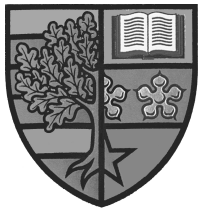
Novelty, Distillation, and Federation in Machine Learning for Medical Imaging

by

Matthew Daykin

Submitted in conformity with the requirements
for the degree of Doctor of Engineering

Heriot Watt University



School of Engineering and Physical Sciences

Submitted April 2020

Abstract

The practical application of deep learning methods in the medical domain has many challenges. Pathologies are diverse and very few examples may be available for rare cases. Where data is collected it may lie in multiple institutions and cannot be pooled for practical and ethical reasons. Deep learning is powerful for image segmentation problems but ultimately its output must be interpretable at the patient level. Although clearly not an exhaustive list, these are the three problems tackled in this thesis.

To address the rarity of pathology I investigate novelty detection algorithms to find outliers from normal anatomy. The problem is structured as first finding a low-dimension embedding and then detecting outliers in that embedding space. I evaluate for speed and accuracy several unsupervised embedding and outlier detection methods. Data consist of Magnetic Resonance Imaging (MRI) for interstitial lung disease for which healthy and pathological patches are available; only the healthy patches are used in model training.

I then explore the clinical interpretability of a model output. I take related work by the Canon team — a model providing voxel-level detection of acute ischemic stroke signs — and deliver the Alberta Stroke Programme Early CT Score (ASPECTS, a measure of stroke severity). The data are acute head computed tomography volumes of suspected stroke patients. I convert from the voxel level to the brain region level and then to the patient level through a series of rules. Due to the real world clinical complexity of the problem, there are at each level — voxel, region and patient — multiple sources of "truth"; I evaluate my results appropriately against these truths.

Finally, federated learning is used to train a model on data that are divided between multiple institutions. I introduce a novel evolution of this algorithm — dubbed "soft federated learning" — that avoids the central coordinating

authority, and takes into account domain shift (covariate shift) and dataset size. I first demonstrate the key properties of these two algorithms on a series of MNIST (handwritten digits) toy problems. Then I apply the methods to the BraTS medical dataset, which contains MRI brain glioma scans from multiple institutions, to compare these algorithms in a realistic setting.

Dedication

This thesis is dedicated to everyone.

Yes. Everyone. Now, before, and after.

Because it was the actions of the people in the past that enabled the opportunities for me to produce this research.

It is the people here now who helped me through every difficulty while completing this doctorate, and these people are shaped by those they are in contact with (and in turn they are shaped by their contacts). In this connected world, everyone has some impact.

And it will be the people of the future that bring out the potential of this thesis and build upon it further. For the good of everyone.

Dedicated to the world...

Acknowledgements

I am immensely indebted to my industrial supervisor, Dr. Ian Poole, whose dedication and expertise have helped at every stage of this doctorate. I would not have achieved anything near to what I have without his help and extensive discussions.

I also would like to thank my academic supervisor, Dr Mathini Sellathurai, for her commitment to the program, useful comments throughout, and especially for her reviews of this thesis in the final year.

I am grateful to my doctoral centre, The Centre for Doctoral Training in Applied Photonics, for their funding and constant support and training; and to the Engineering and Physical Sciences Research Council for their funding and resources.

My thanks also extend to my host company, Canon Medical Research Europe and their Artificial Intelligence Centre of Excellence, for providing the funding and hardware required for this degree. Thank you to the teams I primarily worked with — Image Analysis and Artificial Intelligence Research — for stimulating discussions and help when I needed it.

Further, I am grateful to Alison O'Neil for creating the subsampling code for the lung data, Keith Muir for providing the data for ASPECTS, Aneta Lisowska for producing the model used in the ASPECTS work, Erin Beveridge for being the clinical expert in the ASPECTS methods, creating the ASPECTS atlas, and helping understand early ischemic stroke signs, Corné Hoogendoorn for the discussions around the early theories for soft federated learning, Keith Goatman for helping with the survey in the federated learning chapter, all of the clinical staff who took the time to provide insights in the survey, Richard Moffett and Shadia Mikhael for providing clinical understanding, Vismantas Dilys for aiding with advanced programming and software

engineering throughout, and, of course, my family for supporting me on my doctoral journey.

Finally, I would like to thank all of the many, many people who offered the brilliant advice of writing chapters of your thesis as you go along rather than waiting until the end of you doctorate. This was fantastic advice that I wish I had followed(!).

Declaration

The material contained within this thesis has not previously been submitted for a degree at Heriot Watt University or any other university. The research reported within this thesis has been conducted by the author unless indicated otherwise.

Copyright Notice

Research Thesis Submission

Please note this form should be bound into the submitted thesis.

Name:	Matthew Daykin		
School:	EPS (Engineering and Physical Sciences)		
Version: *(i.e. First, Resubmission, Final)*	Final	Degree Sought:	EngD (Engineering Doctorate)

Declaration

In accordance with the appropriate regulations I hereby submit my thesis and I declare that:

1. The thesis embodies the results of my own work and has been composed by myself
2. Where appropriate, I have made acknowledgement of the work of others
3. The thesis is the correct version for submission and is the same version as any electronic versions submitted*.
4. My thesis for the award referred to, deposited in the Heriot-Watt University Library, should be made available for loan or photocopying and be available via the Institutional Repository, subject to such conditions as the Librarian may require
5. I understand that as a student of the University I am required to abide by the Regulations of the University and to conform to its discipline.
6. I confirm that the thesis has been verified against plagiarism via an approved plagiarism detection application e.g. Turnitin.

ONLY for submissions including published works

Please note you are only required to complete the Inclusion of Published Works Form (page 2) if your thesis contains published works)

7. Where the thesis contains published outputs under Regulation 6 (9.1.2) or Regulation 43 (9) these are accompanied by a critical review which accurately describes my contribution to the research and, for multi-author outputs, a signed declaration indicating the contribution of each author (complete)
8. Inclusion of published outputs under Regulation 6 (9.1.2) or Regulation 43 (9) shall not constitute plagiarism.

* *Please note that it is the responsibility of the candidate to ensure that the correct version of the thesis is submitted.*

Signature of Candidate:	M. Daykin	Date:	12th April 2020

Submission

Submitted By *(name in capitals):*	MATTHEW DAYKIN
Signature of Individual Submitting:	M. Daykin
Date Submitted:	22nd April 2020

For Completion in the Student Service Centre (SSC)

Limited Access	Requested	Yes		No		Approved	Yes		No	
*E-thesis Submitted (**mandatory for final theses)***										
Received in the SSC by *(name in capitals):*						Date:				

HERIOT WATT UNIVERSITY

Inclusion of Published Works

Please note you are only required to complete the Inclusion of Published Works Form if your thesis contains published works under Regulation 6 (9.1.2)

Declaration

This thesis contains one or more multi-author published works. In accordance with Regulation 6 (9.1.2) I hereby declare that the contributions of each author to these publications is as follows:

Citation details	M. Daykin, E. Beveridge, V. Dilys, A. Lisowska, K. Muir, M. Sellathurai, and I. Poole, ``Evaluation of an Automatic ASPECT Scoring System for Acute Stroke in Non-Contrast CT'', in Proceedings of Annual Conference on Medical Image Understanding and Analysis, (Springer, Cham, 2017), pp. 537-547.
Author 1 – M. Daykin	Design, implementation, and evaluation of the ASPECTS algorithm. Modifications and fitting of the atlas.
Author 2 – E. Beveridge	Design of the initial atlas. Clinical support.
Author 3 – V. Dilys	Support on code and concept development.
Author 4 – A. Lisowska	Design of the initial neural network.
Author 5 - K. Muir	Data supplier.
Author 6 - M. Sellathurai	Supervision.
Author 7 – I. Poole	Supervision.
Signature:	M. Daykin
Date:	24th September 2019

Citation details	A. Lisowska, A. O'Neil, V. Dilys, M. Daykin, E. Beveridge, K. Muir, S. McLaughlin, I. Poole, ``Context-Aware Convolutional Neural Networks for Stroke Sign Detection in Non-Contrast CT Scans'', in Proceedings of In Annual Conference on Medical Image Understanding and Analysis, (Springer, Cham, 2017), pp. 494-505.
Author 1 – A. Lisowska	Model design, implementation, and evaluation.
Author 2 – A. O'Neil	Technical support.
Author 3 – V. Dilys	Support on code development.
Author 4 - M. Daykin	Support on model design, implementation, and evaluation.
Author 5 - E. Beveridge	Clinical support.
Author 6 - K. Muir	Data supplier.
Author 7 - S. McLaughlin	Supervision

Author 8 – I. Poole	Supervision.
Signature:	M. Daykin
Date:	24th September 2019

Citation details	M. Daykin, M. Sellathurai, and I. Poole, ``A Comparison of Unsupervised Abnormality Detection Methods for Interstitial Lung Disease'', In Annual Conference on Medical Image Understanding and Analysis, (Springer, Cham, 2018). pp. 287-298.
Author 1 – M. Daykin	Implementation and evaluation.
Author 2 – M. Sellathurai	Supervision.
Author 3 – I. Poole	Supervision and technical support.
Signature:	M. Daykin
Date:	24th September 2019

Contents

List of Tables

List of Figures

Abbreviations

1-SVM	One-Class Support Vector Machine
2D	Two-Dimensional
3D	Three-Dimensional
ABCD	Age, Blood pressure, Clinical features, Duration
AUC	Area Under Curve
AI	Artificial Intelligence
ASPECTS	Alberta Stroke Programme Early CT Score
BG	Background (ground truth)
BraTS	Brain Tumour Segmentation (dataset)
cAE	Convolutional Autoencoder
CFL	Conventional Federated Learning
CNN	Convolutional Neural Network
CSF	Cerebrospinal Fluid
CT	Computed Tomography
ED	(peritumoral) Edema
ET	(gadolinium-) Enhancing Tumour
fAE	Flat Autoencoder
FLAIR	Fluid-Attenuated Inversion Recovery
FMCD	Fast-Minimum Covariance Determinant (estimator)
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GPU	Graphical Processing Unit
HGG	High Grade Glioma
HU	Hounsfield Units
IADL	Instrumental Activities of Daily Living
ID	Identification
IF	Isolation Forest
Inf	Infinity/Infinite
ILD	Interstitial Lung Disease
kPCA	Kernel Principal Component Analysis

Abbreviations

kNN	k-Nearest Neighbours
LGG	Low Grade Glioma
LOF	Local Outlier Factor
MCA	Middle Cerebral Artery
MNIST	Modified National Institute of Standard and Technology (handwritten digits dataset)
MRI	Magnetic Resonance Imaging
MVG	Multivariate Gaussian
NaN	Not a Number
NCCT	Non-Contrast Computed Tomography
NCR	Necrotic (tumour core)
NET	Non-Enhancing Tumour (core)
NF	Novelty Forest
NHS	(United Kingdom) National Health Service
NIHSS	National Institutes of Health Stroke Scale
PCA	Principal Component Analysis
PR	Precision-Recall
ReLU	Rectified Linear Unit
ROC	Receiver Operating Characteristic
SFL	Soft Federated Learning
SGD	Stochastic Gradient Descent
TIA	Transient Ischemic Attack
TN	True Negative
TP	True Positive
TPR	True Positive Rate
STAPLE	Simultaneous Truth and Performance Level Estimation

List of Publications

Year	Type	Reference
2016	Poster	**M. Daykin**, A. Michalska, C. Reines, G. Flockhart, "Optical Communications Line with a Moving Target", presented at the CDT in Applied Photonics Annual Conference 2016.
2017	Internal report	**M. Daykin** and I. Poole, "Novelty Forests: A reworking of probability density forests, fit for anomaly detection", Internal report (May 2017).
2017	Peer-reviewed conference proceedings	**M. Daykin**, E. Beveridge, V. Dilys, A. Lisowska, K. Muir, M. Sellathurai, and I. Poole, "Evaluation of an Automatic ASPECT Scoring System for Acute Stroke in Non-Contrast CT", in Proceedings of Annual Conference on Medical Image Understanding and Analysis, (Springer, Cham, 2017), pp. 537-547.
2017	Peer-reviewed conference proceedings	A. Lisowska, A. O'Neil, V. Dilys, **M. Daykin**, E. Beveridge, K. Muir, S. McLaughlin, I. Poole, "Context-Aware Convolutional Neural Networks for Stroke Sign Detection in Non-Contrast CT Scans", in Proceedings of In Annual Conference on Medical Image Understanding and Analysis, (Springer, Cham, 2017), pp. 494-505.
2017	Conference poster	**M. Daykin**, E. Beveridge, V. Dilys, A. Lisowska, K. Muir, M. Sellathurai, and I. Poole, "Evaluation of an Automatic ASPECT Scoring System for Acute Stroke in Non-Contrast CT", SINAPSE Conference (June 2017).
2017	Conference poster	**M. Daykin**, E. Beveridge, V. Dilys, A. Lisowska, K. Muir, M. Sellathurai, and I. Poole, "Applying a Convolutional Neural Network to generate ASPECTS in Non-Contrast CT", TransMed Annual CDT Conference (June 2017).

List of Publications

Year	Type	Reference
2017	Conference poster	**M. Daykin**, E. Beveridge, V. Dilys, A. Lisowska, K. Muir, M. Sellathurai, and I. Poole, "Detecting Ischemic Stroke in Non-Contrast CT Volumes Using Machine Learning", CDT in Applied Photonics Annual Conference (June 2017).
2018	Peer-reviewed conference proceeding	**M. Daykin**, M. Sellathurai, and I. Poole, "A Comparison of Unsupervised Abnormality Detection Methods for Interstitial Lung Disease", In Annual Conference on Medical Image Understanding and Analysis, (Springer, Cham, 2018). pp. 287-298.
2018	Internal report	**M Daykin**, M Falis, K. Goatman. Learning Healthcare: First Steps. Technical report. 2018
2018	Conference presentation	**M Daykin**, "Federated Learning: Next generation artificial intelligence", CDT in Applied Photonics Annual Conference (June 2018).
2019	Conference presentation	**M Daykin**, "Federated Learning for Medical Imaging Data", CDT in Applied Photonics Annual Conference (June 2019).
2019	Patent	**M Daykin** and I. Poole, Soft Federated Learning, Patent, 2019 (Granted)

Chapter 1

Introduction

Healthcare is facing the perfect storm of staff shortages and aging populations. In 2020 the number of people aged 60+ will outnumber children under 5 globally, and by 2050 they will make up 22% of the global population [237]. The elderly are more likely to be frail or suffer chronic health problems than a younger population, which leads to an increased burden on healthcare services. By the age of 65, most people will have at least one chronic health illness and by 75 they will have two [24], and yet, there is a shortage of seven million health workers worldwide [169, 253, 295]. In the UK the gap between the number of healthcare staff needed and the number being recruited is unsustainable [196, 216, 244], leading to a national emergency [25] with staff shortages looking to double between 2019 and 2024 without radical change [134].

Artificial intelligence (AI) — a broad term for machine learning — has the potential to have a profound, positive, and transformative impact on the health of European and worldwide populations [106] with its use within the National Health Service (NHS) [217] saving up to £12.5 billion a year worth in staff time — or about 10% of the NHS budget [102] — easing the pressures of

staff shortages [113]. AI could change healthcare as we know it with machines making critical decisions on behalf of clinical staff [66]. Hospitals produce 50 petabytes of data every year with 90% of this being medical imaging, but 97% of this goes unused [185]. AI can process these data faster than a radiologist and thus save them time and extract more use from the data.

Machine learning provides a way of automating or aiding in clinical work, but this too faces its own set of challenges. For instance, both pathological and healthy anatomy are highly diverse and, in the case of rare or new diseases, prior examples can be difficult to come across. Medical data are highly sensitive and cannot be easily pooled for machine learning training. Outputs from machine learning methods are often not understandable by humans and must be converted into a form that is. This can be done by distilling the high-dimensional output into a simple, small set of outcomes. This thesis aims to demonstrate solutions to these problems.

For common pathology where many imaging examples from varied sources exist, a wide range of machine learning techniques are available with neural networks providing state-of-the-art in image analysis [123]. There are many neural network methods focusing within the medical field [13, 82, 86, 107, 150, 160, 181, 208, 298]. However, the existence of sufficient data and suitable machine learning algorithms does not equal easy application development. The data are controlled within a clinical environment. Technical and ethical issues prevent the data leaving for a research environment. Laws such as the Data Protection Act 2018 [284], General Data Protection Regulation 2016 [54], and, in some cases, the Health Insurance Portability and Accountability Act 1996 [225] prevent sharing or use of the data without costly and complicated contracts. Despite an increasing regulatory environment, in 2019 the NHS financially encouraged the use of artificial intelligence systems to replace clinical

tasks conventionally performed by humans [141].

1.1 Medical Imaging Data

Through its complexity and size, medical imaging data are a difficult but rich and rewarding type of data to work with due to the extensive information they contain. This thesis focuses on the use of CT and MRI data in the exploration of the aforementioned problems. This section provides an overview of these imaging techniques.

1.1.1 CT

Computed tomography (CT) is a technique to form slicewise volume rendering of organs and structures within the body. It does this by passing x-rays through the body from multiple angles and recording their absorption. The projections from multiple angles are captured in a sinogram where the projection there is the Radon transform of the volume. This projection can then be converted back into a volume by the inverse Radon transform [243, 242]. The voxel sizes in the resulting volume are dependent on the number of angles sampled and the resolution of the x-ray detector.

The Hounsfield scale is used to measure absorption in Hounsfield Units (HU), which are standardised against the absorption of air and distilled water at standard temperature and pressure. Air is defined as -1000 HU and water as zero HU. Thus one HU represents a change of 0.1% of the attenuation coefficient of water. Table 1.1 defines the HU range for common medical materials and Figure 1.1 displays an example CT lung axial slice — note how the bone is bright and air spaces are dark.

Substance	HU Range
Fat	-120 to -90
Cancellous Bone	+300 to +400
Cortical Bone	+1800 to +1900
Unclotted Bood	+13 to +50
Clotted Blood	+50 to +75
Pleural Effusion	+2 to +33
Cerebrospinal Fluid	+15
Lung	-700 to -600
Kidney	+20 to +45
Liver	+54 to +66
Lymph Nodes	+10 to +20
Muscle	+35 to +55
Brain White Matter	+20 to +30
Brain Grey Matter	+37 to +45
Glass (foreign body)	+500
Rocks (foreign body)	+2100 to +2300
Copper (foreign body)	+14 000
Steel (foreign body)	+30 000

Table 1.1: The HU ranges of various structures found within the human body. Data from a range of sources: [64, 88, 91, 148, 168, 176, 188, 228, 305].

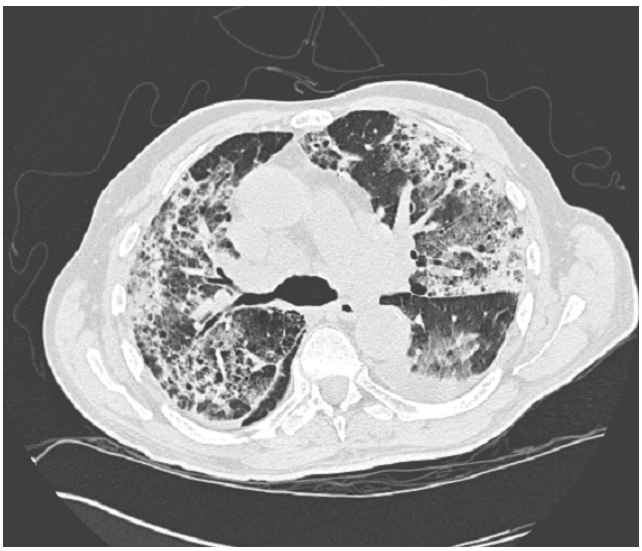


Figure 1.1: Example of a CT axial slice of diseased lungs. The lungs are the darker regions on either side of the centre of the image, the white regions surrounding are bone with the spine at the bottom, and the dark outer region is the region outside of the body (air). The presence of brighter regions (non-air) within the lungs is indicative of disease. Chapter 3 expands on this further.

1.1.2 MRI

MRI — magnetic resonance imaging — utilises the magnetisation properties of atoms within the patient. A powerful magnetic field (3 or 7 Tesla in modern scanners) is applied that aligns the spins of protons along the direction of the applied field. These protons are normally oriented randomly. Most protons are aligned in the direction of the field, however some are aligned opposing it. A second, weaker magnetic field is applied to produce a magnetic gradient across the imaging axis of the patient.

A third energy field - usually electromagnetic radiation in the radio frequency spectrum - is then applied in the form of a short pulse with the frequency of the waves being set to the resonant frequency of the protons in a specific slice of the imaging axis. This frequency, also known as the Larmor frequency, is the product of the magnetic field strength and the gyromagnetic

ratio (a constant for each particle) [78], and hence the gradient of the field allows localisation of the slice. Protons whose spin is aligned with the magnetic field can absorb this energy to flip to become aligned against the field in a higher energy state. The energy of the pulse is set to flip enough protons such that 50% are aligned with the field and 50% oppose it. The application of this field also aligns the spins of all protons, such that they are now in phase.

The result of these two effects is a transverse magnetisation that can be detected. As the pulse ends, the protons gradually fall back into their original state. This happens in two ways. First the protons' spins fall out of phase. This is due to the positive charge of protons repelling each other. This is called the T2 or spin-spin relaxation. As this happens the transverse magnetisation decreases to zero. Next the protons in the high energy flipped state flip back to being aligned with the magnetic field. This is called T1 or spin-lattice relaxation.

To measure the T1 and T2 properties of materials and thus differentiate them, the time between pulses and the delay before measure of the transverse magnetisation is varied. To measure T1 properties, a second pulse is applied shortly after the first pulse, and then the transverse magnetisation signal is recorded almost immediately after this. Molecules whose protons have been able to relax fully in the time between the pulses will absorb the new pulse to create a strong transverse magnetisation (due to 50% of the spins being aligned each way). However, if the protons are still in the excited state when the new pulse arrives, more unexcited protons will absorb the energy and flip to the excited state. This leads to there being more than 50% of the protons now spin-aligned against the magnetic field, which weakens the overall transverse signal.

To measure T2 properties no second pulse is used. Instead the transverse

signal is measured after some time after the original pulse. The strength of the signal informs how in phase the spins are and therefore their T2 properties.

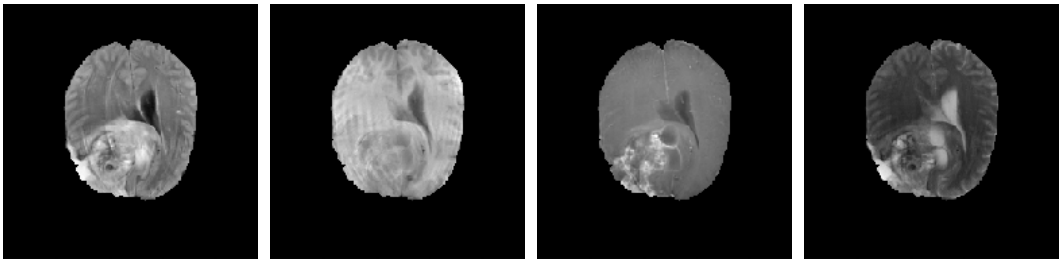


Figure 1.2: Examples of MRI brain slices of a patient with low-grade glioma (a type of brain tumour). From left: FLAIR, T1, T1 contrast enhanced, and T2 imaging. Each modality shows the same slice with the glioma located in the lower left portion. See Chapter 5 for further details.

T1 imaging can be enhanced with the use of a contrast agent — a paramagnetic agent administered to modify the T1 properties of protons to improve the signal.

A third type of imaging exists called FLAIR (fluid-attenuated inversion recovery). This uses a long delay between pulses and between the pulse and recording the signal. Its purpose is to suppress the signal from water and cerebrospinal fluid (CSF) for clearer imaging in organs such as the brain.

Tissue	T1 Appearance	T2 Appearance	FLAIR Appearance
CSF	Dark	Bright	Dark
White Matter	Light	Dark Grey	Dark Grey
Grey Matter	Grey	Light Grey	Light Grey
Bone Marrow Fat	Bright	Light	Light
Demylinated Tissue	Dark	Bright	Bright

Table 1.2: The general appearance of head tissue for T1, T2, and FLAIR MRI scans [258].

1.2 Machine Learning

Machine learning is a wide collection of techniques that take a dataset and produce some output of useful information about the data [97]. This is called training. Once trained, the technique is able to take novel data (not seen before) and provide information about it based on what it has learned during training. In this work I cover unsupervised clustering and supervised classification machine learning.

In the unsupervised approach, the machine learning technique learns to cluster its training data in the sample space. Then, novel data samples are compared to their nearest cluster to determine how similar or different each sample is. This is useful for abnormality detection.

For my supervised approaches, the training data is fed in with ground truth for each sample. This ground truth is some accurate set of information about the sample and is manually added to the data by a human. For instance, in a CT scan it could be a label attached to each voxel indicating the presence or absence of a disease; or for a sampled region, it could be a single label covering

the entire region stating if a disease is present or not.

During training in a supervised approach, the machine learning technique learns to map the data samples to the ground truth. Then, as novel data are fed in (without ground truth), the trained technique can provide the ground truth for them, and thus from this allows us to understand what the data is showing. For instance, the technique could tell us the region of disease within a CT dataset. The technique chooses the ground truth by comparing the samples to the training data to see what ground truth is the best match.

1.3 Research History

Research into machine learning for medical imaging data has existed since the late 1980s and 1990s [33, 147, 173, 193, 203, 210, 286], although it was around 2006 that deep models suitable for complex medical data were designed [127]. Over the past 5 – 10 years advancements in graphical processing units have allowed for rapid prototyping of models of sufficient capacity for medical data thus reducing barriers to development.

With this overcoming of major barriers, the field has experienced a rapid increase in the number of publications of medical imaging data. Figure 1.3 shows the number of matches on Google Scholar [104] for the search term $<<$ *"medical imaging" "machine learning"* $>>$ over the past 30 years from 1988 to 2019. The number of publications exceeded 10 000 in 2018 and has increased at a rate of an order of magnitude every ten years during this time period.

