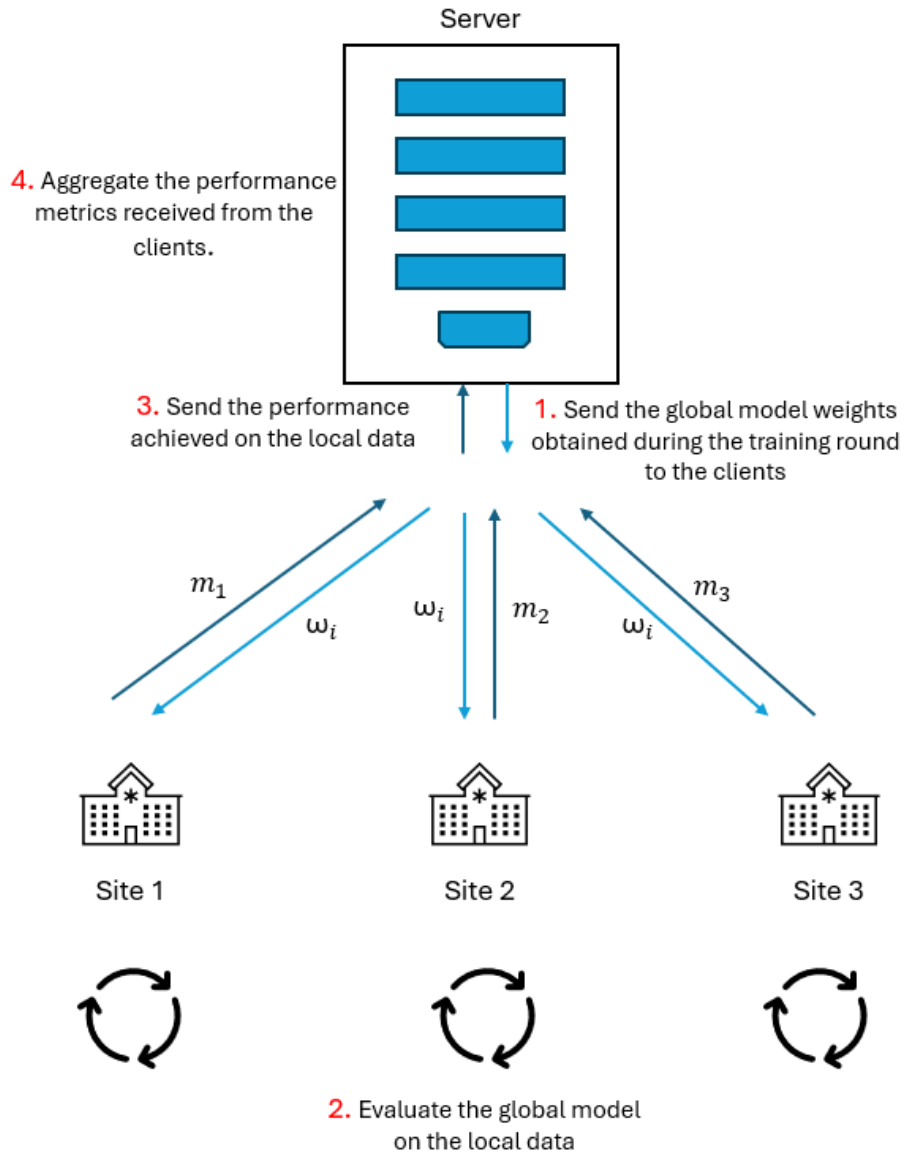


Figure 2.3: A diagram illustrating the process of selecting the best model in the FedAvg framework.

1. $loss_i^m$ is the loss of the model computed in the traditional way, i.e. the cross-entropy loss as described in the section 3.3.1.
2. $\mu$ a hyperparameter defining the strength of the regularisation.
3. $\omega_i$ denotes the weights of the local model at the local training epoch $i$.
4. $\omega_g$ are the weights of the global model.

It is important to note that choosing the correct value of $\mu$ is crucial. [33] suggests setting it empirically.

### 2.3.4 Model Evaluation in Federated Learning

The evaluation procedure in FL is different from the centralized approach. To evaluate the performance of the model robustly, one has to have a validation dataset with a distribution that is representative of the population. If it is not the case, the evaluation might be biased and not applicable to the unseen data. Getting such a dataset might be challenging in a FL setting, as the data resides on different sites, and there might be no global dataset. Moreover, the variance between the sites might be high, different data biases might be present, etc. Getting a good validation dataset is especially cumbersome in the medical domain, as the variance between the distributions of the data from the different centers might be significant there [70].

One possibility to tackle this problem is to create a shared, preferably public dataset. It should be done using the data from various, preferably diverse sites [72]. If done right, such a dataset can be a good proxy for simulating the population distribution [64, 61]. It can be used by the server to validate the global model after each global round. Additionally, this approach improves the reproducibility of the research, as the data can be shared with the community. Nevertheless, in order to do it, parties need to share their private data. Thus, creating such a dataset often is impossible due to issues like privacy concerns, lack of incentive to share the private data or the lack of technical capabilities. Additionally, it may be difficult to build a uniformly distributed dataset, as the population distribution across the clients is not known. Thus, often other evaluation schemas must be employed. [12, 72, 52, 40, 64]

Another approach for the model evaluation is to perform Leave-One-(institution)-Out testing. This approach proposed in [52], is a variant of leave-one-out cross-validation. In this approach, one leave-one-out score is calculated for each institution taking part in the training. In order to calculate this score for an institution, during fitting, this institution is left out from the training set. Then, the obtained global model is sent to that institution so that it evaluates the model on its local data. After calculating the leave-one-out score for each institution, the average is calculated and reported as the model score. This validation method can be performed without sharing the data by the clients, which makes it more flexible. Also, it is a good estimator of the model performance [52]. Nevertheless, it is a very costly approach, as one must fit one model for each left-out institution.

An alternative, less time-consuming approach is to designate a portion of the data at each institution as a validation set. After each training round, the server can transmit the global model to each client. Then, the clients can assess the model on their local validation sets and report the results to the server.

Apart from the model validation during and after training, one also needs to evaluate if a methodology used for FL yielded good results. Thus, a baseline is needed. One of the

most commonly used baselines is to compare the model created by a FL approach to the model yielded by Collaborative Data Sharing (CDS) [72, 12]. CDS in this context describes a scenario where all the institutions share their data with one center, and a centralized training is performed there. This evaluation method compares a FL approach to the "easiest" scenario, so one where the data is not distributed. Using this technique, one can see how much of a performance decrease one has to expect, if FL approach is used. Nevertheless, private data of each institution needs to be shared for this method to work, which might be impossible. Another idea, widely used to estimate the performance improvement of a novel FL algorithm, is to compare its results to the FedAvg framework [70].

## 2.4 Main Challenges in Federated Learning

Even though FL achieves good results in many real-world applications, there are still many challenges and issues to work on. The main fields of research in FL were first defined in [40]. They are often recognized and addressed in the literature [12, 72]:

- The data across nodes is unbalanced and statistically heterogeneous.
- The data distributions across different clients are Non-IID.
- Communication across different nodes might be limited and unreliable.
- Often the number of nodes is very high compared to the average number of data points at a node.

In addition, two other difficulties are often mentioned in the literature [72]. The first is that even if the private data is not directly shared, some personal information may still be leaked during a FL process [54]. For example, in [45], it was shown that just sharing of the model parameter updates may pose a threat to the privacy of the data in FedAvg. Secondly, the architecture and scalability of a FL system can be cumbersome [38].

### 2.4.1 Challenges Arising from Non-IID Data across the Clients

This thesis focuses on coping with the challenges coming from statistically heterogeneous and Non-IID data. Those properties are often encountered in the real world datasets; thus, it is important to study them [12, 72, 64, 70, 16, 66, 71]. Even though FedAvg can deal with some degree of statistical heterogeneity between the clients [40], degradation of the performance is rather unavoidable when working with heterogeneous or highly Non-IID datasets [71].

Firstly, it is crucial to understand why the performance of FedAvg deteriorates when the data across the clients is heterogeneous. The main challenge is that the weights of the local models might diverge significantly from the global model. In other words, even though at the beginning of each round every local model starts from the same set of

weights, they can diverge quickly during the local fine-tuning. It is caused by the fact that each model overfits to the training data available at a site, which might have a different distribution than the global dataset. The grade of divergence depends strongly on the degree of the data heterogeneity. This phenomenon may be even stronger if the number of epochs for the fine-tuning is high, as then the local models overfit to the client-specific distribution even more. If the local models are significantly different, the global model obtained by averaging might diverge significantly from the "perfect" model (the model that would be obtained if trained by CDS). It leads to a much slower convergence of the training, and in the case of strongly Non-IID data it might even fail to converge at all [65, 72]. For example, in [71] it was shown that the weights of the CNN model diverge more in the Non-IID setting.

There are several categories of Non-IID data. Amongst those, there are two, which are commonly encountered in the real-world settings [72]. The first one is the attribute skew. It refers to a situation where the distribution of features differs between clients, resulting in different distributions of the model inputs $X$. This can be caused by factors such as different populations that are available to the clients or different data acquisition methods. Another frequent Non-IID data type is the labels skew. It occurs when the distributions of the target $y$ are different between the clients.

Although these biases are often present in existing datasets, it is often necessary to simulate Non-IID data in order to evaluate novel FL methods. Many papers, due to the data scarcity, simulate the FL by splitting a dataset between several virtual clients [12]. There are multiple approaches to emulating both the attribute and label skews in such an artificial split. Often it is possible to analyze the dataset properties and try to identify clusters of samples with similar attributes. For instance, samples can be clustered by the method of data collection, or the location where the sample was taken. Each cluster can then be assigned to a different site. This approach facilitates the creation of meaningful differences between clients, similar to what occurs in the real world. However, datasets often lack metadata and do not have enough data points to create a good proxy for a real-world scenario. Thus, simpler methods of Non-IID data simulation are needed, particularly if the dataset comes from only a single institution.

When dealing with image data, the simulation of label skew is typically easier [72, 71]. For instance, in [70], the data heterogeneity was solely simulated through the introduction of a label skew between clients. The paper outlines a series of experiments, each of which incorporates a different level of data heterogeneity between the client. The label skew was achieved there by dividing the dataset into imbalanced subsets, and then assigning them to different clients. The researchers used a two-sample Kolmogorov–Smirnov test (KS test) to quantify the degree of data heterogeneity. The test statistic, called Kolmogorov–Smirnov test statistic (KS test statistic) was calculated between each pair of institutions [47]. The higher the statistic, the greater the difference between the clients' distributions. This approach is a simple, yet highly interpretable method for simulating label skew. It might also simulate the attributes skew indirectly; nevertheless, it is not guaranteed [72].

### 2.4.2 Methods to Reduce the Impact of Non-IID Data

As mentioned above, there is a lot of research aimed at solving challenges arising from the Non-IID distribution across the clients. There are few main directions how the researches approach the problem.

#### Data Augmentation

Data augmentation can be used to enrich the datasets available at the client sites to solve the data imbalance issue. One approach using this technique was proposed in [15]. In the paper, it was proposed to first calculate a median number of samples for each class. Then, each client with a lower sample size for a class than the median for this class, augmented its private training dataset to match this value.

Another approach to augment the clients' datasets is to use a GAN. For example, in [8] and [4] Dual-GANs were utilized to improve the classification of skin lesions. In order to augment the private datasets, a generator was initially trained on the server site. This was done using a set of discriminators, one on every participating client. All the sites trained the generator simultaneously, thus enabling the generator to access indirectly the distributions residing on every site. Once the generator training completes, then it can be used to augment the private datasets with samples that aim to simulate the global distribution. The paper reported a satisfactory performance of the method.

#### Modification of the Weights Averaging Method

The methodology used for the aggregation of weights on the server side can significantly impact the performance of a FL framework. This is especially true in the Non-IID setting [71]. There are multiple approaches in the literature that aim at improving the weighting algorithm used in FedAvg.

One such method is the Inverse Distance Aggregation algorithm, which was first introduced in [63]. The method is similar to that of the FedAvg framework, with the exception that it de-weights updates coming from models that deviate significantly from the average weight update. The update from each model is weighted by the inverse distance of the client parameters to the average model of all clients. This helps to minimise the negative impact (also called model poisoning), coming from diverging models. Additionally, the paper proposes an extension of this weighting called Inverse Training Accuracy. This extension penalises overfitted models based on their training accuracy.

A similar idea is also proposed in [26]. The paper defines two approaches called RegAgg and SimAgg that modify the weighting function used in FedAvg. The collaborators are assigned a weighting based on their similarity to the average model parameters. Furthermore, the paper introduces a sliding window for selecting the clients participating in each global round. It is demonstrated that this approach can stabilize the training process and improve the results in the Non-IID scenario.

The authors of [22] present another framework that addresses model parameters divergence. They propose to add momentum to the weight update on the server side to improve the convergence. To demonstrate the impact of Non-IID data distribution between clients and to evaluate the novel method, the authors use synthesised datasets.

#### Modification of the Training Methodology

The literature proposes several training methodologies to improve the results of FL in the Non-IID scenario. Their objective is to modify the training process in order to account for the statistical heterogeneity across the clients. One of such approaches, called FedProx, was proposed in [33]. The framework imposes a penalty on the local Stochastic Gradient Descent (SGD) if it deviates from the global model. To achieve it, the paper proposes to modify the loss function on the client side by including a *proximal term* to it. The proximal term is calculated as a square root of a sum of deviations of the local model parameters from the global model. This number is then multiplied by a hyperparameter $\mu$, which defines the strength of the regularization. The researches showed that this alternation improves the stability of the training, thereby improving its results on synthetic datasets with statistical heterogeneity across the clients. Additionally, the paper provides guidance on how to select the optimal $\mu$.

Model ensembling techniques were also often used in the literature to address statistical heterogeneity across the clients. In [19], the authors propose using an ensemble to reduce the communication cost between server and clients in a density estimation tasks. The paper assumes that the server has a set of pre-trained models from each client. Then, the weights to be assigned to each model in the ensemble are learned in a FL manner to improve the performance of the global model. However, the paper focuses on the ensemble weighting methodology and does not provide guidance on training the initial pre-trained models.

Another interesting approach to achieving better results for client-specific data is using personalization layers. It was introduced in [1, 34]. The proposed method divides the layers of the model into two categories: shared base layers and personalized layers, with each client having their own set of personalized layers. The shared layers are trained using a classical FL approach, while the personalized layers are fine-tuned on the client side to match its local data distribution. In the methodology defined in [34], the shared layers are the shallow layers of a DL model which focus on extracting high level features from the data. The personalized layers are deep layers that use those features to make the inference. The results reported in the papers were promising; however, the focus of these methodologies is to improve the performance of each client, rather than creating a single global model.

## 2.5 Federated Learning in Medical Imaging

FL is a promising method that could help overcome the scarcity of annotated datasets in the medical domain. It could provide a way to train better and more generalizable

models, as the data from multiple centres could be used. Also, it could help overcome the privacy issues connected to sharing the data [12]. There are several examples of the use of FL in cancer research in the literature. Most of them simulate the decentralized setting by artificially partitioning the dataset; however, there are several projects using real-world multicenter data.

### 2.5.1 Performance of Federated Learning in Cancer Imaging

Several papers compare the performance of FL approaches to the performance of collaborative data sharing. They implement multiple FL frameworks that deliver satisfactory results for various tasks in Medical Imaging. One example of such frameworks was implemented in [52]. There, the performances of three different methodologies to train a DL model without sharing private the data between institutions were compared. Even though the basic FedAvg was used, FL achieved almost as good results as CDS and yielded the best results out of those 3 methods. Moreover, it was shown that the model trained using FL performs significantly better than the models trained only using the private data of each institution. The paper utilized the BraTS dataset [41]. It contains multi-modal brain tumor scans together with their segmentations that were gathered from multiple institutions. This dataset is widely used as a benchmark for medical imaging segmentation in oncology [12].

[7] utilized the same dataset to evaluate a FL framework. The framework utilizes the concept of GANs to train a global model. The method proposes to train a generator of synthetic images on the server side using discriminators on all the clients. The discriminators residing at every client learn to differentiate between its local data and the synthetic images generated by the server, while simultaneously training the server's generator. Once the generator has been trained, it is then utilized to generate images on which a conventional U-Net is trained.

In [51], the authors present a comparison between a U-Net trained using the FL approach and U-Nets trained by each participating institution separately. The models are evaluated on a glioblastoma MRI scans segmentation task. The paper shows that the FL approach produces superior results, with the final model reaching 99% of the performance of a model trained by CDS.

Multiple large, multi-institutional projects have been developed to facilitate and streamline the collaboration on FL projects, both in academia and industry [48]. For example, Melloddy project [43] focuses on orchestrating end-to-end FL training for drug discovery. Another interesting initiative is German Cancer Consortium's Joint Imaging Platform [49], which aims to support the treatment and diagnosis studies in the field of cancer research using FL.

Another interesting large-scale project in the field of medical imaging is FetS [44] FL tool. In addition to providing support for FL, the tool offers many utilities, including generation of automatic annotation or built-in augmentation methods. The tool is currently tuned specifically to the tasks on neuro-radiological mpMRI scans; however,

it is intended to be extended to other applications. Additionally, the paper points out several challenges associated with conducting a FL training across multiple institutions, such as different image acquisition protocols across institution or systems heterogeneity. The researchers have also organised Federated Tumor Segmentation Challenge[3]. It is a competition focused on advancing the FL methods for medical imaging and dealing with the statistical heterogeneity of the medical image data across different institution. More details on copying with the statistical heterogeneity across institutions are presented in the next section.