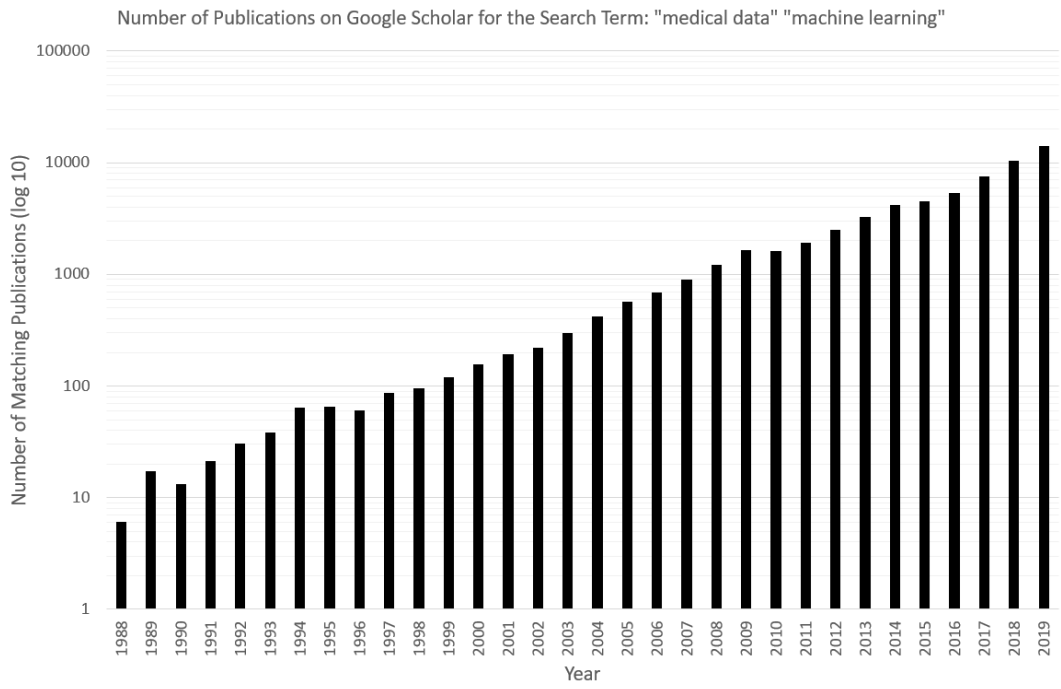


Figure 1.3: Log scale of the number of publications found on Google Scholar [104] for the search term '*"medical imaging" "machine learning"*' each year from 1988 to 2019.

1.4 Taking it Further

This thesis has three key contributions that advance machine learning in medical imaging further by tackling the challenges at the beginning of this section. This section explains the high level contributions in overview, while Section 1.5 discusses the chapters in detail.

1. I test five abnormality detection machine learning methods on weakly labelled medical data (CT interstitial lung disease) to compare their speed and accuracy and derive useful results that could be applicable to other medical imaging problems (Chapters 2 and 3). This addresses the issue of dealing with rare pathology where training examples for that pathology do not exist.

2. I next take a supervised classifier's voxel-level output on acute ischemic stroke head CT and process this into a form suitable for a medical application — the noting of presence/absence of ischemic stroke signs in different regions in the brain — in Chapter 4. This form effectively distills the information from the classifier's output, and thus the output becomes interpretable at a patient level. This addresses distillation of a model output.

3. Finally, I explore ways around some of the data sharing laws by removing the need to move data around. This is achieved using a federated learning training algorithm in Chapter 5. This enables wider access to data and therefore easier development of machine learning applications in the industry. I introduce soft federated learning, a novel evolution of the conventional federated learning algorithm, and compare it to conventional federated learning and a number of baseline measures.

Where work has been previously published, it has been reproduced in this thesis with permission from the publisher.

1.5 Chapter Overview

This section provides a detailed overview of each research chapter of this thesis, starting from the first research chapter introducing the Novelty Forest and progressing in order through to the chapter on Federated Learning near the end of the thesis.

Chapter 2: Here I introduce the novelty forest, a novel form of the decision forest — a binary tree ensemble approach — which efficiently clusters data by using a subset of the features within the dataset. This is per-

formed in an unsupervised manner for abnormality detection. I scrutinise the behaviour of the novelty forest using Gaussian distributions and the Modified National Institute of Standard and Technology (MNIST) handwritten digits dataset to demonstrate its strengths and weaknesses, particularly with regards to its node division and growing behaviour. Success and failure examples are discussed and I show that the novelty forest is susceptible to noise.

Chapter 3: This chapter takes the novelty forest along with four other unsupervised classification techniques and using CT interstitial lung disease data extracted as either healthy or pathological patches — an abnormality detection problem — I compare the accuracy and runtime speed of these five methods on a range of data embeddings.

The new methods are the one-class support vector machine, isolation forest, local outlier factor, and fast-minimum covariance determinant estimator, which each represent a different family of techniques. The embeddings I use are the salient components from principal component analysis and its kernel form, and the embeddings from the condensed layer of a flat autoencoder and a convolutional autoencoder. The data consist of patches of healthy, emphysema, fibrosis, ground glass opacity, and micronodule pathology.

Chapter 4: I explore a specialised convolutional neural network model on non-contrast head CT of suspected ischemic stroke patients. This model aims to highlight areas of early ischemic change. Non-contrast CT (NCCT) is often the first scan that gets taken of a patient due to its speed and availability in ambulances (particularly, mobile stroke units). Therefore being able to classify it accurately is crucially important.

The model produces voxel-level confidence masks of ischemic stroke by comparing the brain hemispheres. Then I use a brain region atlas to deliver the Alberta Stroke Programme Early CT Score (ASPECTS) (a clinically meaningful measure of stroke severity) from the predictions. The atlas is a typical brain volume with the ASPECTS regions overlaid and is rigidly aligned to my other samples. I use rules saying for a region to be classed as ischemic it requires either a certain fraction of its volume as ischemic prediction or it requires an ischemic prediction of any size in any one region. This ensures an accurate conversion from the prediction masks to regions.

This chapter focuses on the conversion of predictions to this score and compares the scores the model's predictions provide to those provided clinically, those from the ground truth segmentations used to train the model, and those by visual examination of the CT volumes by the creator of the ground truth segmentation. This is done at both the volume level and region level.

Chapter 5: I expand on neural network training by using a series of model copies in a federated setup. In this setup, each model copy sees a subset of the total data and can only communicate that data to the other copies through its learned parameters. A regular sharing of parameters is used to develop all models over training cycles. This happens by periodically aggregating the shared parameters into a new model and then replacing all of the current model copies with copies from this new model.

After introducing the concept of federated learning, I evolve it to a new algorithm called soft federated learning that is able to recognise when there are differences between the data each network sees (covariate shift) and can adjust the parameter sharing accordingly to account for these

differences. Soft federated learning does this in two steps. First it discovers how similar these data subsets are by training a model within each one and evaluating it on each subset, including its own. The evaluation happens on data not seen during training and the performance provides a measure of similarity. The second step involves adjusting the parameter aggregation step to now create a unique model for each data subset based on how similar that subset is to others — models from similar subsets are weighted higher than those from dissimilar.

I demonstrate the ability of this new method on transformed MNIST data, and then compare both methods on the Brain Tumour Segmentation (BraTS) medical dataset, which contains Magnetic Resonance Imagine (MRI) brain glioma scans from multiple institutions. The tranformed MNIST data takes the MNIST dataset and applies one of a range of affine transformations or noise to each sample in a data subset, such that each subset has one transformation applied to all of its data. For the BraTS dataset, its splits in source institution provide the model training data subsets.

I show that soft federated learning is effective in a range of crafted scenarios and is able to match the accuracy performance of conventional federated learning on the BraTS medical dataset.

Chapter 2

The Novelty Forest: An Efficient Decision Forest Unsupervised Clustering Technique

2.1 Abstract

The novelty forest, an evolution of the decision forest, is a decision tree ensemble approach for efficient abnormality detection in imaging data. It sparsely evaluates the features in a sample to allow for efficient clustering using a specified information gain. Features in this case are the pixel intensities in a set of images. Once clustered, each sample delivers an abnormality score indicating how well it fits within its cluster. I introduce the novelty forest and explore its behaviour on the MNIST dataset divided into even and odd digits representing normality and abnormality. We discuss issues surrounding the use of its information function and edge cases involving singular covariance matrices

and a large Mahalanobis distance relative to the number of data dimensions.

Success and failure examples are shown and I explore the novelty forest on a simple imaging dataset of handwritten digits to discover how the node division during growth and the stopping criteria affect the accuracy when tested on clean and noisy versions of the images. The noisey versions are generated by adding Gaussian noise to the original images on a per pixel basis. A more detailed evaluation of the novelty forest occurs in the next chapter (Chapter 3).

2.2 Introduction

In this chapter I introduce the Novelty Tree — a novelty (abnormality) detection method based on decision trees [128, 61] and use it as an ensemble method called the Novelty Forest. This efficient method seeks to operate at a low computational cost while not sacrificing accuracy, especially on high-dimensional medical imaging datasets. It does this by utilising a narrow range of features in a dataset — the fewer features, the higher the efficiency. The Novelty Forest is introduced in this chapter and demonstrated on a simple problem. In the next chapter I evaluate it against four common novelty methods.

This chapter starts with an introduction to the concept of decision trees and decision forests (Section 2.3) and a literature overview (Section 2.4) before introducing the Novelty Forest in Section 2.5. Following this, specific design details of the Novelty Forest and its pseudocode are described in Section 2.6, and a proof of concept on simple artificial data is found in Section 2.7. Section 2.8 then notes weaknesses of the Novelty Forest. I then introduce a set of data in Section 2.9 and record the performance of the novelty forest with various growing and stopping criteria on this data in Section 2.10. Finally, discussion

and conclusions are in Sections 2.11 and 2.12.[1]

The novelty forest is used in this chapter on MNIST data to highlight its strengths and shortcomings, and then used in the next chapter (Chapter 3) for medical dataset evaluation alongside several other methods.

2.3 Decision Forests

A decision tree is a structure made up of nodes, each of which make some decision. The nodes can be branch nodes, which connect to two or more deeper nodes — called child nodes; or they can be leaf nodes, which are the terminal points of a tree and do not perform any decision making. The first node of the tree is called the root node, although it will also be a branch node or a leaf node. Figure 2.1 illustrates a simple decision tree with each branch node leading to two child nodes. While a decision tree's branch nodes can feature two or more child nodes, the novelty tree introduced later focuses on binary branching as it is convenient and the most common way of designing a decision tree. Decision trees have been shown to have a better overall performance than a conventional single-stage classifier with the same number of features because the different feature subsets can be selected at different levels within a tree [308].

[1] **Acknowledgement:** The original idea and theory presented in the section introducing the novelty forest (Section 2.5) and the content in Section 2.6.1 was devised solely by Dr. Poole, my industrial supervisor, and was adapted from an internal report written by Dr. Poole [236]. My contribution to this work is the remainder of this chapter.

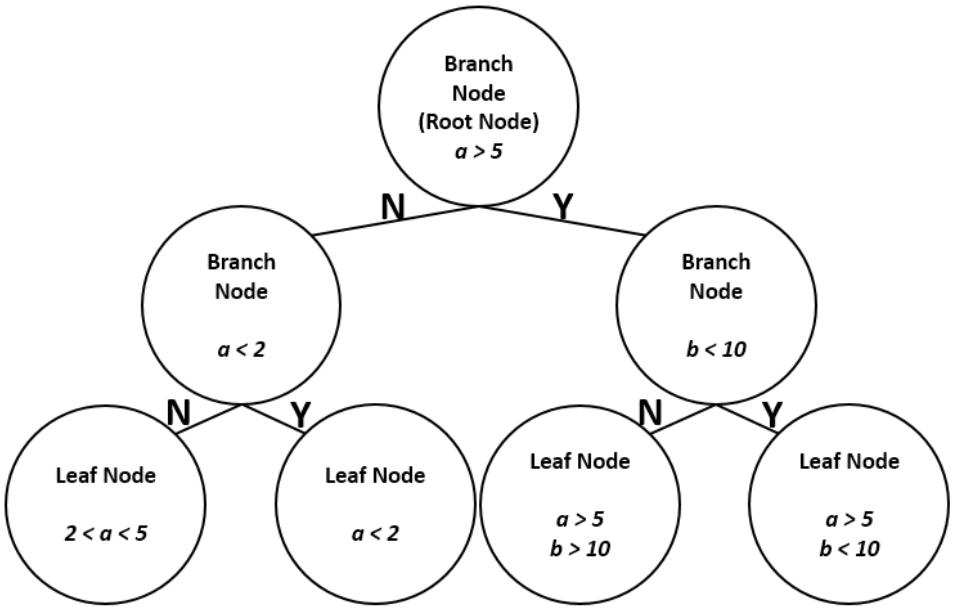


Figure 2.1: A simple 7-node binary decision tree showing the root node, branch nodes with two child nodes each, and leaf nodes. Example decision choices are displayed in the branch nodes where a feature (represented by a or b) is compared against a threshold. The Y/N labels show the Yes/No path to follow based on whether or not a sample follows the decision in that node. The leaf nodes show the set of all decisions that lead to that node. Data samples enter through the root node and are individually processed down the tree until a leaf node is reached.

Decision trees are used to sort and classify data. A data sample begins at the root node and, if this node is a branch node, will be passed to one of the child nodes based on a simple decision. The root node itself can be a leaf node, in which case a data cohort is not divided. This *decide-and-pass-on* method is applied at every branch node the sample encounters until a leaf node is reached.

Decision trees feature a *growth stage* followed by a *prediction stage*. During the growth stage, the tree is created from a set of training data and some predetermined rules on how to process this data. A tree grows from its root node to its leaf nodes. The rules may include the information gain of a decision, or

restrictions on depth, sample size, or tree complexity, and so on. During the prediction stage, samples of unseen data (i.e. data not used during the growth stage) are fed individually into a grown tree, where they then are processed into a leaf node and evaluated.

Decision trees can be collated into an ensemble called a *decision forest* [175]. Each tree in the forest gets a random subset of samples from the training cohort, and so it grows differently. These differences can be structural in the number of nodes and depth of the tree, or decisional in the decision rules the nodes learn[2]. See Figure 2.2 for an example of different (fictional) trees in a forest.

When the forest is used for prediction, each sample is fed through each tree and the outputs — the leaf nodes the samples ends up in — are aggregated for the final result.

[2]To expand on this, the structural differences can also include factors such as the number of child nodes per branch, tree truncation, or any other factor that is outside a node. Decisional differences focus on the nodes themselves, changing how the exact node behaves and processes the data. These two types capture all possible differences between trees.

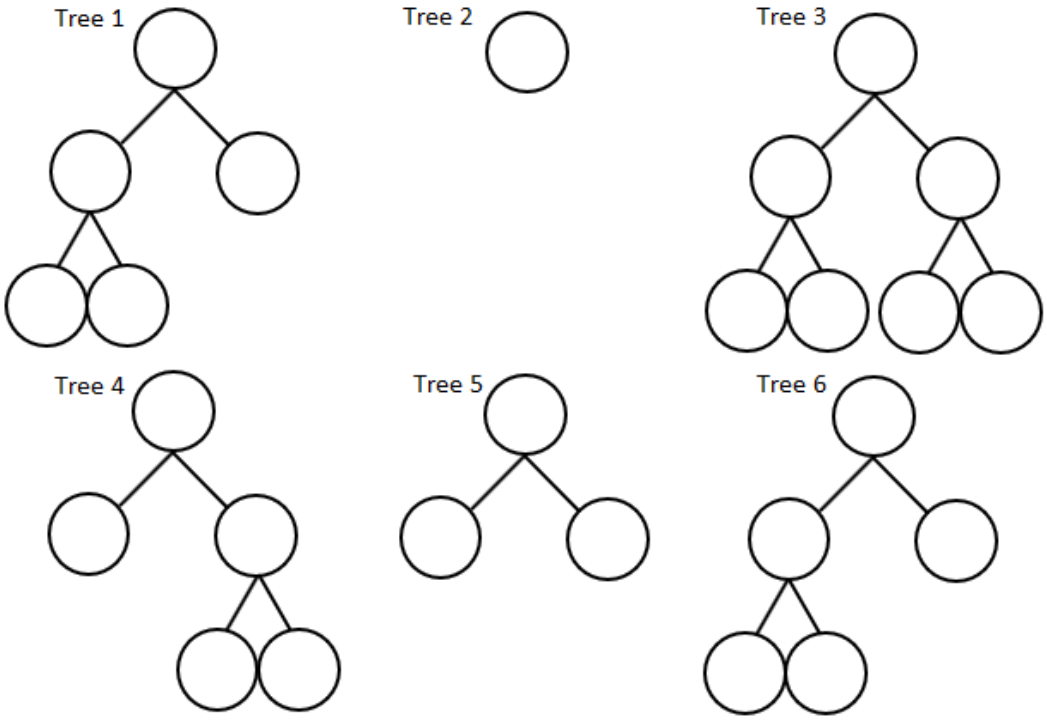


Figure 2.2: A forest consists of multiple trees, which may have a varying structure. Six example trees are shown here, but a forest may have any number. A large forest may be comutationally inefficient, while a small one may risks bias or poor accuracy.

2.4 Previous Work

Decision trees are a effective algorithm for multistage decision making. Common alternatives are table look-up rules [119], which can be inefficient due to checking redundant choices, optimal decision trees [35, 29, 121, 152, 204, 303] that use the final objective of the decision tree to choose its form (as opposed to growing the tree one node at a time maximising the division), but require additional complexity in finding the optimal form, and sequential approaches that evolve on samples one at a time [100, 288].

Optimality criteria for tree design include: minimising the error, having

min-max path length (minimising the maximum number of nodes in path to a decision), achieving a minimum number of nodes or expected length of path, and maximising the information gain [251]. Some trees only use a priori knowledge and do not optimise via any other means. For example, Argentiero *et al.* used a set of 2D canonical transforms and Bayes table look-up decision rules based on a priori information [9]. Similar approaches can be found for noisily printed Chinese character recognition [112], and white blood cell classification [170]. Casey and Nagy used characters with some pixels within each character either on or off (black or white) with some probability. They then used these a priori class probabilities and number of black (on) bits in each pixel position in each class to design the tree to yield a prescribed error probability [46].

Other methods of designing the tree have been proposed. Diday and Moreau used a nearest neighbours hierarchical algorithm that served to cluster nearby (similar) samples into a node, and then clustered nearby nodes into a higher level node, and repeated until all samples were contained within a single node, effectively building the tree from the leaf nodes up to root [75]. Payne and Meisel designed their tree from root to leaf by optimising min-max path length, minimum number of nodes in the tree, and expected path length at each node [230]. The Kolmogorov-Smirnov criterion [155] was used by Rounds to optimise the classification decision at each tree node [248], while Sethi and Sarvarayudu maximised mutual information gain at each node [267]. Scheurmann and Doster showed how using a soft (probabilistic) decision strategy could be used to build a tree [259].

Other interesting research related to decision trees include Sethi who showed the equivalence of a decision tree and a neural network and how to convert from a tree to a neural network [266], Wang and Suen who analysed a range of theories for constructing a tree top-down using entropy as the node opti-

misation task [289], and Criminisi *et al.* provide a good overview of different decision forest designs and optimisations in [60].

2.5 An Introduction to the Novelty Forest

The novelty tree is inspired by the density tree (introduced in [59]; see also Section 2.5.1) and seeks to sparsely evaluate the features in a sample to allow for efficient clustering and evaluation speed. A feature here refers to a single parameter in a data sample — for example, a pixel in an image. Each node looks at a single feature to make a decision. A novelty tree seeks to cluster the training data using as few features as possible to minimise memory usage and improve speed.

The novelty forest – an ensemble of novelty trees – can be thought of as a clustering algorithm followed by abnormality detection.

During test time, an unseen sample is fed through a forest, passing through each tree and, within each, being channeled to the nearest cluster of training data. A z-score distance metric (explained shortly in Section 2.5.2) is determined based on how close the novel sample is to the training cluster, which gives a measure of abnormality. The Novelty Forest is implemented in the Python programming language (version 3.6) [93].

2.5.1 The Density Tree

The density tree is a binary decision tree that learns nodes that maximise some form of information gain at each node to classify the sample space by sample density. All features are evaluated at each node. At each leaf node the samples are modelled as a Gaussian distribution, such that a single density tree can be viewed as a special case of a Gaussian mixture model. Samples

are evaluated against the mean and variance of this Gaussian to deliver an abnormality score.

The novelty tree differs from the density tree in its use of a minimal number of features for the clustering and Gaussian evaluation, making it more efficient. The runtime complexity of the novelty tree is not significantly affected by the number of features in the data, while the density forest has a $\mathcal{O}(fn)$ relation where f is the number of features and n is the number of nodes. This is because each additional feature causes an additional calculation in each node.

2.5.2 Defining the Novelty Tree

The novelty tree's branch nodes learn a threshold-feature pair that seeks to optimise the *information gain*[3] (Equation 2.1) of the data division between exactly two children nodes:

$$I_j = H(\mathcal{S}_j) - \sum_{i=L,R} \frac{|\mathcal{S}_j^i|}{|\mathcal{S}_j|} H(\mathcal{S}_j^i) \tag{2.1}$$

where I_j is the information gain of a split at node j, $|\mathcal{S}_j^i|$ is the number of samples in the left or right ($i = L$ or R) child node, $|\mathcal{S}_j|$ is the total number of samples in the node, and H being the entropy — defined as the log determinant of the covariance matrix:

$$H(\mathcal{S}) = \log(|\Lambda(\mathcal{S})|) \tag{2.2}$$

where Λ is the covariance matrix — in my case this collapses down to the standard deviation as I only evaluate one feature at a time. Since $|\Lambda(\mathcal{S})|$ is related to the volume occupied by a Gaussian distribution, the above criteria partitions a dataset into tight distributions.

[3] "Information gain" is a broad term and the form I use here should not be confused with other common forms - see [96].

During the prediction stage, each sample is fed down a tree to end in a leaf node. The training data in that leaf node is used to deliver an abnormality score. The abnormality score is a measure of how outlying a sample is relative to the others in its leaf node. My abnormality score is the z-score, which is the number of standard deviations a sample is from the mean of all samples within that cluster [161]. Although the z-score can take negative values denoting a sample is below the mean, throughout this work I use the z-score as a distance metric that takes the absolute value of the true z-score (Equation 2.3).

$$z = \left| \frac{x - \mu}{\sigma} \right| \tag{2.3}$$

where z is the z-score, x is a data value being evaluated against a distribution, μ is the distribution's mean, and σ the distribution's standard deviation.

The novelty forest aims to deliver a z-score from the leaf nodes computed by assuming Gaussian distributions for only the features used in the path from the root node to leaf node — all other features are ignored for efficiency purposes. A novelty tree requires only a single feature to be measured at each branch node as opposed to the full feature vector used in decision trees [35].

As the z-score is strictly 1D and the novelty tree will usually deliver multidimensional clusters, I use the multidimensional equivalent of the z-score: Mahalanobis distance [194], and later convert it to the z-score (Section 2.5.3). At each leaf node, the Mahalanobis distance is calculated based on the set of unique features measured in the path from the root node to that leaf node. Equation 2.4 shows the Mahalanobis distance where x is a sample, μ is the mean vector for a multivariate Gaussian (MVG), and Λ^{-1} is the inverse of the covariance matrix for the MVG. T is the transpose [297]. The covariance matrix in the case of the novelty forest consists of only the features used in the path from the root to the leaf node.

$$\mathcal{D}(x;\mu,\Lambda) = \sqrt{(x-\mu)^T\Lambda^{-1}(x-\mu)} \tag{2.4}$$

The novelty tree has one key advantage over the density tree.

- The full mean vector and covariance matrix need not be estimated at all nodes of the novelty tree. The use of every feature dimension in a sample is computationally costly. There is also likely to be redundancy between features. One feature's value may be correlated strongly with another feature. Hence evaluating every feature is an inefficient use of resources. A node should only evaluate the most important features.

2.5.3 Converting from Mahalanobis Distance to Z-Score via χ^2

The Mahalanobis distance is itself a measure of abnormality and works with any number of dimensions. However, its interpretation is dependent on the number of dimensions it is calculated in. I convert it to a z-score via the χ^2 distribution to provide comparable distances. The z-score for a test sample is the mean of the z-score produced by the trees in the novelty forest. The χ^2 distribution comes about because it is the Euclidean distance of normally-distributed independent random variables from the origin for a given number of degrees of freedom.

A k-dimensional MVG distribution with respect to $\mathcal{D}^2$ follows a χ^2_k distribution, where k is the *degrees of freedom parameter* — the number of features in my case. I denote the cumulative χ^2 distribution as $F_k()$. The interpretation of $F_k(\mathcal{D}^2)$ is: *the proportion of the probability mass lying inside the hyper-ellipsoid defined by Mahalanobis distance* $\mathcal{D}$.

By using the inverse of this, it is possible to transform a Mahalanobis distance measured in k-dimensional feature space, $\mathcal{D}_k$, into an equivalent 1-

dimensional Mahalanobis distance, or z-score, z.

$$z(\mathcal{D}_k, k) = \sqrt{F_1^{-1}(F_k(\mathcal{D}_k^2))} \tag{2.5}$$

To explain:

1. A (squared) Mahalanobis distance $\mathcal{D}_k$ is obtained for some k-dimensional pattern vector of interest, based on μ and Λ, as in equation 2.4.
2. I convert this to a cumulative distribution percentile via $F_k()$.
3. Then convert back to a (squared) Mahalanobis distance *appropriate to a 1D Gaussian*, via $F_1^{-1}()$.
4. I obtain a z-score by taking the square root.

The above expression allows us to convert a Mahalanobis distance measured in any number of dimensions into a z-score.

2.6 Design Parameters

2.6.1 Growing a Novelty Forest

Information Gain: The objective function used during the growth stage is a form of information gain that we defined in Equation 2.1. The gain is based on the single feature being tested. This encourages splits which, along the feature being tested, result in child nodes having small standard deviations compared with the parent.

Division Method: We have two division methods for finding the optimal split in a branch node: the best of a selection of random feature/threshold pairs, and an optimised random feature split. The randomness creates a

large degree of diversity between trees in the forest, which is beneficial for the ensemble approach. In my experiments, we trial both to learn their effectiveness.

Random Feature/Threshold : A set of 1000 randomly chosen feature-threshold pairs are evaluated for information gain. The highest gain is chosen for the division.

This technique allows for a high probability of finding a high information gain split, ensuring the most relevant features are used for growing the tree. However, the downside is that the threshold chosen may not be the most optimal for the chosen feature.

Optimised Feature : A randomly chosen feature is evaluated across its range with 1000 equally-spaced interval steps to identify the optimal threshold for information gain. This threshold and feature is selected regardless of the information gain it offers.

The benefit of this technique is that it does not require searching the entire feature/threshold space and instead focuses on achieving a perfect split along a single feature. Although this feature may not be informative, across a forest of trees where each tree randomly selects a feature, the average of all these features should approximate the full feature space. Of course, this technique can fail if a large number of features are uninformative.

Figure 2.3 shows an example of the Random Feature/Threshold method of selecting features and thresholds, while Figure 2.4 shows the Optimised Feature method.

Leaf Size Checking: A practical method of stopping tree growth is ensuring a minimum number of samples in a leaf node. If there are too few sam-

ples, then the covariance matrix and mean will have a large uncertainty, which leads to problems when seeking an accurate z-score during prediction. This is done by setting a minimum value for the number of samples required in a leaf node. This value may be a constant to use throughout the tree (e.g. 100 samples minimum per leaf), or it can be a function of depth (e.g. d^2 minimum samples in a node, where d is the depth of the node). The choice of d^2 arises from the number of values in the covariance matrix, which is a square of the number of features used, which is the depth of the node. With the number of samples increasing with the size of the covariance matrix, the sampling error of the covariance matrix remains constant.

Sample Size and Feature Count: To grow a forest of different trees, each tree must be grown differently. While there is some randomness introduced in the division method, additional variation can be added by growing each tree with a random subsample of samples and features present in the full data. This also reduces the growth and prediction times of each tree as they grow smaller.

Tree Count: Finally, the number of trees in a forest can be adjusted. Having few trees allows for speed and minimal memory usage at the cost of accuracy, while many trees allows z-scores to be determined to higher precision.