### 2.5.2 Federated Learning Frameworks Reducing the Impact of Non-IID Data

The challenges with the statistical heterogeneity across the clients outlined in the Section 2.4.1 are also relevant to medical imaging tasks. In [66] and [52], it was shown that DL is prone to overfitting on differences and biases between different institutions. The models often tend to use the subtle differences between scanners, different patient populations and other confounding factors, rather than the pathological information available in a scan [44, 66, 16]. These biases bias often lead to poor performance of the DL models, especially when applied on the data from an institution that did not participate in the training. Therefore, much of the research in the medical imaging field, focuses on the problems posed by Non-IID data across the clients. [66, 12, 44]

[12] summarizes several methods that aim to overcome these challenges. Many focus on improving the weight aggregation methodology to account for the statistical heterogeneity across the network. For example, in [60], the researchers modify the weight aggregation of FedAvg to one in which the weights of each client are set dynamically. The weights are set depending on the data distribution across the institutions. The work has been evaluated on two multi-institutional medical imaging tasks.

There are also papers that focus on improving the training methodology already on the client side. In [25], the authors show that the order in which the samples are presented to the optimizer affects the final model. They propose a memory-aware algorithm, which optimises this ordering and reduces the likelihood that the final global model will forget part of the samples presented to it during training. The method was evaluated on a multi-site breast cancer classification task.

[18] applies FL to several MRI reconstruction tasks. It used an adversarial domain identifier to align latent features extracted by the clients without sharing the data between them. The method was evaluated on four different public MRI datasets and was shown to achieve a performance close to a model trained with CDS for all of them.

Another heterogeneity-aware framework called SplitAvg was proposed in [70]. It suggests splitting the network into two subnetworks. A copy of the first subnetwork, which contains the shallow layers of the model, resides on each client. The second subnetwork, which

[3] https://github.com/FETS-AI/Challenge

contains the deeper layers of the model, resides on the server and is shared between the clients. During the forward pass, the low-level features are extracted by each subnetwork on the clients' side, aggregated and passed to the server. The server then uses them to train the subnetwork responsible for classification. The paper evaluates the method using the BraTS dataset [41], which contains the data from multiple different institutions. It was shown that although the performance of the basic FedAvg plummets in the Non-IID scenario, the performance of the SplitAvg framework is only slightly lower than in the IID scenario.

The papers often compare their novel methods with other FL frameworks and the centrally trained models to evaluate their performance. In particular, the basic FedAvg and the more sophisticated FedProx[33] are often used as the baseline FL frameworks.

### 2.5.3 Ensemble Federated Learning in Cancer Imaging

The concept of ensemble models, which is widely used in ML, can also be applied to FL. There are some studies that explore this idea. In [16], FL is used to create a model to classify COVID from X-ray scans. The researchers propose to use a family of existing FL frameworks to train one model using each framework on the same dataset. These models can then be used as a majority voting ensemble. The authors claim that this approach can mitigate the biases introduced by each FL framework and improve the performance in the Non-IID scenarios. However, it requires performing the training multiple times, which can be resource intensive.

[31] proposes to use Shapley Values to rank the contributions of different institutions to the final model. The first step, is to train a global model using the classical FedAvg algorithm. Then, each site trains a logistic regression classifier using the feature vector created by applying the last layer of the global model to its data. Those local classifiers can then be used to perform a majority voting classification and calculate the Shapley values. In the experiments, the method achieved a similar performance to the FedAvg approach.

## 2.6 Summary

In recent years, DL has proven to be a powerful tool in image recognition, including medical imaging. For example, it has been used to classify the WHO grading of meningiomas. However, the task is challenging, mainly due to the scarcity of annotated training data. To increase the amount of training data, FL is emerging as a promising alternative to the standard CDS. The chapter outlines several existing FL frameworks, two of which (FedAvg and FedProx) were evaluated and compared experimentally in the course of the thesis.

In addition, the chapter describes the problems arising from the heterogeneity of the data across the clients participating in FL. These challenges were investigated in the

thesis, with a focus on their impact on the accuracy of the models and ways to minimise their negative effects.

# CHAPTER 3

# Methodology

This chapter describes the details of the methods used in the experiments. It builds on the background and state of the art presented in the previous chapter. The chapter starts with the definition of the prediction targets and the classification tasks. Then, the data preprocessing methods are presented. Finally, the chapter details the methods used to overcome the challenges presented in the problem statement and to enable meningioma grading prediction in the centralized and FL environments. The main goal of the experiments is to compare the performance of the centralized training with FL frameworks in different data heterogeneity settings. The code used to run the experiments is available at [1].

## 3.1 Input Data and Prediction Targets

The dataset consists of T1 weighted brain MRI sequences of meningiomas. Each scan is accompanied by a WHO grading of the tumour obtained from histopathological examination, which is considered to be the most reliable measure for grade assessment [39]. The grading is denoted by an integer between 1 (benign tumour) and 3 (the most aggressive form). Additionally, the tumour segmentation annotated by one clinician is available for each scan. The segmentations include the tumour together with the edema (if present). An example of a middle slice from one of the 3D volumes is shown in Figure 3.1.

Two classification tasks were defined: three-class classification and two-class classification. In both tasks, the WHO grading of the meningiomas is the prediction target, the difference is how it is mapped to the sample labels.

**Three-class classification scenario:** In this scenario, all three WHO grades had to be discriminated.

[1] https://github.com/Fidelisus/meningioma_dl

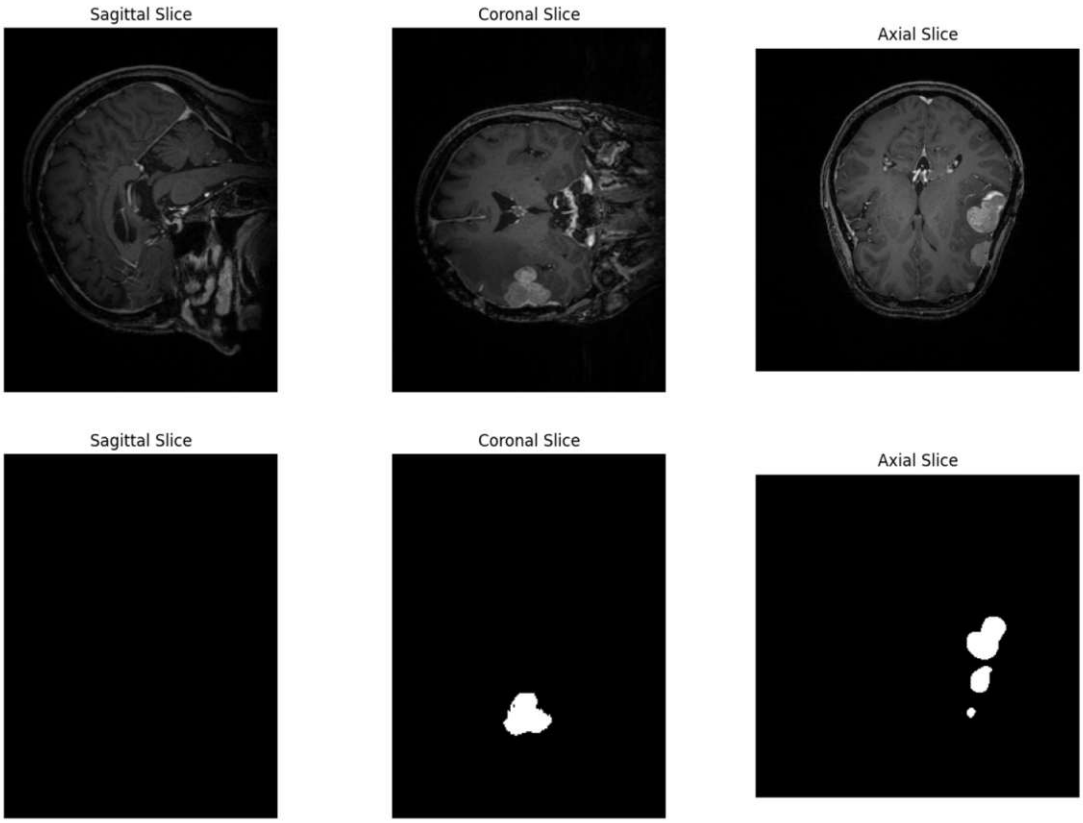


Figure 3.1: The middle slice for each plane of a randomly selected scan. At the top is the MRI scan and at the bottom is the segmentation of the lesion as marked by a clinician.

**Two-class classification scenario:** As the three-class classification task is often challenging [23], a simpler scenario with only two classes as the prediction target was also defined. After consultation with a clinician from the AKH, samples graded 2 and 3 were grouped as *label 2*, while scans graded 1 formed *label 1*. This grouping was chosen because grade 2 and 3 meningiomas often require medical intervention, whereas grade 1 meningiomas usually only require observation to monitor disease progression.

Let's denote those components as:

- $\mathbf{I}_i \in \mathbf{R}$: Three-dimension MRI sequence of meningioma.
- $\mathbf{S}_i \in \{0, 1\}$: Tumour segmentation of sample $i$.
- $y_i^{3C} \in \{1, 2, 3\}$: Label of sample $i$ in the three-class classification scenario.
- $y_i^{2C} \in \{1, 2\}$: Label of sample $i$ in the two-class classification scenario.
- $f^{3C}(\mathbf{I}, \mathbf{S})$: Three-class classifier.

- $f^{2C}(\mathbf{I}, \mathbf{S})$: Two-class classifier.

## 3.2 Image Preprocessing

Medical images are often noisy and the intensity of the pixels varies greatly depending on the type of scanner. Similarly, the image resolution can vary between the scanners. In addition, images can sometimes be corrupted. Therefore, the scans had to be preprocessed in order to unify their characteristics and prepare them for the modelling.

First, a simple visual check of the images was carried out. This was done by visualising the centre slice of each of the 3 planes in the image and assessing whether there were any significant data errors, such as corrupted scans. Then, a preprocessing pipeline was applied to each scan. The image preprocessing can be described by the following formula:

$$g(\mathbf{I}_i, \mathbf{S}_i) = \mathbf{I}_i^p$$

where $\mathbf{I}_i^p$ is the preprocessed image and $g$ denotes the preprocessing function. The parameters of $g$ were chosen in a series of preliminary experiments. The final preprocessing pipeline consisted of the following steps executed sequentially:

1. **Reorientation:** This operation unifies the image orientation across all scans.
2. **Resampling:** This ensures that each voxel in the image has the same size in each scan. In this case, the voxel size was set to $1mm * 1mm * 1mm$. Without this step, the volume represented by one voxel could be different in different samples, making modelling more difficult.
3. **Foreground cropping using tumour segmentation:** In this step, the image was cropped using the tumour segmentation. First, the tumour mask was increased by 20% in order to include the tissue around the tumour, as it may also contain valuable information. Then the intensity of the whole area outside the mask was set to 0. This step is important to remove the unimportant parts of the image to make it easier for the model make a correct classification.
4. **Image padding:** This step pads the cropped images so that their dimensions are the same as the dimensions of the largest segmentation mask. The scans were padded with zeros. This is necessary because the ResNet model requires all input samples to have the same shape.
5. **Intensity normalisation:** The final step in the pipeline, which normalises the intensities of the images to reduce the differences between the scanners.

The preprocessing was done using an open source Monai library [5]. Examples of the preprocessed scans can be seen in Figure 3.2.

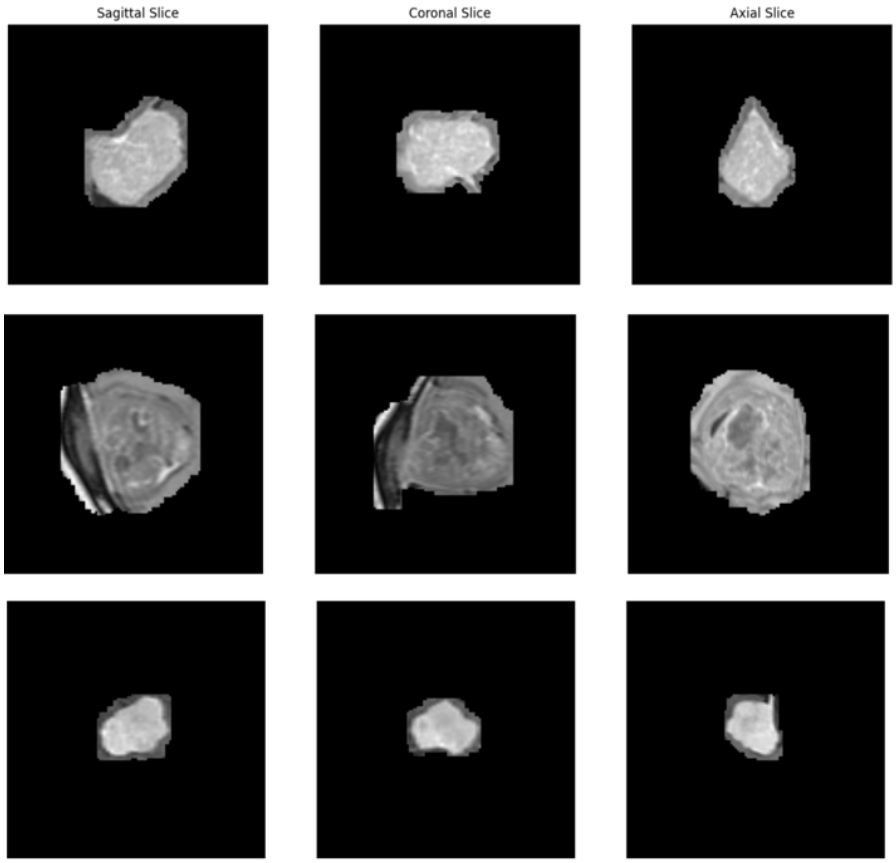


Figure 3.2: The figure shows three randomly selected examples of MRI scans of meningioma after preprocessing. The central slices of each plane are shown for each scan.

## 3.3 Deep Learning Model

The preprocessed image $\mathbf{I}_i^p$ serves as an input to model that predicts the label $y_i$. The prediction procedure can be expressed by the following formulas:

$$f^{3C}(\mathbf{I}_i^p) = \hat{y_i^{3C}} \text{for the three-class scenario}$$
$$f^{2C}(\mathbf{I}_i^p) = \hat{y_i^{2C}} \text{for the two-class scenario}$$

The thesis uses a classical 3D-ResNet model implemented in the Pytorch library[2]. ResNet is a well-established and flexible CNN model that has proven its high accuracy in many tasks, including meningioma classification [39]. MedicalNet from [10] was used as the

[2]https://pytorch.org

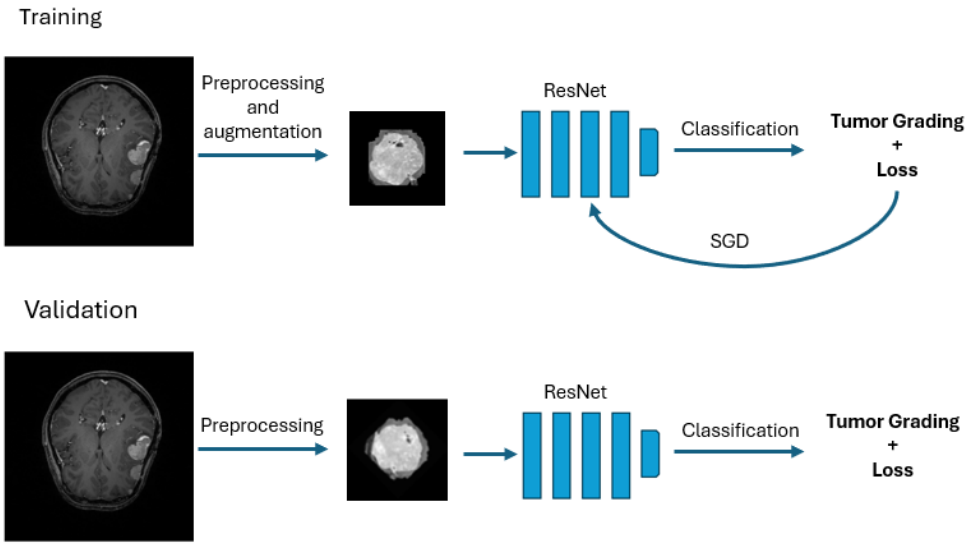


Figure 3.3: A simplified diagram of the training process. At the top is the pipeline used for training. At the bottom is the validation pipeline.

pre-trained model. It was chosen because the paper reported promising results on several medical imaging tasks.

The models $f$ were trained using a standard SGD procedure. A simplified diagram of the training process is shown in Figure 3.3.

### 3.3.1 ResNet Model Hyperparameters

After the initial experiments, the main parameters of the model and the training process were defined. The ResNet10 model, which is one of the smallest in the family of ResNet models, was selected. It was chosen because a smaller model meant that the training time was reduced and therefore more experiments could be carried out. In addition, as high classification accuracy was not the main goal of the thesis, the additional predictive power of the larger models was not needed.

A fully connected layer responsible for classification was added as the last layer of the ResNet10 model. It had 2 output neurons for $f^{2C}$ and 3 output neurons for $f^{3C}$. The network used the Adam optimiser and the batch size was set to 4. Loss function $loss_m$ was set to cross-entropy weighted by the number of samples in each class. It could not be set higher than 4 due to GPU memory limitations. The learning rate $lr$ was decayed over epochs using an exponential learning rate scheduler. It sets the learning rate at each epoch to $lr_t = \gamma * lr_{t-1}$, where $\gamma$ was set to 0.99.

As transfer learning was used in the experiments, the number of shallowest layers to be frozen during fine-tuning had to be defined. As shown in Figure 3.4, the ResNet model consists of four main layers as described in [21]. Experiments were carried out for zero,

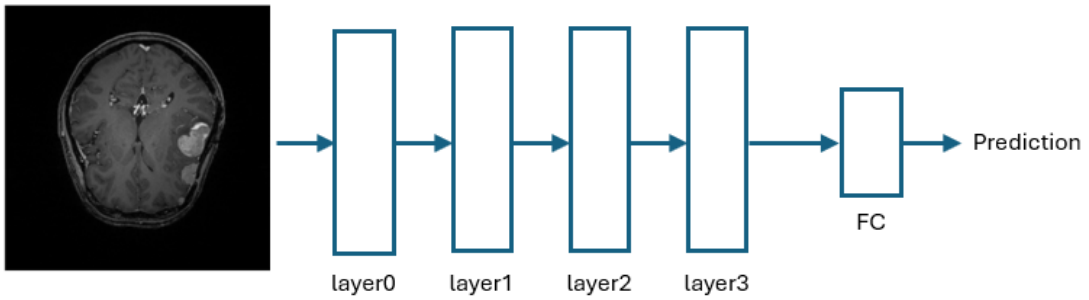


Figure 3.4: The diagram illustrates the main building blocks of the ResNet10 classifier. It consists of 4 main layers and ends with a fully connected layer that is responsible for the classification.

one, two and three frozen layers. The layers were frozen starting with the shallowest layers, so the layers responsible for the extraction of the low-level features. For example, freezing two layers meant that *layer0* and *layer1* from Figure 3.4 were frozen during training. This parameter will be referred to as $l_{frozen}$ in the following chapters.

## 3.4 State-of-the-art Federated Learning Frameworks

Two state-of-the-art FL frameworks were implemented in the thesis: FedAvg and FedProx. They were implemented as described in the sections 2.3.2 and 2.3.3 and used to train the $f^{2C}$ model.

## 3.5 Federated Localized Ensemble Framework

The main idea behind the framework was inspired by the concept of model ensembling. It is widely used in DL and sometimes adapted for FL. However, to the best of my knowledge, the proposed approach has not yet been described in the literature. There are some approaches similar to this idea, such as SplitAvg [70] and Personalisation Layers [1], but the details of the method differ.

When it comes to model ensembling, there are two main questions to answer. The first is how to train the models that form the ensemble, and the second is how their predictions are aggregated.

### 3.5.1 Training of the Site-Specific Models

Training consists of the following two steps:

**Pre-training of the base model.** First, any existing FL framework, such as FedAvg, is used to train the base model $f_{base}$, as described in the section 2.3.2. It has a similar goal to the idea behind transfer learning.

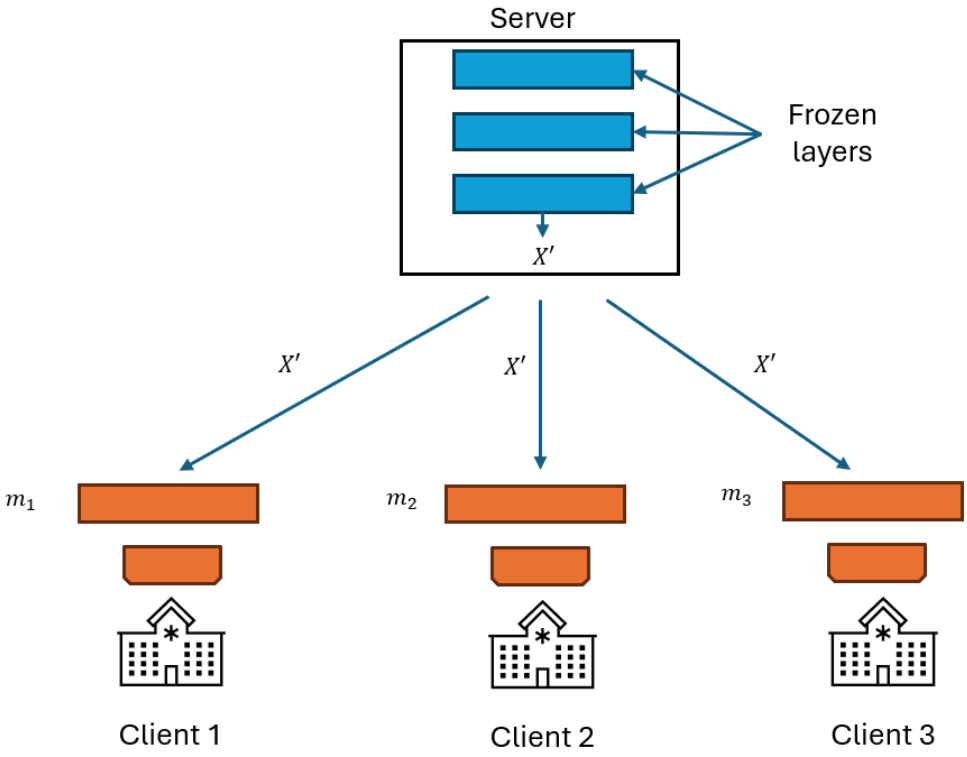


Figure 3.5: A diagram illustrating the process of fine-tuning of the *site-specific* models in Federated Localized Ensemble framework.

**Fine-tuning of the site-specific models.** In this step, $f_{base}$ is used as the initial model to train *site-specific* models $f_c$ for each client $c$. Each client fine-tunes its $f_c$ for $n_{tuning}$ epochs. Then the model is returned to the server. A diagram of this process is shown in Figure 3.5. In addition, $l_{frozen}^{tuning}$ layers of $f_{base}$ are frozen during the tuning process, similar to transfer learning. This is done to avoid overfitting to the client's local data.

The final ensemble model consists of all $f_c$ trained during fine-tuning, one for each client involved in the training.

### 3.5.2 Weighting of the Models in the Ensemble

The next step is to aggregate the predictions made by all the *site-specific* models. Majority voting is often used for this purpose. However, as this approach is often too simple, another idea has been proposed. It takes into account the performance of the individual *site-specific* models.

Two types of weights are derived. The first set of weights, called *global weights*, is

optimised to work well on any client's data. The second is called *local weights* and is optimised for the local data of the client.

### Global Weights

The first step is to calculate the performance of each $f_c$ on the local data of each client. To calculate this, all models $f_c$ are sent to all clients, which evaluate them on the local data. They then send a score for each model $f_c$ back to the server. Let's call this score $score_c^{f_c}$.

Then the weight $\omega_{f_c}$ of each model is calculated as

$$\omega_{f_c} = \frac{1}{C} \sum_{c=1}^{C} score_c^{f_c}$$

where $C$ is the total number of clients. Therefore, the better a model $f_c$ performs on each client, the higher its weight will be in the ensemble model. The final prediction of the ensemble model can then be expressed by the formula

$$y = \arg\max_l (\omega_{f_c} * logit_l^{f_c})$$

where $logit_l^{f_c}$ is the logit function returned by the output layer for label $l$.

### Local Weights: the Weights Optimised for the Client's Local Data

The local weights are calculated in a similar way to the *global weights*. The only difference is that the model $f_c$ is only evaluated on the client for which the weights are optimised. Therefore, the models that perform better on the client's local data will be weighted more in the ensemble.

CHAPTER 4

# Experimental Setup and Evaluation

## 4.1 Dataset

The dataset used in the project consists of 186 brain MRIs of patients diagnosed with meningioma. All the data comes from the AKH hospital in Vienna. The classes are relatively balanced, with 73 WHO grade 1, 71 WHO grade 2 and 42 WHO grade 3 samples, as shown in Table 4.1. Scans come from different patients, except for two patients for which there are two examinations. It is noteworthy that the class distribution in the dataset differs significantly from the real population distribution. In the real world, grade 1 meningiomas make up about 90% of meningioma tumours [46]. Therefore, the class distribution is much more skewed in the underlying population.

The data was split into training and test sets. As the classes are slightly unbalanced, stratified random sampling was used. From all the samples, 41 samples were randomly selected to become the test set. This number would preferably be higher in order to assess the performance of the classifier more reliably. However, given the rather small size of the dataset, this number was the most that could be afforded. The test set images are only used at the end of the experiment to assess the performance of the final models. A separate validation set created from the training set is used for model evaluation during training and hyperparameter tuning.

## 4.2 Image Augmentation

Several augmentations were applied to the preprocessed scans to improve the generalisability of the model. The following augmentation operators were defined:

| WHO Grade | 1 | 2 | 3 | **Total** |
|---|---|---|---|---|
| Number of samples | 73 | 71 | 42 | **186** |

Table 4.1: Table shows the number of MRI scans with each WHO grade.

**Random intensity shift.** It shifts the intensity of the image by $\alpha * std(I)$ where $std(I)$ is the standard deviation of an image and $\alpha$ is a randomly sampled factor.

**Adding random Gaussian noise.** It adds gaussian noise to the image, with mean $m = 0$ and standard deviation $std$ with the value depending on the pipeline.

**Random affine transformations: rotation, zoom and translation.** The parameters of these transformations were randomly sampled from the following domains: $z$ defining possible zoom values, $\beta$ containing the possible rotations and $t$ determining the translation.

**Adding random bias field.** The bias field is a linear combination of varying basis functions and is often present in MRI scans. It is described by two parameters, as described in the open source Monai library [5]: degree of the polynomials, which was set to 3 and the coefficients randomly sampled from a domain $c$.

**Random image contrast adjustment with gamma transform.** It randomly changes image intensity with gamma transform, which strength is regulated by the $\gamma$ parameter.

The augmentation pipeline consisted of several operators. It was employed to each sample during training; however, not every step in the pipeline was always applied. The probability in which an augmentation step was applied to a sample was specified by the *augmentation steps probabilities* hyperparameter, termed $aug_{prob}$.

Three augmentation pipelines were tested, each implementing different augmentation method $h_{aug}$:

**Basic pipeline** containing all the operators excluding random bias field and random image contrast adjustment.

**Strong pipeline** using the same operators as the *Basic pipeline*, but the parameters of each step have been set to ones that apply stronger augmentation to the scans.

**Extended pipeline** applying all the operators.

The parameters of the augmentation operators for different pipelines can be seen in Table 4.2.

| | *Basic* | *Strong* | *Extended* |
|---|---|---|---|
| Intensity shift | $\alpha \in [-0.05, 0.05]$ | $\alpha \in [-0.2, 0.2]$ | $\alpha \in [-0.2, 0.2]$ |
| Gaussian noise | $std = 0.1$ | $std = 0.15$ | $std = 0.15$ |
| Zoom | $z \in [0.9, 1.1]$ | $z \in [0.8, 1.2]$ | $z \in [0.8, 1.2]$ |
| Rotation | $\beta \in [-\pi/4, \pi/4]$ | $\beta \in [-\pi/4, \pi/4]$ | $\beta \in [-\pi/4, \pi/4]$ |
| Translation | $t \in [-5, 5]$ | $t \in [-20, 20]$ | $t \in [-20, 20]$ |
| Bias field | - | - | $\gamma \in [0.8, 1.5]$ |
| Contrast Adjustment | - | - | $c \in [0.0, 0.1]$ |

Table 4.2: The parameters of augmentation operators for different pipelines.

Augmentations were only applied to the scans during model training. For model validation, only image preprocessing was used. This process can be described by the following formulas:

$$f(h_{aug}(\mathbf{I}_i^p)) \rightarrow \hat{y}_i \text{ during training}$$
$$f(\mathbf{I}_i^p) \rightarrow \hat{y}_i \text{ during validation}$$

Figure 4.1 shows the results produced by each pipeline on the same randomly selected scan.

## 4.3 Federated Learning Scenario Simulation

All the samples in the dataset come from one medical institution. Thus, the FL environment had to be simulated. Three different scenarios were defined, one assuming the IID distribution across clients and two simulating the Non-IID distribution. All the scenarios were simulated for 3 FL clients. This number of clients is often higher in the real world, but this lower number was chosen due to the small size of the dataset. All the FL experiments were carried out for the two-class classification task, as this is an easier task than the three-class classification. In other worlds, the target variable for all the experiments will be $y^{2C}$, as described in the section 3.1.

### 4.3.1 Simulation of the IID Data

The IID data was simulated by distributing the training samples across the clients using stratified random sampling. Thus, each client had a similar class distribution. It often leads to an IID distribution of attributes; however, it is not guaranteed [72]. This strategy was often used in the literature as described in section 2.4.1.

### 4.3.2 Simulation of Label Frequency Difference

The first Non-IID scenario was simulated by distributing the samples in way that yielded different class distributions at each client. This idea has often been used in the literature

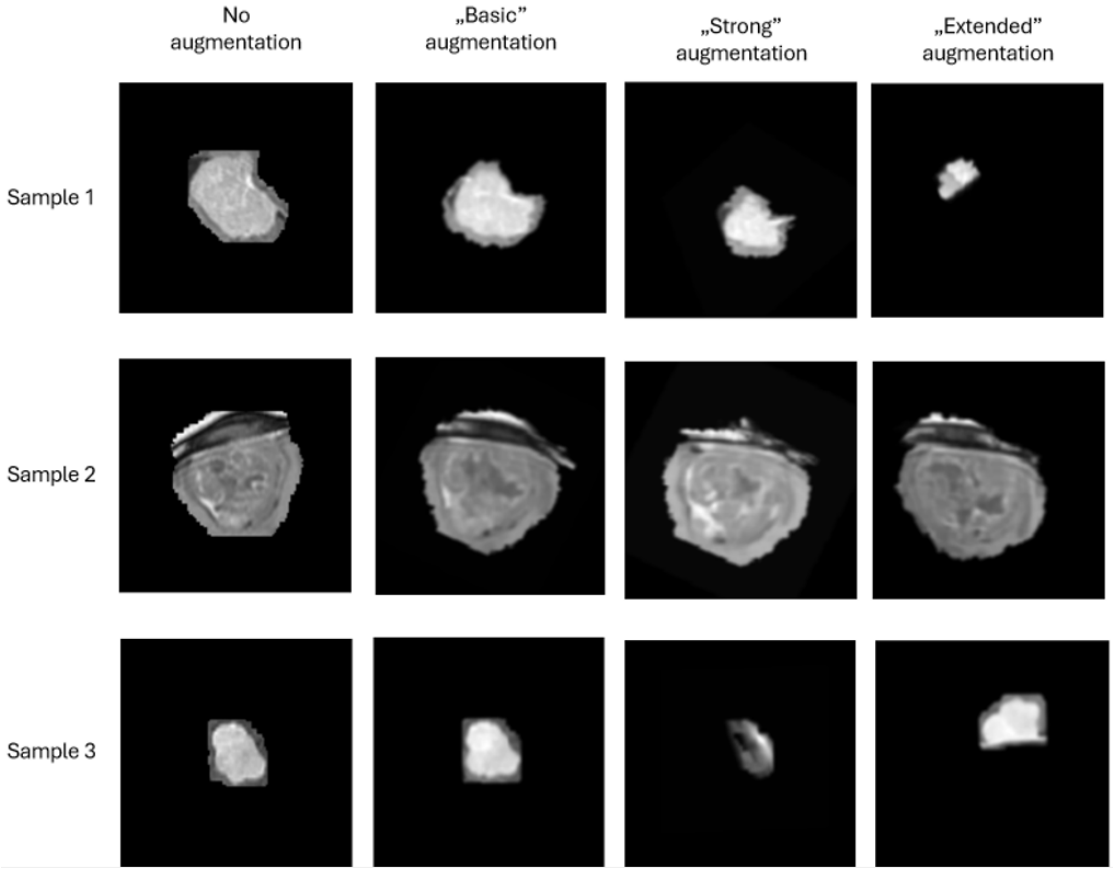


Figure 4.1: The figure shows the result of different augmentation pipelines for three randomly chosen scans. The central slice of the Saggital plane is shown for each scan.

for simulating Non-IID scenarios, as described in section 2.4.1. The idea of quantifying the degree of label difference was inspired by [70]. A two-sample KS test was used to quantify the degree of data heterogeneity. The test statistic, called KS test statistic, was calculated between each pair of institutions [47] and averaged. The higher the statistic, the greater the difference between the client distributions. It was decided to use a split with a KS test statistic equal to 0.4. Figure 4.2 shows an example bar chart of the class distributions for each client.

### 4.3.3 Simulation of Imaging Difference

The second Non-IID scenario simulated a difference in imaging technology between clients. This is common in practice, as different medical facilities often use different types of scanners. This scenario is more difficult to simulate as the imaging differences are hard to quantify systematically. Therefore, it was decided to use a simple approach that focuses on only one image characteristic. One of the image properties that often depends on the type of scanner is the voxel intensities, and in particular their distribution [53].

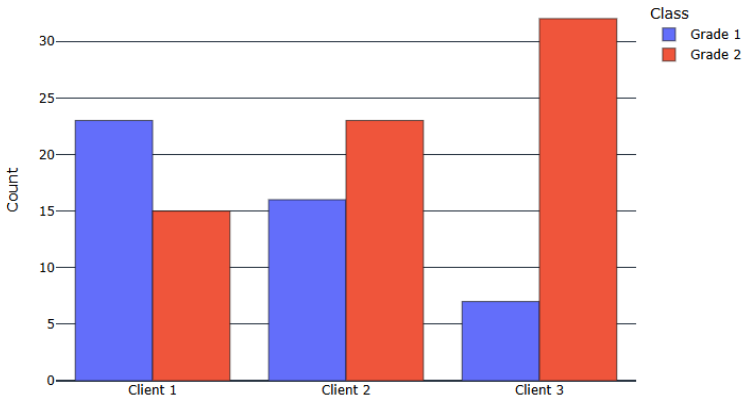


Figure 4.2: Figure display an exemplary bar chart of class distributions at each client.