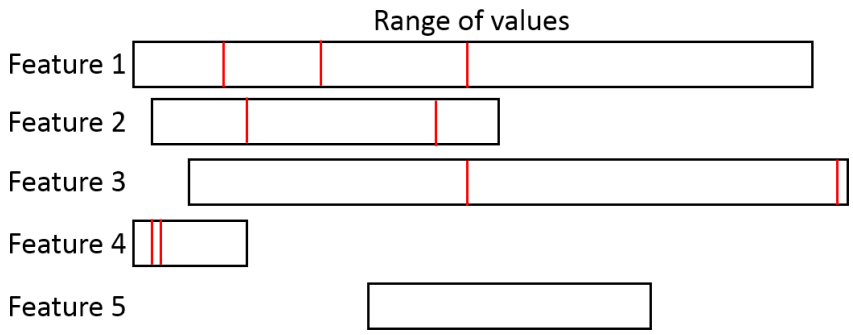


Figure 2.3: An example of Random Feature/Threshold selection for a dataset with five features whose value ranges are represented by the bars. The marks within the bars show ten possible choices for the feature and threshold this method might select.

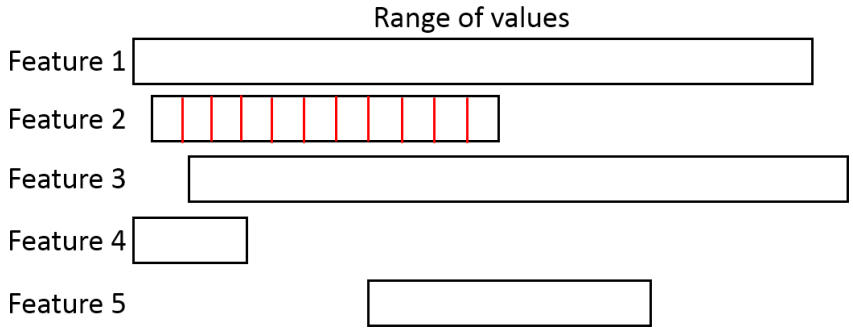


Figure 2.4: An example of Optimised Feature selection for a dataset with five features whose value ranges are represented by the bars. If this method is used to select ten thresholds and Feature 2 was chosen, the marks within this bar show the choices for the threshold.

2.6.2 Pseudocode

Pseudocode for growing and evaluating with a Novelty Forest can be found in Algorithms 1 and 3.

Algorithm 1 Novelty forest growth stage

for Tree in Forest **do**

Subsample samples and features

Call Algorithm 2 createNode(Data, Root Node)

Return set of Trees

Algorithm 2 createNode(Data, Node)

if Stopping criteria met **then**

Set node to leaf node

Return

else

Find threshold and feature to divide

Create Left Child Node

Create Right Child Node

Split Data into Left Node Data and Right Node Data

Call Algorithm 2 createNode(Left Node Data, Left Child Node)

Call Algorithm 2 createNode(Right Node Data, Right Child Node)

Algorithm 3 Novelty forest prediction stage.

for Tree in Forest **do**

for Sample in Test Data **do**

Call Algorithm 4 predictionNode(Sample, Root Node)

Return average z-score for each sample

Algorithm 4 predictionNode(Sample, Node)

```
if Node is Branch Node then
    if Sample at Node.divFeature ≤ Node.divThreshold then
        Append Node.divFeature to Node.featurePath
        Call Algorithm 4 predictionNode(Sample, Left Child Node)
    else
        Append Node.divFeature to Node.featurePath
        Call Algorithm 4 predictionNode(Sample, Right Child Node)
else
    Calculate z-score for Sample using unique Features in Node.featurePath
    Return z-score
```

2.7 Proof of Concept

In this section I explore the behaviour of the Novelty Forest on a series of simple artificial tests using Gaussian distributions.

2.7.1 Dividing Two Gaussians

When presented with a simple one-dimensional case of two sampled Gaussian distributions separated by some distance such that they are fully separated and there is space between them, the novelty forest divides these with a single branch node, such that each child node contains a full Gaussian.

For the random feature/threshold division method, the threshold is selected at random in the empty space between the Gaussians. For the optimised feature method, the division is the lowest evaluated threshold in this gap as this is the first threshold checked that produces maximal information gain.

This behaviour remains when there is a partial overlap between two Gaus-

sians. The midway point between the peaks of the Gaussians is used for the division.

2.7.2 Dividing N Gaussians

In the case of N Gaussians, where $N > 2$, each separated by an equal distance, the novelty forest divides them as above, but the first split is between the two most central Gaussians. The child nodes will contain the same number of Gaussians as each other, or, if the parent data contains an odd number of Gaussians, one child node will have one more Gaussian than the other. This is because the information gain of dividing the Gaussians evenly is higher than for any other division.

2.8 Issues with the Novelty Forest

This section provides an overview of problems that became apparent during early testing of the Novelty Forest. Carefully designed examples are used to show the failings of the algorithm.

2.8.1 Issue 1 - Impossibility of Finding Optimal Information

An optimal feature/threshold split in a dataset can only be found if the distribution for each feature is continuous. When there are gaps in a dataset, which many datasets will have to some extent, a single optimal threshold cannot be identified. This means that even if the optimal feature is found, the precise threshold will be unknown. See Figure 2.5 for an emphasised example using two sampled Gaussians.

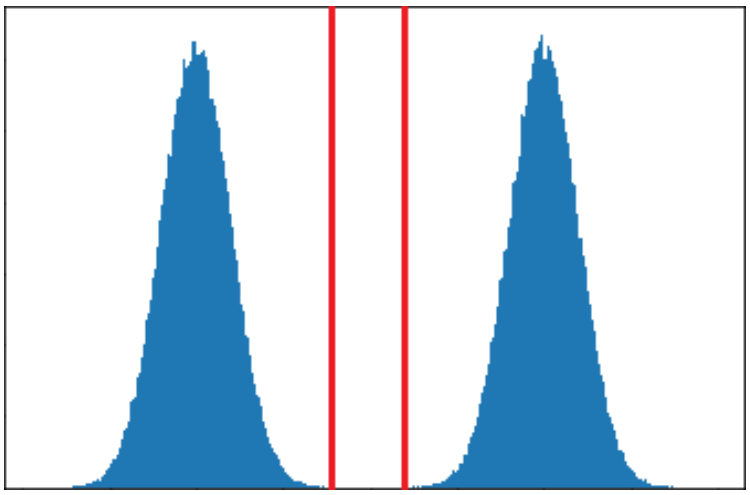


Figure 2.5: Two Gaussians separated by some distance will produce identical (and maximal) information gain for any division that takes place in the space between them - annotated by the central vertical lines.

2.8.2 Issue 2 - Singular Covariance Matrices

During the prediction stage, the covariance matrix of samples in a leaf node may be singular in cases where there is a strong correlation between features, which is problematic during the calculation of Mahalanobis distance (Equation 2.4) as the inverse of the covariance matrix is taken to calculate the distance metric. A singular matrix does not have an inverse because of its zero determinant.

This is resolved by using the *Pseudo Inverse* function [275] from the SciPy library [264]. This allows for the inverse to be taken at a minimal cost to overall speed and works by using a least squares generalisation to solve for the matrix.

2.8.3 Issue 3 - Very Large Mahalanobis Distances

Upon calculating the Mahalanobis distance, this value is converted to the z-score via Equation 2.5. If the Mahalanobis distance is large - with the definition

of *large* being relative to the number of dimensions - the χ^2 value saturates and $F_k(\mathcal{D}_k^2)$ becomes one. This is due to the limited precision of floating point numbers, leading to a value very close to one becoming rounded to one. This causes an issue when converting this value to a z-score as the z-score becomes infinity. This is both false and unworkable.

To resolve this, the program checks for a value of one when a Mahalanobis distance is converted to the χ^2 domain. If found, it sets the z-score to be the highest allowed non-infinite value for a floating point number. This approximation highlights the test sample as highly abnormal while allowing for a numerical value that can be used in the forest.

2.8.4 Issue 4 - Failure Case: Missing Unseen Features

Here I present a more complex problem. Imagine a case where there are two classes in the training data, and each sample has two binary features (see Figure 2.6). Referring to the figure, a novelty tree will learn that the first (left) feature in the training data is important as it differs between the classes, whereas the right feature does not. Thus a branch node will grow that sends these two classes of training data to separate leaves. This is expected behaviour.

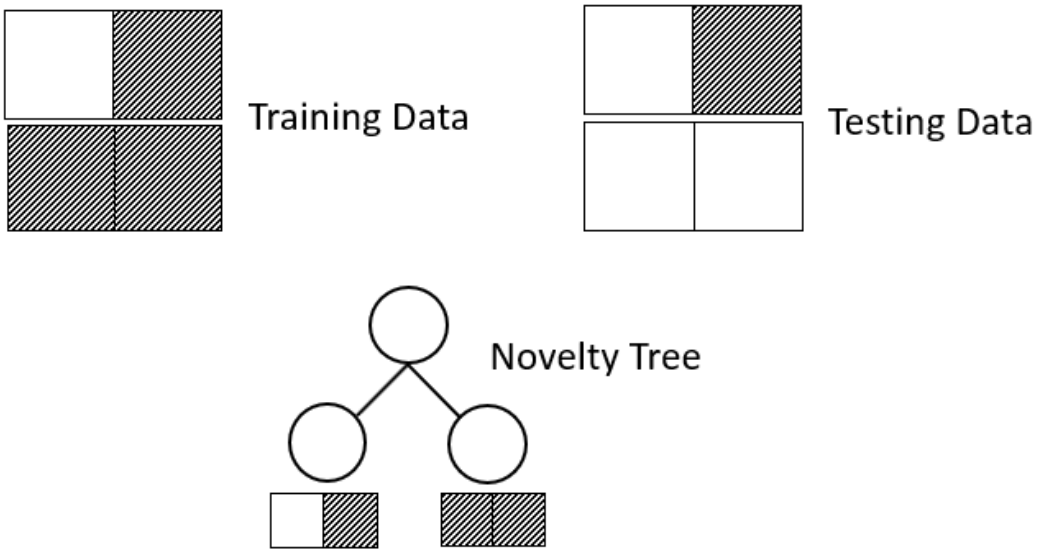


Figure 2.6: If the training data contains many examples of two distinct classes of data, the novelty tree will — in this example — learn to divide the data using the left feature as this is the descriptive feature. When using the testing data for prediction, which has one class the same as one class different to the training data, all of the data will be sent to the left leaf node as only the first feature is checked. This means the novelty tree fails to separate the classes. This failure occurs because the testing data has an unseen class within it.

However, when the two test cases are input to the tree, they will both be channeled down the left node and evaluated against the first feature. The first case is correctly classified as belonging to that leaf, but the second case is incorrect as the second feature - ignored during the evaluation - is different to the others. This failure is caused by the tree not learning to split at the second feature, which in turn is caused by there not being any training data with a variation with this feature.

While a lack of training examples of a certain class is an issue for most trained classifiers[4], the novelty forest is particularly vulnerable to them as it only factors in a subset of all features during training. This makes its use as

[4]Other abnormality detection methods, as described in Chapter 3, are a notable exception.

an abnormality detection method questionable.

The argument against this failure case is that the gain in efficiency by only checking a small subset of features outweighs the cost in accuracy by potentially mis-evaluating cases in the testing data that do not follow the patterns in the training data. The novelty forest is designed with efficiency in mind.

2.9 Data

I use the Modified National Institute of Standards and Technology (MNIST) dataset [306] as a simple dataset to explore the Novelty Forest's behaviour. This dataset is a collection of 70 000 2D images of handwritten digits. Each image is normalised and centred to a 28 x 28 pixel area and created by applying an anti-aliasing filter to the original black and white version of the digit. Each digit appears in approximately equal proportions.

From this dataset I create two sets of data called MNIST Even and MNIST Odd:

MNIST Odd Dataset: I take the odd-valued digits (1, 3, 5, 7, 9) and from these separate out 5000 at random. The 5000 are placed in the *normal test set*, while the remaining around 30 000 are put in the *training set*. These are used for training and testing the novelty forest in Section 2.10. I then take the even-valued digits (0, 2, 4, 6, 8) and form the *abnormal test set*.

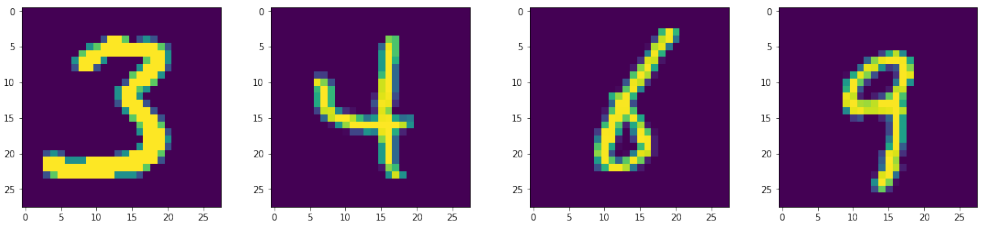
MNIST Even Dataset: The same as above, but with the even and odd digits swapped. Thus the training set and normal test set consist of even digits, while the abnormal test set holds odd digits.

The training data serves to grow the novelty forest, the normal test set provides a baseline for the forest's performance on data similar to its training data

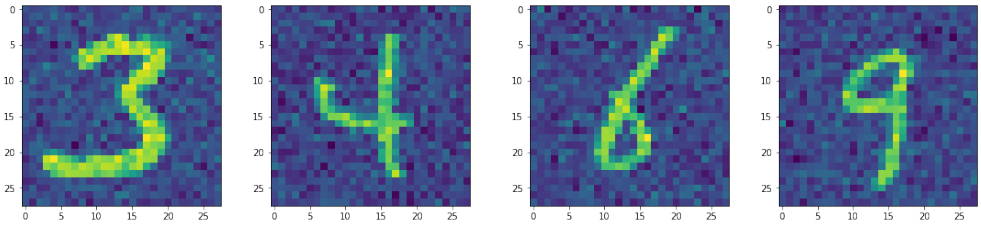
(i.e. normal data samples), and the abnormal test set gives the performance when evaluating on data different to its training data (i.e. abnormal data samples). This simulates abnormality detection where normal data are plentiful, but some rare examples, represented by the abnormal test set here, are not available during train time. Chapter 3 goes into more detail on abnormality detection.

I also use a *noisy* variant of these datasets. With noisy, I add Gaussian noise to each digit with the Gaussian having zero mean and a variance of 1000. The noise is sampled from this Gaussian and added to the intensity value of each feature (pixel). The noisy data will strain the model. These four datasets are chosen as they offer a sufficiently high complexity to explore the behaviour of the novelty forest, but are simple enough for the results to be easily interpreted. My results, shown later, demonstrate this.

Figure 2.7 shows examples of MNIST digits and their noise-added variants.



3 (original) 4 (original) 6 (original) 9 (original)



3 (noise added) 4 (noise added) 6 (noise added) 9 (noise added)

Figure 2.7: Examples of MNIST digits and their Gaussian noise-added variants.

2.10 Experimentation

To understand how the novelty forest's performance is affected by the division method and stopping criteria on various datasets, I perform a comprehensive evaluation under these parameters. I use the MNIST Odd and MNIST Even datasets and the Gaussian noise-added variant of them introduced previously. Each test digit is assigned a z-score and labelled as normal or abnormal according to which test set it came from.

On each dataset I apply a novelty forest with either the Random Feature/Threshold division method or the Optimised Feature method, leading to eight experiments. For each experiment the novelty forest is tested with three stopping criteria: a minimum number of samples of 10 or 100 in each leaf node, or a minimum number equal to the square of the node's depth. The novelty forest is evaluated for its area under the curves for the Receiver Operating Characteristic (ROC) curve and the Precision-Recall (PR) curve when tested on the normal and abnormal test data sets. The ROC curve measures the true positive rate (TPR) against the false positive rate (FPR) (both defined below) with 0.5 representing a performance that could be obtained by randomly assigning class labels to the data samples; scores range from 0 to 1 with higher scores indicating more TPs or less FPs. The PR curve measures precision against recall with these terms defined below. The value can be between 0 and 1 with 1 being perfect.

$$TPR = \frac{TP}{TP + FN} \tag{2.6}$$

$$FPR = \frac{FP}{FP + TN} \tag{2.7}$$

$$Precision = \frac{TP}{TP + FP} \tag{2.8}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.9}$$

where TP, TN, FP, FN represent the true positives, true negatives, false positives, and false negatives respectively.

The forests are grown with 100 trees, each with a random 10% of the samples and a random 10% of the features. These parameters were found in prior tests to provide a suitable ensemble of trees in the forest without costing excessive computational resources. Each experiment is repeated three times to gauge the variance and the results averaged. Table 2.1 displays the accuracies.

Dataset	Division Method	Min 10	Min 100	Min Depth[2]
MNIST O	Random	**0.86 ± 0.01**	0.73 ± 0.01	0.78 ± 0.01
	Feat./Thres.	**0.97 ± 0.00**	0.93 ± 0.01	0.95 ± 0.00
MNIST O	Optimised	0.79 ± 0.01	**0.81 ± 0.01**	0.80 ± 0.01
	Feature	0.94 ± 0.00	**0.95 ± 0.01**	**0.95 ± 0.00**
MNIST E	Random	**0.45 ± 0.02**	0.44 ± 0.04	0.43 ± 0.02
	Feat./Thres.	**0.85 ± 0.01**	**0.85 ± 0.01**	0.84 ± 0.01
MNIST E	Optimised	**0.54 ± 0.03**	0.45 ± 0.06	0.50 ± 0.02
	Feature	**0.88 ± 0.01**	0.85 ± 0.06	0.86 ± 0.00
MNIST O	Random	0.52 ± 0.02	**0.85 ± 0.00**	0.84 ± 0.00
(Noisy)	Feat./Thres.	0.86 ± 0.01	**0.97 ± 0.00**	0.96 ± 0.00
MNIST O	Optimised	0.50 ± 0.01	**0.75 ± 0.02**	0.70 ± 0.01
(Noisy)	Feature	0.86 ± 0.00	**0.94 ± 0.01**	0.92 ± 0.00
MNIST E	Random	**0.54 ± 0.01**	0.43 ± 0.01	0.41 ± 0.01
(Noisy)	Feat./Thres.	**0.87 ± 0.00**	0.85 ± 0.01	0.84 ± 0.00
MNIST E	Optimised	**0.51 ± 0.00**	0.45 ± 0.02	0.41 ± 0.01
(Noisy)	Feature	**0.86 ± 0.00**	0.85 ± 0.01	0.83 ± 0.00

Table 2.1: The performance of the novelty forest across a variety of algorithmic parameters. The error is the standard deviation of the three experiments of 100 trees with each tree using a random 10% subset of features and samples. The top row in each cell is the area under the ROC curve; the bottom is the area under the PR curve. *MNIST E/O* represents the even and odd datasets respectively and the *(Noisy)* label represents the dataset with added Gaussian noise.

2.11 Discussion

My two division methods first select a feature and then choose a threshold within this feature. This means each feature is selected with equal probability, but the thresholds within a feature are searched inversely proportionally to the feature's value range.

The number of leaf samples used in the stopping criteria should not be a constant. The covariance matrix used for the Mahalanobis distance is a function of the $depth^2$ as it is the number of features used squared. To estimate the covariance matrix's values, which is a 2D square matrix of the features used, there should be at least as many samples as there are values to estimate. This is where the $depth^2$ stopping criteria comes from. This enables larger training sets to form deeper trees, but the number of samples would need to increase quadratically with depth. This limits the depth of a tree and thus keeps it efficient.

Despite this, the best performing stopping criteria were the minimum 10 or 100 samples in leaf nodes. This means the increase in accuracy from forming deeper trees and smaller clusters outweighed the loss in accuracy from uncertainty in the covariance matrix. This may be because many of the features in MNIST are uninformative, so most nodes in the tree are useless in the case of the Optimised Feature method. With Random Feature/Threshold the number of useful combinations found is likely to be low, so even the best found split may not be great. I believe having more nodes in the tree is a way around this as more feature/threshold splits will be explored, leading to a higher chance of finding the useful combinations. This may not be an issue when using data with a higher proportion of informative features. In general, the Random Feature/Threshold method performed stronger on the MNIST datasets. If the proportion of useful features increased, I would expect the Optimised Fea-

ture method to improve because on average it will then find better splits (by optimising features as before but having a higher chance of targeting useful features).

The MNIST Even dataset (even training digits) had much poorer performance than the Odd dataset with the area under the ROC curves around 0.5 in all cases, which is random chance accuracy. The features learned on the even digits were not useful for distinguishing the even and odd digits during test time. The even digits: 0, 2, 4, 6, 8 have lower interclass variability than the odd digits 1, 3, 5, 7, 9. With the even digits, 0, 2, 6, and 8 all show similar curves at the top or bottom of the digits. Only the 4 is relatively unique. The trees will be splitting according to this digit alone, at least at the high levels, and these splits may not be informative for distinguishing the odd digits. On the other hand, the set of odd digits have higher interclass variability. 3, 5, and 9 have similar structures in them, but have differences too. 1 and 7 are incomparable to these digits. The trees will be forming more informative splits with this data and this in turn leads to a greater discriminative power on the test data.

2.12 Conclusions and Further Work

There is a need to explore the Novelty Forest with a wider range of data, especially data that has a higher proportion of informative features. This will aid in understanding the strengths of the two division methods. I would also like to explore what would happen if every feature was equally informative. I would expect the Optimised Feature method to perform strongly here, as each node will be optimally thresholded. The Random Feature/Threshold method would be unlikely to find the optimal threshold for any feature, so performance could be lower. Likewise if there was only a single informative feature in a large

range of features, the Random method should be able to find it and find an acceptable split, while the Optimised method would struggle here.

There is also a case for introducing a smarter division method. One that randomly samples the features and thresholds at first to determine which may be the most useful and then focuses searches within these areas. This will come with a cost during the growing stage, but will not affect the prediction stage. It is also possible that prior knowledge about the distribution, if available, could be used to improve the division method by providing areas to search in or clues about finding optimal thresholds. However, as decision forests depend on a substantial degree of randomness to search the solution space, this may not be the best way forward.

In a similar line, the criteria for optimising (the information gain) could be modified to better identify optimal areas. For instance, the derivative of the information gain would provide a gradient, which could be used by the division method to shift its search towards a likely higher gain area. Alternatively, the criteria could be chosen specifically for the dataset at hand as the information gain might not be the most suitable quantity for all datasets. Information gain is about minimising the standard deviations of child nodes, which might not be the best choice for every dataset. More research is needed here to explore this.

Next Steps: I take the Novelty Forest, along with four soon-to-be-introduced abnormality detection techniques, and evaluate them on medical data under a range of embedding methods in the next chapter.

Chapter 3

An Accuracy and Runtime Comparison of Abnormality Detection Techniques for Unsupervised Classification in Medical Imaging Data

3.1 Abstract

Abnormality detection, also called outlier detection and novelty detection, seeks to identify data that do not match an expected distribution. This may be done by learning a model of normality, against which new samples are evaluated. In this chapter the novelty forest introduced previously and four other abnormality detection methods, each representing a different family of techniques, are compared. These methods are local outlier factor, the one-class support vector machine, isolation forest, and fast-minimum covariance

determinant estimator.

Each method is evaluated on patches of CT interstitial lung disease where the patches are encoded with one of four embedding methods: principal component analysis, kernel principal component analysis, a flat autoencoder, or a convolutional autoencoder. These methods seek to capture the salient trends of the data, allowing the outlier detection method to perform with higher accuracy and speed. The methods are also evaluated on the non-embedded (raw) data for reference.

The dataset consists of 5500 healthy patches from one patient cohort defining normality, and 2970 patches from a second patient cohort with emphysema, fibrosis, ground glass opacity, and micronodule pathology representing abnormality. From this second cohort 1030 healthy patches are used in the evaluation for a comparison to the abnormal patches.

Evaluation occurs in both the accuracy (area under the receiver-operator curve) and runtime. The fast-minimum covariance determinant estimator shows fair time scaling with dataset dimensionality, while the isolation forest and one-class support vector machine scale well with dimensionality. The one-class support vector machine is the most accurate, closely followed by the isolation forest and fast-minimum covariance determinant estimator. The novelty forest is found to be fast with good accuracy. The embeddings from kernel principal component analysis are the most generally useful.

3.2 Introduction

Abnormality detection – also known as outlier detection or novelty detection – is a type of unsupervised data classification process whereby a model of *normal* data is created against which unseen data samples are compared. The class labels of the samples are not used (beyond normal/abnormal). The unseen

samples that match the model are classed as normal, while those that do not are classed as abnormal. Normal and abnormal samples may be referred to as inliers and outliers respectively.

An outlier is defined by Hawkins [122] as "an observation point, which deviates from the other observation points so much that it is caused by the suspicion that it is generated by different mechanisms." Barnett *et al.* [23] provide a similar, but simpler, definition: "An observation (or subset of observations) which appears to be inconsistent with the remainder of that set of data."

Abnormality detection finds use in situations where normal data are plentiful or easy to obtain and represents the space of normality well, but abnormal data are logistically difficult, expensive, or otherwise challenging to obtain; or where it is unreasonable to collect data on the full space of abnormality because the space required to be sampled to capture all possibilities is large or infinite. Abnormality detection can be used in medical imaging to, for example, find patient data with possible pathology or, more generally, to exclude data that are normal (healthy). Every patient is unique and being able to model the range of healthy patients is crucial to being able to detect rare diseases that may not appear in any training data.

There are a broad range of methods for abnormality detection. The most popular ones fall into four categories: random forest, hyperplane separation, Gaussian fit, and distance/density measurement. For each of these categories I select one of the most influential algorithms for my evaluation: The Isolation Forest [182] for the random forest, the One-Class Support Vector Machine [256] for the hyperplane separation, Fast-Minimum Covariance Determinant Estimator [249] for the Gaussian fit, and Local Outlier Factor [37] for the distance/density measurement. I also take the novelty forest introduced in the

previous chapter (Chapter 2) for comparison.

These methods ultimately assign a value to each sample that describes its abnormality – or in other words how likely that sample is to *not* belong to the normal population. I call this the *abnormality score*. The abnormality score is a measure of how abnormal a sample is, with a lower score (bounded by zero) indicating a perfectly normal sample, while an increasing abnormality score – which has no upper bound – shows an increasingly more abnormal sample. The abnormality score for the outlier detection methods is discussed in more detail in Section 3.5.2. In practice, these scores are thresholded at a sensible level, such as 95% confidence of the sample being normal, to classify samples into normal or abnormal. The abnormality scores produced by one method are not directly comparable with the scores produced by a different method.

In this work I do not use a single threshold to classify the samples but instead threshold at all possible thresholds to determine the false positive rate and true positive rate at these thresholds. These two rates then form a receiver operating characteristic (ROC) curve from which I evaluate the performance by the area under the curve (AUC).

My evaluation is implemented in the Python programming language (version 3.6) [93].

3.3 Previous Work

Identifying outliers in a dataset is an active area of interest in statistical fields. This section describes the most popular and interesting techniques for outlier detection. These techniques broadly fall into two categories: density based or distance based. Density-based techniques focus on using the density of clusters within the data to determine outliers. Distance-based techniques put less emphasis on cluster density and more on local distances between samples.