To obtain a different distribution of voxel intensities, the image intensity histogram was shifted differently between clients. The transformation was carried out using a *RandHistogramShift* method implemented in the well-established Monai library[1]. The method applies a random non-linear transformation to the intensity histogram. The strength of the distribution shift is defined by the *number of control points* parameter, which defines the non-linear intensity mapping. A smaller value of this parameter results in larger intensity shifts.

The following transformations were applied to the client as the last step of the preprocessing pipeline described in the section 3.2:

- **Client 1:** No histogram shifts.
- **Client 2:** Medium histogram shift with *number of control points* equal to 10.
- **Client 3:** Strong histogram shift with *number of control points* equal to 5.

Figure 4.3 shows a visual comparison of example preprocessed samples for these clients.

[1] https://monai.io/

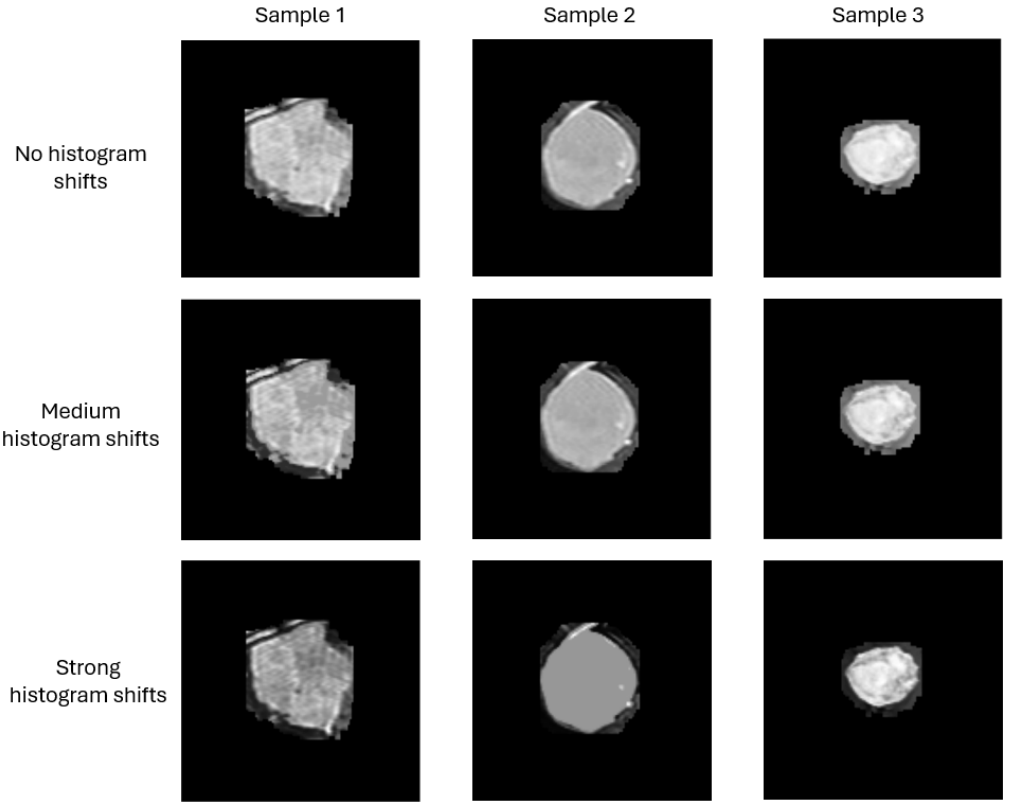


Figure 4.3: The figure shows a visual comparison of the preprocessed samples for different clients in the imaging difference simulation. The central saggital slice is shown for each scan.

## 4.4 Evaluation of the Resulting Classifiers

All methods were evaluated on the same test set defined in 4.1. This test set was only used as a final form of evaluation, it was not used for hyperparameter tuning. It consisted of only 41 samples, which is a relatively small number. In such cases, the inter-model variability might be high. In this context, inter-model variability is the difference between models trained on the same data. It occurs when the training dataset is small and small differences in the input data significantly alter the convergence, resulting in high variability in model performances. This phenomenon has been studied extensively in [24].

In order to understand the origin of this phenomenon, it is first necessary to define a methodology for selecting the best model at the end of training. This is not trivial, as the last model may not be the best due to the overfitting phenomenon. To make the selection, the F1 score of the model on the validation set was calculated for each epoch. The model with the highest F1 score was then returned as the best model. Therefore, the selection of the best models depends strongly not only on the training set but also on the validation set.

To provide a more robust estimate of model performance, multiple models were trained and evaluated for each method. To calculate the performance of a method in each experiment, 10 models were trained. They were trained using 5-fold Cross-validation (CV), with two models with different seeds trained for each fold. The model were then evaluated on the **test set** and the average score of these 10 models was calculated. The following three metrics were used: $\overline{F1-score}$, $\overline{Precision}$ and $\overline{Recall}$. A diagram of this process is shown in Figure 4.4. The metrics were computed differently for the two and three class classification tasks:

**Three-class classification:**

$$m = \frac{\sum_l^3 v_l n_l}{\sum_l^3 n_l}$$

where $v_l$ is a metric derived separately for the label $l$ and $n_l$ is the number of samples with label $l$.

**Two-class classification:**

$$m = \frac{v_1 + v_2}{2}$$

Thus, in this case, no weighting by the number of samples for a label was performed.

In addition, a baseline model was introduced to help understand the meaning of the metrics. It was set to a classifier predicting the majority class.

## 4.5 Experimental Setup

### 4.5.1 Hyperparameter Tuning

The experiments began with hyperparameter tuning. First the hyperparameters were tuned for the $m_{3C}$ model and then for the $m_{2C}$ model, both trained centrally. The second step was to optimise the hyperparameters of each FL framework. All models were trained over 200 epochs and evaluated using the same validation set.

#### ResNet Model

The search space of the model's hyperparameters consisted of the following four hyperparameters:

- $lr \in \{10^{-4}, 10^{-3}, 10^{-2}, 2 * 10^{-2}, 0.1, 0.2\}$
- $aug_{prob} \in \{0.5, 0.8, 1.0\}$
- $h_{aug} \in \{\text{Basic}, \text{Strong}, \text{Extended}\}$
- $l_{frozen} \in \{0, 1, 2, 3\}$

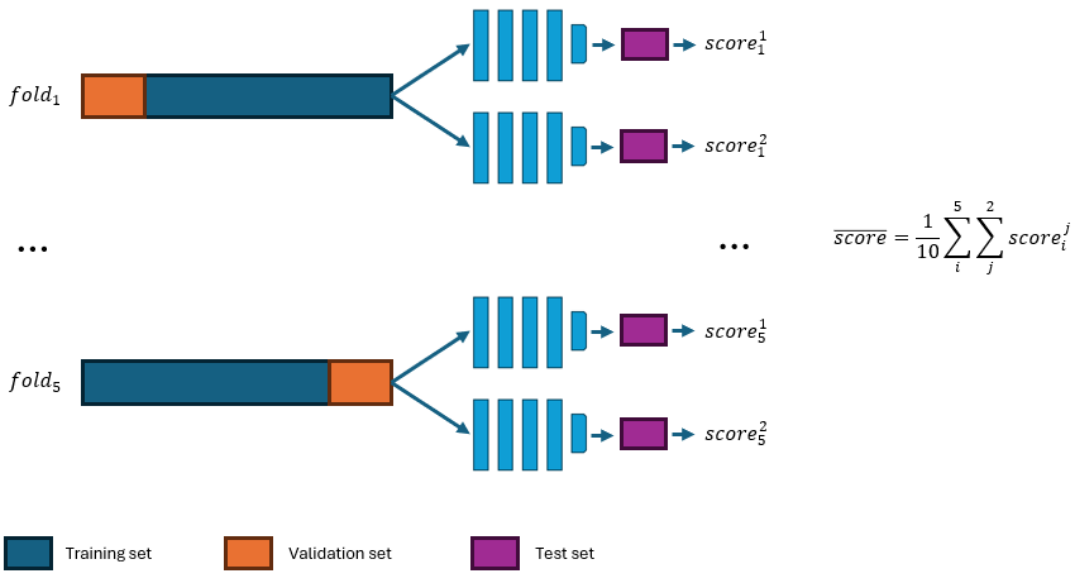


Figure 4.4: The diagram illustrates the evaluation process. In the first step, 10 models are fitted using CV and each of these models is evaluated against the same test set. The metrics are then calculated as the average of the scores obtained by the models.

Due to limited computational resources, a grid search for the best hyperparameters could not be performed. Therefore, different hyperparameters to try were iteratively selected manually based on the results for other hyperparameters. The process is described in detail in the section 5.7. The hyperparameters for which the model achieved the highest score on the validation set were used in the remaining experiments. The hyperparameters for the $m_{3C}$ and $m_{2C}$ models were tuned separately, so both could have different optimal hyperparameters.

### Federated Learning Frameworks

The search for optimal hyperparameters was carried out in the scenario with IID data between clients, for both frameworks separately.

**Federated Averaging:** It started with the identification of the optimal $n_{\text{local epochs}}$ and proportion of clients per round. These are the main parameters identified as important to optimise for FedAvg in [20]. The search space for the number of local epochs consisted of four possible settings: 1, 2, 5 and 20. The proportion of clients per round hyperparameter was set to either 1.0 or 2/3. Therefore, either all or 2 clients were sampled in each global round. In all experiments, the total number of epochs was 200.

**FedProx:** Additionally, for the FedProx framework, the hyperparameter $\mu$ had to be set. The guidelines proposed in [33], which introduced FedProx, were followed. Six different $\mu$ values were evaluated: $10^{-1}$, $10^{-2}$, $10^{-3}$, $3 * 10^{-3}$, $10^{-4}$ and $10^{-5}$.

#### Federated Localized Ensemble Framework

The two main hyperparameters configured for the Federated Ensemble method were the number of frozen layers during fine-tuning $l_{frozen}^{tuning}$ and the number of fine-tuning epochs $n_{tuning}$. The search space of the $l_{frozen}^{tuning}$ hyperparameter was 0, 1, 2 and 3 deepest layers. In addition, one experiment was run where 2 shallowest layers were frozen. All experiments were run with $n_{rounds}$ set to 100, $n_{\text{local epochs}}$ set to 1 and $n_{tuning}$ set to 50 or 100.

Additionally, as no client-specific test data was available, only the *global weighting* method was evaluated.

### 4.5.2 Comparison of Centralized Training and Federated Learning in the IID Scenario

The goal of this experiment was the comparison between the models trained centrally and with different FL frameworks. All the models were trained and evaluated on the two-class classification tasks. Evaluation was carried out on the same test data for each method, using the evaluation process described in the section 4.4.

The experiments used the optimal hyperparameters of all the methods obtained during the procedure described in the section 4.5.1 Performance was reported for centralised training and three FL frameworks: FedAvg, FedProx and Federated Localized Ensemble approach. The FL frameworks were evaluated in the IID scenario.

### 4.5.3 Comparison of Federated Learning Frameworks in the Non-IID Scenario

The aim of this experiment was to observe the impact of Non-IID data across clients on the performance of FL. Although in the IID scenario the performance of models trained with FL can match that of centrally trained models, the Non-IID may be more challenging [71].

Two Non-IID scenarios were simulated as described in section 4.3. To analyse the differences between the methods, the three FL described in the chapter 3 were evaluated and compared in each Non-IID scenario. The models were evaluated on the two-class classification task.

# CHAPTER 5

# Results

This chapter presents the results of the experiments. It starts with the comparison of the models trained centrally and using the FL frameworks. Then, the comparison of the prediction accuracy of the FL frameworks in different Non-IID scenarios is presented. It is followed by the results achieved by the classifier trained centrally. Lasty, additional insights into the methods and hyperparameter tuning are presented. All models were evaluated on the same test set defined in section 4.1. A detailed evaluation methodology is presented in the section 4.4.

## 5.1 Comparison of Centralized Training and Federated Learning in the IID Scenario

The goal of this experiment is to compare the models trained centrally with those trained using different FL frameworks. All models were trained and evaluated on the two-class classification tasks.

Table 5.1 displays $\overline{F1}$, $\overline{Precision}$ and $\overline{Recall}$ achieved on the test set by the models trained with different methods. The metrics are reported together with their standard deviations. The F1-scores of the models trained during evaluation process are also displayed as violin plots in Figure 5.1. Table 5.2 presents the confusion matrices created by aggregating the predictions generated by all the models grouped by the training framework.

Given the similarity of the scores, two-sample t-tests were conducted to quantify the differences. Their aim was to assess whether the performance of the FL methods is comparable to that achieved through centralized training. The test showed that the performance of the models trained using FedProx and the novel method is significantly worse than for the models trained centrally. However, they still achieved much better F1-score than the baseline.

| | F1-score | | Recall | | Precision | |
|---|---|---|---|---|---|---|
| | AVG | SD | AVG | SD | AVG | SD |
| Two-class Baseline | .2400 | N/A | .5000 | N/A | .1578 | N/A |
| Centralized training | .5796 | .0380 | .5859 | .0396 | .5942 | .0410 |
| FedAvg | .5664 | .1099 | .5783 | .0967 | .5720 | .1096 |
| FedProx | .4571 | .0865 | .4849 | .0779 | .4634 | .1141 |
| Federated Localized Ensemble | .4201 | .0744 | .4955 | .0472 | .4125 | .1393 |

Table 5.1: The figure present $\overline{F1}$, $\overline{Precision}$ and $\overline{Recall}$ achieved on the test set by the models trained with different methods. The means are reported together with their standard deviations. Scores achieved by the baseline model predicting majority class are displayed alongside the predictions.

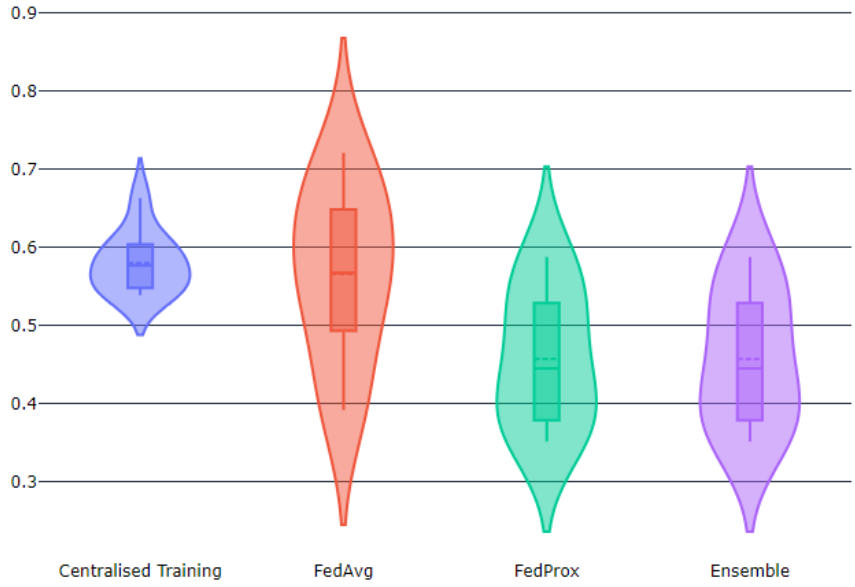


Figure 5.1: Violin plots of $\overline{F1}$ achieved on the test set by the 10 models trained during the evaluation process using different training methodologies. Blue plot shows the model trained centrally, red displays FedAvg, green FedProx and purple the Federated Localized Ensemble framework.

The test didn't show a significant difference between FedAvg and the model trained centrally, with a p-value of 0.723. Therefore, it is probable that FedAvg had a similar $\overline{F1}$ on the test set to the model trained centrally.

The discrepancy between recall and precision for each training method is small. It is therefore not straightforward to define whether the model is returning a greater number of False Positives (FP) or False Negatives (FN). It is confirmed by the structure of the

| | | Centralized training | | FedAvg | | FedProx | | Federated Ensemble | |
|---|---|---|---|---|---|---|---|---|---|
| Predicted | | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| True | 1 | 76 | 84 | 66 | 94 | 31 | 113 | 12 | 148 |
| | 2 | 76 | 174 | 64 | 186 | 72 | 194 | 21 | 229 |

Table 5.2: Confusion matrices created by aggregating the predictions on the test set of all the models throughout the evaluation process. Every column corresponds to the models trained by a different method.

confusion matrices in Table 5.2, which indicate that the frequency of each type of error is in a similar range. All the models are prone to a considerable number of errors, with the ratio of FP and FN frequently exceeding the number of True Positives (TP).

The standard deviation of the performance metrics of the centrally trained model was found to be lower than for all the FL methods. This behaviour is clearly visible in Figure 5.1, where all the violin plots of the FL methods are much wider than the one of centralized training. Therefore, although FedAvg can achieve a similar mean performance to the centralized training, the models created by this method are less stable. This behavior was also visible in the learning curves plotted during training, where validation losses of FL trainings were much noisier.

## 5.2 Comparison of Federated Learning Frameworks in the Non-IID Scenario

This section presents a comparative analysis of the results obtained using different FL frameworks when the data was Non-IID across the clients. As presented in Section 5.1, the performance of the models trained using FL can reach that of the models trained centrally. Nevertheless, in practice, the data is often heterogeneously distributed between the clients, which may result in a decline in the models' performance. In the experiments described in Section 5.1 the data was randomly split between the clients. Given that all the data came from the same institution and was labelled by the same clinician, it was assumed that the data is IID across the clients.

In this section, experiments on two scenarios of Non-IID data across the network were carried out. The first scenario simulated label frequency difference between the clients, while the second imitated a situation when imaging technology varies across the sites. The methodology employed for the simulation is described in detail in Section 4.3.

Table 5.1 presents the results achieved for all the three scenarios in which the FL frameworks were tested. $\overline{F1}$, $\overline{Precision}$ and $\overline{Recall}$ are reported. Figure 5.2 depicts same results in the form of grouped bar charts, where each bar chart illustrates the F1-score of the model together with its standard deviation.

| | | F1-score | | Recall | | Precision | |
|---|---|---|---|---|---|---|---|
| | | M | SD | M | SD | M | SD |
| Baseline Model | | .2400 | N/A | .5000 | N/A | .1578 | N/A |
| IID | FedAvg | .5664 | .1099 | .5783 | .0967 | .5720 | .1096 |
| | FedProx | .4571 | .0865 | .4849 | .0779 | .4634 | .1141 |
| | Federated Ensemble | .4201 | .0744 | .4955 | .0472 | .4125 | .1393 |
| Non-IID: Label frequency | FedAvg | .4637 | .0832 | .4848 | .0734 | .4717 | .1007 |
| | FedProx | .4961 | .0894 | .5404 | .0686 | .5095 | .1312 |
| | Federated Ensemble | .4147 | .0373 | .4961 | .0283 | .4847 | .1620 |
| Non-IID: Imaging difference | FedAvg | .5458 | .1109 | .5861 | .0940 | .6213 | .1416 |
| | FedProx | .5605 | .1039 | .5764 | .1054 | .5763 | .1088 |
| | Federated Ensemble | .3902 | .0259 | .4954 | .0153 | .3704 | .1116 |

Table 5.3: The table presents the results achieved for the two-class classification in all the three scenarios in which the FL frameworks were tested. $\overline{F1}$, $\overline{Precision}$ and $\overline{Recall}$ are reported. Scores achieved by a baseline model predicting majority class are displayed on top to facilitate comparison.

The label frequency difference scenario yielded the lowest scores across the models. In other the two scenarios (IID and imaging difference), the models performed visibly better, achieving similar average performances. However, it is important to note that in the imaging difference scenario, the metrics had higher standard deviations.

The FedAvg framework proved to be the best in the IID scenario, with the $\overline{F1}$ of 0.5664. In the Non-IID scenarios it performed worse; however, it still achieved relatively high scores in comparison to the other FL frameworks. The FedProx framework performed visibly worse that FedAvg in the IID data setting. It achieved the $\overline{F1}$ of 0.4571 that is 20% lower than for FedAvg. In the Non-IID scenarios, FedProx trained model achieved the highest F1-scores of all the methods, with the performance of FedAvg being only slightly worse.

Given the high standard deviations of the metrics, a t-test was performed to statistically compare the $\overline{F1}$ of the methods in different scenarios. The test was conducted pairwise for all FL frameworks in each scenario. The results are presented in Figure 5.3. Each matrix represents a single scenario. The colour red indicates that the $\overline{F1}$ is statistically different, whereas green means that it is not. The results show that there was no statistically significant difference between FedAvg and FedProx methods in any of the scenarios. Therefore, despite that the $\overline{F1}$ of FedAvg seemed much higher in the IID scenario, due to the high variance of the scores, it was statistically insignificant. In the Non-IID scenarios, both methods showed almost identical performance. Thus, it can be stated that FedAvg and FedProx demonstrated comparable performance across all scenarios. The highest difference between the two was in the IID setting, where FedAvg performed better.

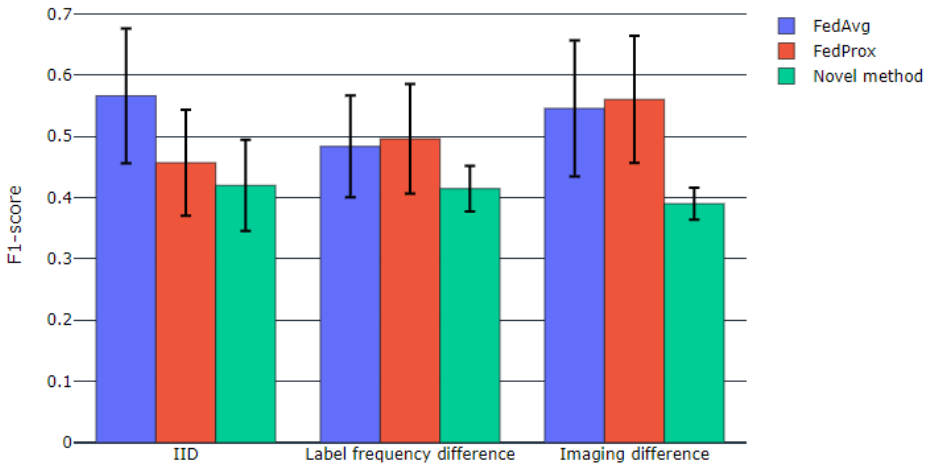


Figure 5.2: The bar chart presents the results achieved for all the three scenarios in which the FL frameworks were tested. Bars are grouped by the scenario, and each bar chart shows the $\overline{F1}$ achieved for a FL framework.

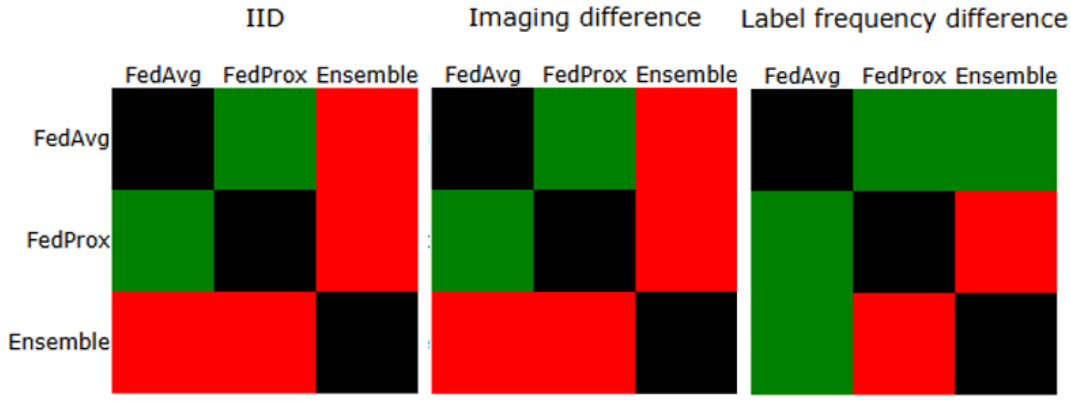


Figure 5.3: The figure shows the results of a t-test conducted to compare the $\overline{F1}$ of the FL frameworks in different scenarios. The test was conducted pairwise for all FL frameworks in each scenario. The colour red indicates that the $\overline{F1}$ is statistically different, whereas green means that it is not.

The novel ensemble framework performed significantly worse than the other two frameworks. As illustrated in 5.3, despite having similar performance across all the scenarios, its metrics were almost always significantly lower than the other frameworks. It achieved better results than the baseline, yet they were much worse than the other two frameworks. To investigate it, the learning curves of the process of fine-tuning on the client side were analysed. They were flat, with minimal classification accuracy improvements visible

over the epochs. The highest F1-score was typically achieved in the initial stages of fine-tuning; specifically, on average around the $10th$ epoch.

## 5.3 Performance of Centrally Trained Classifiers

This section presents the results of the classification achieved by models trained centrally. The results are reported for both two-class and three-class classification.

Table 5.4 shows the results achieved by the best models. To facilitate the comparison, they are reported together with the scores achieved by a baseline model predicting majority class. Each metric is reported along with its standard deviation to show the model stability. Table 5.5 displays confusion matrices produced by aggregating the predictions generated by all 10 models on the test set throughout the evaluation process. Alongside the matrices, sensitivity and specificity for each class are displayed.

The prediction accuracy is higher in the 2-class classification than in the 3-class classification setting (with the test set $\overline{F1}$ of 0.5796 and 0.4960 respectively). Additionally, standard deviations of the metrics are higher in the 3-class case; thus, signalling higher differences between performances of the models.

In the 3-class setting, the model tends to predict class 3 less often than classes 1 and 2, even though the dataset imbalance is not that significant. This leads to the fact that the sensitivity for class 3 is only 0.2111 compared to 0.7125 and 0.5188 achieved for classes 1 and 2. Yet specificity for class 3 is 0.9533; thus, much higher than 0.6240 and 0.6560 obtained for classes 1 and 2 respectively. An opposite behaviour is visible in the 2-class classification case. There, the higher specificity and lower sensitivity was achieved for class 1. Nevertheless, in both settings, many of the predictions are correct signalling the predictive power of the model.

To reliably verify that the $\overline{F1}$ of the ResNet model is higher than the majority voting baseline model, a one-sample t-test was performed. It showed that the performance metrics were statistically higher than the baseline for both models.

The models were also evaluated against the validation data, with each model assessed against its own validation set used in the evaluation procedure. The confusion matrices obtained from this procedure were aggregated and can be seen in Table 5.6 It is visible that the models perform less accurately on the test data than on the validation data. Nevertheless, the confusion matrices have a similar structure. The three-class classification model rarely attempts to predict class 3, which makes sensitivity obtained for this label the lowest. A considerable number of samples of class 3 were erroneously predicted by the model as label 1, further indicating problems in correctly identifying the grade 3 tumours.

Figure 5.4 shows examples of the scans from the test set that were identified correctly and incorrectly by the two-class model with the highest validation score. The images are arranged in the form of a confusion matrix. Images in the top-left corner are TP, in

| | F1-score | | Recall | | Precision | |
|---|---|---|---|---|---|---|
| | M | SD | M | SD | M | SD |
| Two-class Baseline | .2400 | N/A | .5000 | N/A | .1578 | N/A |
| Three-class Baseline | .2191 | N/A | .3902 | N/A | .1523 | N/A |
| Two-class ResNet | .5796 | .0380 | .5859 | .0396 | .5942 | .0410 |
| Three-class ResNet | .4960 | .0805 | .5268 | .0642 | .5114 | .0981 |

Table 5.4: An overview of the results achieved on the test data by the models trained centrally. The numbers are reported as a mean of the metric of the 10 models trained during the evaluation procedure. Scores achieved by a baseline model predicting majority class are displayed alongside the predictions.

Two-class (a)

| | | Predicted | |
|---|---|---|---|
| | Label | 1 | 2 |
| True | 1 | 76 | 84 |
| | 2 | 76 | 174 |

Three-class (b)

| | | Predicted | | |
|---|---|---|---|---|
| | Label | 1 | 2 | 3 |
| True | 1 | 114 | 43 | 3 |
| | 2 | 66 | 83 | 11 |
| | 3 | 28 | 43 | 19 |

Sensitivity and Specificity (c)

| Label | Sensitivity | Specificity |
|---|---|---|
| 1 | .4750 | .6960 |
| 2 | .6960 | .4750 |

Sensitivity and Specificity (d)

| Label | Sensitivity | Specificity |
|---|---|---|
| 1 | .7125 | .6240 |
| 2 | .5188 | .6560 |
| 3 | .2111 | .9533 |

Table 5.5: The tables (a) and (b) show confusion matrices created by aggregating the predictions on the test set of all 10 models trained during the evaluation procedure. Table (c) and (d) display sensitivity and specificity achieved for each distinct class.

the top-right corners there are FN, on the bottom-left there are FP and in the bottom-right True Negatives (TN) are displayed. Only a few examples were selected; nevertheless, one can observe similarities and differences between the tumours of different severity.

## 5.4 Analysis of the Predictive Features

This section presents the results of an analysis of the features that are important for classification. It begins with a description of the results obtained using the Grad-CAM method. The second part presents the result of the analysis of the relationship between the location of the tumour in the brain and its WHO grading.

### 5.4.1 Class Activation Mapping Visualization

Figure 5.5 shows a comparison of Grad-CAM visualisation for the same MRI scan as applied to different models trained during the evaluation procedure. The Grad-CAM was

Two-class (a)

| | | Predicted | |
|---|---|---|---|
| | Label | 1 | 2 |
| True | 1 | 68 | 47 |
| | 2 | 31 | 144 |

Three-class (b)

| | | Predicted | | |
|---|---|---|---|---|
| | Label | 1 | 2 | 3 |
| True | 1 | 82 | 27 | 6 |
| | 2 | 25 | 78 | 6 |
| | 3 | 37 | 19 | 10 |

Table 5.6: Table shows aggregated confusion matrices of the predictions made by each model trained during the evaluation procedure on its own validation set. Figure (a) shows the 3-class setting, while figure (b) portraits 2-class classification.

applied to the last layer of the ResNet model. Red shows the negative contribution to the predicted label, while green shows the positive contribution. All the models shown in the figure predicted the same label that matched the true label of the sample. However, they identified different parts of the image as important for the classification. For two models on the right, the centre of the tumour provided most of the contribution, while for the other two, its edge was the most informative. Similar differences were identified for several samples analysed with Grad-CAM.

### 5.4.2 Positions of the Tumours

During the experiments, only the part of the MRI that was segmented as the tumour and its surroundings was used in the model, as described in the section 3.2. This means that the position of the tumour in the brain was not taken into account in the modelling. Therefore, the distributions of the positions of the tumours of different grades were analysed. Figure 5.6 shows a visualisation of the position of the tumour segmentation centre coloured according to its WHO grading. A few vague clusters could be visually identified during the analysis.

## 5.5 Correlation of the Scores on the Test and Validation Sets

The validation set is used to select the best model during training and hyperparameter tuning. This is because the model's results on the validation set should serve as a "proxy" for its results on the test set. So, in theory, the better the score on the validation set, the better the model should generalise to the unseen data. To check it, a linear correlation was calculated between the validation and test F1-scores for each method. The result is illustrated in Figure 5.7. It presents a single scatter plot for each training methodology. Each scatter plot shows the F1-scores on the test set vs on the validation set, together with the correlation coefficient. A negative correlation indicates that a higher validation score results in a lower test score. The correlation was negative for the models trained centrally and the Ensemble FL method, with coefficients of $-0.2034$ and $-0.3176$ respectively. Therefore, for these methods, a higher validation F1-score yields,

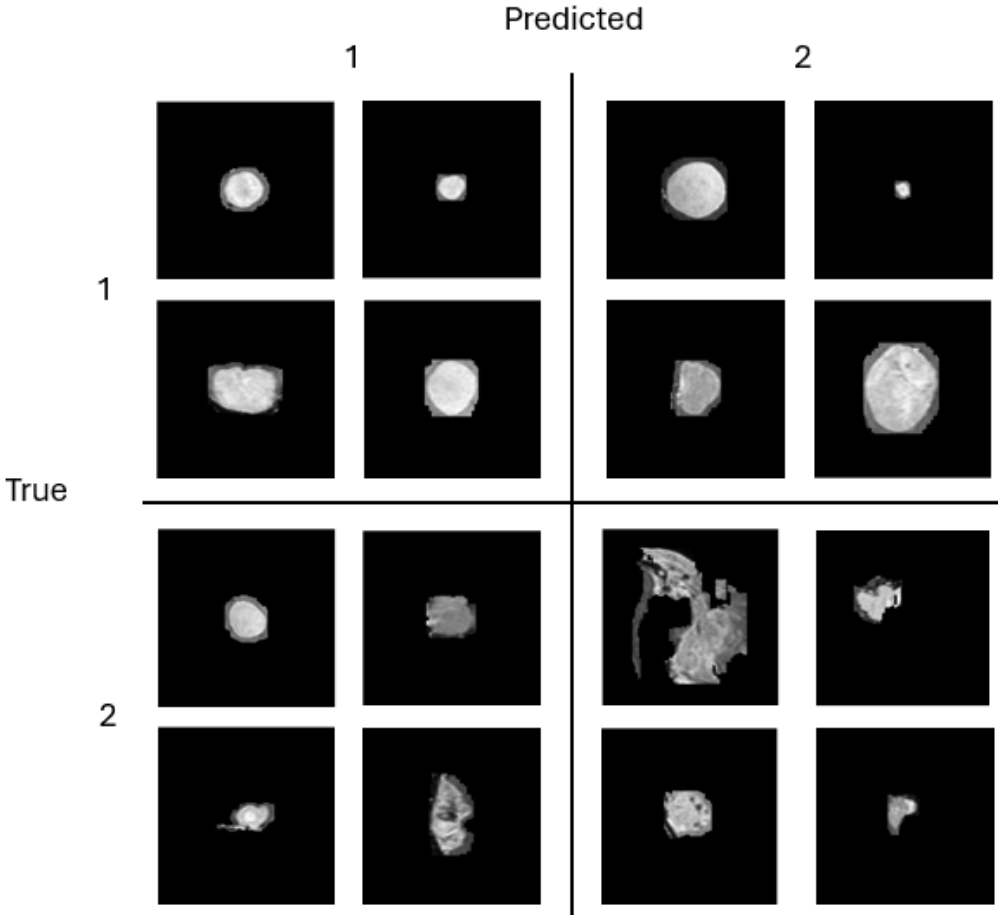


Figure 5.4: The figures show examples of the scans from the test set that were identified correctly and incorrectly by the two-class model with the highest validation score. The images are arranged in the form of a confusion matrix. Images in the top-left corner are TP, in the top-right corners there are FN, on the bottom-left there are FP and in the bottom-right TN are displayed. The scans displayed in the figure were selected at random and all show the middle sagittal slice of the tumour.

on average, a lower score on the test set. For FedAvg and FedProx, the correlation was positive, showing the expected relationship between validation and test set performance. Specifically, a better performance on the validation set resulted in a higher prediction accuracy on the test set.