This can be thought of as looking at the local density around a point, instead of looking at broader density patterns. These categories are broad and arguably have some degree of overlap, but they aim to serve a coherent structure to this section. I also include a short section at the end for techniques that do not fit into either of the aforementioned definitions cleanly.

The techniques introduced previously for this chapter's experiments (Section 3.2) are not discussed here, but are described in detail in Section 3.5.2.

3.3.1 Density Techniques

These techniques directly use the density of clusters as a means of determining abnormality. This may involve using the density directly or by clustering the data according to densities.

Examples of clustering algorithms that seek to omit outliers are Shared Nearest Neighbour by Ertöz *et al.* [84], DBSCAN (Density-Based Spatial Clustering of Applications with Noise) by Ester *et al.* [85], and OPTICS (Ordering Points To Identify the Clustering Structure) by Breunig *et al.* [36]. Other work aims to cluster data and determine the abnormality of each data sample by its distance to the centre of its closest cluster. Kohonen used self-organising maps [154] for this task, while Barbará *et al.* [21] designed a novel algorithm for intrusion detection based on intersecting segments of unlabelled data and using the intersection as the base data for clustering.

Some research has taken a more sophisticated approach where a sample's abnormality score factors into how its local cluster is structured, such as the size of the cluster. He *et al.* proposed a technique called Find Cluster-Based Local Outlier Factor [124] that uses the size of clusters to influence the degree of abnormality of a sample.

The Isolation Forest I introduced earlier in this chapter has been shown

by Hariri *et al.* to have issues with artifacts [120]. These are artifacts in the abnormality score assignment in the feature space. Figures 1 – 3 of their publication ([120]) demonstrate that the Isolation Forest is unable to pick up intricate design within the feature space and instead acts to blur a pattern of samples into a single cluster. It also increases the abnormality score of regions in feature space that are perpendicular to the training samples. The authors present two extensions to the Isolation Forest which allow for finer details to emerge within the training set and reduces the artifacts seen perpendicular to the training samples. The first is the Rotated Isolation Forest, which rotates training data randomly per tree during the growing stage; the second is the Extended Isolation Forest that rotates the hyperplane that dissects the data within each node, essentially allowing a division to exist across any number of features at once. Both methods are shown to reduce data artifacts in the original Isolation Forest, but they have a higher runtime. For the purpose of evaluating the simplest version of the Isolation Forest and with the expectation that the artifacts mentioned here will be rare in my dataset evaluation, I will not be using these proposed extensions in my work.

Zhou *et al.* take a different approach. With their CBMIR (Content-Based Image Retrieval) method use the Isolation Forest to represent the original data in a new form: a vector of relevance features [311]. In other words, they use the Isolation Forest as an embedding space. This new space is shown to be richer in information than the original space, and this in turn allows for a higher performance on detection tasks. The authors claim CBMIR has linear time complexity with respect to the dataset size when training, constant time complexity for number of features in the dataset if the number of features to use for training is fixed, and is tolerant to irrelevant features due to its use of the Isolation Forest in determining outliers.

Other researchers have focused on a trade-off between efficiency and accuracy. The SCiForest by Liu *et al.* is an example of this [183]. It maintains the favourable time complexities of the Isolation Forest to allow it to outperform other methods in runtime, but it is also of superior performance to the Isolation Forest in most of the tasks the authors trialled. The SCiForest grows Isolation Trees but the authors examines the post-split dispersal for each node during the growing stage and focus on minimising this. They do this by introducing random hyperplanes for outliers that are undetectable by single attributes. When the outliers depend on multiple attributes for detection, a higher number of attributes being used in the hyperplanes yields an improved detection performance. The authors also claim the SCiForest is particularly good at detecting outliers that are close to the edge of normal clusters.

Ting's *et al.* Mass Estimation is another example where efficiency has been central to the method [283]. Mass Estimation uses the *mass* (i.e. number of surrounding samples) samples in feature space have to perform abnormality detection with constant time and space complexities. The mass of a sample is independent of other characteristics of the region it occupies, such as density, shape, and volume; such that mass is a rectangular function with the same value for the region it is measured in. The authors show how this measure can be used alongside a concave function to effectively cluster samples together and provide an ordering in terms of abnormality rather than a probability density estimation.

Ho developed the Random Decision Forest [128] that was later developed by Barandiaran [20] and Breiman [34] into the modern version of the Random Forest. An example of the random forest in action can be found by Peerbhay *et al.* who use the random forest to map an invasive plant species from satellite imagery data [232].

Müller *et al.* present a novel outlier detection method, OUTRES (OUTlier RElevant Subspace), for samples deviating in subspace projections of the original data [213]. This approach computes the deviation from the local density but only takes into account a selection of subspaces for each sample, where subspaces that are not distributed uniformly random are used for the outlier detection. A density measurement that adapts to each subspace is used for comparable production of abnormality scores.

Latecki provides a method based on non-parametric density estimation with a variable kernel to yield a robust local density estimation, which in turn leads to outliers being revealed [172].

3.3.2 Distance Techniques

These techniques focus on the distance between samples to decide abnormality. This is similar to the density techniques discussed previously, but these pay less attention to the density of the overall cluster and more to local density, which acts as a proxy for distance.

Harada *et al.* apply the Local Outlier Factor method , which I introduced earlier as one of the methods I use in my evaluation, to cyber physical systems [118]. They show that while many outliers were detected, this method also found anomalous samples that appeared too frequently were missed (false negatives), as well as featuring a number of false positives, although the authors note that these false positives share some similarities with the anomalies.

Su *et al.* develop an efficient local outlier factor method that operates in two stages [276]. The authors argue that traditional local outlier factor is inefficient as it must evaluate every data sample, despite the majority of the samples being inliers and therefore uninteresting to an outlier detection method. To account for this, the authors develop E2DLOS (Efficient Density-based Local Outlier

detection for Scattered data), a technique able to identify the most obvious inliers and remove most of them from a dataset. The authors then perform a modified local outlier factor to find the outliers. In this way the method only factors in a subset of the inliers when evaluating samples. Agyemang provide a similar technique through their LSC (Local Sparsity Coefficient) method where non-outlying data are removed before applying a modified local outlier factor method [4].

Jin *et al.* provide an improvement to the accuracy of a k-nearest neighbours (kNN) [58, 7] method when density distributions are required to determine outliers, for instance in the case of a sparse cluster of outliers being close to a denser cluster of inliers [138]. The authors do this by looking at the symmetric neighbourhood relationship to consider both the neighbours and the reverse neighbours of an object when estimating its density distribution. Djenouri *et al.* also use a kNN approach, but in this case to detect outliers within a dataset and remove them, thus enriching the information remaining in the dataset [76].

Cai *et al.* adapt the kNN approach to work with time data [43]. This adaption is called RD-kNN (Real-time Detection based on kNN). First this method explores the historical data of the system to determine an outlier threshold and then uses a modified kNN approach to detect outliers in real time data.

Zhang *et al.* demonstrate LDOF — Local Distance-based Outlier Factor — a distance-based outlier detection method that is particularly effective on scattered datasets where there are implicit data patterns [309]. This method is shown to outperform kNN and local outlier factor on such a dataset.

3.3.3 Other Techniques

Some proposed techniques do not fit cleanly into the previous definitions. I have kept these separate here.

Paulheim *et al.* take a different approach entirely for outlier detection and effectively reformulate unsupervised outlier detection as a set of supervised learning problems, specifically supervised regression learning problems, using their ALSO (Attribute-wise Learning for Scoring Outliers) method [229]. The authors create a predictive model for each feature in a dataset using the other features as attributes. Each model learns the relations between features. The models produce a weighting for each feature and use these alongside the difference between the predicted value and the original value for a sample feature to determine the degree of outlierness. That is, a feature value that is very different to what is predicted by the model based on the other features is likely to be abnormal as it does not follow the pattern of other values for that feature. This method comes with a significant computational cost, especially when the number of features is large.

While most methods focus on some concept of distance between samples (be it density, clustering, projected distances, or otherwise) to determine outliers, Kriegel *et al.* take a different approach with their ABOD (Angle-Based Outlier Detection) technique and use the variance from angular vectors between samples to find outliers [163]. This allows them to avoid the curse of dimensionality in high-dimensional datasets to accurately determine outliers. In a later publication [162], Kriegel *et al.* formalise the process of determining whether a sample is an outlier or inlier from its abnormality score using a method named LoOP (Local Outlier Probabilities).

Schneider *et al.* propose a new kernel-based outlier detection method, EXPoSE (EXPected Similarity Estimation) which is efficient for very large

datasets and operates with linear time complexity during training and constant time and space complexity during testing [255].

3.4 Datasets

My evaluation uses two publicly available datasets: MedGIFT [73] and the Emphysema dataset [274]. The MedGIFT dataset is used for testing data, while the Emphysema dataset is used to generate training data. This ensures there is no potential for overfitting to the evaluation dataset and mimics training a system in a research environment and deploying it to an external environment, such as a hospital.

MedGIFT Dataset [73]: 93 volumetric scans of interstitial lung disease (ILD) collected at the University Hospitals of Geneva where patients had a history of ILD and radiographic evidence consistent with the diagnosis. Patients received high-resolution CT imaging of the thorax. The scan slices are annotated by a radiologist with 2D regions of interest for pathological lung patterns.

Emphysema Dataset [274]: High-resolution CT scans of a study group of 39 patients. The scans were acquired at the Gentofte University Hospital in an exploratory study. A set of 61 x 61 pixel patches is provided, extracted for regions of healthy (non-pathological) tissue, centrilobular emphysema, and paraseptal emphysema. Healthy tissue was marked only on non-smoking patients. Patches are labelled with the leading diagnosis based on the consensus of an experienced chest radiologist and an experienced CT pulmonologist.

These datasets feature segmented ground truth that is used to define sampling regions. Each sampling region is either healthy tissue (representing nor-

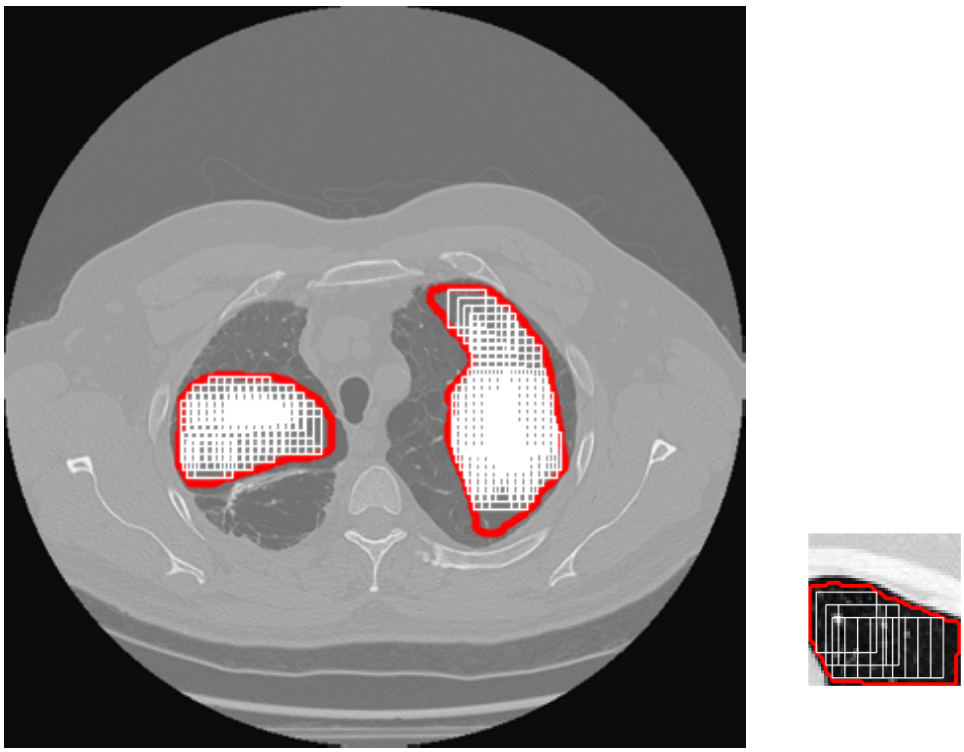


Figure 3.1: **Left:** Example of a lung scan slice from the Emphysema dataset showing an emphysema sampling region in each lung (outlined region) and extracted patches (squares). **Right**: A zoomed-in example of a small sampling region.

mal samples) or pathological (representing abnormal samples). Regions are defined by a single ground truth label: Healthy, Emphysema, Fibrosis, Ground Glass Opacities, or Micronodules and samples are extracted fully within the regions. The MedGIFT dataset has all the full range of labels while the Emphysema dataset has Healthy and Emphysema only.

2D patches of 20 x 20 mm size are sampled at a resolution of 1 pixel/mm^2 from these regions on scan slices. This means the patches have 400 pixels (or features). Once the samples are extracted, their ground truth labels are removed to ensure this is an unsupervised approach.

5500 healthy patches are taken from the Emphysema dataset. Due to limited data availability, these patches are permitted to overlap by 80% to

increase the number of samples taken. This is a form of data augmentation. Figure 3.1 shows the sampling process. These samples are used for training the model of normality. No samples of pathological tissue are taken from the Emphysema dataset.

1030 healthy patches and 2970 pathological patches are taken from the MedGIFT dataset for testing the model of normality. There is no overlap between these patches. The pathological patches consist of the following ground truth labels: 231 emphysema, 557 fibrosis, 266 ground glass opacities, and 1916 micronodule samples. The relative proportion of these pathologies derives from the relative abundance of the labels in the MedGIFT dataset. Figure 3.2 shows a set of patches extracted from the MedGIFT dataset.

These patches are divided into a training set and two testing sets: normals and abnormals. From each of these sets, a small subset is removed and placed into a corresponding *optimisation* set. Figure 3.3 illustrates the division of the data into these six sets. The optimisation sets are used for optimising the hyperparameters of the abnormality methods only and are then discarded.

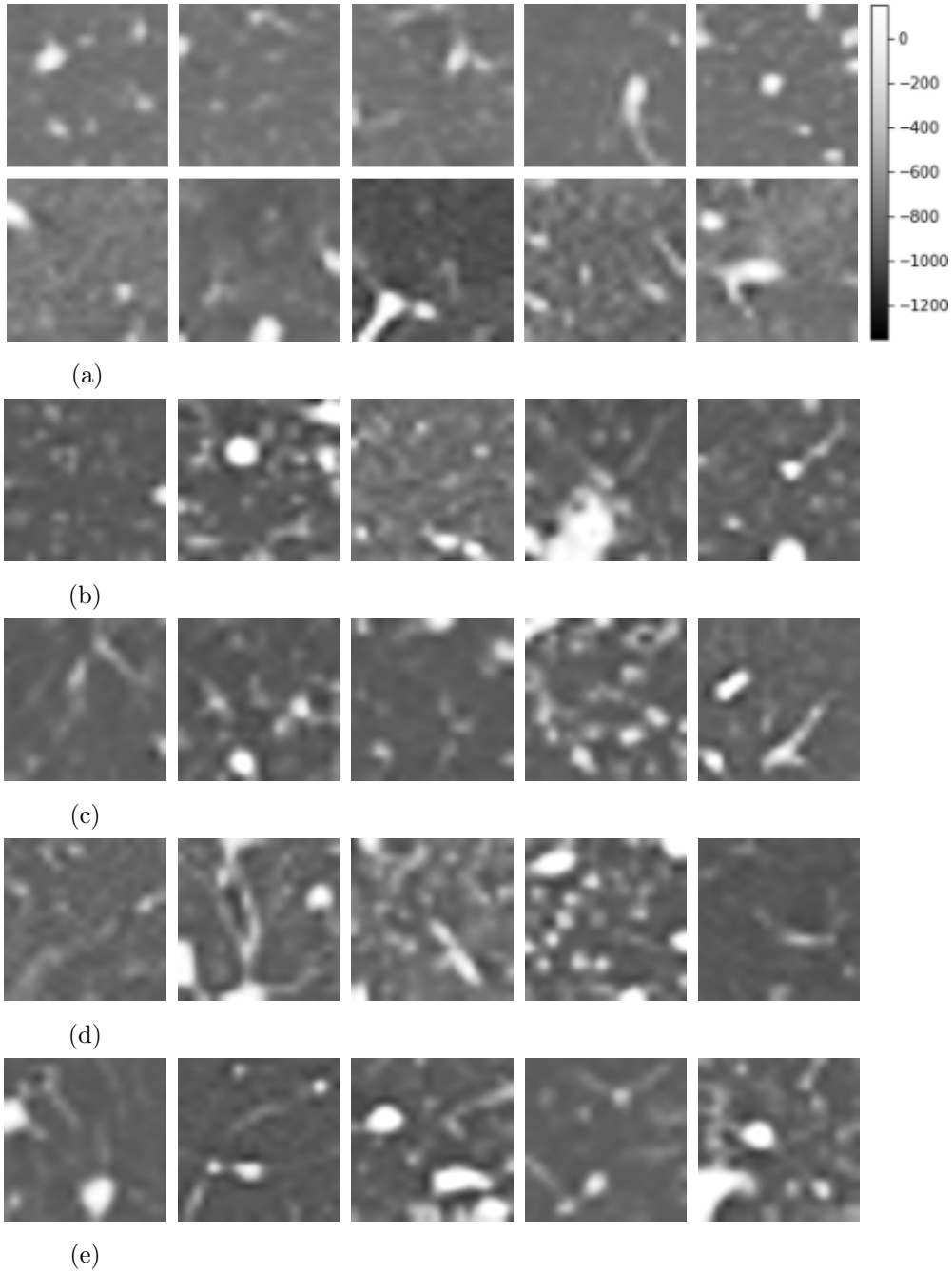


Figure 3.2: Examples of extracted patches of lung pathology windowed at a level of -600 HU with a width of 1500 HU, as recommended by Radiopaedia [241]. From top to bottom: **a)** Healthy (upper row from the training set (Emphysema dataset) and lower row from the test set (MedGIFT)) **b)** Emphysema **c)** Fibrosis **d)** Ground glass opacities **e)** Micronodules. Rows b-e are from the MedGIFT dataset.

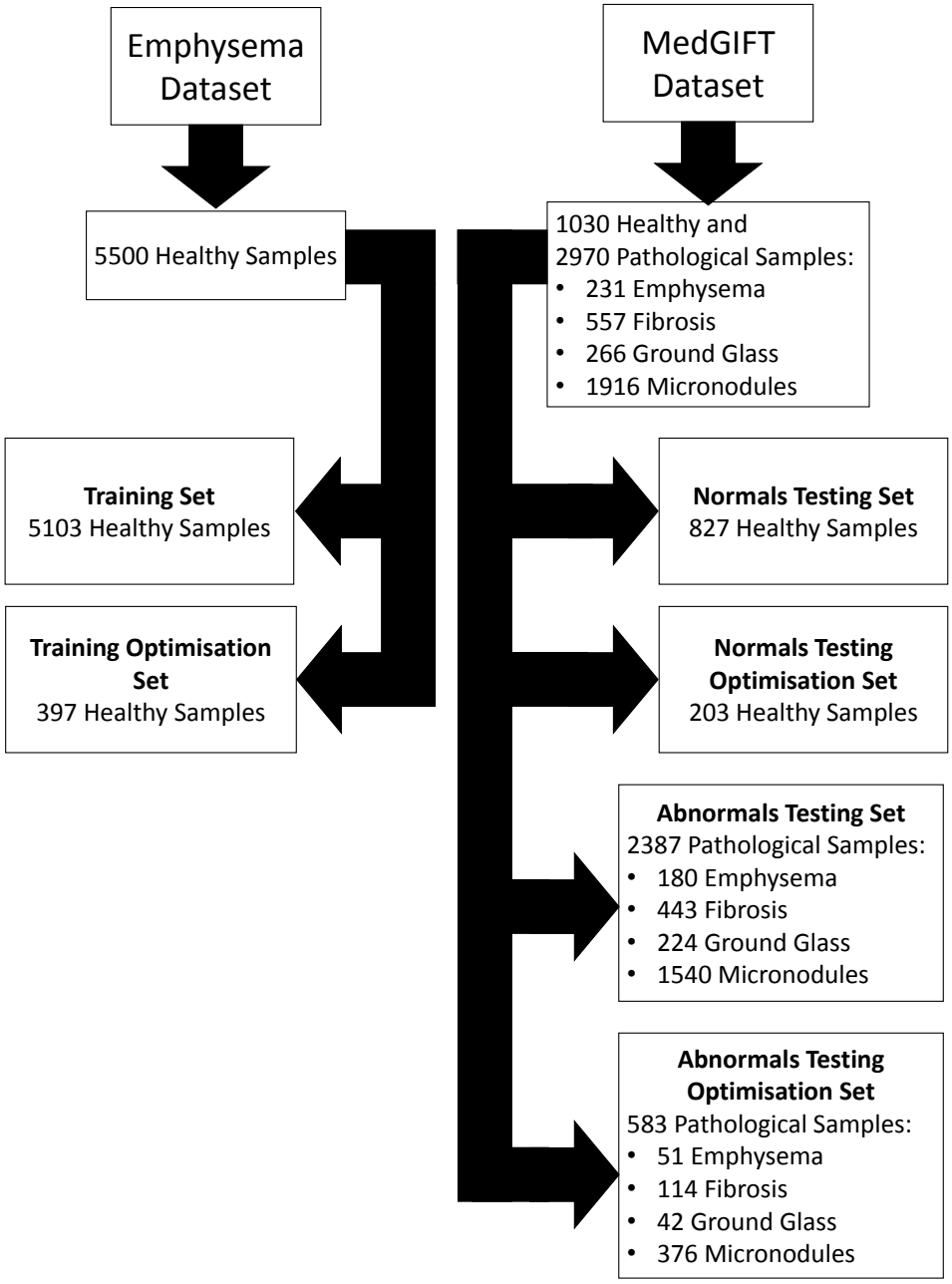


Figure 3.3: A training dataset is extracted from the Emphysema dataset, and normals and abnormals testing datasets are extracted from the MedGIFT dataset. Each of these three datasets has an optimisation version that is used for optimising the parameters of the abnormality methods.

3.5 Experiment Design

In this section I describe the experimental setup used starting with the embedding spaces (Section 3.5.1), then the abnormality detection methods (Section 3.5.2), the optimisation of each pair of embedding and abnormality detection method (Section 3.5.3), and the overall pipeline setup (Section 3.5.4).

3.5.1 Embedding Spaces

Each of the abnormality methods have their own strengths and weaknesses. To highlight these a number of embedding spaces are utilised for the input data samples, as well using the raw data (no embedding). These embeddings are: Principal Component Analysis, Kernel Principal Component Analysis, Flat Autoencoder, Convolutional Autoencoder. I denote the number of features these embedding spaces produce by x, which I explain how to get and its final values in Section 3.5.3 and Table 3.2. Each embedding space transforms the data in a unique way and it is unclear without experimentation which will be the best.

The parameters of each method (e.g. batch size, number of epochs) were chosen based on prior experimentation with the optimisation datasets.

None: The data samples are not projected into a embedding space and retain their 400 original features.

Principal Component Analysis (PCA) [140]: Principal component analysis converts a set of samples of possibly correlated features into a set of linearly uncorrelated features called components using an orthogonal transformation. It is a dimensionality reduction technique using a linear mapping in such a way that the variance of the data is maximised in the transformed domain. Each of the components explain some fraction of

the variance seen in the original data samples. In this work the components are sorted by the amount of variance they explain, starting with the highest variance, and the first x components are kept for analysis.

Kernel Principal Component Analysis (kPCA) [257]: An extension of PCA where the data samples are mapped into a higher dimensional space using a supplied kernel. PCA is then performed in this space. A third order polynomial kernel is used in this work as this was found to be effective at capturing explained variance in a small number of features. Like the PCA method, the components are sorted by explained variance and the first x components are used in the analysis.

Flat Autoencoder (fAE): An autoencoder is an artificial neural network whose output is learned to match its input. Thus, the input and output layer dimensions must be identical. In the case of an autoencoder for dimensionality reduction, which is the purpose of the one used in this work, the intervening layers have a lower dimension than the input/output layer. This forces the network to learn an encoding of the data with a lower dimension than the original data such that minimal information is lost (i.e. the data can be reconstructed accurately from this encoding).

A simple 5-layer dense neural network is used for the autoencoder with the central layer providing the encoded data of dimensionality x, which is then used for abnormality method evaluation (Figure 3.4). The flat part of this autoencoder refers to the use of dense neural layers. The fAE is trained for 200 epochs with a batch size of 256, adadelta optimisation [79], and the mean square error loss function.

Convolutional Autoencoder (cAE): Similar to the fAE but with 2D con-

volutional and max pooling layers instead of dense layers. Figure 3.5 shows the structure. The constraints of this structure means the dimensionality of the data extracted from the central layer (x) is a multiple of 25. Training is carried out identically to the fAE with 200 epochs, a batch size of 256, adadelta optimisation [79], and mean square error loss.

The fAE and cAE are implemented through Keras [50] (version 2.1.2).

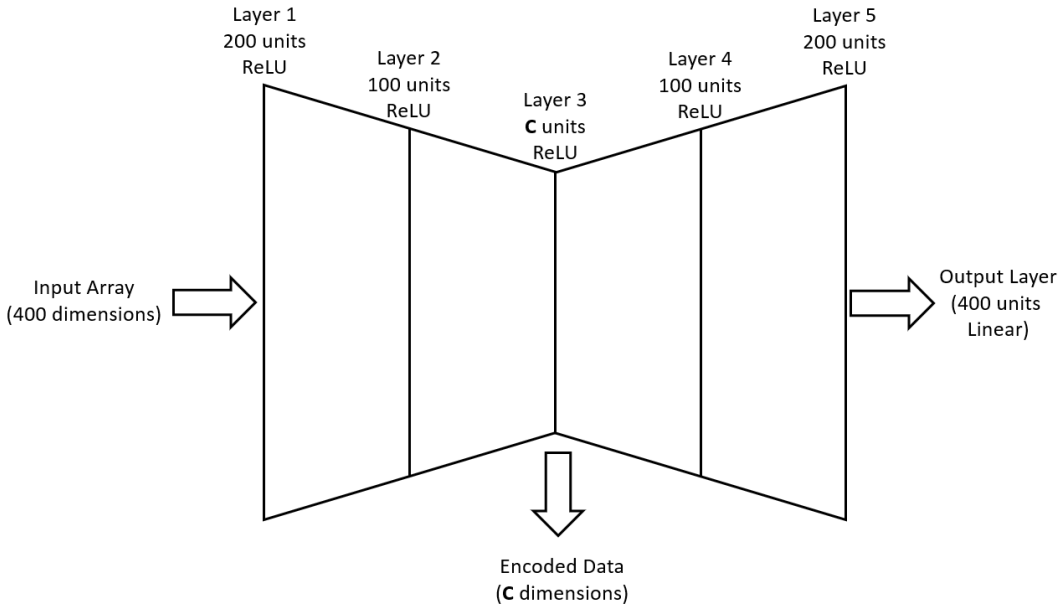


Figure 3.4: The structure of the flat autoencoder showing the number of units in each layer and the activation function used. ReLU is the Rectified Linear Unit. Layer 3 acts as the input to the outlier detection methods and has a number of units that varies between the methods (see Table 3.2). The number of extracted dimensions is labelled C.

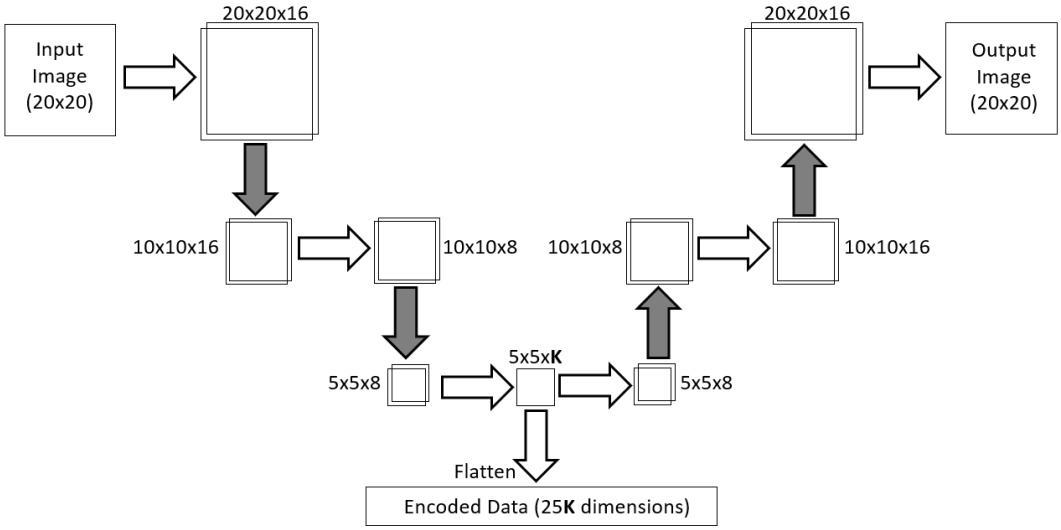


Figure 3.5: The structure of the convolutional autoencoder showing the dimensionality after each layer. The horizontal arrows represent 2D convolutions using a filter size of 3 x 3 pixels and a padding of zero surrounding the convolved images (using Keras [50] padding=*same*). The shaded arrows represent max pooling or up-sampling (both by a factor of 2) for downwards-facing or upwards-facing respectively. The vertical plain arrow is the flattening of the image for the encoded output. The flattening occurs as the outlier detection methods require a 1D input. All layers use the rectified linear unit activation function with the exceptions of the flatten and final layer, which use a linear activation function. The number of extracted dimensions is labelled K.

3.5.2 Outlier Detection Methods

In addition to the novelty forest, there are a wide variety of algorithms for abnormality detection. I utilise four of the most influential algorithms here, each representing a different family of solutions: local outlier factor, the one-class support vector machine, isolation forest, and fast-minimum covariance determinant estimator. By doing this, I aim to broadly capture the strengths and weaknesses these methods offer relative to each other and provide a useful evaluation against the novelty forest.

Novelty Forest (NF) is a forest ensemble technique that rapidly and intel-

ligently divides data based on a subset of features. Chapter 2 describes the novelty forest in detail.

Local Outlier Factor (LOF) [37] is a nearest-neighbour-based approach that determines the distance to the k^{th} nearest neighbour for each data sample in feature space. This distance is then compared to the distances other nearby samples gave to calculate the final abnormality score. LOF effectively judges each sample relative to the density of its local area in the feature space.

Mathematically speaking, the abnormality score for a sample a for the k^{th} nearest neighbour is

$$\text{abnormality}(a) = \frac{\sum_{b \in N_k(a)} \frac{\text{lrd}_{(b)}}{\text{lrd}_{(a)}}}{|N_k(a)|} \tag{3.1}$$

where $N_k(a)$ is neighbour k to sample a, $|N_k(a)|$ is the number of neighbours to a, and *lrd(a)* is the *local reachability density* of sample a as defined as the inverse of the average *reachability distance* of sample a from its k neighbours:

$$\text{lrd}(a) := 1/\left(\frac{\sum_{b \in N_k(a)} \text{reachability-distance}_k(a,b)}{|N_k(a)|}\right) \tag{3.2}$$

and this *reachability distance* is the L2 distance [94] in feature space between two samples bounded by a minimum of the distance to the k^{th} nearest neighbour from sample b:

$$\text{reachability-distance}(a,b) = \max\{\text{k-distance}(b), \text{distance}(a,b)\} \tag{3.3}$$

One-Class Support Vector Machine (1-SVM) [256, 235] is a support vector machine-based method for working with one training class. It

transforms the data samples to a higher dimensional space than they were initially in and seeks to build a hyperplane decision boundary that maximises the distance from this hyperplane to the origin. This is effectively building a non-linear region around the training samples. This is not to be confused with the support vector data description by Tax *et al.* [281], which is a one-class support vector machine method that constructs a hypersphere dividing surface of minimal volume around the training samples.

The kernel trick [282] is used to calculate the pairwise distances for the sample vectors in the transformed space without needing an explicit projection to that space. This improves the efficiency of the method. Equation 3.4 shows how the abnormality of a sample, a, is determined. α_i are the Lagrange multipliers, also known as the supports for the machine, ρ is related to the margin between the origin and the decision hyperplane, n is the number of samples, $K(a, a')$ is the kernel function, which in my case is the Gaussian Radial Base Function defined in Equation 3.5.

$$\text{abnormality}(a) = \sum_{i=1}^{n} \alpha_i K(a, a_i) - \rho \tag{3.4}$$

$$K(a, a') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \tag{3.5}$$

Isolation Forest (IF) [182, 49] is a binary forest approach whereby each node randomly selects a feature, and within this feature chooses a random threshold for splitting between the minimum and maximum value along that feature. Equations 3.6 and 3.8 This continues until each node has a single sample in it. These nodes form the leaf nodes. The greater the number of nodes between a leaf node and the root node, the greater the

abnormality score for the sample in that node. This is repeated over an ensemble of trees (a *forest*) and the scores averaged on a per sample basis. At each node that is not a leaf node:

$$f_{split} = \text{randint}(0, |F|) \tag{3.6}$$

$$t_{split} = \text{randfloat}(\ \min(A(f_{split})), \max(A(f_{split}))\ ) \tag{3.7}$$

where *randint(n,m)* selects a random integer between n and m inclusively, and *randfloat(n,m)* selects a float between n and m exclusively. F is the set of all features and $|F|$ is the cardinality of this set. f_{split} is the feature corresponding to the random integer selected. $A(fsplit)$ is the set of sample values along feature f_{split}. The min and max functions select the minimum and maximum value in the set. t_{split} is the threshold selected for the node split, while f_{split} is the feature selected for splitting.

The samples in the node are then split by evaluating each sample, a at the chosen feature and threshold:

$$\text{Child Node choice} = \begin{cases} \text{left} & \text{if } a(f_{split}) < t_{split} \\ \text{right} & \text{otherwise} \end{cases} \tag{3.8}$$

The isolation forest works in a probabilistic fashion with samples not belonging to a cluster being more likely to be separated out early in the growing phase in the tree, and thus they will end up in leaf nodes in a fewer number of steps than samples within a cluster. This technique is useful if there are many useful features in the data, otherwise it can be splitting along meaningless features and this leads the abnormality score to be meaningless.

Fast-Minimum Covariance Determinant Estimator (FMCD) [249] is a Gaussian fit model that is robust to outliers in the training data, thus leading to a robust fit centred around clusters. The mean and covariance of the nearest fitted Gaussian to a sample provides a means of determining the abnormality score of any sample in terms of standard deviations from the mean using the Mahalanobis distance (defined in the previous chapter – see Section 2.5. The formulation of FMCD goes as follows [249]:

For a dataset **a** of samples a_i and size n, let $H_1 \subset \{1, ..., n\}$. μ_1 is the mean and Λ_1 is the covariance matrix of this subset. Define the relative distances using the formula for Mahalanobis distance previously defined (Equation 2.4) but restated here for completeness:

$$d_1(i) = \sqrt{(a - \mu)^T \Lambda^{-1} (a - \mu)} \tag{3.9}$$

where $d_1(i)$ is the Mahalanobis distance for sample a_i.

Now a set H_2 is taken such that $\{(d_1)_{1:n}, ..., (d_1)_{h:n}$, where $(d_1)_{1:n} \leq (d_1)_{2:n} \leq \cdots \leq (d_1)_{n:n}$ are the ordered distances. The covariance matrix is calculated for this H_2. It follows that Equation 3.10 holds true. The *det* function is the matrix determinant.

$$\det(\Lambda_2) \leq \det(\Lambda_1) \tag{3.10}$$

The above step now repeats iteratively with H_3 onwards until the convergence criterion of the determinants being equal between two successive iterations or the determinant reaching zero. At this point, the mean and covariance matrix of the final subset, H_{final}, is used as the model of normality and the abnormality scores are calculated using the Mahalanobis

distance with these parameters.

The time complexity of these methods against the number of features (f) and samples (s) is found in Table 3.1.

Method	Train Time Scaling	Test Time Scaling
NF	$\mathcal{O}(log(s))$	$\mathcal{O}(log(s))$
LOF	$\mathcal{O}(f^2 s)$	$\mathcal{O}(fs)$
1-SVM	$\mathcal{O}(fs)$	$\mathcal{O}(fs)$
IF	$\mathcal{O}(s)$	$\mathcal{O}(log(s))$
FMCD	$\mathcal{O}(fs)$	$\mathcal{O}(s)$

Table 3.1: The time complexities of the abnormality methods for the training and testing stages in terms of the number of features, f, and number of samples, s. Worst case scenarios are shown.

With the exception of the NF, the methods are implemented by Scikit-Learn [231] (version 0.19.1). Direct Scikit-Learn references follow: LOF [262], 1-SVM [263], IF [261], FMCD [260]. The NF is implemented in the Python programming language (version 3.6) [93].

3.5.3 Parameter Optimisation

For every pair of data embedding with abnormality method there is a set of parameters that must be optimised for the problem being solved. Fair and reasonable attention to parameter optimisation is essential to be able to draw conclusions on the relative efficacy of the methods. By having a separation between the optimisation and experiment datasets (see Section 3.4) I prevent overfitting to the experiment dataset. I use a gridsearch method over the

optimisation datasets to find the optimal parameters for each pair. No cross-validation is used during optimisation due to time constraints. The following parameters were tuned for each abnormality method (all ranges are inclusive):

- NF: The node division method (Random Feature/Threshold vs Optimised Feature) only.
- LOF: The size the leaf and the number of neighbours. The leaf size was explored between 1 - 10 in steps of 1 and 10 - 100 in steps of 10; the number of neighbours was explored between 1 - 10 in steps of 1 and 10 - 100 in steps of 10.
- 1-SVM: The upper bound on the fraction of training errors (and lower bound on the fraction of support vectors) and the kernel coefficient. Both were explored between 0.1 - 0.9 in steps of 0.1.
- IF: The number of trees in forest was explored between 1 - 10 in steps of 1 and between 10 - 100 in steps of 10.
- FMCD: The support fraction between 0.1 - 0.9 in steps of 0.1.

For the NF the stopping criteria is chosen to be the $depth^2$, the number of trees is 100, and the sample and feature fraction used for each tree is 0.1. These choices were found to provide high accuracy with minimal speed trade-off in initial research not included in this thesis.

For the encoding methods, the following output dimensionality values were explored:

- PCA: The number of components kept after performing PCA. This was first explored broadly from 10 to 390 in steps of 10, and then followed by a higher precision evaluation of: y-9 to y+9 in steps of 1. y represents

the best performing value from the initial broad search. Computational limitations prevented searching the full range in steps of 1. The number of components is the dimensionality of the encoded data.

- kPCA: The number of kept components and the kernel used. Identically to the PCA method, the number of kept components for kPCA was first explored broadly from 10 to 390 in steps of 10, and then followed by a higher precision evaluation of: y-9 to y+9 in steps of 1. y represents the best performing value from the initial broad search. The kernel had five possible options: *linear*, *poly*, *rbf*, *sigmoid*, and *cosine* with the *poly* kernel having a range of possible degrees. The degree was explored between 2 and 10 in this case. This covers all common choices for the kernels.

- fAE: The encoding dimension (the number of neurons in the central layer). This was explored broadly between 10 - 90 in steps of 10, followed a closer inspection of y-9 to y+9 in steps of 1 for the best performing value from the broad stage, y.

- cAE: The encoding dimension (the number of neurons in the central output). This was explored from 25-400 in steps of 25. The multiple of 25 is required due to the design restrictions of the cAE (see Figure 3.5).

Table 3.2 shows the number of features selected by the optimisation of each method. The kernel selected for the kPCA method was the third degree polynomial. For the NF method, the Optimised Feature division method was chosen. Again, this choice comes from prior experimentation.

Embedding	NF	LOF	1-SVM	IF	FMCD
None (Fixed)	400	400	400	400	400
PCA	340	5	20	237	347
kPCA	400	3	22	15	2
fAE	15	40	26	31	10
cAE (Multiple of 25)	50	25	25	50	25

Table 3.2: The number of features used (per sample) by each embedding-method pair. With the exception of the None method, these values are from optimising each pair on the optimisation data sets. The None method uses the raw data and so uses the full set of features (400).

3.5.4 Experiment Pipeline

My evaluation consists of four main stages. These stages are sketched in Figure 3.6 which shows input/output of each stage. First, I take the pairs of embedding space/abnormality detection method and determine their optimal parameters (Section 3.5.3). Second I train a model of normality on the training set (Section 3.4) and use this to generate a set of abnormality scores for the testing sets (third step). Finally, these abnormality scores and the class labels of the testing data — which were not used previously — are converted to a ROC curve and the area under this provides the final evaluation score.

This final stage of converting from abnormality score to the ROC curve is shown in Figure 3.7. I order the samples in terms of their abnormality score, from lowest to highest. The class label is assigned to each sample representing either a normal sample (0) or abnormal (1). The ROC is created by scanning through the full range of samples and at every possible threshold making a note of the number of true positives (TP) — normal cases below the threshold, false

positives (FP) — abnormal cases below the threshold, true negatives (TN) — abnormal cases above the threshold, and false negatives (FN) — normal cases above the threshold. From these I calculate the true positive rate (TPR) and false positive rate (FPR) as defined:

$$TPR = \frac{TP}{TP + FN} \tag{3.11}$$

$$FPR = \frac{FP}{FP + TN} \tag{3.12}$$

With these two numbers, a ROC curve can be plotted and the area under it follows [98].

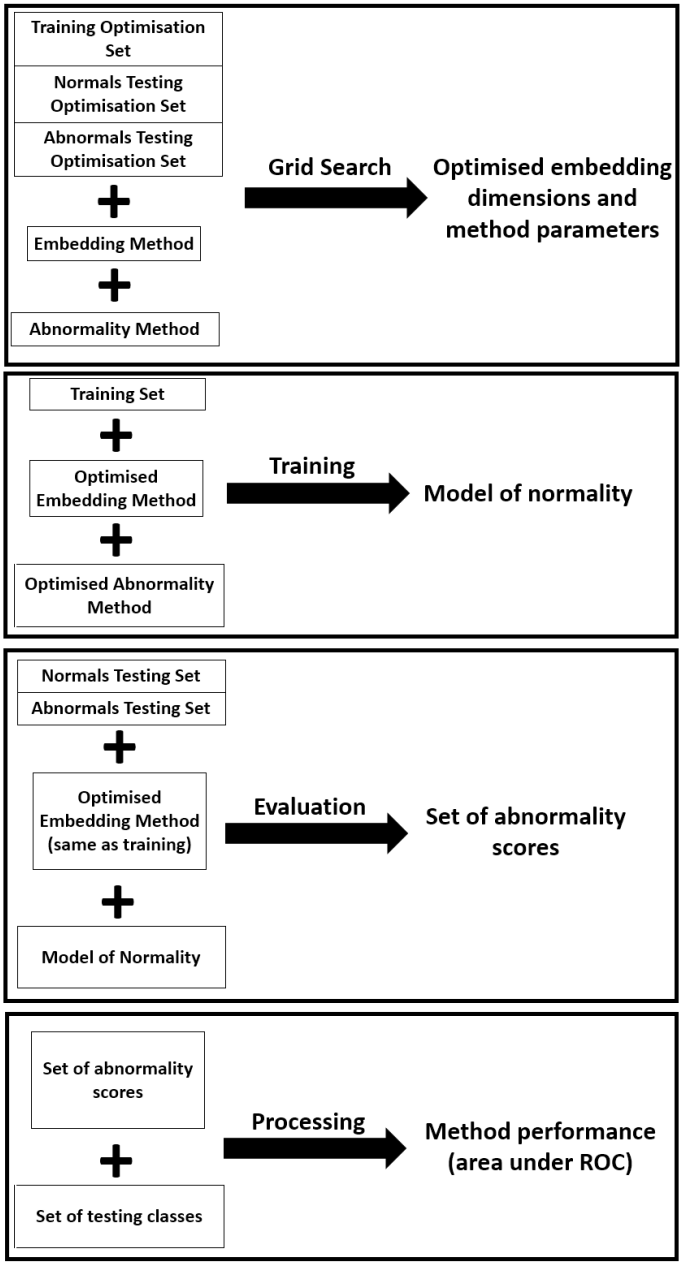


Figure 3.6: The full process of determining the performance of a method. First, the optimisation datasets are used to find the optimal number of embedding features and abnormality method parameters (see Section 3.5.3). Once optimised, the training data can be passed through the embedding method and then the abnormality method to develop a model of normality. This model of normality acts upon the testing data to give an abnormality score to every testing sample. Finally, the abnormality scores are evaluated to give overall performance of that embedding/abnormality pair.

Abnormality Score	True Class
1.0	0
2.0	0
5.0	1
5.1	0
8.0	1

TP = Below threshold with class 0
TN = Above threshold with class 1
FP = Below threshold with class 1
FN = Above threshold with class 0

Threshold < 1.0 TP = 0, TN = 2, FP = 0, FN = 3 TPR = 0 FPR = 0	**1.0 < Threshold < 2.0** TP = 1, TN = 2, FP = 0, FN = 2 TPR = 0.33 FPR = 0
2.0 < Threshold < 5.0 TP = 2, TN = 2, FP = 0, FN = 1 TPR = 0.67 FPR = 0	**5.0 < Threshold < 5.1** TP = 2, TN = 1, FP = 1, FN = 1 TPR = 0.67 FPR = 0.5
5.1 < Threshold < 8.0 TP = 3, TN = 1, FP = 1, FN = 0 TPR = 1 FPR = 0.5	**8.0 < Threshold** TP = 3, TN = 0, FP = 2, FN = 0 TPR = 1 FPR = 1

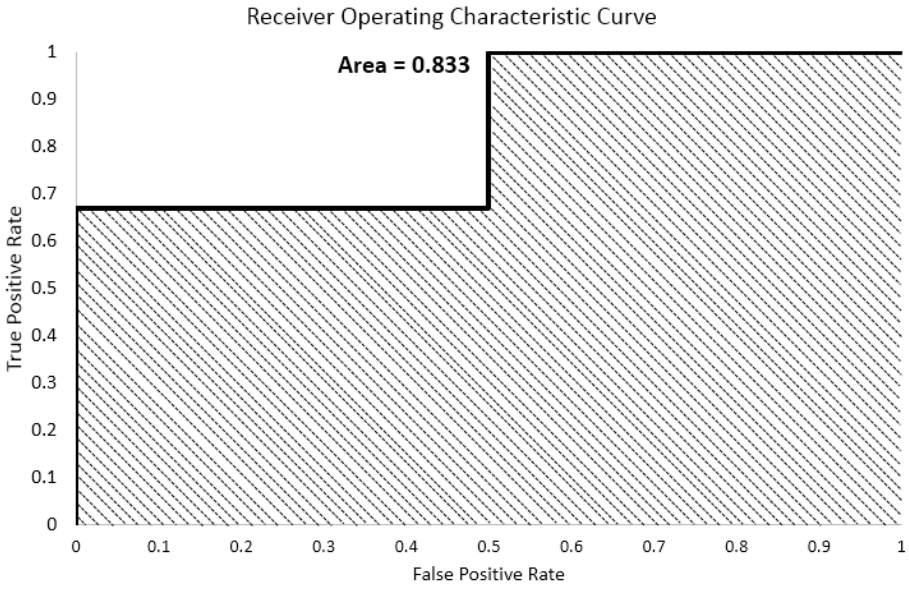


Figure 3.7: An example calculation of area under the ROC on fictional results. Each sample is assigned an abnormality score by the outlier detection method. The samples are sorted by these scores and have a true class assigned which is 0 for a normal sample, or 1 for an abnormal sample. At each possible division of the ordered list, the TPR and FPR is calculated. The pairs of TPR and FPR are plotted to create an ROC curve and from this the area under the curve is trivially determined.

3.6 Results

The aim of this research is to demonstrate how the abnormality methods perform on a medical dataset when trained on normal samples taken from one patient cohort and tested on normal and abnormal samples taken from a second cohort. This mimics training a system in a research environment and deploying it to an external environment such as a hospital. The objective is to correctly distinguish abnormality from normality and to do this in a time-efficient manner.

Each abnormality detection method is evaluated on the four embedding methods plus the non-embedded raw data. The methods are trained with healthy patches from one patient cohort and evaluated with a healthy test set and a pathological test set both from a second cohort to produce an abnormality score for all test samples. These scores are used to construct a ROC curve. The test scores for the healthy (normal) samples provide the true positive rate, while the test scores for the pathological (abnormal) samples give the false positive rate. The AUC is used for the reported accuracy (Table 3.3). The time required for the algorithm to embed and fit to the data (training stage) and predict abnormality scores (testing stage) is detailed in Table 3.4[1]. Figure 3.8 summarises my experiment pipeline.

[1]The predict times do not include the time taken to run the embedding method on the data being predicted on.

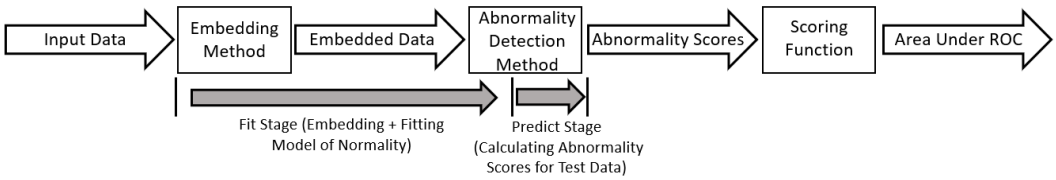


Figure 3.8: The experiment pipeline showing the order of data processing and the fit and predict stages. The scoring function takes the abnormality scores and produces an ROC curve from them, from which the area under the ROC curve follows. The abnormality detection method consists of first fitting a model of abnormality followed by testing of this model on the testing datasets.

Embedding	**NF**	**LOF**	**1-SVM**	**IF**	**FMCD**	**Average**
None	**0.740**	**0.697**	0.776	0.771	0.657	0.728
PCA	0.691	0.620	**0.823**	0.620	0.650	0.681
kPCA	0.678	0.612	0.817	**0.807**	**0.809**	0.744
fAE	0.546	0.627	0.742	0.790	0.771	0.696
cAE	0.626	0.539	0.816	0.766	0.758	0.695
Average	0.656	0.619	0.795	0.751	0.729	

Table 3.3: The AUC ROC curve for all embedding methods and outlier detection methods.

Embedding		NF	LOF	1-SVM	IF	FMCD
None	Fit	3.2	848	18.3	3.5	69.1
	Pred.	3.9	678	6.3	0.9	0.1
	Total	7.1	1526	24.6	4.4	69.2
PCA	Fit	3.4	1.2	2.3	2.8	49.0
	Pred.	4.0	0.7	0.7	0.6	0.1
	Total	7.4	1.9	3.0	3.4	49.1
kPCA	Fit	3.3	5.1	20.4	19.1	6.6
	Pred.	4.0	2.7	3.0	2.7	2.4
	Total	7.3	7.8	23.4	21.8	9.0
fAE	Fit	61.0	71.1	85.5	65.1	64.6
	Pred.	2.3	3.6	0.8	0.3	0.1
	Total	63.3	74.7	86.3	65.4	64.7
cAE	Fit	906	1012	959	954	1012
	Pred.	5.6	4.6	1.2	0.9	0.4
	Total	912	1017	960	955	1012

Table 3.4: The time in seconds required for each experiment combination to fit to the training data and predict on both test sets. Sample size is 5500 patches for fitting and 3000 patches for predicting. The NF results are per tree to aid comparison.

The highest accuracies are for the 1-SVM with a PCA or kPCA embedding where the AUC ROC curve is around 0.82. The IF and FMCD achieve similar to this with accuracies of 0.81 with the kPCA embedding. Overall the kPCA embedding gave the best results, PCA produced the poorest on average, and the autoencoder methods gave similar average AUC ROC. The lowest accuracy is LOF operating on the cAE embeddings where it roughly equates to random

chance. LOF has the poorest accuracy on average and the 1-SVM has the highest.

For the PCA embedding, the NF, IF and FMCD methods are most accurate when using a high number of components (>50%). kPCA used fewer than 6% of the total components for all methods except for the NF, where it used all. For the IF and FMCD methods, the kPCA embedding improves the accuracy over PCA. For the NF, LOF, and 1-SVM methods there is little difference in the two embeddings.

3.7 Discussion

My results demonstrate the key differences between the NF and four selected unsupervised abnormality detection methods when explored in terms of accuracy and runtime efficiency (relative time taken to complete computation) on CT lung data and embeddings of it. The highest accuracy for each method is:

- NF: 0.740 area under the ROC curve in 7.1 s using the raw data.
- IF: 0.807 area under the ROC curve in 21.8 s using kPCA.
- FMCD: 0.809 area under the ROC curve in 9.0 s using kPCA.
- LOF: 0.697 area under the ROC curve in 1525.6 s using the raw data.
- 1-SVM: 0.823 area under the ROC curve in 3.0 s using PCA.

However, the runtimes vary greatly — by about three orders of magnitude — between these methods. Of course, with any method there will be a trade-off between runtime efficiency (speed) and accuracy. In some clinical settings, such as stroke (see next chapter), speed is crucial and any method that takes many minutes or longer may not be considered acceptable. More generally,

these methods will be tying up compuational resource while completing, which is a limited resource at hospitals. If a method takes too long and is run too often, there could be a shortage of computational resource here. Other use cases, such as in the detection of an illness that needs to be diagnosed quickly to have a good prognosis, accuracy is highly valued and so a longer runtime is more acceptable.

3.7.1 Accuracy

The most accurate method is the 1-SVM, closely followed by the IF and FMCD. NF comes next and LOF has the poorest accuracy. Dimensionality reduction methods have a positive effect on accuracy for FMCD and 1-SVM and a negative impact on LOF and NF.

LOF relies on a distance measure to points in space. If dimensionality is reduced, information on distance between points is lost, and this has a noticeable impact on the accuracy.

The NF on the other hand should perform well when working with fewer features and information concentrated in these features. This appears to be down to the choice of stopping criteria. As the trees grow to a similar depth regardless of the number of features used, with few features I get a higher probability of some features being split multiple times as a tree grows. This serves to break larger clusters along one feature into smaller ones, which may not be optimal for finding abnormality.