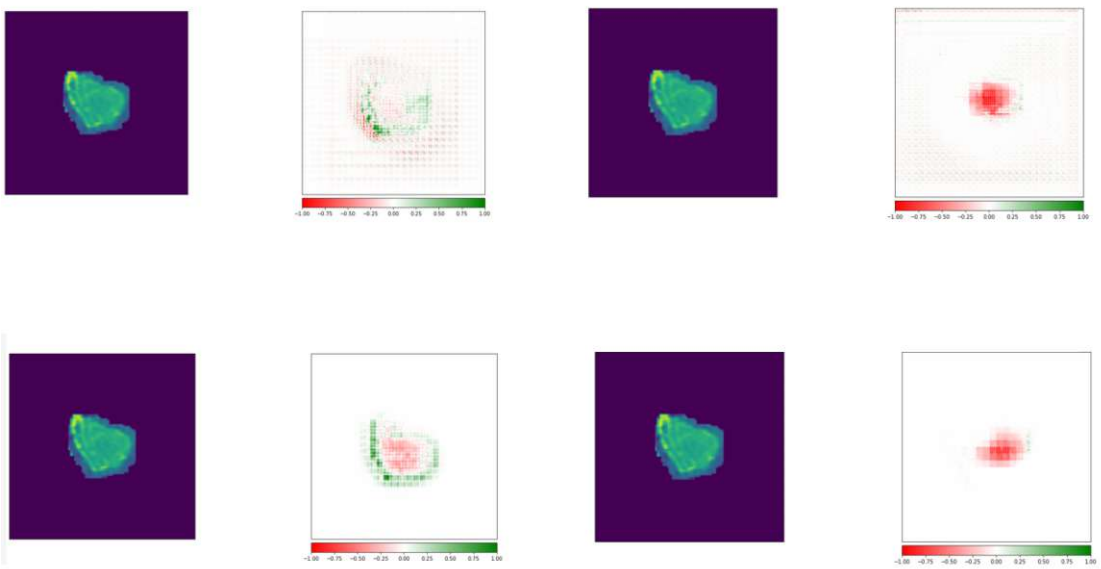

Figure 5.5: A comparison of Grad-CAM visualization for the same MRI scan for different models trained during the evaluation procedure. It was applied on the last layer of the ResNet model. Red colour shows the negative contribution to the predicted label, while green shows positive contribution. All the displayed models predicted the same label that matched the true label of the sample.

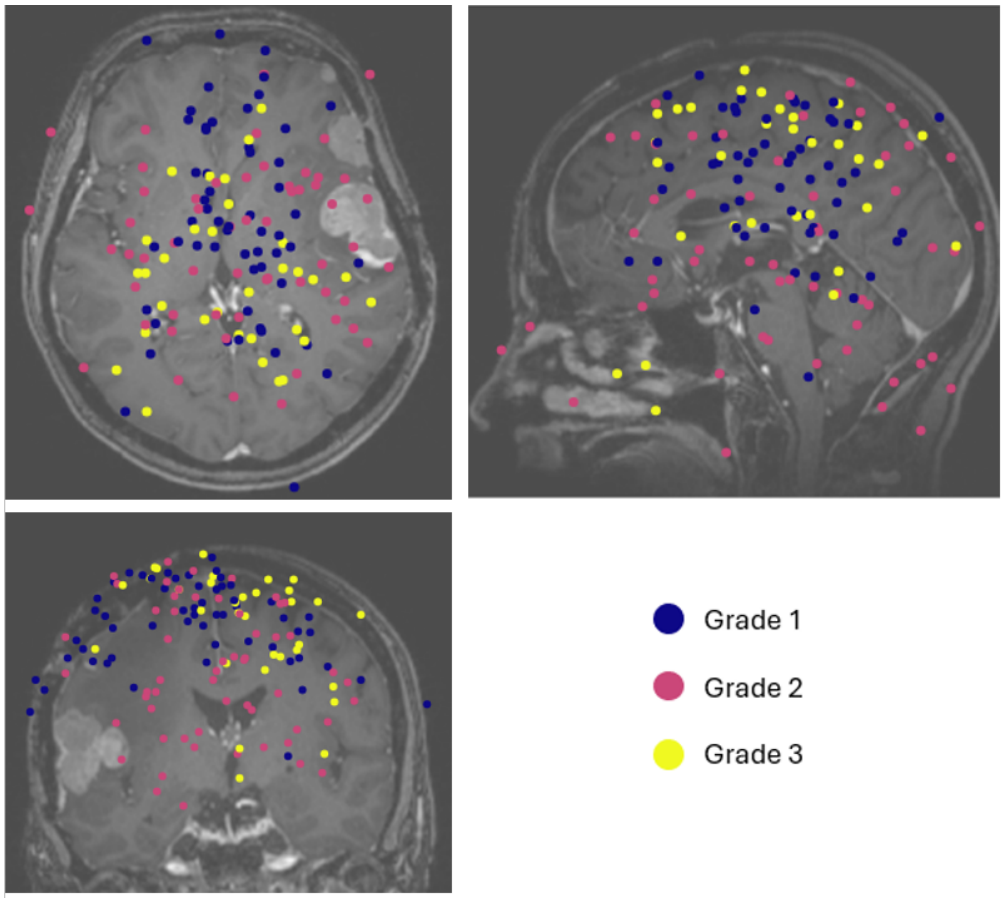


Figure 5.6: Figure shows a visualisation of the position of the tumour segmentation centre coloured according to its WHO grading. In the background, the middle slice for each plane of a randomly selected scan is displayed.

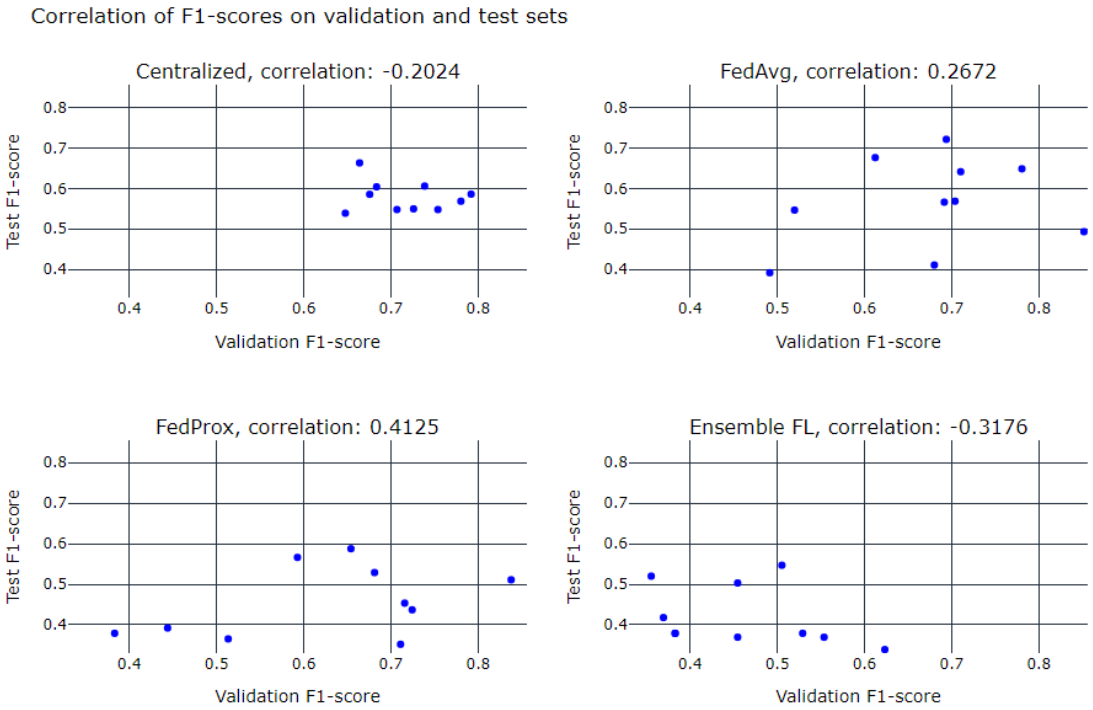


Figure 5.7: Figure presents a single scatter plot for each training methodology. Each scatter plot shows the F1-scores on the test set vs on the validation set, together with the correlation coefficient. A negative correlation indicates that a higher validation score results in a lower test score.

## 5.6 Weights Assigned to the Models in the Federated Ensemble Method

This section presents insights into the weights assigned to each ensemble model in the Federated Ensemble framework. Both local and global weights were analysed, but only the latter was evaluated in the experiments.

### 5.6.1 Global Weighting

The global model is an ensemble of the models trained by each client together with their weights, where the weights should favor the models that perform better on the validation data. In most cases, the method assigned a similar weighting to the models within the global ensemble, with a median of 0.33. There were no instances in which a weight was set to a value exceeding 0.44 (where all 3 weights sum up to 1.0).

#### Local Weighting

In addition to creating a set of weights for the models in the global ensemble, each client also creates a set of the weights that optimise the ensemble for the client's local data. In this context, the weight can be assigned either to the model trained on the client's local data or to the model fine-tuned by other clients. Figure 5.8 presents a box plot of the weights assigned to the models fine-tuned on the client's local data. In the IID scenario, the models fine-tuned to the local datasets of the clients appear to be weighted the same as the models fine-tuned to the data of the other clients. The maximum weight assigned to the local model was 0.45, which is not much higher than its median of 0.33. While the median of the weights in the Non-IID scenarios shows a similar behaviour, the variance is higher. Thus, there are some ensemble models that give more weight to the local model.

## 5.7 Insights from Hyperparameter Tuning

### 5.7.1 ResNet Hyperparameter Tuning

This section describes the experiments that were carried out to find the best hyperparameters for the ResNet model. All the models were trained over 200 epochs and evaluated using the same validation set. Initially, the hyperparameters were first tuned for the three-class setting, and subsequently for the two-class setting.

#### Learning Rate

First, the learning rate hyperparameter was tuned. Six distinct learning rates were evaluated, and other hyperparameters were set to values deemed reasonable based on preliminary experiments. The $aug_{prob}$ hyperparameter was fixed at 0.2 and $l_{frozen}$ was set to 3.

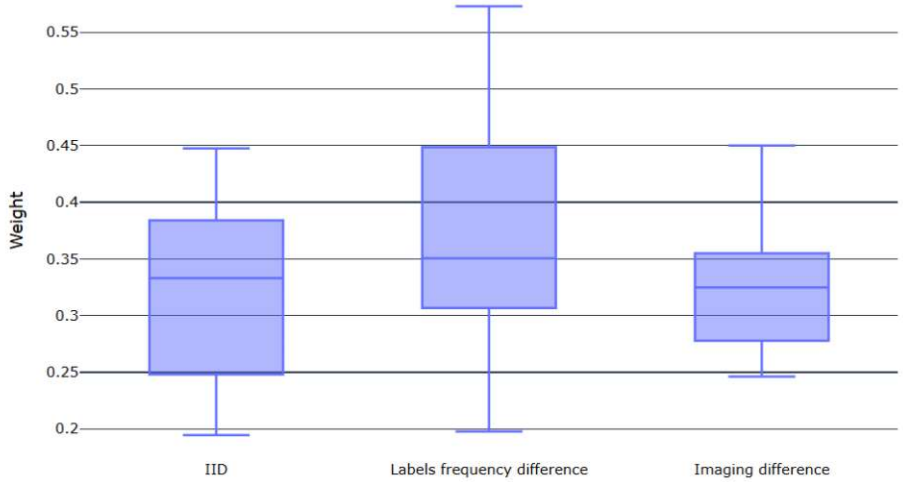


Figure 5.8: Figure presents a box plot of the weights assigned to the models fine-tuned on the client's local data. One weight was assigned to each of the 3 models in the ensemble, and the weights summed up to 1.0.

Figure 5.9 illustrates the comparison of F1-scores for varying learning rates. The highest F1-score of 0.5527 was achieved for the learning rate of $10^{-3}$. Notably, relatively high scores were also achieved for the learning rates of 0.01 and 0.2. However, for the higher learning rates, the learning curves were very noisy, which may indicate that the learning rate is too high. In the case of lower learning rates (equivalent to or below 0.01), the learning curves were more stable. Nevertheless, despite the initial decrease in validation loss in line with training loss during the initial epochs, validation loss increased rapidly later on, signalling overfitting. An exemplary learning curve for a learning rate of $10^{-3}$ is presented in Figure 5.10.

Even though overfitting was visible in the training process, the resulting models demonstrated a predictive power and achieved higher results than the baseline model.

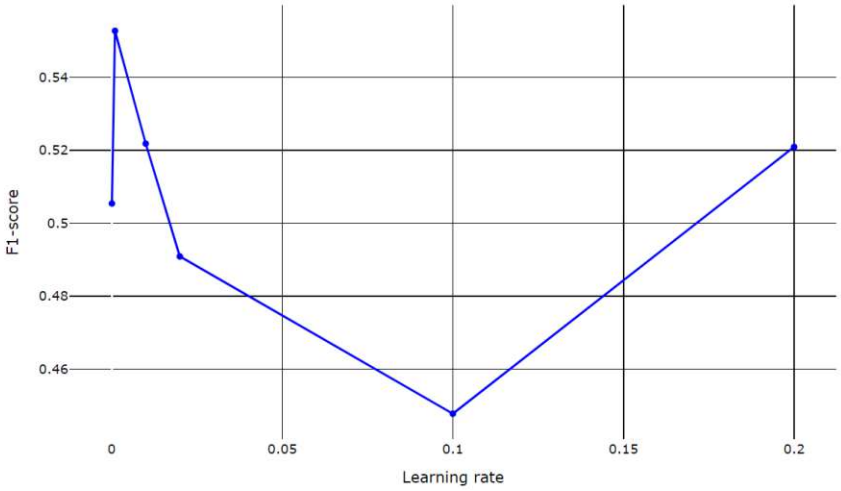


Figure 5.9: Comparison of F1-scores for different learning rates.

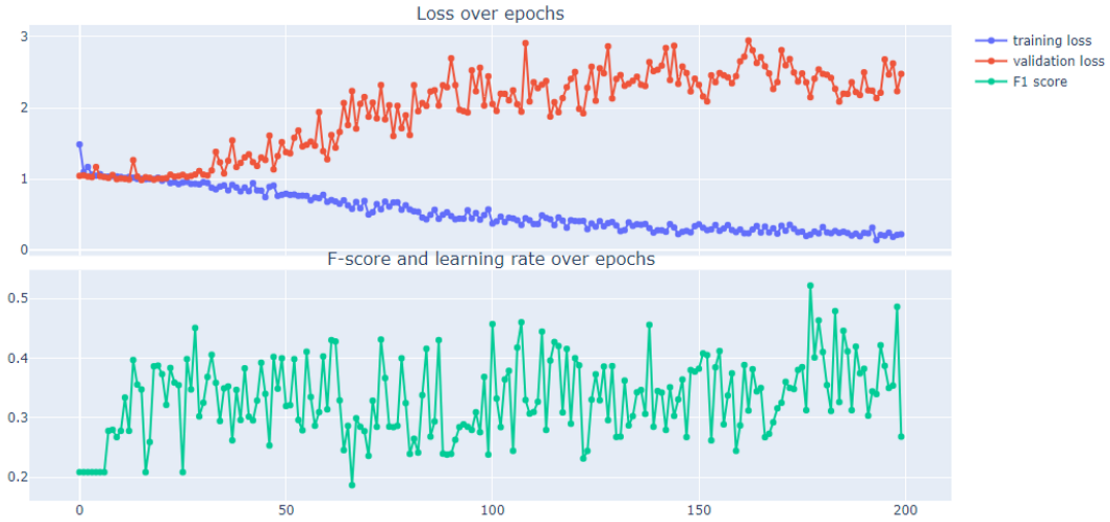


Figure 5.10: Learning curve for a run with a learning rate of $10^{-3}$ that signals overfitting when there is too little image augmentation. The red curve illustrates the training loss, and the blue curve illustrates the validation loss.

### Image Augmentation

Given the high degree of overfitting observed, experiments were conducted to assess if image augmentation could reduce it. Initially, a number of image augmentation procedures were evaluated visually, as outlined in 4.2. Then different $aug_{prob}$ and $h_{aug}$ settings were tried out. An in-depth description of the augmentation techniques can be found in Section 4.2.

| Augmentation Pipeline | Augmentation Probability 0.5 | 0.8 | 1.0 |
|---|---|---|---|
| Basic | .5219 | .5449 | .5513 |
| Strong | .5021 | .5375 | .5634 |
| Extended | .5057 | .5348 | .5777 |

Table 5.7: Table shows an overview of the F1-scores across different settings of augmentation. Every row of the matrix corresponds to a different augmentation type and every column to a different augmentation probability. The $\overline{F1}$ for each cell is calculated using the results of experiments with three different learning rates, namely $10^{-4}, 10^{-3}, 10^{-2}$.

Table 5.7 presents an overview of the scores across different hyperparameter settings. Every row of the matrix corresponds to a different augmentation methods and every column to a different augmentation step probability. The $\overline{F1}$ for each cell is calculated using the results of experiments with three different learning rates, namely $10^{-4}, 10^{-3}, 10^{-2}$. It can be observed that a higher augmentation probability leads to an improved performance of the model. Setting the augmentation probability correctly is a more influential factor for optimizing performance than the specific augmentation method. The *Extended* augmentation method with the augmentation probability of 1.0 achieved the best results, with a $\overline{F1}$ of 0.5777.

The learning curves in the majority of the experiments indicated less overfitting in comparison to those with minimal augmentation described in the Section 5.7.1. One of the learning curves is displayed in Figure 5.11. It can be observed that the validation loss begins to increase much later than in Figure 5.10. Based on those results, the *Extended* augmentation method with an augmentation probability of 1.0 and a learning rate of 0.001 was selected for use in the remaining experiments.