The PCA and kPCA embeddings selected a large number of features for NF (340 and 400 respectively) but remained worse than using the raw data. This could be because in the medical imaging data the information is spread widely between the features, while in PCA and versions of it, the information becomes concentrated in a small number of features. The random nature of

feature selection can lead the novelty trees to have several low-information features in its path as opposed to using the raw data where features are more likely to be useful.

3.7.2 Speed

These times are the total of the fit and predict times. It is assumed that the fitting stage of the methods would take place in a research environment and the predicting stage in a clinical environment. This means the predict time is much more critical to the successful functioning of the method, provided the fit time remains feasible.

The time taken to complete an experiment is based on two key factors: The speed of the embedding method and the dimensionality of the data.

PCA is the fastest of the tested embedding methods (excluding None), followed by kPCA, then fAE, and finally cAE. For PCA, all experiments are faster than using no embedding, meaning the speedup from the dimensionality reduction outweighed the time increase from performing PCA. kPCA is the most accurate, on average, of all the methods, and PCA is the least accurate. This may be due to non-linear structure present in my data that can only be captured effectively by kPCA. The lower or comparable number of features selected for kPCA relative to PCA supports this idea - kPCA is more efficiently capturing the information in the data.

The LOF method is the most affected by the number of features due to the exponential increase in the number of calculations required with increasing features. The IF method is the least affected as it has no dependence on the number of features.

PCA has a substantial drop in performance for the IF method. This is likely due to PCA focusing the variance along a small number of features.

This affects the IF method as it relies on all features having useful variance so it can divide the data effectively no matter which features is selected at each node. Despite this, the IF method selected 237 features for its analysis. This means that having fewer features was not enough to usefully split the data as there was not sufficient information, but at the same time adding more features failed to improve the splitting more due to the lack of useful information in some features. This means the features of the sample patches were mostly of useful features.

The NF is generally the fastest method, although the stated times are normalised for the number of trees. When factoring in the use of 100 trees in the forest, NF becomes the slowest method generally.

The fast training time for the NF is what enables its overall speed. The IF has similar runtime as the NF for the raw data, but when the autoencoder embeddings are used the speed reduces by 1–2 orders of magnitude for the reasons stated above. The NF grows in the same way regardless of how useful the features are, so its fit time is not affected.

3.8 Conclusion

This chapter reviewed the novelty forest and four of the most influential abnormality detection algorithms, each representing a different family of solutions: local outlier factor, the one-class support vector machine, isolation forest, and fast-minimum covariance determinant estimator. These methods were evaluated on CT interstitial lung pathology imaging data under five embeddings: none (raw data), salient components from principal component analysis, salient components from kernel principal component analysis, embeddings from a flat autoencoder, and embeddings from a convolutional autoencoder. The aim was to correctly distinguish abnormality from normality and to do this in a

time-efficient manner.

Local outlier factor method had the lowest accuracy and was poorly suited to datasets with a large number of features. This is due to it calculating distance as a vector through all dimensions. The fast-minimum covariance determinant estimator with its Gaussian fitting showed better scaling with the number of features but the effect of the number of features on speed was still noticeable. Its overall performance varied depending on the embedding from poor to good. The isolation forest and one-class support vector machine were the least affected by the number of features. The isolation forest uses a number of divisions equal to the number of samples minus one, so has no significant dependence on the number of features. Finally, the novelty forest was fast, but the performance poor due to its inability to capture all dimensions — even relative to the isolation forest, which is an entirely stochastic method, the novelty forest's accuracy was much weaker on average.

The one-class support vector machine was the most accurate, closely followed by the isolation forest and fast-minimum covariance determinant estimator, then the novelty forest and local outlier factor.

Kernel principal component analysis was the most effective embedding technique, leading to the highest average accuracy, but the effect of each embedding varied across the methods. The isolation forest appeared to be well-suited to complex medical datasets. However, concern is noted over the isolation forest's ability to explore all features in datasets with a large number of features relative to number of samples, which may lead to it missing small pathologies. The novelty forest was fast with good general accuracy.

Next Steps: This chapter concludes my work on novelty. The next chapter examines the problem of distillation for a voxel level output into a single, interpretable quantity.

Chapter 4

Ischemic Stroke Severity Analysis in Non-Contrast CT using an ASPECTS Atlas

4.1 Abstract

Determining the severity of early ischemic stroke in non-contrast computed tomography (CT) is critical to ensuring a patient is treated effectively and rapidly. However, due to a low signal to noise ratio and confounding factors from other pathologies, past events, and natural age-related changes within the brain, this is not an easy task. These issues contribute to a variable interpretation of the ischemic stroke severity. Incorrect or delayed treatment can lead to an adverse outcome for the patient.

I investigate the level of agreement between four methods, including the use of an automated system, with the aim of identifying early ischemic changes within the brain. For the evaluation I divide the Middle Cerebral Artery (MCA) territory of each hemisphere into ten regions defined according to the

Alberta Stroke Programme Early CT Score (ASPECTS). These regions allow us to calculate the ASPECTS – a score indicating the severity of the stroke – for each method, and effectively distills the outcome into something interpretable at the patient level.

The automatic system uses a specialised Convolutional Neural Network (CNN) to produce voxel-level confidence masks showing which voxels are suspected of showing early ischemic change. From this I compute the score. I also obtain the score from three other methods that involve trained human graders, and compare the level of agreement between these methods at both a patient level and a territory level through Simultaneous Truth and Performance Level Estimation (STAPLE) and Cohen's kappa coefficient. I analyse possible causes of disagreement between the methods and statistically validate the performance of the CNN model against the performance of clinical staff (the professional standard). I find that the CNN produces scores that correlate the greatest with its training data at the patient level, but the training data could be improved to strengthen the correlation with the professional standard. Improvements would come from improved ground truth.

4.2 Introduction

This chapter focuses on the problem of distillation in medical image analysis. Using non-contrast CT (NCCT) brain scans of patients with suspected ischemic stroke, I utilise a specially designed neural network to provide voxel-level probability maps of the stroke-affected regions. I then distill these probability maps through a brain region mapping and processing rules to provide a single clinically meaningful, easily interpretable, and reliable quantity for each patient that can be used to influence that patient's treatment.

During an ischemic stroke, blood supply is lost to a region of the brain

due to a vessel clot. This requires immediate medical attention as the affected tissue will die if blood flow is not promptly restored. To select the appropriate treatment for a patient exhibiting stroke symptoms, the cause of the symptoms must be identified - ischemic stroke, hemorrhagic stroke, or stroke mimic - and the severity determined [171]. Stroke symptoms include lateral weakness manisfesting as, for example, drooping of one side of the face with the patient having reduced ability to smile or speak clearly, inability to lift both arms and keep them in position due to weakness or numbness in one arm, or other unilateral symptoms. Other potential symptoms of stroke include paralysis of one side of the body, vision loss, blurring, or double vision, dizziness, confusion, difficulty interpreting speech, problems with balance and coordination, dysphagia (swallowing difficulties), a sudden and severe headache, and loss of consciousness [218].

A similar condition to stroke is the transient ischemic attack (TIA) [5]. This is essentially a more minor version of a stroke resulting in similar symptoms but that clear within minutes or hours. A TIA is a warning sign of stroke as its root causes are the same [14].

Strokes and TIAs are considered medical emergencies requiring immediate medical attention. There are several tests for strokes, such as FAST [223], FASTER [224], and BE-FAST [12], which evolved from the Cincinnati scale [159] used previously. These methods focus on the rapid identification of symptoms. During a stroke the average patient loses as many brain neurons per hour as 3.6 years of normal aging would cause, and over the duration of the stroke the patient may lose almost 2 million neurons, 14 billion synapses, and 12 km of anonal fibres [252]. 15 million people globally are affected by stroke each year with 5 million of these resulting in death and a further 5 million leading to permanent disability [47]. The number of stroke cases is increasing,

driven primarily by people living longer. The number of stroke incidents in developed countries may double by 2050 from 2010 numbers (16.9 million [90]) with those aged 75+ contributing the most to this rise [130].

NCCT is a widely available and rapid means of imaging the brain. It is frequently used as the first step in identifying the stroke cause and classifying its severity [167]. However, early ischemic changes are challenging to detect and the boundary between affected and normal brain tissue is poorly defined [293]. A range of features with different pathophysiological bases may present including hyperdense vessels, loss of the insular ribbon, obscuration of the lentiform nucleus, loss of grey-white matter differentiation, sulcal effacement, and hypoattenuation [111]. Conflation of these features contributes to variable interpretation of stroke severity [109, 291]. This makes it difficult to compare the performance of an external automated system for detecting early ischemic changes to clinical diagnosis.

The Alberta Stroke Programme Early CT Score (ASPECTS) is a clinically meaningful system of scoring stroke severity that concentrates on hypoattenuation as this is the most relevant indication of the severity of ischemia [233]. ASPECTS determines the presence or absence of acute hypoattenuation in ten regions within the middle cerebral artery territory of each hemisphere and can influence treatment choice [22] The score starts at 10 for a normal brain volume and reduces by one point for each affected territory in the symptomatic hemisphere. Patients with a score of 7 and higher are most likely to respond positively to treatment [126].

ASPECTS is a means of distilling down the information from a patient NCCT brain scan into: first a brain territory level, then a patient level score. The issue of distillation appears frequently in the field of medical imaging. Every stroke is unique and to summarise one as a single number on a stan-

dardised scale is not straightforward. Throughout this chapter I will present and discuss four methods of determining ASPECTS, and statistically compare them using Simultaneous Truth and Performance Level Estimation (STAPLE) [294] and Cohen's kappa [51]. The methods consist of three based on manual reading, and one using an automatic method. The performance is judged at two interpretation levels: a dichotomised patient level and an ASPECTS territory level.

The remainder of this chapter is structured as follows: Section 4.3 provides further details on ASPECTS. Section 4.4 details previous work on ischemic stroke classification. Section 4.5 provides the sources and information of my datasets. Section 4.6 explains four scoring methods. Section 4.7 shows my performance evaluation. Sections 4.8, 4.9, and 4.10 are my results, discussion, and conclusions respectively.

My evaluation is implemented in the Python programming language (version 2.7) [92].

4.3 ASPECTS

The carotid arteries carry blood up from the torso to the brain, neck, and face. Within the head they divide into the internal carotid arteries, which supply the brain with blood. These internal carotids divide at the Circle of Willis where a pair of major arteries called the Middle Cerebral Arteries (MCAs) begin. The MCAs each supply blood to most of a hemisphere of the brain – Figure 4.1 shows the areas they supply as the large central region.

ASPECTS is a clinically meaningful score from 0 to 10 in integer steps where 10 indicates a healthy brain and 0 indicates severe lateral ischemia. It is determined by ten regions supplied by the MCA in each hemisphere consisting of the subganglionic nuclei, supraganglionic nuclei, and basal ganglia regions.

These territories are given one or two character reference identifications (IDs) detailed in Table 4.1, and marked on a NCCT brain scan in Figure 4.2.

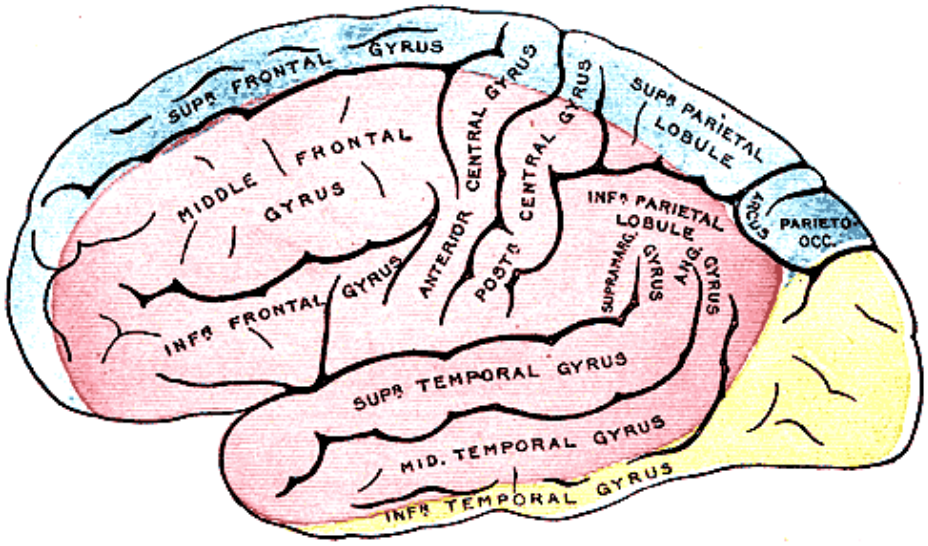


Figure 4.1: A schematic of the brain showing the region supplied by the middle cerebral arteries as the large central region, extending from the middle frontal gyrus (upper left of centre) across to the inferior parietal lobule and angular gyrus (right of centre) down to the mid temporal gyrus (lower centre). This is the pink region on a colour print. Figure from the public domain.

To determine the ASPECTS of a patient I count the number of territories affected by early ischemic change in each hemisphere. I then take the highest (most affected) of the hemispheres and subtract the number of its affected territories from ten. The result is the ASPECTS. See Equation 4.1.

$$\text{ASPECTS} = 10 - \max(T_{left}, T_{right}) \tag{4.1}$$

where T_{left} is the number of ischemic territories in the left hemisphere, and T_{right} is the number of ischemic territories in the right hemisphere.

ID	Region	Full Clinical Name
M1	Subganglionic Nuclei	Frontal Operculum
M2	Subganglionic Nuclei	Anterior Temporal Lobe
M3	Subganglionic Nuclei	Posterior Temporal Lobe
M4	Supraganglionic Nuclei	Anterior Cortex immediately rostal to M1
M5	Supraganglionic Nuclei	Lateral Cortex immediately rostal to M2
M6	Supraganglionic Nuclei	Posterior Cortex immediately rostal to M3
C	Basal Ganglia	Caudate
L	Basal Ganglia	Lentiform Nucleus
I	Basal Ganglia	Insula
IC	Basal Ganglia	Internal Capsule

Table 4.1: The ten ASPECTS territories showing their shortened ID name, general brain region, and full clinical name.

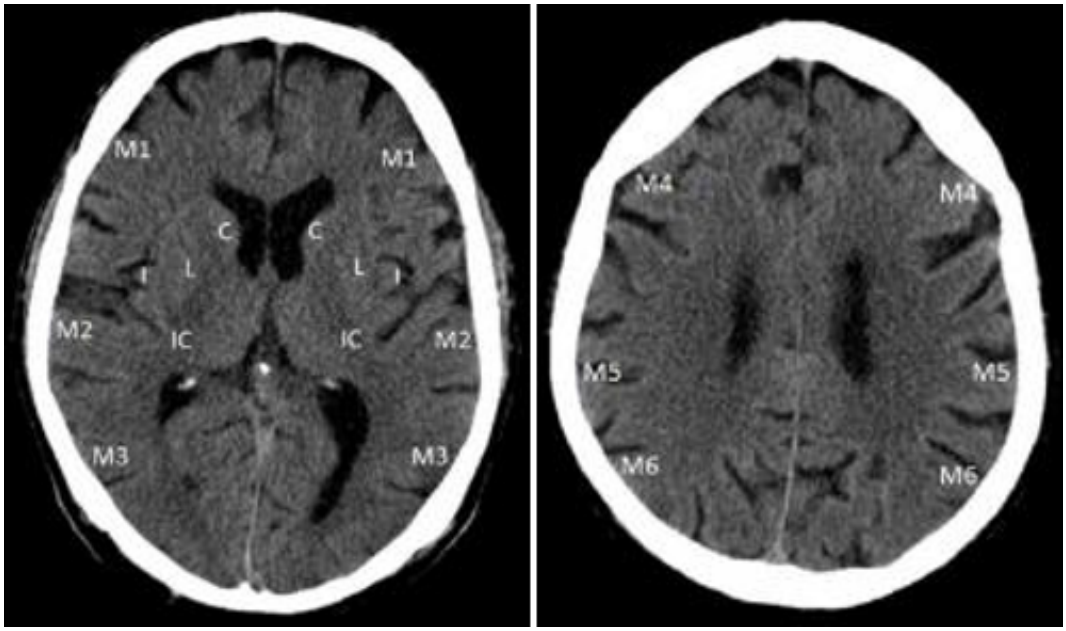


Figure 4.2: Two axial slices of a healthy human brain showing the ganglionic level (left) and supraganglionic level (right) with the ASPECTS territories labelled with their IDs.

4.4 Previous Work

This section covers previous work in classifying stroke. I start by exploring the 1/3 MCA rule, which is an alternative to ASPECTS. After this I detail other work on ASPECTS and follow with other techniques for measuring stroke severity.

4.4.1 The 1/3 MCA Rule

Prior to ASPECTS clinicians used the 1/3 MCA rule [287] to determine whether or not to treat patients with thrombolysis (breaking down the clot using alteplase). This rule states that one must determine if the affected volume of the MCA territory is greater than a third of the total MCA volume. If it is, thrombolysis is not recommended as it is associated with worsened outcome. ASPECTS on the other hand recommends thrombolysis for scores of 8 or higher.

Contra-indications for thrombolysis include hemorrhage or history of hemorrhage, time of symptom onset being greater than 4.5 hours (the more time that passes, the less effective this treatment is), neurosurgery, head trauma, or stroke in the past three months, hypertension, endocarditis, a low blood glucose level, and the patient taking anticoagulants (although this may be countered by administrating coagulants). The use of thrombolysis for severe strokes, which are usually caused by a larger blockage, leads to an increased risk of hemorrhagic transformation (bleeding) [101], which is a serious complication that can be fatal [15].

In cases where thrombolysis is not suitable, an alternative treatment is available: thrombectomy [219]. This involves using surgery to remove the clot. This treatment has risks and may lead to bleeding (potentially causing death), infection, or damage to the blood vessel at the site of the clot [202]. There is

also a risk of the clot breaking into smaller pieces, which may in turn cause further clots in other vessels. Research by El Tawil and Muir indicate that using thrombolysis after thrombectomy treatment can improve outcome by breaking down these smaller pieces [83].

A study by Mak *et al.* showed that the 1/3 MCA rule was more reliable in detecting significant early ischemic change on CT brain within 6 hours of stroke onset, whereas ASPECTS was able to detect significant early ischemic change in a higher proportion of these scans [197]. Further research has shown that a more systemic approach improves the reliability of the 1/3 MCA rule [142, 270]. Dzialowski showed that for patients where the 1/3 MCA threshold is reached, the median ASPECTS is 4 [81].

4.4.2 Work on ASPECTS

Kobkitsuksakul *et al.* studied interobserver agreement in 2018 and found the ASPECTS interpretation varied among raters [153]. Using 43 patients, the Cohen's kappa coefficient [51, 103, 271] varied from 0.49 to 0.68 for consensus between two neuroradiologists and a neuroradiology fellow, and from 0.20 to 0.49 for consensus between two neuroradiologists and a senior radiology resident. These ranges derive from three window level settings.

Coutts *et al.* also studied the interobserver variation for ASPECTS [57]. They took 214 patients presenting acute ischemic stroke or transient ischemic attack and who had a head CT scan performed within 12 hours of symptom onset. Each scan was read by the treating physician and later by an expert reader, and the ASPECTS determined. The Cohen's kappa coefficient was 0.69, and the mean score difference between the two raters was 0 with a standard deviation of 1.1.

Aviv *et al.* took the ASPECTS of 36 patients with acute stroke and NCCT

taken within 3 hours of symptom onset and found ASPECTS is as predictive of radiologic outcome as cerebral blood volume from the the NCCT, however ASPECTS may have the additional benefit of predicting patients with major neurological improvement [16]. The ASPECTS were assigned by 3 neuroradiologists.

Puetz *et al.* have a detailed report from 2009 titled *The Alberta Stroke Program Early CT Score in clinical practice: what have we learned?* that explains a range of practical findings related to the score. Findings include the presence of focal brain swelling confounding the interpretation of the score, the 1/3rd rule providing similar outcomes to ASPECTS, and the lack of evidence proving the effectiveness of using ASPECTS to inform thrombolysis treatment [238].

4.4.3 Other Techniques for Measuring Stroke Severity

ASPECTS is one of several techniques to measure stroke severity. Here I list other popular alternatives that are used alongside or instead of ASPECTS in practice. These can be divided into three categories: Prehospital, Acute, and Outcome.

Prehospital measures are assessments completed before taking a patient to the hospital. The purpose here is to reduce the number of false stroke cases (false positives) or categorise the severity of these cases to speed up treatment decisions later.

Acute assessments examine patients in the acute phase of a stroke — when it is happening or shortly after. ASPECTS is an example of an acute assessment. These assessments focus on the symptoms of the stroke, which can include physical changes immediately visible to the clinician, such as unilateral weakness or inability to perform certain tasks, or they may focus on results

from medical scans, such as ASPECTS.

Outcome measures look long-term into the level of disability the patient has after recovering as much as possible. These cannot be used during the acute phase, but can serve as input to future decisions on the patient should they experience a further stroke. They serve as a means of translating a complex set of symptoms into something interpretable at the patient level. This is similar to the aims of ASPECTS.

4.4.4 Prehospital Assessments

ABCD Score [247] The ABCD — or Age, Blood pressure, Clinical features, and Duration — score is a simple means of classifying the likelihood that someone is having a stroke based on a small number of characteristics [247]. These are the age and blood pressure of the patient, the presence of any clinical features such as motor weakness or speech disturbance, and the duration of these symptoms. This information may only display a risk factor, but may inform clinical treatment should the risk be particularly high or low.

Cincinnati Stroke Scale [159]: This is a short physical assignment consisting of two tasks: Raising arms and speaking. The patient is assessed on three criteria and on meeting all three is sent for stroke treatment: (1) One side of face does not move when speaking; (2) speech is slurred, inappropriate, or mute; (3) one arm drifts compared to the other. A study by Kothari *et al.* showed the scale had high reproducibility and validity [159] and in earlier work showed it had high sensitivity and specificity when identifying patients for thromolysis [158].

Los Angeles Prehospital Stroke Screen (LAPSS) [149]: LAPSS is an as-

sessment to determine the probability that a patient is having a stroke. It is designed around a number of risk factors and examines the lateral asymmetry of the face, grip, and arm strength to determine if a patient is likely having a stroke. The risk factors are an age over 45, no history of seizures or epilepsy, symptoms present for less than 24 hours, patient is not wheelchair bound or bedridden, and a blood glucose level between 60 and 400 [149]. Although these do not capture every possible stroke patient, it does help to highlight contraindications to stroke that may require further investigation.

Chen *et al.* reviewed the LAPSS diagnosis with 1130 patients and found that out of the 795 patients this criteria identified as having a stroke, 782 (98.4%) were clinically determined to be stroke patients [48]. However, out of the 335 patients it did not identify as stroke patients, 215 (64.2%) were in fact experiencing a stroke. The Los Angeles Prehospital Stroke Screen provides a low false positive rate, but has a high false negative rate. The author states that if the criteria is not met, the patient should continue to be assessed following standard medical protocol [149]. This will allow false negative cases to be caught.

4.4.5 Acute Assessments

National Institutes of Health Stroke Scale (NIHSS) [40, 214, 296]: NIHSS consists of 15 items that are scored on a scale of 0 to 2, 3, or 4 each, depending on the item. The maximum score is 42. This is a largely physical assessment and features areas such as vision, speech, arm and leg movement, sensation, and memory. It does not factor in other neurological or physical issues into the assessment, which can lead to inaccurate analysis. This test takes time to perform, unlike ASPECTS which can

be automated from NCCT data in near real time.

Schlegel *et al.* showed that each point on NIHSS decreased the likelihood of "excellent outcome" by 24% at one week and 17% at three months [254]. Meyer *et al.* found that using a simpler scale or 11 items improved interobserver reliability [206] over the original scale.

Six Signs and Symptoms [240]: This is a newer method from 2014 derived from NIHSS that uses six signs of stroke to determine its score between 0 (no symptoms) and 15 (severe stroke). Its advantage over NIHSS is its simplicity, allowing it to be carried out faster and with less prior training. It has shown a similar performance to NIHSS [240].

European Stroke Scale [117]: The European Stroke Scale is used when a patient has recently had a stroke involving the area of brain supplied by the middle cerebral artery — the same as for ASPECTS. This scale measures the impact a stroke has had on a patient's consciousness. comprehension, speech, visual ability, and a range of physical activities involving the face, arms, wrist, fingers, legs, feet, and gait. A final score out of 100 captures the severity of the stroke.

Hemispheric Stroke Scale [3]: This is a comprehensive survey that, like the European Stroke Scale, scores the patient out of 100. The areas covered range from language — such as comprehension, naming, repetition, and fluency; to visual fields and gaze; facial expression; ability to draw shapes; arm and leg movement; muscle tone; gait; and touch recognition and sensitivity.

Orgogozo Stroke Scale [227]: Similar to the Hemispheric Stroke Scale, this is a comprehensive survey covering the same general areas but separates muscle tone into upper limb and lower limb, and does not examine mental

capacity beyond the level of consciousness. This effectively make it a faster, although weaker, assessment. It predates the Hemispheric Stroke Scale by four years.

Canadian Neurological Scale [56]: This assessment measures the patient's mentation for levels of alertness, orientation, and speech, and then checks for weakness or asymmetric strength in the face, arms, and legs. Cote *et al.* showed the interobserver reliability on each scale item was "good" [55]. A later study compared it to the NIHSS where Bushnell *et al.* showed that for retrospectively assigned NIHSS and Canadian Neurological Scale scores, levels of interobserver agreement were almost perfect [42].

Hunt & Hess Scale [132, 133]: This quick survey is aimed at non-traumatic sub-arachnoid haemorrhage patients and consists of a single evaluation. The five-point scale ranges from mild headache and neck stiffness to more serious symptoms such as drowsiness/confusion and focal neurologic deficit and through to the most severe score for a comatosed patient.

Scandinavian Stroke Scale [110]: Focusing on ischemic stroke, the patient is scored for level of consciousness, eye movement, and motor power of the arms and legs. The severity of the symptoms indicates the severity of the stroke.

4.4.6 Outcome Assessments

Modified Rankin Scale [19, 277, 285]: Introduced in 1957 by Dr. John Rankin and modified into scale form by Van *et al.* [285], this seven-point scale rates patient outcome by perceived disability from a score of

0 meaning no symptoms, through to a score of 6 — patient is deceased. The in-between scores are 1: No significant disability despite symptoms; able to carry out all usual duties and activities; 2: Slight disability; unable to carry out all previous activities, but able to look after own affairs without assistance; 3: Moderate disability requiring some help but can walk without assistance; 4: Moderately severe disability; unable to walk and attend to bodily needs without assistance; 5: Severe disability; bedridden, incontinent and requiring constant nursing care and attention.