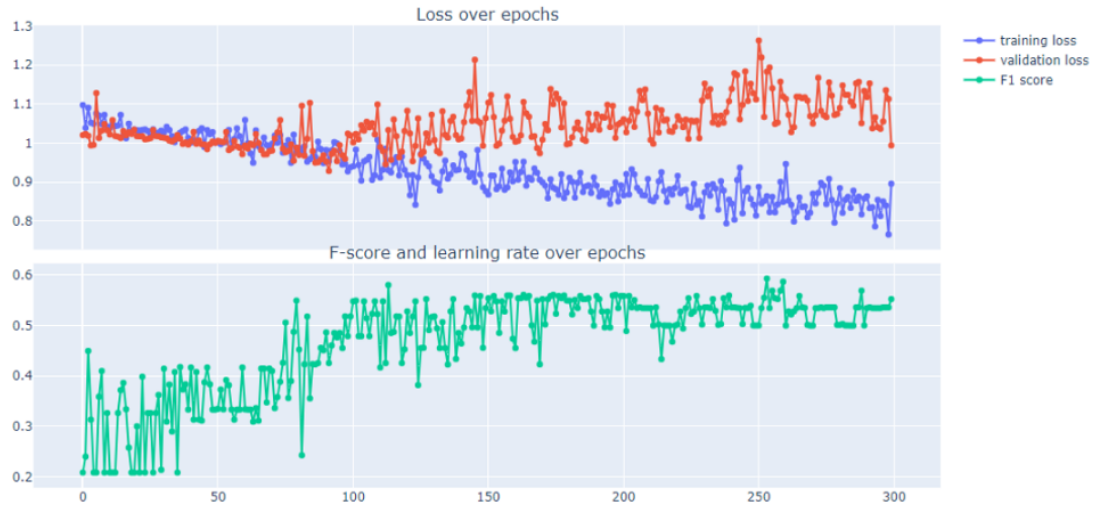


Figure 5.11: Learning curve of a run with a higher augmentation. The red curve illustrates the training loss, and the blue curve illustrates the validation loss.

### Number of Layers to Freeze

The final stage of the hyperparameter tuning process was to determine the optimal $l_{frozen}$. Four different settings were experimented with, namely 0, 1, 2 or 3. Two different learning rates were tried out: 0.01 and 0.001. Other ResNet hyperparameters were set to those obtained in the previous section.

It was observed that freezing two or three layers resulted in underfitting. The model with those settings achieved relatively low scores of 0.5143 and 0.4190 respectively. The models with one and no frozen layers achieved a better performance, with F1-scores of 0.5549 and 0.6197 respectively. The model with no frozen layers yielded more stable training and validation losses, leading to more accurate predictions. Figure 5.12 shows the comparison between the learning curves of the models with three and zero frozen layers. Training loss of the model with three frozen layers converged worse than the model with no frozen layers, indicating underfitting. Thus, it was decided that no layers will be frozen in the later experiments.

Considering the results, the following optimal hyperparameters were selected for the three-class classification ResNet: the learning rate of 0.001, the *Extended* $h_{aug}$ with an $aug_{prob}$ of 1.0 and $l_{frozen}$ equal to 0.

Furthermore, it was observed that the results depend strongly on the random seed used for the model training. Consequently, a considerable inter-model variability, leading to a considerable variance in results, is to be expected.

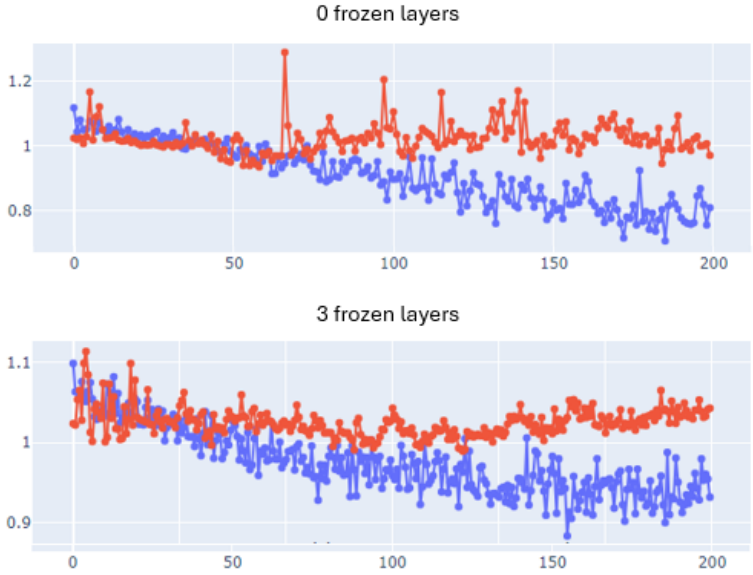


Figure 5.12: The figure presents a comparison between the learning curves of the models with three and zero frozen layers. The model with zero frozen layers is displayed at the top, while the model with three frozen layers is displayed at the bottom. The red curve illustrates the training loss, and the blue curve illustrates the validation loss.

#### Two-class Classification

A similar hyperparameter tuning procedure was conducted for the two-class problem. An analogous hyperparameter space was explored. Considering the results, the following optimal hyperparameters were selected for the three-class classification ResNet: the learning rate of 0.01, the *Extended* $h_{aug}$ with an $aug_{prob}$ of 1.0 and $l_{frozen}$ equal to 0.

### 5.7.2 Federated Learning Hyperparameter Tuning

This section presents the results of hyperparameter tuning of the FL frameworks. All the experiments evaluated the frameworks in the two-class classification scenario. The optimal ResNet hyperparameters identified in the previous section were used across all the subsequent experiments. FL was simulated using the *flower* library[1] with the number of clients set to 3.

#### Federated Averaging

The optimal hyperparameters search started with the identification of the optimal $n_{\text{local epochs}}$ and proportion of clients per round for FedAvg. The hyperparameters search

[1] https://flower.ai/

| $n_{\text{local epochs}}$ | $n_{rounds}$ | Proportion of Clients per Round | F1-score |
|---|---|---|---|
| 1 | 200 | 1.0 | .6801 |
| 1 | 200 | 2/3 | .6900 |
| 2 | 100 | 1.0 | .6362 |
| 2 | 100 | 2/3 | .6480 |
| 5 | 40 | 1.0 | .6557 |
| 5 | 40 | 2/3 | .6524 |
| 20 | 10 | 1.0 | .6136 |
| 20 | 10 | 2/3 | .6649 |

Table 5.8: The table shows an overview of the $\overline{F1}$ across different hyperparameters of FedAvg. For each set of the hyperparameters, the $\overline{F1}$ of two runs with different seeds is reported. Due to the observed high inter-model variability, the experiments were run twice for each setting.

space was defined as described in the section 4.5.1. Similarly to the previous experiments, a high degree of inter-model variability was observed. Consequently, the experiments were run twice with different seeds for every hyperparameters' setting. The results of the search are displayed in Table 5.8.

The highest F1-score of 0.6900 was achieved by the run with $n_{\text{local epochs}}$ set to 1, and the proportion of clients per round equal to 2/3. This score was only slightly worse than the score achieved in the centralized approach. The experiments conducted with the higher $n_{\text{local epochs}}$ yielded inferior results. The learning curves demonstrated that increasing $n_{\text{local epochs}}$ resulted in a slower convergence of the loss.

The proportion of clients for each round set to 2/3 achieved better results and higher validation loss stability than using all the clients for each round. Additionally, setting this parameter to the lower value accelerates the training process. Thus, the proportion of clients for each round will be set to 2/3 in the subsequent experiments.

Nevertheless; it is notable that despite some differences, the F1-score was similar across all the hyperparameter settings.

### FedProx

After identifying the optimal parameters for FedAvg, the optimal $\mu$ for the FedProx framework was set. The best F1-score of 0.7029 was achieved for the $\mu$ of $10^{-5}$. This setting applies the lowest penalty on deviations from the global model.

### Federated Localized Ensemble Framework

The best results were achieved by the models with $l_{frozen}^{tuning}$ equal to 2. The mean validation F1-score achieved by them was 0.5066, which is inferior to the results obtained by the other FL frameworks. Nevertheless, it was still better than the baseline.

Additionally, an experiment was carried out with $n_{tuning}$ set to 50. The achieved F1-score was 0.5132, which was not significantly higher than the score of the run with $n_{tuning}$ equal to 100. Nevertheless, this setting was selected, as it also speeds up training.

CHAPTER 6

# Discussion

## 6.1 Dataset

The dataset used for experiments is one of the most important parts of any ML project. In this case, the experiments were performed on a real-world dataset from one institution. The dataset was small, consisting of only 186 samples. It is difficult to collect a lot of data for this particular disease because meningioma is rare and annotating the data is a costly process. Thus, the results may not be generalisable to other datasets as the data may not be representative. For example, the class distribution in the dataset differs significantly from the population distribution.

The true label of each sample was obtained by biopsy, which is a very precise measurement. However, the segmentations were labelled by only one clinician, so some errors are possible.

## 6.2 Comparison of Centralized Training and Federated Learning in the IID scenario

The first experiment focused on comparing the models trained with FL to those trained centrally. Even if the models trained using FL were slightly worse, the benefits of training them on more data might outweigh the performance drop [72]. This is particularly true for rare diseases, where each sample may be very difficult to obtain.

According to the experiments and statistical tests, FedAvg performed similarly to the centralized training, indicating the potential of the method. However, the standard deviation of the achieved scores was higher. This means that the models trained with FL had higher inter-model variability. It was also visible on the feature importance maps generated using Grad-CAM. Variability may happen due to the fact that the dataset size is small. In general, FL is better suited to working on larger datasets, especially as the method itself is designed to allow easier inter-institutional collaborations [72, 12].

The FedProx framework achieved a lower score than the basic FedAvg, but the t-test did not show the difference to be significant. As the main aim of this framework is to reduce the impact of the Non-IID on the training, a slightly lower score could indicate that the data in the first simulation was indeed IID, and there is no benefit to changing the training process from the basic FedAvg.

In addition, the validation sets used to select the best model obtained during training were too small and not representative enough. It was shown in the section 5.5 that sometimes higher scores on the validation set correlated with lower performance on the test set. This also means that the hyperparameter tuning procedure may have been misleading, as the hyperparameters were chosen based on the score on the validation set.

## 6.3 Federated Learning in the Non-IID Scenario

The experiments showed that the prediction accuracy of the models decreased in the Non-IID scenarios, which is expected [71]. It was most affected in the first scenario, where the label frequency difference was simulated. The Non-IID scenario, which simulated the imaging difference had little effect on the accuracy of the models. However, although the performance of FL was affected,the FedAvg and FedProx frameworks achieved not much worse results than the centrally trained models.

Apart from the accuracy drop, the inter-model variability increased. This shows that in the Non-IID case the training is less predictable and there is a higher risk of poor training convergence and overfitting.

In both Non-IID scenarios, the FedProx framework produced better results than the other FL frameworks, although it was worse in the IID setting. This was the case even though $\mu$ may not have been set optimally. It shows its usefulness in dealing with data heterogeneity across the institutions.

The performance of the Federated Ensemble framework was significantly worse than the other frameworks. It performed similarly in all the FL scenarios, but the prediction accuracy was low most of the time. Also, the framework seemed to usually assign similar weights to all models in the ensemble, so the weighting method did not bring much benefit. However, the weighting method optimised for the client's local data was not evaluated.

Additionally, during the fine-tuning of the site-specific models, almost no improvement of the validation loss was observed. This may be because the client-side dataset was too small to train the model by itself. This is likely as the number of model parameters is significant and therefore a training dataset in the range of 30 samples may not be sufficient. In addition, the validation set available at each site was very small, often around 8 samples. Therefore, in most cases it was unlikely to be representative of the dataset. If this was the case, the selection of the best model during training was highly biased and the calculation of *global weights* was almost impossible with the current methodology. However, it could also be that the method itself simply does not work well.

Although Federated Localized Ensemble did not produce satisfactory results, it does have some advantages. First, an institution can join after the initial training by simply adding its model to the ensemble. Also, institutions can set the weights of the ensemble models to optimise the model for their local data distribution. Finally, the method requires lower communication costs as there is no need to exchange models during the second part of the training.

## 6.4 Performance of the Centrally Trained Classifiers

The models trained centrally showed predictive accuracy and managed to achieve higher scores than the baseline model, both for two- and three-class classification, even though the task of meningioma WHO grading classification is known to be very hard [23]. However, they still made a lot of mistakes.

The three-class classification task was more difficult for the classifier. The WHO grade 3 tumours were most difficult to identify correctly, while benign tumours were the easiest to classify.

The automatic grading of meningiomas from MRIs has also been addressed in the literature. The papers report different results, with accuracies ranging from 55% to 95% [42, 39]. Most studies evaluated the methods on small datasets, which is not surprising as meningioma is a rare disease. Therefore, the datasets are likely to be very different to the one used in this paper, for example in terms of label distribution, making it difficult to compare accuracy between methods. This points to what probably was the hardest issue in the thesis - the small size of the dataset. Given that the task is very challenging [23], this small number of samples may have been enough to achieve very good results. Given that the task is very challenging [23], this small number of samples may have been enough to achieve very good results.

Some possible improvements to the method were identified. Firstly, the decision to use the 3D data representation may have been detrimental. It resulted in a large number of parameters in the model and a long training time. Also, cropping by the segmentation mask often left a large empty space, which could be problematic for the model. Finally, another transfer learning network could be tried.

In addition, some additional features could be added as input to the model, such as the position of the tumour. Many studies also use multiple MRI sequences for classification to give the model more information. For example, T1-weighted contrast-enhanced (T1CE) sequences are often used [42, 39].

CHAPTER 7

# Conclusion and Future Work

The aim of the study was to investigate the possibility of classifying meningioma WHO grading from MRI images, which is an important but challenging task. A DL model, called ResNet, was used to approach the task. It was trained and evaluated in different environments. The first environment, which was also the simplest, was centralised training. It assumes that all the data can be collected on the same machine. Although the accuracy of the model trained in this way was not very high, it showed predictive accuracy. This suggests that ML has the potential to help in the non-invasive grading of meningiomas. This was the case even though the dataset size was rather small.

In the medical field, small datasets are often an issue, making it difficult to develop accurate models. This is also the case for meningioma, as it is a rare disease (like all brain tumours), and the collection of large datasets often requires inter-institutional collaboration. Therefore, the main focus of the thesis was FL. Experiments showed that the models trained in this way performed similarly to centrally trained models. This shows that training models using FL may be an interesting alternative to CDS for meningioma grading classification, although this idea has not been widely explored in the literature.

Nevertheless, the dataset used in the experiments was collected at one institution, so the FL environment was simulated. Thus, the frameworks were evaluated in a simplified environment. Therefore, the experiments were also conducted in two environments that simulated heterogeneous data distribution across the FL clients. Although the simulation was artificial, it was shown that the performance of the models was negatively affected. Nevertheless, the models still produced reasonable results, not much worse than the centrally trained models. In particular, the FedProx framework was promising and could probably achieve even better results if more time was spent on hyperparameter tuning.

Moreover, a novel framework was proposed to address the negative impact of Non-IID data across clients. However, it did not produce satisfactory results, mainly because it was

not well suited to the small amount of samples available at each client. Nevertheless, it would be interesting to evaluate it on a larger dataset, as its design has some advantages.

As the models and the FL framework itself showed promising results, several future directions were identified. First, the underlying DL model could probably be improved. For example, additional input data, such as a different MRI sequence or tumour location, could be used. Also, the 3D volumes could be transformed into 2D slices to reduce the number of parameters in the model. Another idea would be to also test different DL networks. Secondly, although the evaluation methodology addressed the small size of the dataset to some extent, a larger dataset, preferably including data from several institutions, would be recommended. Finally, other state-of-the-art FL frameworks could be evaluated, especially those that address the issue of statistical heterogeneity.

# Overview of Generative AI Tools Used

**DeepL Write:** It was used only to check and improve the grammar. It was not used for any actual content generation.

# Bibliography


[1] Arivazhagan, M. G., Aggarwal, V., Singh, A., and Choudhary, S. Federated learning with personalization layers.

[2] Banzato, T., Causin, F., Della Puppa, A., Cester, G., Mazzai, L., and Zotti, A. Accuracy of deep learning to differentiate the histopathological grading of meningiomas on mr images: A preliminary study. *Journal of Magnetic Resonance Imaging 50* (03 2019).

[3] Brown, J. R. G., Mansour, N. M., Wang, P., Chuchuca, M. A., Minchenberg, S. B., Chandnani, M., Liu, L., Gross, S. A., Sengupta, N., and Berzin, T. M. Deep learning computer-aided polyp detection reduces adenoma miss rate: a united states multi-center randomized tandem colonoscopy study (cadet-cs trial). *Clinical Gastroenterology and Hepatology 20*, 7 (2022), 1499–1507.

[4] Cai, X., Lan, Y., Zhang, Z., Wen, J., Cui, Z., and Zhang, W. A many-objective optimization based federal deep generation model for enhancing data processing capability in iot. *IEEE Transactions on Industrial Informatics 19*, 1 (2023), 561–569.

[5] Cardoso, M. J., Li, W., Brown, R., Ma, N., Kerfoot, E., Wang, Y., Murrey, B., Myronenko, A., Zhao, C., Yang, D., Nath, V., He, Y., Xu, Z., Hatamizadeh, A., Myronenko, A., Zhu, W., Liu, Y., Zheng, M., Tang, Y., Yang, I., Zephyr, M., Hashemian, B., Alle, S., Darestani, M. Z., Budd, C., Modat, M., Vercauteren, T., Wang, G., Li, Y., Hu, Y., Fu, Y., Gorman, B., Johnson, H., Genereaux, B., Erdal, B. S., Gupta, V., Diaz-Pinto, A., Dourson, A., Maier-Hein, L., Jaeger, P. F., Baumgartner, M., Kalpathy-Cramer, J., Flores, M., Kirby, J., Cooper, L. A. D., Roth, H. R., Xu, D., Bericat, D., Floca, R., Zhou, S. K., Shuaib, H., Farahani, K., Maier-Hein, K. H., Aylward, S., Dogra, P., Ourselin, S., and Feng, A. Monai: An open-source framework for deep learning in healthcare, 2022.