Quinn *et al.* show the scale may not be reliable with significant interobserver variability observed [239]. An average Cohen's kappa coefficient of 0.42 — or "moderate" correlation — is noted in interobserver agreement over ten studies that range from 0.25 ("poor") to 0.72 ("good") [28, 44, 115, 207, 215, 239, 285, 301, 302, 304]. Wilson *et al.* found that using structured interviews to rate a patient on the scale improved the reliability and decreased the variability and bias between observers [301]. Cohen's kappa coefficient improved from 0.25 ("poor") to 0.74 ("good") when using a structured interview [302]. Bruno *et al.* showed that using a simplified set of interview questions consisting of two to three questions before determining the modified Rankin score reduced assessment time from 5 minutes for a regular questionnaire to 1 minute and 40 seconds [41]; interobserver agreement was good with a Cohen's kappa of 0.72.

Glasgow Outcome Scale [136]: The scale is very similar to the Modified Rankin Scale. This scale scores a patient from 1 to 5 based on disability. 5: No significant disability, 4: disabled but independent, 3: conscious but disabled, 2: persistent vegetative state, 1: patient is deceased. This scale combines the lowest two categories of the Modified Rankin Scale and is focused on simple, broad categories. This can allow it to be used

faster and to a similar effect as the Modified Rankin Scale.

Barthel Index [52, 108, 184, 195]: The Barthel index measures the patient's independence after having a stroke. A number of activities are scored on a scale and then the total score provides the impact of the stroke on the patient. Activities include toilet use, bathing, feeding, dressing, transfers (bed to chair), and stairs. None of these activities are completed on site, but are instead discussed with the patient.

Berg Balance Scale [26, 27]: The Berg Balance Scale has a number of motor activities that are completed under the supervision of the clinician. These include actions involving standing, sitting, reaching down or forward, and turning around. These are scored between 0 and 4 and then summed. The final score is interpreted into three categories: wheelchair bound, walking with assistance, and independent.

Lawton IADL Scale [174]: Instrumental Activities of Daily Living (IADL) is a high level set of categories for the mental and physical capacity of the patient. It measures the functional impact of emotional, cognitive, and physical impairments. There are eight areas that are discussed with the patient: the ability to use a telephone, shopping, food preparation, housekeeping, laundry, transportation, medication, and finance management.

Stroke Specific Quality of Life Measure [299, 300]: This measure focuses on the patient's interactions with their environment as well as their own abilities. These topics are discussed rather than tested in the clinical setting. Categories of questions cover energy, family roles and social roles, but also language, mobility, and self care. Personality, mood, and productivity are also discussed. This makes this measure one of the most

comprehensive outcome assessments for stroke and needs post-stroke.

Other assessments of note that do not fit into the above categories are the Mathew Stroke Scale [198] which is a combination of other tests and physical and mental assignments, the Oxfordshire Community Stroke Project Classification (Bamford) [18, 201] that uses a broad set of patient symptoms to determine the size/severity of the stroke, the Action Research Arm Test [45, 62, 68, 187] — a short assessment looking at the arm and hand actions of the patient. It is used at various stages of recovery to quantify the recovery rate, although does not quantify stroke symptoms beyond the arms and hands. The test consists of grasp, pinch, grip, and gross movement of the hand or arm. The Hachinski Ischaemia Score [116, 209] examines the speed of onset and progress of symptoms as well as the history of a patient to find the stroke's impact and risk.

4.5 Datasets

Ie use data from three previous studies on patients with suspected stroke: ATTEST [131], POSH [190], and WYETH [292] (detailed below) provided by the Institute of Neuroscience and Psychology, University of Glasgow, Southern General Hospital. Specifically, I made use of the NCCT volumes, radiology reports, and ASPECTS for 156 patients. The NCCT scans were taken within 6 hours of symptom onset. The ASPECTS provided by these studies represents the professional standard.

Ground truth was collected on the NCCT scans using manual segmentation with the program 3D Slicer (4.5.0) [89] by an internal clinical expert (see Section 4.6) on a per-slice voxel-level basis.

ATTEST [131] (Alteplase versus Tenecteplase for Thrombolysis af-

ter ischaemic Stroke) The aim of this study was to comparatively assess two drugs: alteplase and tenecteplase. These can be used to treat acute ischemic stroke. Patients eligible for intravenous thrombolysis within 4.5 hours of stroke onset were recruited. The outcome of this study showed that of the 71 patients (35 assigned tenecteplase and 36 assigned alteplase) no significant difference was found in terms of the efficacy and safety of the treatments.

POSH [190] (Post Stroke Hyperglycemia) This study investigated whether post stroke hyperglycaemia has an impact on the infarct growth and brain arterial patency. The patients recruited for this study presented with acute ischemic stoke (within 6 hours of stroke onset) and were divided into three groups: 17 normoglycaemic patients at all time, 59 patients hyperglycaemic on admission and 32 patients that became hyperglycaemic 6 or more hours after admission. Findings suggested that the patient hyperglycaemic on admission tended to have larger infarct volumes at the 24-48 hour stage than normoglycaemic and late hyperglycaemic patients [189] .

WYETH [292] This study verified the predictive value of perfusion and angiographic imaging for clinical outcomes. 83 patients with potentially disabling stroke were recruited in three stroke centres. Two sets of imaging took place. First 76% were imaged with CT and 24% with MRI within 6 hours from the stroke onset. Second, 72 hours later, they were imaged with the same modality again. Researchers found that recanalisation at 72 hours on the angiography volumes predicted clinical outcome more directly than tissue reperfusion.

4.6 Scoring Methods

Further to the ASPECTS I have calculated the score via three other means, described below, leading to four overall methods of determining the ASPECTS for each patient:

1. From clinical studies, as taken from the datasets (defined in Section 4.5) themselves.
2. From the manual segmentation of ischemic and non-ischemic regions by an expert.
3. From visual inspection of the NCCT volumes by an expert.
4. From an automatic segmentation using a CNN algorithm.

Clinical Score: These scores were generated as part of the aforementioned studies (Section 4.5). ASPECTS was calculated by two clinicians reviewing the data independently. A third clinician resolved discrepancies. The experience and background of the reviewers varied.

Ground Truth Score: At my host company there is an in-house member of staff, who I will refer to as the clinical expert throughout, with 16 hours of training in detecting stroke signs. They segmented regions of acute ischemia in the NCCT volumes under the supervision of an experienced neuroradiologist. The clinical expert additionally created an atlas volume of the twenty ASPECTS territories (refer to Section 4.3) on a patient whose regions were clearly defined and otherwise unremarkable. These two tasks were completed in the program 3D Slicer (version 4.5.0) [89] at the slice level.

The atlas is rigidly aligned to each ground truth volume via landmark registration using automatic landmarks [65] to identify the territories for

that patient. Rigid alignment is chosen to prevent non-rigid distortions of the atlas creating artifacts that could undermine clinical validity. However, rigid alignment does introduce an error in the boundary locations of the ASPECTS territories.

To convert the ground truth to ASPECTS two rules are applied. These rules are based on work by Kosior *et al.* [157] in the absence of an accepted standard for calculating ASPECTS.

1. The first rule – which I call the 5% rule – states that any territory with ground truth ischemia present in more than 5% of its volume is classified as ischemic. This is defined mathematically in Equation 4.2.

$$\text{Territory Ischemia} = \text{Present if } \left(\frac{1}{|D|}\sum_{i\in D} GT_i\right) \geq 0.05 \qquad (4.2)$$

   where D is the set of voxels in a territory and GT_i is the ground truth label for voxel i.

2. The second rule - the small volume rule - is applied only if no territories meet the 5% rule but there is ground truth ischemia present in at least one territory. This rule states that the territory with the highest volume of ischemia ground truth present in it, and only this territory, is classified as ischemic. This results in an ASPECTS of 9. The total volume of the territory is not taken into account – only the volume of ischemia ground truth in each territory is used. Equation 4.3 demonstrates this.

$$\begin{aligned}\text{Territory Ischemia} = \text{If } \forall \text{ territories } \left(\frac{1}{|D|}\sum_{(i\in D)} GT_i\right) < 0.05 \\ \text{and } \left(\sum_{i\in D_{all}} GT_i\right) > 0 \\ \text{then present in } max\left(\sum_{i\in D} GT_i\right) \text{ only.}\end{aligned} \tag{4.3}$$

To explain: The first line checks to see if any territories meet the 5% rule (in which case Equation 4.2 is used), the second line checks for the set of voxels in all territories (D_{all}) if any ground truth above zero is present, and if so, the third line says the territory with the highest total value of ground truth is classed as having ischemia present.

Observed Score: Six months after the completion of the ground truth segmentations, the clinical expert visually examined each of the data volumes by using 5 mm slabbed slices and noted which territories presented acute ischemia. The ASPECTS came directly from the number of affected territories.

CNN Score: A CNN architecture based on previous work by Lisowska *et al.* [180, 179] is used to produce a voxel-level confidence mask of early ischemic change in each volume. The CNN features two identical intensity channels consisting of a series of three orthogonal 1D convolutions with kernel size of 5 x 1 x 1 , 1 x 5 x 1 , and 1 x 1 x 5 and 16 kernels. The two channels then merge, and atlas coordinates are input. The set of convolutions is repeated with two kernels. Figure 4.3 demonstrates the design. The use of the left hemisphere and right hemisphere channels provides a hemispherical (bilateral) comparison, which is used by clini-

cians to diagnose stroke — stroke usually affects only a single hemisphere due to the brain's major blood vessels each only connecting to a single hemisphere. This comparison helps rule out false positives caused by age-related changes in the brain. These changes exhibit a high degree of symmetry.

I use samples of approximately 20 million voxels from folded slices. Voxels are given positive confidences if ischemia is suspected by the CNN, and negative confidences if no ischemia is suspected. There is no upper or lower bound on the confidence values.

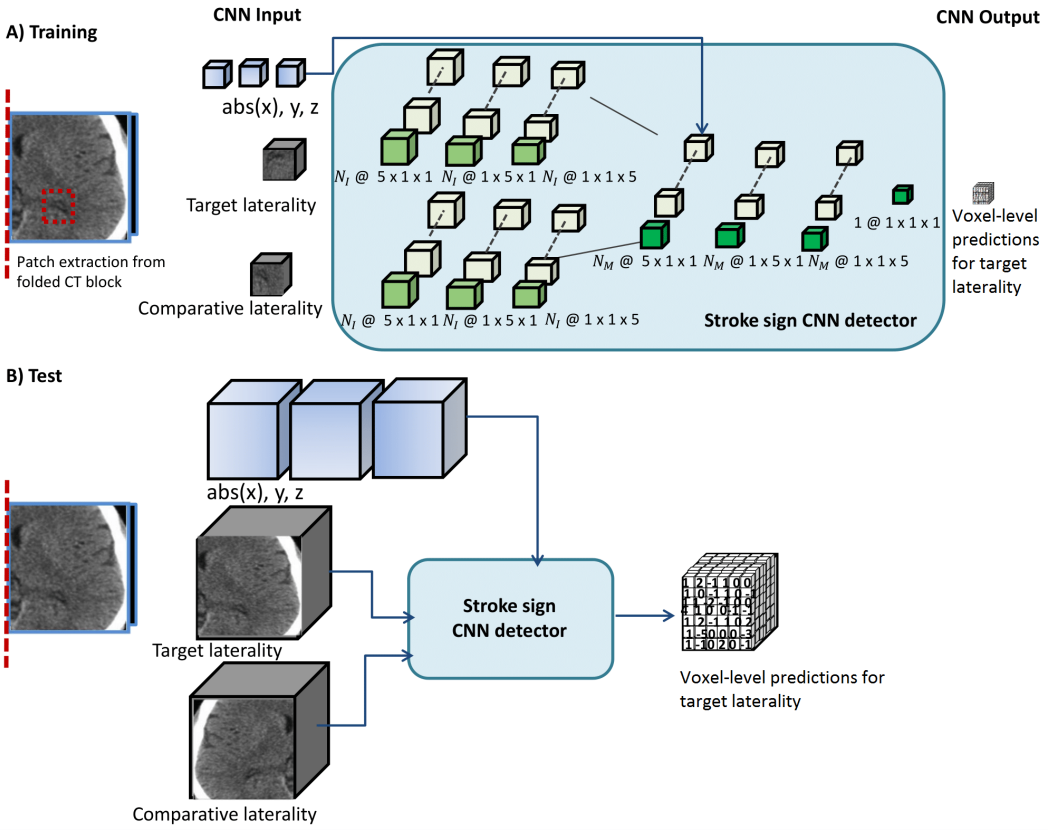


Figure 4.3: **A)** Schematic of the CNN showing the filter sizes and the number of layers. Pairs of contralateral 3D image intensity patches are input to the network at training time. Atlas coordinate inputs are then fused at the merge point of the intensity channels. **B)** Application of the network at test time. Whole folded slices are input to the network, but predictions are generated separately for each hemisphere. Modified from [180].

The ground truth segmentations are used to train and validate the CNN with five-fold cross-validation. For each fold 40% of the training data is kept separate as a validation set. The ASPECTS atlas is aligned to the voxel confidence masks in an identical manner to the Ground Truth Score. Figure 4.4 shows an example detection with ground truth and the atlas.

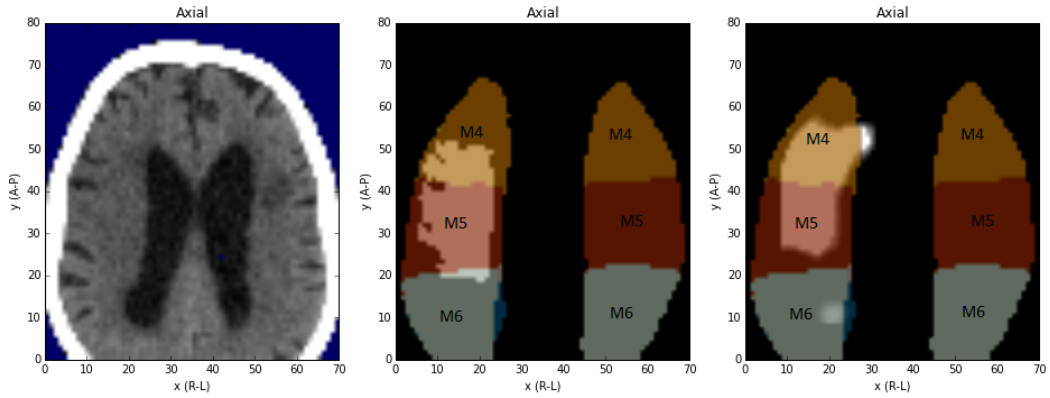


Figure 4.4: **Left:** NCCT volume axial slice of a dataset showing the M4 to M6 regions of the MCA. Centre and **Right:** The ASPECTS atlas for the shown NCCT slice with the M4-M6 territories marked for each hemisphere. The brighter regions in the left of the images show segmented ground truth ischemia by the clinical expert (Centre) or suspected ischemic voxels from the CNN (Right) for that slice. A-P refers to the Anterior to Posterior axis; R-L references the Right to Left axis.

The voxel confidences are converted into ischemic territories via Equation 4.4, which averages across the suspected ischemic (positive) voxels in a territory and applies a threshold to determine if this average is sufficient for the region to be classed as ischemic. The threshold is selected by receiver operating characteristic (ROC) curve analysis to balance false positives with false negatives relative to the ground truth in the training set, resulting in a score that is unbiased across a large number of datasets.

$$\text{Territory Ischemia} = \text{Present if } \left(\frac{1}{|D|}\sum_{i \in D} c_i\right) \geq \text{t} \tag{4.4}$$

where D represents the set of voxels comprising the region, c_i is the confidence score of voxel i, and t is the threshold.

I dichotomise the four method scores at $\geq$7. A dichotomised ASPECTS can be clinically useful to weight treatment decisions and the use of a dichotomised

score is increasingly common in clinical trials [233]. For all the methods except the Clinical score I also have a territory breakdown. This level of detail was not provided with the Clinical score. I refer to this as the territory data.

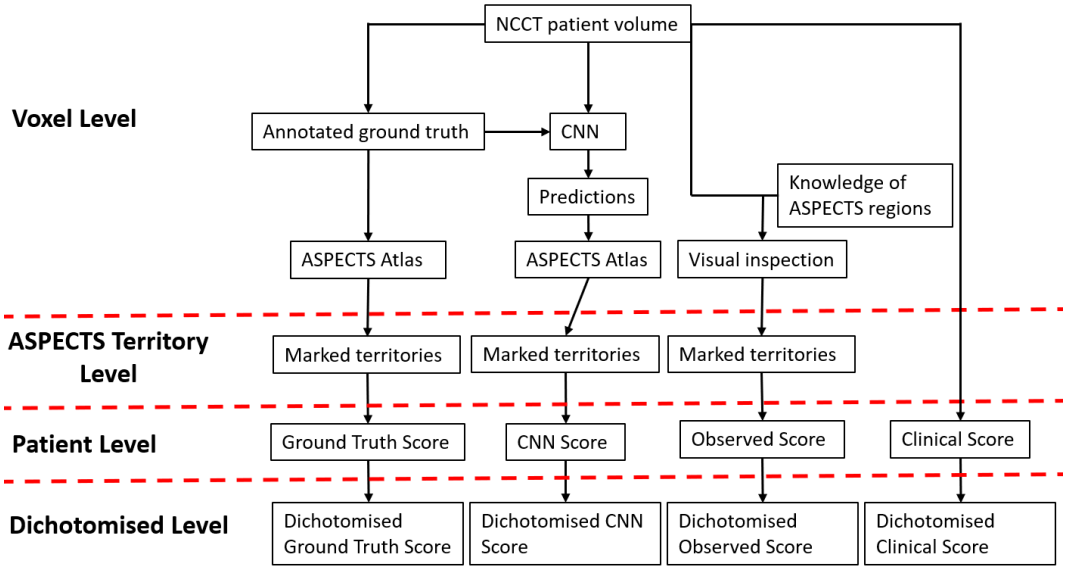


Figure 4.5: A high level overview of how each of the scores are arrived at from the original volume and the different interpretation levels each step exists within.

4.7 Evaluation Procedure

I cannot take any of these methods as a definitive gold standard, however the clinical score can be accepted as the closest. Statistical measures are required to compare the overall accuracy of each method to the clinical score and to each other. I apply two measures: Cohen's kappa [51, 103, 271] and STAPLE [294]. Each measure is applied to pairs of methods at a time for all possible pairings for the dichotomised ASPECTS and the territory results.

Cohen's kappa is a statistic to measure agreement between two sets of data while taking into account the probability of random chance agreement — therefore accounting for class imbalance. Cohen's kappa is defined as

$$\mathcal{K} = \frac{p_o - p_e}{1 - p_e} \tag{4.5}$$

where p_o is the observed agreement among methods, and p_e is the probability of chance agreement:

$$p_o = \frac{TP + TN}{n} \tag{4.6}$$

$$p_e = \frac{(TP + FN) * (TP + FP) + (TN + FN) * (TN + FP)}{n^2} \tag{4.7}$$

where n is the number of patients or regions in the data and *TP*, *FN*, *FP*, and *TN* are the number of True Positives, False Negatives, False Positives, and True Negatives respectively.

STAPLE is an algorithm that compares the performance of two or more results, in this case my four methods, with each other. STAPLE is an extension of the expectation-maximisation algorithm [211]. It does this by estimating a *true score* for each patient based on the score from each *expert* (the methods) and the perceived ability of each expert. The algorithm then calculates the performance of each expert against the true score, which gives the sensitivity and specificity of each expert. The process then repeats from the first stage until convergence.

The sensitivity is the fraction of positive cases correctly identified as being positive. The term positive here refers to an ASPECTS on the "more severe" side of the dichotomised scale (i.e. ASPECTS < 7) in the case of the patient-level evaluation, or to an ischemic territory in the case of the territory-level evaluation. The specificity is the fraction of negative cases correctly identified as being negative.

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{4.8}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{4.9}$$

4.8 Results

I use all available data from the studies (156 patients) and for each patient I collect the ASPECTS via the four methods at the patient level using the conversions explained in Section 4.6, and three methods at the territory level — the Clinical Score did not provide a territory-level break down. This is because it was not scored by the hospitals involved.

Figure 4.6 illustrates the distribution of the scores for each method. From the 156 patients, 52 of these reported a Clinical Score of 10, however our clinical expert recorded an ASPECTS of 10 in 62 cases (Observed Score), and there were 66 cases of such in the Ground Truth Score. Each method generally tails off towards the lower scores, but there are notable exceptions. The Clinical Score features a clustering of patients around the central scores (particularly scores 4–7) that the other methods do not exhibit. The Ground Truth score has a peak at 9 that is 33% larger than the method with the next highest bar of that score (CNN Score) and almost three times that of the clinical standard (Clinical Score). The CNN Score is the most likely to predict no ischemia within the territories.

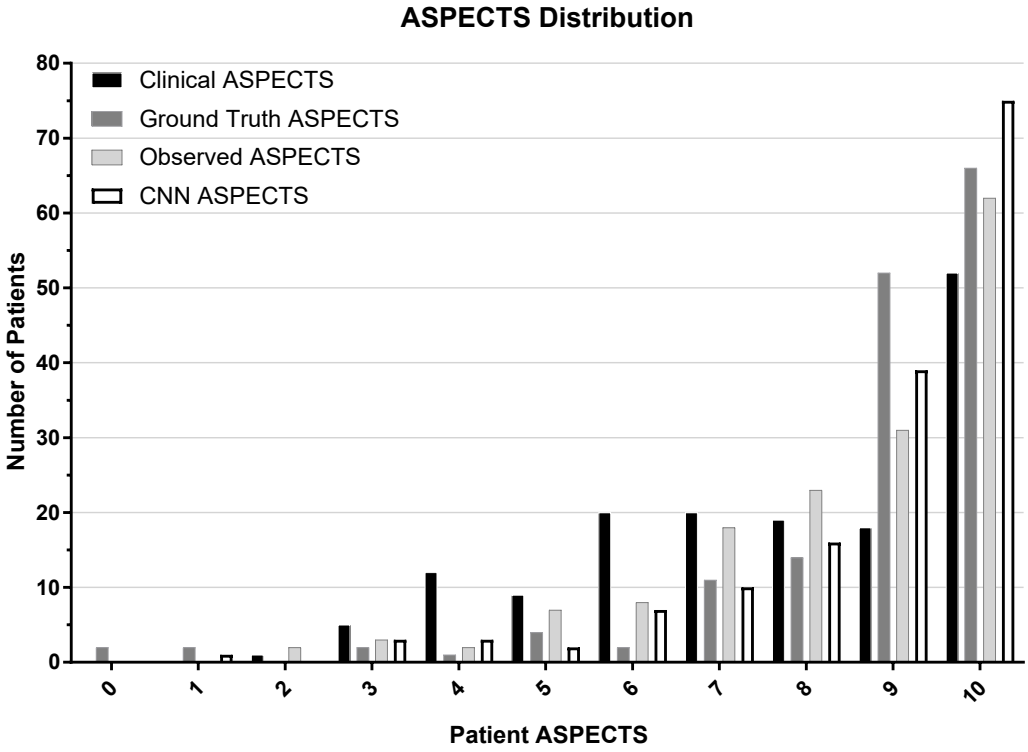


Figure 4.6: The distribution of the raw ASPECTS for each method.

At the patient level, the fraction of patients above the threshold for thrombolysis treatment ($\geq$7) varies between the methods: 70% of the Clinical Scores, 92% of the Ground Truth Scores, 86% of the Observed Scores, and 90% of the CNN Scores.

A total of 3120 (156 x 20) ASPECTS territories were scored by each of the three methods (Ground Truth, Observed, and CNN scores) as presenting ischemic signs or not. I calculated Cohen's kappa coefficient between pairs of the dichotomised scores and each pair of the territory scores. Figure 4.7 displays these values. The highest correlation is seen between the Ground Truth and CNN dichotomised scores, while the lowest set of correlations at the dichotomised level are the three comparisons with the Clinical Score. At the territory level there is a modest correlation between the territories calculated

as ischemic between the Ground Truth and the Observed methods, while a weaker kappa is present between the CNN methods and both of these.

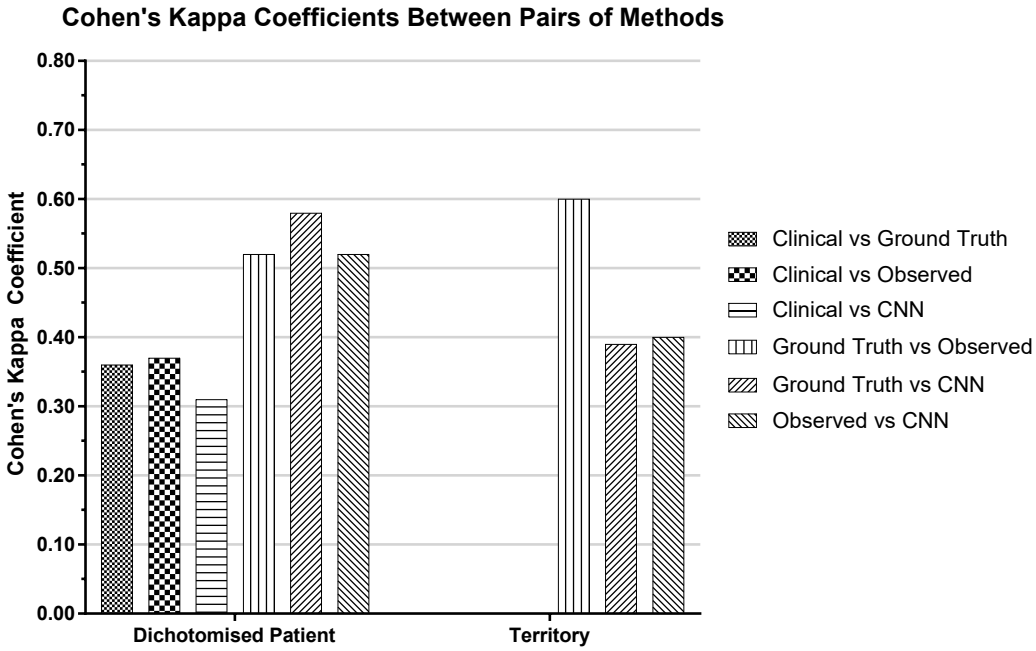


Figure 4.7: Cohen's kappa coefficient between pairs of ASPECTS methods for the dichotomised per-patient data and territory data.

For the voxel-level masks (CNN method) I evaluated these against the ground truth segmentations in terms of Area Under Curve (AUC) for the Receiver Operating Characteristic (ROC) curve and the Precision-Recall (PR) curve to get ROC AUC = 0.94 and PR AUC = 0.41.

The STAPLE algorithm determines the sensitivity and specificity of each of the methods at the dichotomised and territory levels.

Table 4.2 shows these values. Further to the evaluation against the STAPLE true score, each of the methods are evaluated against each other to produce sensitivity and specificity for each. Table 4.3 shows the evaluation at the dichotomised level, while Table 4.4 displays it at the territory level. In both cases the assumed true class is on the left and the assumed predicted class is

along the top. Sensitivity for methods evaluated against the Clinical score is low in all cases, but specificity is high. Specificity is expected to be high due to the large number of true negatives in the data relative to true positives. This means any false positives only have a minimal effect on the specificity each. On the other hand, the sensitivity is more prone to false negatives due to the low number of true positives in the data relative to true negatives.

The most likely regions to be disagreed between methods (i.e. where one method claims they are affected territories and another does not) are the small central ASPECTS territories: Caudate, Lentiform Nucleus, Insula, and the Internal Capsule. I believe this disagreement comes from the small size, close proximity, and irregular and variable shaping of these territories making them more challenging than the others for the methods to correctly classify.