[6] Chang, K., Bai, H., Ly, I., Su, C., Agbodza, E., Kavouridis, V., Senders, J., Beers, A., Zhang, B., Capellini, A., et al. Residual convolutional neural network for determination of idh status in low-and high-grade gliomas. In *SNO 2017 Annual Meeting* (2017), SNO.

[7] Chang, Q., Qu, H., Zhang, Y., Sabuncu, M., Chen, C., Zhang, T., and Metaxas, D. Synthetic learning: Learn from distributed asynchronized discriminator gan without sharing medical image data. pp. 13853–13863.

[8] Chang, Q., Qu, H., Zhang, Y., Sabuncu, M., Chen, C., Zhang, T., and Metaxas, D. N. Synthetic learning: Learn from distributed asynchronized discriminator gan without sharing medical image data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 13856–13866.

[9] Chen, C., Guo, X., Wang, J., Guo, W., Ma, X., and Xu, J. The diagnostic value of radiomics-based machine learning in predicting the grade of meningiomas using conventional magnetic resonance imaging: A preliminary study. *Frontiers in Oncology 9* (2019).

[10] Chen, S., Ma, K., and Zheng, Y. Med3d: Transfer learning for 3d medical image analysis. *arXiv preprint arXiv:1904.00625* (2019).

[11] Chlap, P., Min, H., Vandenberg, N., Dowling, J., Holloway, L., and Haworth, A. A review of medical image data augmentation techniques for deep learning applications. *Journal of Medical Imaging and Radiation Oncology 65*, 5 (2021), 545–563.

[12] Chowdhury, A., Kassem, H., Padoy, N., Umeton, R., and Karargyris, A. A review of medical federated learning: Applications in oncology and cancer research. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries* (Cham, 2022), A. Crimi and S. Bakas, Eds., Springer International Publishing, pp. 3–24.

[13] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database, 2009.

[14] Draelos, R. L., and Carin, L. Use hirescam instead of grad-cam for faithful explanations of convolutional neural networks.

[15] Duan, M., Liu, D., Chen, X., Tan, Y., Ren, J., Qiao, L., and Liang, L. Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications. In *2019 IEEE 37th International Conference on Computer Design (ICCD)* (2019), pp. 246–254.

[16] Elshabrawy, K. M., Alfares, M. M., and Salem, M. A.-M. Ensemble federated learning for non-ii d covid-19 detection. In *2022 5th International Conference on Computing and Informatics (ICCI)* (2022), pp. 057–063.

[17] Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics 36*, 4 (Apr. 1980), 193–202.

[18] Guo, P., Wang, P., Zhou, J., Jiang, S., and Patel, V. M. Multi-institutional collaborations for improving deep learning-based magnetic resonance image reconstruction using federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2021), pp. 2423–2432.

[19] Hamer, J., Mohri, M., and Suresh, A. T. FedBoost: A communication-efficient algorithm for federated learning. In *Proceedings of the 37th International Conference on Machine Learning* (13–18 Jul 2020), H. D. III and A. Singh, Eds., vol. 119 of *Proceedings of Machine Learning Research*, PMLR, pp. 3973–3983.

[20] Hard, A., Rao, K., Mathews, R., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., and Ramage, D. Federated learning for mobile keyboard prediction. *CoRR abs/1811.03604* (2018).

[21] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), 770–778.

[22] Hsu, T.-M. H., Qi, H., and Brown, M. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335* (2019).

[23] Huang, R. Y., Bi, W. L., Griffith, B., and Kaufmann. Imaging and diagnostic advances for intracranial meningiomas. *Neuro-Oncology 21*, Supplement 1 (01 2019), i44–i61.

[24] Isensee, F., Wald, T., Ulrich, C., Baumgartner, M., Roy, S., Maier-Hein, K., and Jaeger, P. F. nnu-net revisited: A call for rigorous validation in 3d medical image segmentation, 2024.

[25] Jiménez-Sánchez, A., Tardy, M., Ballester, M. Á. G., Mateus, D., and Piella, G. Memory-aware curriculum federated learning for breast cancer classification. *CoRR abs/2107.02504* (2021).

[26] Khan, M. I., Jafaritadi, M., Alhoniemi, E., Kontio, E., and Khan, S. A. Adaptive weight aggregation in federated learning for brain tumor segmentation. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries* (Cham, 2022), A. Crimi and S. Bakas, Eds., Springer International Publishing, pp. 455–469.

[27] Khawaldeh, S., Pervaiz, U., Rafiq, A., and Alkhawaldeh, R. S. Noninvasive grading of glioma tumor using magnetic resonance imaging with convolutional neural networks. *Applied Sciences 8*, 1 (2017), 27.

[28] Kim, M., Yun, J., Cho, Y., Shin, K., Jang, R., Bae, H.-j., and Kim, N. Deep learning in medical imaging. *Neurospine 16*, 4 (2019), 657.

[29] Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* (2012), F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25, Curran Associates, Inc.

[30] Kshettry, V. R., Ostrom, Q. T., Kruchko, C., Al-Mefty, O., Barnett, G. H., and Barnholtz-Sloan, J. S. Descriptive epidemiology of world health organization grades II and III intracranial meningiomas in the united states. *Neuro Oncol 17*, 8 (May 2015), 1166–1173.

[31] Kumar, S., Lakshminarayanan, A., Chang, K., Guretno, F., Mien, I. H., Kalpathy-Cramer, J., Krishnaswamy, P., and Singh, P. Towards more efficient data valuation in healthcare federated learning using ensembling. In *Distributed, Collaborative, and Federated Learning, and Affordable AI and Healthcare for Resource Diverse Global Health* (Cham, 2022), S. Albarqouni, S. Bakas, S. Bano, M. J. Cardoso, B. Khanal, B. Landman, X. Li, C. Qin, I. Rekik, N. Rieke, H. Roth, D. Sheet, and D. Xu, Eds., Springer Nature Switzerland, pp. 119–129.

[32] Lee, J.-G., Jun, S., Cho, Y.-W., Lee, H., Kim, G. B., Seo, J. B., and Kim, N. Deep learning in medical imaging: general overview. *Korean journal of radiology 18*, 4 (2017), 570.

[33] Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems 2* (2020), 429–450.

[34] Liang, P. P., Liu, T., Ziyin, L., Salakhutdinov, R., and Morency, L.-P. Think locally, act globally: Federated learning with local and global representations. *ArXiv abs/2001.01523* (2020).

[35] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I. A survey on deep learning in medical image analysis. *Medical Image Analysis 42* (2017), 60–88.

[36] Liu, X., Faes, L., Kale, A., Wagner, S., Fu, D., Bruynseels, A., Mahendiran, T., Moraes, G., Shamdas, M., Kern, C., Ledsam, J., Schmid, M., Balaskas, K., Topol, E., Bachmann, L., Keane, P., and Denniston, A. A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis. *The Lancet Digital Health 1* (09 2019).

[37] Liu, X., Faes, L., Kale, A. U., Wagner, S. K., Fu, D. J., Bruynseels, A., Mahendiran, T., Moraes, G., Shamdas, M., Kern, C., Ledsam, J. R., Schmid, M. K., Balaskas, K., Topol, E. J., Bachmann, L. M., Keane, P. A., and Denniston, A. K. A comparison of deep learning performance against

health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis. *The Lancet Digital Health 1*, 6 (Oct. 2019), e271–e297.

[38] Ludwig, H., Baracaldo, N., Thomas, G., Zhou, Y., Anwar, A., Rajamoni, S., Ong, Y., Radhakrishnan, J., Verma, A., Sinn, M., Purcell, M., Rawat, A., Minh, T., Holohan, N., Chakraborty, S., Whitherspoon, S., Steuer, D., Wynter, L., Hassan, H., and Abay, A. Ibm federated learning: an enterprise framework white paper v0.1.

[39] Maniar, K. M., Lassarén, P., Rana, A., Yao, Y., Tewarie, I. A., Gerstl, J. V., Recio Blanco, C. M., Power, L. H., Mammi, M., Mattie, H., Smith, T. R., and Mekary, R. A. Traditional machine learning methods versus deep learning for meningioma classification, grading, outcome prediction, and segmentation: A systematic review and meta-analysis. *World Neurosurgery 179* (2023), e119–e134.

[40] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *International Conference on Artificial Intelligence and Statistics* (2016).

[41] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M.-A., Arbel, T., Avants, B. B., Ayache, N., Buendia, P., Collins, D. L., Cordier, N., Corso, J. J., Criminisi, A., Das, T., Delingette, H., Demiralp, C., Durst, C. R., Dojat, M., Doyle, S., Festa, J., Forbes, F., Geremia, E., Glocker, B., Golland, P., Guo, X., Hamamci, A., Iftekharuddin, K. M., Jena, R., John, N. M., Konukoglu, E., Lashkari, D., Mariz, J. A., Meier, R., Pereira, S., Precup, D., Price, S. J., Raviv, T. R., Reza, S. M. S., Ryan, M., Sarikaya, D., Schwartz, L., Shin, H.-C., Shotton, J., Silva, C. A., Sousa, N., Subbanna, N. K., Szekely, G., Taylor, T. J., Thomas, O. M., Tustison, N. J., Unal, G., Vasseur, F., Wintermark, M., Ye, D. H., Zhao, L., Zhao, B., Zikic, D., Prastawa, M., Reyes, M., and Van Leemput, K. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging 34*, 10 (2015), 1993–2024.

[42] Neromyliotis, E., Kalamatianos, T., Paschalis, A., Komaitis, S., Fountas, K. N., Kapsalaki, E. Z., Stranjalis, G., and Tsougos, I. Machine learning in meningioma mri: Past to present. a narrative review. *Journal of Magnetic Resonance Imaging 55*, 1 (2022), 48–60.

[43] Oldenhof, M., Ács, G., and Pejó. Industry-scale orchestrated federated learning for drug discovery. *Proceedings of the AAAI Conference on Artificial Intelligence 37*, 13 (Jul. 2024), 15576–15584.

[44] Pati, S., Baid, U., Edwards, B., Sheller, M. J., Foley, P., Anthony Reina, G., Thakur, S., Sako, C., Bilello, M., Davatzikos, C., Martin, J., Shah, P., Menze, B., and Bakas, S. The federated tumor segmentation (FeTS) tool: an open-source solution to further solid tumor research. *Phys Med Biol 67*, 20 (Oct. 2022).

[45] Phong, L. T., Aono, Y., Hayashi, T., Wang, L., and Moriai, S. Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Transactions on Information Forensics and Security 13*, 5 (2018), 1333–1345.

[46] Poulen, G., Vignes, J.-R., Le Corre, M., Loiseau, H., and Bauchet, L. Who grade ii meningioma: Epidemiology, survival and contribution of postoperative radiotherapy in a multicenter cohort of 88 patients. *Neurochirurgie 66*, 2 (2020), 73–79.

[47] Pratt, J. W., and Gibbons, J. D. *Kolmogorov-Smirnov Two-Sample Tests*. Springer New York, New York, NY, 1981, pp. 318–344.

[48] Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., Bakas, S., Galtier, M. N., Landman, B. A., Maier-Hein, K., et al. The future of digital health with federated learning. *NPJ digital medicine 3*, 1 (2020), 1–7.

[49] Scherer, J., Nolden, M., Kleesiek, J., Metzger, J., Kades, K., Schneider, V., Bach, M., Sedlaczek, O., Bucher, A. M., Vogl, T. J., Grünwald, F., Kühn, J.-P., Hoffmann, R.-T., Kotzerke, J., Bethge, O., Schimmöller, L., Antoch, G., Müller, H.-W., Daul, A., Nikolaou, K., la Fougère, C., Kunz, W. G., Ingrisch, M., Schachtner, B., Ricke, J., Bartenstein, P., Nensa, F., Radbruch, A., Umutlu, L., Forsting, M., Seifert, R., Herrmann, K., Mayer, P., Kauczor, H.-U., Penzkofer, T., Hamm, B., Brenner, W., Kloeckner, R., Düber, C., Schreckenberger, M., Braren, R., Kaissis, G., Makowski, M., Eiber, M., Gafita, A., Trager, R., Weber, W. A., Neubauer, J., Reisert, M., Bock, M., Bamberg, F., Hennig, J., Meyer, P. T., Ruf, J., Haberkorn, U., Schoenberg, S. O., Kuder, T., Neher, P., Floca, R., Schlemmer, H.-P., and Maier-Hein, K. Joint imaging platform for federated clinical data analytics. *JCO Clin Cancer Inform 4* (Nov. 2020), 1027–1038.

[50] Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., and Batra, D. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR abs/1610.02391* (2016).

[51] Sheller, M., Edwards, B., Anthony Reina, G., Martin, J., and Bakas, S. Nimg-68. federated learning in neuro-oncology for multi-institutional collaborations without sharing patient data. *Neuro. Oncol. 21*, Supplement_6 (Nov. 2019), vi176–vi177.

[52] Sheller, M. J., Edwards, B., Reina, G. A., Martin, J., Pati, S., Kotrotsou, A., Milchenko, M., Xu, W., Marcus, D., Colen, R. R., and Bakas, S. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Sci Rep 10*, 1 (July 2020), 12598.

[53] Shinohara, R. T., Sweeney, E. M., Goldsmith, J., Shiee, N., Mateen, F. J., Calabresi, P. A., Jarso, S., Pham, D. L., Reich, D. S., and Crainiceanu, C. M. Statistical normalization techniques for magnetic resonance imaging. *NeuroImage: Clinical 6* (2014), 9–19.

[54] Shokri, R., and Shmatikov, V. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (New York, NY, USA, 2015), CCS '15, Association for Computing Machinery, p. 1310–1321.

[55] Shorten, C., and Khoshgoftaar, T. M. A survey on image data augmentation for deep learning. *Journal of big data 6*, 1 (2019), 1–48.

[56] Suzuki, K. Overview of deep learning in medical imaging. *Radiological physics and technology 10*, 3 (2017), 257–273.

[57] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. *CoRR abs/1512.00567* (2015).

[58] Valverde, J. M., Imani, V., Abdollahzadeh, A., De Feo, R., Prakash, M., Ciszek, R., and Tohka, J. Transfer learning in magnetic resonance brain imaging: A systematic review. *Journal of Imaging 7*, 4 (Apr. 2021), 66.

[59] van Griethuysen, J. J. M., Fedorov, A., Parmar, C., Hosny, A., Aucoin, N., Narayan, V., Beets-Tan, R., Fillion-Robin, J.-C., Pieper, S. D., and Aerts, H. J. Computational radiomics system to decode the radiographic phenotype. *Cancer research 77 21* (2017), e104–e107.

[60] Xia, Y., Yang, D., Li, W., Myronenko, A., Xu, D., Obinata, H., Mori, H., An, P., Harmon, S., Turkbey, E., Turkbey, B., Wood, B., Patella, F., Stellato, E., Carrafiello, G., Ierardi, A., Yuille, A., and Roth, H. Auto-fedavg: Learnable federated averaging for multi-institutional medical image segmentation, 2021.

[61] Yang, Q., Liu, Y., Chen, T., and Tong, Y. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol. 10*, 2 (jan 2019).

[62] Yarabarla, V., Mylarapu, A., Han, T. J., McGovern, S. L., Raza, S. M., and Beckham, T. H. Intracranial meningiomas: an update of the 2021 world health organization classifications and review of management with a focus on radiation therapy. *Front Oncol 13* (Aug. 2023), 1137849.

[63] Yeganeh, Y., Farshad, A., Navab, N., and Albarqouni, S. Inverse distance aggregation for federated learning with non-iid data. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning: Second MICCAI Workshop, DART 2020, and First MICCAI Workshop, DCL 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4–8, 2020, Proceedings 2* (2020), Springer, pp. 150–159.

[64] Yoshida, N., Nishio, T., Morikura, M., Yamamoto, K., and Yonetani, R. Hybrid-fl for wireless networks: Cooperative learning mechanism using non-iid data. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)* (2020), pp. 1–7.

[65] Yu, F., Rawat, A. S., Menon, A., and Kumar, S. Federated learning with only positive labels. In *Proceedings of the 37th International Conference on Machine Learning* (13–18 Jul 2020), H. D. III and A. Singh, Eds., vol. 119 of *Proceedings of Machine Learning Research*, PMLR, pp. 10946–10956.

[66] Zech, J. R., Badgeley, M. A., Liu, M., Costa, A. B., Titano, J. J., and Oermann, E. K. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLOS Medicine 15*, 11 (11 2018), 1–17.

[67] Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., and Gao, Y. A survey on federated learning. *Knowledge-Based Systems 216* (2021), 106775.

[68] Zhang, H., Mo, J., Jiang, H., Li, Z., Hu, W., Zhang, C., Wang, Y., Wang, X., Liu, C., Zhao, B., Zhang, J., and Zhang, K. Deep learning model for the automated detection and histopathological prediction of meningioma. *Neuroinformatics 19*, 3 (Jul 2021), 393–402.

[69] Zhang, H., and Qie, Y. Applying deep learning to medical imaging: A review. *Applied Sciences 13*, 18 (2023).

[70] Zhang, M., Qu, L., Singh, P., Kalpathy-Cramer, J., and Rubin, D. L. Splitavg: A heterogeneity-aware federated deep learning method for medical imaging. *IEEE Journal of Biomedical and Health Informatics 26*, 9 (2022), 4635–4644.

[71] Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. Federated learning with non-iid data. *ArXiv abs/1806.00582* (2018).

[72] Zhu, H., Xu, J., Liu, S., and Jin, Y. Federated learning on non-iid data: A survey. *Neurocomputing 465* (2021), 371–390.