Evaluation	**Clinical**	**Observed**	**Ground Truth**	**CNN**
Dichotomised Sensitivity	**0.87**	0.72	0.48	0.50
Dichotomised Specificity	0.82	0.98	**1.00**	0.97
Territory Sensitivity	N/A	**0.84**	0.72	0.49
Territory Specificity	N/A	0.98	**0.99**	0.97

Table 4.2: The sensitivity and specificity from binary STAPLE analysis on the dichotomised and territory data.

		Clinical	Observed	Ground Truth	CNN
Clinical	Sensitivity:		0.37	0.28	0.28
	Specificity:		0.96	1.00	0.97
Observed	Sensitivity:			0.46	0.50
	Specificity:			0.98	0.96
Ground Truth	Sensitivity:				0.69
	Specificity:				0.95
CNN	Sensitivity:				
	Specificity:				

Table 4.3: The sensitivity and specificity between pairs of scores at the dichotomised patient level.

		Observed	Ground Truth	CNN
Observed	Sensitivity:		0.57	0.40
	Specificity:		0.98	0.97
Ground Truth	Sensitivity:			0.43
	Specificity:			0.96
CNN	Sensitivity:			
	Specificity:			

Table 4.4: The sensitivity and specificity between pairs of scores at the territory level.

4.9 Discussion

I present results from two statistical measures applied to methods of scoring ASPECTS to compare three methods of determining ASPECTS to the professional standard and to explore the methods against each other. I use Cohen's kappa as a method resilient to dataset imbalance to compare the correlation between pairs of methods, and STAPLE to determine the sensitivity and specificity of each method relative to the others.

4.9.1 Comparison of Score Methods

This section describes and constrasts the score methods.

CNN Score: At the patient level, the CNN Score correlates the strongest with the Ground Truth Score, which is expected as the CNN is trained on the ground truth. The similarity between the CNN and Ground Truth Scores is influenced by the choice of the CNN Score threshold, which exists to balance false positives and false negatives, and thus minimise the bias between the two scores. The quality of the ground truth provides an upper bound on the ability of the CNN. At the territory level the performance of these methods diverge. A STAPLE sensitivity of 0.5 for the CNN Score indicates that there are as many false negatives as true positives.

Ground Truth Score and Observed Score: The Kappa coefficient of 0.6 between the Ground Truth territory scores and the Observed territory scores is low given that both were marked by the same clinical expert. It can be reasonably assumed that the clinical expert will have chosen similar territories for both methods, however the Observed territories are identified directly the clinical expert, while the Ground Truth territories

came from an indirect route through manual segmentation, atlas alignment, and thresholding rules. There are associated errors with these processing steps such as ground truth segmentation accuracy, atlas accuracy and misalignment, threshold rules inconsistently applied due to human judgement, and different tools being available at the time of the method - namely the availability of a slabbing tool which was used for the Observed Scores, but was not available for the Ground Truth Scores. Slabbing makes the identification of ischemic stroke signs easier by improving contrast, so should result in a lower ASPECTS. This is observed in Figure 4.6, although with the number of other sources of error here I cannot conclude that the lower ASPECTS is caused by slabbing alone.

The high frequency for the Ground Truth Score at an ASPECTS of 9 in Figure 4.6 is an indication that the small volume rule could be being applied too regularly or inconsistently with the clinical expert's manual observations. This could be due to the 5% threshold being set too high, which in turn may be due to underestimating the extent of ischemia due to the absence of a slabbing tool when marking.

Clinical Score: The Clinical Score returns the lowest STAPLE specificity of all the methods. Figure 4.6 shows this method is biased towards lower scores, which leads to a lower specificity. The Cohen's kappa between the Clinical Score and each of the other methods is poor. A possible reason for this is that the hospital clinicians assigning the score may have been able to see subtle signs due to their expertise level and use of clinical tools.

4.10 Conclusion

Stroke is a serious condition and is becoming more common in developed countries. This chapter examined the reliability of ASPECTS obtained from four observers at two levels of interpretation: the ASPECTS territory level, and patient level. I showed how a voxel-level output from a CNN can be distilled into these levels and how comparable these results were to other observers.

My results show the CNN is an effective means of determining stroke severity; however it is most closely correlated to its ground truth rather than to the professional standard. This is because my ground truth does not capture the level of detail visible to expert clinicians, which leads to a higher average ASPECTS. The performance of the CNN is similar to the ground truth itself when interpreted at the patient level, but the performance diverges at the territory level due to my rules on converting the voxel predictions to territory score being applied broadly.

4.10.1 Future Work

In the future it will be possible to improve the ground truth used in my experiments. If an ensemble of experienced radiologists could mark accurately the affected regions, my ground truth quality would be improved and would match the professional standard better. This would lead to a CNN that can predict closer to the professional standard. However radiologists often lack the free time to complete this exercise.

A simpler improvement could come from an improved atlas and registration to better align the territory scores from the CNN to the observed scores. Brains are unique with variations due to age, historical pathology, and head shape, among other factors, affecting the scan. The atlas, especially rigidly aligned, does not approximate well to every brain. Improving this approxi-

mation through the use of better landmarking, the use of different atlases for different ages/pathologies/heads, or careful non-rigid alignment of the atlas are areas for future research.

Improvements may also be made to the CNN by slabbing training data to provide additional context that clinicians already have access to. Similarly, the follow-up scan data taken for surviving stroke patients could be used to add context, as could scans taken in other modalities. While these data could be used for training, they cannot be available in practice, so the CNN must only use it as an additional ground truth aid and not as an input.

Next Steps: With distillation explored for ischemic stroke in CT, I move on to the final problem of federation where I train a neural network within a collection of hospitals simultaneously.

Chapter 5

Federated Learning

5.1 Abstract

Federated learning is an approach that allows a machine learning model, such as a neural network, to train on two or more isolated datasets without any data samples transferring between these datasets. Instead, the model exists as a copy for each dataset that trains locally and returns their trained parameters to a central location, where they are aggregated to form a new model. This new model is then copied to each dataset and replaces the previous copy. The cycle then repeats.

Federated learning has uses in domains where data privacy or sensitivity are concerns, such as in the medical domain. In addition to this conventional form described, I present a novel evolution called soft federated learning which accounts for covariate shift between the isolated datasets to yield specialised model aggregation for each model copy.

To analyse these two federated learning algorithms, I use three baseline implementations: *Global Pooling* where all datasets are pooled together and trained on; *Local* where a model copy is trained and tested within each dataset

(no model aggregation stage); and *Ensemble* in which the models from the Local method are evaluated on every dataset and the average prediction for a dataset is taken.

I use two datasets to evaluate the performance of these algorithms and the baselines. First is the MNIST dataset introduced earlier in this thesis, which consists of handwritten digits, but for this chapter I divide it into a series of subsets to represent distinct datasets. Each of these subsets may have an additional transformation applied to simulate inter-dataset differences. These transformations consist of a range of affine transformations and varying levels of noise. Sample level classification accuracy forms my evaluation.

The second dataset is the BraTS medical imaging dataset, which are MRI scans of brain glioma from multiple institutions. Each institution forms an isolated dataset. I evaluate using the Dice coefficient on slicewise pixel-level segmentations.

The aim of my experiments is to evaluate conventional federated learning against its soft counterpart and to compare these against the baselines. My results show soft federated learning is more effective than its convential counterpart at accounting for differences in the data domains.

5.2 Chapter Overview

To finish the thesis I tackle the issue of federation for machine learning in the medical domain. Federated techniques seek to bypass the need to move data around, allowing training across multiple cohorts. They do this by sending copies of model parameters between data providers, such as hospitals. In this chapter I introduce federated learning in its most popular form currently used in industry, and later develop soft federated learning, which is a new federated technique that addresses some of the weaknesses of the current form.

Federated learning is a model-agnostic means of training using data that are divided between distinct locations called *institutions* without any data samples being transferred between institutions. An institution has some number of data samples contained within. In the most popular form currently used commercially [17, 135], federated learning achieves this by training a model at each institution and then transferring trained model parameters to some central location where they are synchronised and aggregated to form a new model. The new model then replaces the model used in each institution and the cycle repeats. The aggregated model can be considered to have learned on all data present despite not directly observing any data.

After introducing this form of federated learning, which I will refer to as Conventional Federated Learning (CFL), I describe the difficulties this algorithm faces when dealing with domain shift between institutions — a common issue. I then introduce a novel development called Soft Federated Learning (SFL). SFL addresses the issues of CFL by modifying the model aggregation stage based on the relative differences between data in different institutions. It also removes the need for a central location, which while not an issue in itself, is a potential point of failure or attack.

Federated learning (both CFL and SFL) is useful in situations where I want to learn from a large amount of data, but this data cannot be pooled into a central location because of, for example, costs, technical barriers, legal concerns (privacy, data protection, consent, etc.), and so on. For these reasons the data must remain in their respective institutions and not be transferred from them. Medical data are one of the most protected forms of data and transferring it is often costly or unfeasible. This makes it a prime area for federated learning.

This chapter starts with a short reference list of the mathematical nota-

tion used throughout this chapter (Section 5.3). I then move into a technical description of CFL detailing the algorithm and its strengths and weaknesses (Section 5.4), and provide a literature review on this topic (Section 5.5). The results of a survey on clinical staff for federated learning for a clinical decision support application are examined in Section 5.6.

In Section 5.7 I describe the novel evolution of this algorithm, SFL, that addresses the major issues with federated learning. CFL and SFL are comprehensively compared against each other and a series of baseline results on simple problems involving the MNIST dataset and a multi-institutional medical dataset known as BraTS to emphasise their differences. The datasets used are described in Section 5.8, experiment design in Section 5.9, model training and evaluation in Section 5.10, an explanation of the baseline measures Section 5.11, and results in Section 5.12. Discussions and conclusions finish this chapter in Sections 5.13 and 5.14 respectively.

The programs used in the experiments in this chapter were written from scratch in Python 3.6 [93]. There now exists a framework for (conventional) federated learning using TensorFlow [105], but this was not available at the time of this research.

The SFL algorithm has been filed as a Canon patent [67].

5.3 Notation

Here I define the mathematical notation I will be using throughout this chapter. This acts as a reference list that the reader may refer back to throughout this chapter.

Symbol	Definition
N_I	The total number of data institutions. This does not count the central server used in CFL.
N_C	The total number of classes across all institutions.
$N_{d,i}$	The number of data samples at institution i.
N_D	The total number of data samples across all institutions ($= \Sigma_{i=0}^{N_I} N_{d,i}$).
d_i	The dataset within institution i. A dataset is made up of training and testing data: $d_i = (d_i^{train}, d_i^{test})$
x	A data sample. $x \in \mathbb{R}^{dim}$ where dim is the number of dimensions in the data — the number of features (such as pixels or voxels), not the number of spatial dimensions.
y	A class. $y \in \mathbb{N}^+$ with, for example, MNIST being the set $\{0, ..., 9\}$. y_{pred} refers to the predicted class, y_{gt} refers to the true class (ground truth), and y_c is the correct class c of a data sample.
D	The set of all data across all institutions. $D = (\mathbf{x}, \mathbf{y})$ where $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_i\}$, $\mathbf{y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_i\}$. $\mathbf{x}_i = \{x_1, \ldots, x_{N_{d,i}}\}$, $\mathbf{y}_i = \{y_1, \ldots, y_{N_{d,i}}\}$ This can also be expressed as $D = \cup_i d_i$ where $\cup$ is the set union.
T	The number of cycles for a given experiment run.
t	The current cycle in the experiment run.

M_i^t	A neural network model (at cycle t and within institution i) The model consists of a set of weights ($\mathbf{w}$) and biases ($\mathbf{b}$) $M_i^t = (\mathbf{b}_i, \mathbf{w}_i)$ where $\mathbf{b}_i = \{b_1, \ldots, b_{j,i}\}$ and $\mathbf{w}_i = \{w_1, \ldots, w_{k,i}\}$ where j is the number of biases in the model and k is the number of weights. $M^{t=0}$ is the initial model state.
I_i	An institution. i can take the values $\{1, \ldots, N_I\}$ An institution has data and a model: $I_i = (d_i, M_i^t)$
I_{cent}	A special institution called the *central server* or *central location*.

Table 5.1: Definitions of the symbols used in this chapter.

5.3.1 General Model Training

Here I define model training mathematically for a general case. A cycle of training takes a model, M^t to M^{t+1}:

$$M_i^{t+1} \leftarrow \text{SGD}(M_i^t, L(M_i^t, d_i^{train})) \tag{5.1}$$

where *SGD* is Stochastic Gradient Descent and the loss function, L, is the cross-entropy loss:

$$L(M, d) = -\sum_{c=1}^{N_c} y_c \log(y_{pred}) \tag{5.2}$$

where y_{pred} in this case is the probability of that class prediction.

Expanding, I get:

$$M_i^{t+1} \leftarrow M_i^t - \alpha L(M_i^t, d_i^{train}) \tag{5.3}$$

The optimisation function (SGD) and loss function (cross-entropy) can be

trivially substituted by any other corresponding function. In my work on the MNIST data I use the above functions, while for the BraTS data I use a custom loss function (Section 5.10.2) and the Adam optimiser [151].

5.3.2 General Model Evaluation

Now for the evaluation, I take the class predictions generated by applying the model, M, to each data sample in the test dataset, x^{test}:

$$P(\mathbf{y}|x^{test}, M) = \{P(y_0|x, M), \ldots, P(y_{max}|x, M)\} \tag{5.4}$$

and then I take the class with the highest probability to be the class prediction:

$$y_{pred} = \arg\max_{c} P(y_c|x) \tag{5.5}$$

with the predicted class contained within N_C.

To calculate the accuracy of an evaluation at an institution, i, I count the number of predictions matching the ground truth and divide by the number of samples.

$$\text{Acc}(\mathbf{y}_{pred}, \mathbf{y}_{gt}) = \frac{1}{N_d^{test}, i} \sum_{j=1}^{N_d^{test}} \text{correct}(y_{pred,j}, y_{gt,j}) \tag{5.6}$$

$$\text{correct}(y_{pred}, y_{gt}) = \begin{cases} 1, & \text{if } y_{pred} = y_{gt} \\ 0, & \text{otherwise} \end{cases} \tag{5.7}$$

The evaluation function can be trivially substituted for another. In my BraTS work I use the Dice score [74] instead of simple accuracy (see Section 5.10.2, Equation 5.15) as it is a better suited evaluation method for image segmentation tasks.

5.4 An Introduction to Federated Learning

Federated learning is a model-agnostic approach to training a machine learning model, in this case a neural network, across two or more distinct institutions without there being any transfer of data samples between the institutions. The collection of institutions is called the *federation*.

Federated learning is generally inferior in terms of achievable accuracy to the traditional model training technique of pooling all available data in a central location and training on this pooled data. It falls behind in terms of time, performance, resource usage (memory and network), and computational complexity — experiments later in this chapter confirm this. This is because the added complexity of federated aggregation and fitting error of training on subsets of the total data reduce the performance of federated learning. However, pooling data is not always easy or possible to do. For example, in the medical domain, a hospital is a location that is generating data (through taking scans of patients, for instance), but I cannot simply remove these data from the hospital. To obtain data in such a case I require contractual agreements, patient consent, plenty of time (often months), and financial payment, and even if the data are obtained, there will likely be restrictions on their use. These issues make it difficult to develop a central pooled resource of medical data for model training.

Federated learning overcomes this by removing the need for any transfer of sensitive data. Federated learning generally operates under four key assumptions about the data [156, 200].

1. The data are sensitive and cannot leave their institution.

2. The data at each institution have some degree of "similarity". That is, a model trained on a dataset at one institution will learn something

applicable to a dataset at a different institution.

3. However, the datasets are *not* independently and identically distributed. This means each institution's data may not be representative of the global distribution.

4. The datasets are class unbalanced both within themselves and between each other. This means some classes occur much more frequently than others and some institutions do not have all of the classes of other institutions. An example would be a particularly rare pathology label that is only observed at a minority of institutions.

5.4.1 The CFL Algorithm

Here I introduce the CFL algorithm, which is the most popular federated learning algorithm currently used in commercial applications [17, 135].

I start with one of two options. Either a model that has been pre-trained on a task similar to what I expect to see in the institutions, or an untrained (randomly initialised) model. The model is chosen to be suitable for the desired task. Using a pre-trained model accelerates convergence to good performance within the institutions and ensures that the first few iterations have an acceptable performance. Using an untrained model leads to a slower training but convergence is still reached.

With this model I create a copy of it for each institution. Within these institutions the model then trains locally on the data present in the usual way. After some set amount of time or number of epochs the models send their parameters (their "model") — for example, the bias and weights of neurons in a neural network — to some central location. This may be the location where the model originally came from, or any other secure and accessible location.

The time before parameter return must be long enough for the models to specialise to the local features of their data, but not too long that the models substantially diverge from each other. Federated learning requires that the models remain somewhat synchronised, otherwise aggregating the parameters can result in a nonsensical model.

Depending on the resources available, a subset or all of the institutions can return their models — in my work I always return all of the models. If a federation is very large (lots of clients), it may be applicable to return a random subset of models under the assumption that the models in this subset will approximate the models in the full federation. This improves the time efficiency of the algorithm.

The returned parameters are aggregated with a weighting equal to the amount of data each was trained on — Equation 5.8. The new model is then copied to each institution, replacing the previous version, and the cycle repeats for the lifetime of the software. See Algorithm 5 for pseudocode of CFL using a neural network, which is the machine learning algorithm I will use in this chapter.

$$M_g^t(\mathbf{b}_g, \mathbf{w}_g) \leftarrow \sum_{i=1}^{i=N_I} \left((\mathbf{b}_i, \mathbf{w}_i) * \frac{N_{d,i}^{train}}{N_D^{train}} \right) \tag{5.8}$$

The models at the institutions can be used for evaluation of local data once their performance is deemed good enough by some defined metric.

Figure 5.1 shows non-federated data transfer where data from a number of institutions (sometimes only one) are transferred to a central location. Figure 5.2 shows the CFL approach where the data is not transferred (remains within each red ring), but the model parameters are passed back and forth between the central location and each institution. Finally, Figure 5.3 shows an overview of a cycle of CFL at a single institution.

CFL essentially allows knowledge transfer between institutions without any raw data leaking out, especially when this knowledge is encrypted. Knowledge here refers to the patterns and mappings implicitly learned in the parameters of the models. The models are essentially learning on all data without any data being directly moved into or out of an institution. It achieves this by each model being an encoding of the data it trained on and these abstract data encodings (the models) are transferred across institution boundaries.

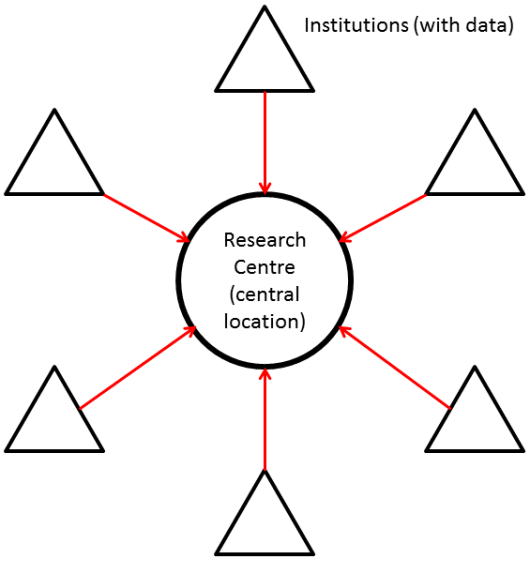


Figure 5.1: An overview of a traditional (non-federated) setup showing a central location (central server) where data from numerous institutions are transferred to (red arrows) and pooled. A model is then trained at the research centre on the pooled data. Any number of data institutions may be used.

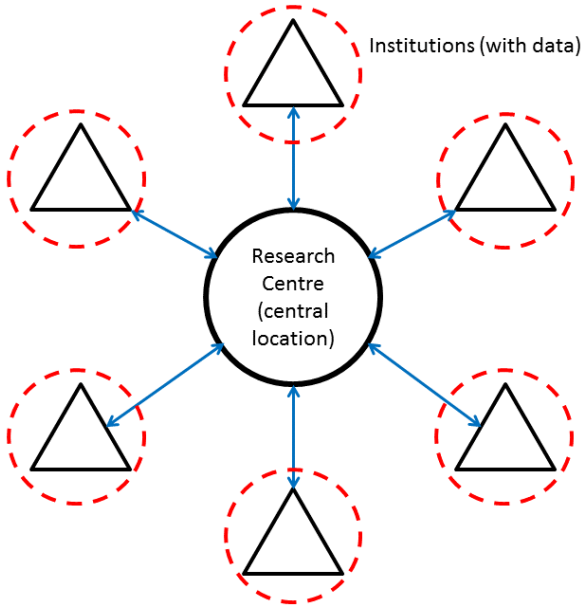


Figure 5.2: An overview of a federated learning setup showing a central location (central server) connected to numerous data institutions. The data do not leave their respective institutions (as indicated by the dashed rings), but instead the model parameters are passed back and forth (blue arrows). To utilise the CFL algorithm, at least two institutions are needed.

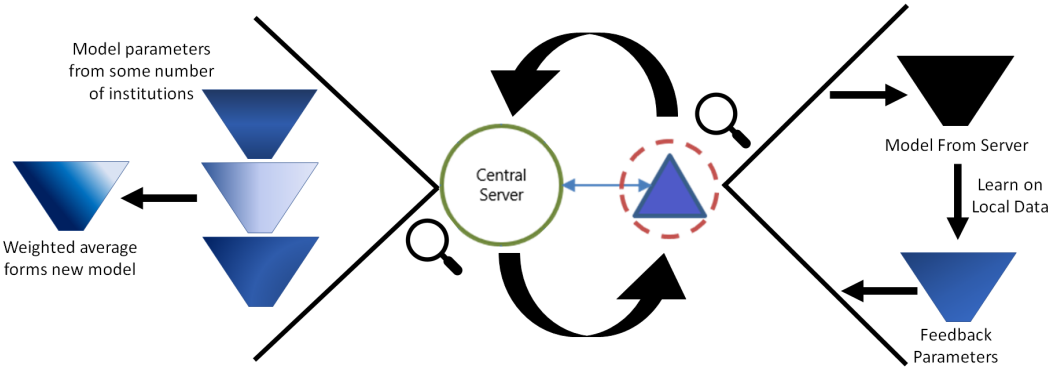


Figure 5.3: A cycle of federated learning at a single institution. Starting from the central server and going to the right a model is copied to each institution. The model learns locally and is returned. On the left of the figure a series of learned models are averaged to form a new model. The cycle then loops back to the first step.

Algorithm 5 Conventional Federated Learning Pseudocode

// Randomly initialise and then pre-train the global model:

$M_{cent}^{t=0} \leftarrow \text{rand init}(M_g)$

$M_{cent}^{t=1} \leftarrow \text{SGD}(M_{cent}^{t}, L(M_{cent}^{t=0}, d_{cent}^{train}))$

// For each cycle of federated learning

while t $\leq$ T **do**

For each institution

for i in 1 to N_I **do**

// Copy the central model to the institution

$M_i^t \leftarrow M_{cent}^t$

// Train locally

$M_i^{t+1} \leftarrow \text{SGD}(M_i^t, L(M_i^t, d_i^{train}))$

// Parameter feedback and weighted averaging.

// The parameters of each institution's model are weighted based on the fraction of the total training data samples seen when training.

$M_g^t(\mathbf{b}_g, \mathbf{w}_g) \leftarrow \sum_{i=1}^{i=N_I} \left((\mathbf{b}_i, \mathbf{w}_i) * \frac{N_{d,i}^{train}}{N_D^{train}} \right)$

5.4.2 Issues with CFL

Federated learning works well if the data within each institution are representative of the whole population. This means any knowledge learned within one institution can be directly applied to all other institutions.

Unfortunately, this is rarely the case. Each institution will have its own characteristics which may or may not leave its data similar to those in other institutions. These characteristics can be placed into two categories:

Data Acquisition: These are differences in how the data are collected. In the medical domain a hospital, for instance, that is collecting CT data may

have a different CT scanner make, model, and age, or may use different scanner settings, or scan protocol to another hospital. There can also be differences between how clinicians record data. These differences affect the data samples that in turn affect what a model at that institution learns.

Population Sampling: This forms differences in which data are observed at a hospital, rather than within the data themselves. Some hospitals, such as specialist clinics, or hospitals in specific geographical locations, may see a much higher number of a certain type of pathology than other, more general, hospitals. This leads to a shift in the class balance which means the model will focus on different features between institutions.

Furthermore, institutions may see a different volume of data: a large hospital is likely to generate more data than a small hospital. The model is at risk of overfitting on the smaller hospital's data — where it trains for an extended period on a few data samples — leading to non-useful parameters being learned. Meanwhile, the larger hospital is less likely to overfit and more likely to learn good general parameters, but its new model at a new cycle will be influenced by the overfitted one from the small institution, thereby weakening its ability.

Finally, federated learning is at risk of compromised institutions. When deployed to a large number of institutions around the world, there is a risk of an institution malfunctioning, or being attacked in such a way that its data becomes invalid or the model returned is maliciously modified to weaken the federation during aggregation. One can imagine a scenario where an institution is compromised and its model is designed to perform poorly on the task by, for example in the medical domain, always returning negative (healthy) labels for pathology. If this compromised model is sent to the central location alongside

a parameter saying it has been trained on a very large number of datasets, then it will dominate the aggregation (see Equation 5.8) and thus result in a compromised aggregated model that is then sent to every institution, thereby rendering the entire system useless.

Later in this chapter (Section 5.7) I detail my new algorithm, SFL, to solve these issues:

1. SFL factors in overfitting when deciding how much a model should be weighted during aggregation.

2. SFL also factors in the differences between institutional data cohorts when weighting models, and it does this for every pair of institutions.

3. SFL is able to recognise outlier models (such as from rogue institutions) and weaken their influence in the federation (potentially to zero).

4. SFL removes the need for a central location.

5.5 Previous Work

In recent years there have been substantial advances in the computational abilities of graphical and computational processing units for machine learning. This has allowed higher complexity models to be trained [69], which offer improved performance. Performance on ImageNet — an image classification challenge used as a benchmark for machine learning [72] — continues to improve every year. With these advances, one of the largest constraints in the field is now a lack of data. Sun *et al.* showed (empirically) that there is a logarithmic improvement achieved in model performance with increasing data [278]. As such, finding new ways to obtain data and reduce the difficulty of

collecting large sets together is important to continue improving model performance. Federated learning is one way this is being addressed [199]. McMahan *et al.* developed federated learning and found ways of efficiently communicating between many datasets [200] and showed that there are diminishing returns with the number of institutions used in each update. Bonawitz *et al.* discuss some of the key challenges with high-level federated learning design [32], some of which will be discussed below.

Attempts have been made to create open-source advanced federated learning technology, such as OpenMined [226], which started in 2017 [307]. Closed-source federated learning systems include WedGLORE [137] and EXPLORER [290] that enable privacy-preserving construction of a global logistic regression model from distributed sensitive datasets, GWAS [53] specifically for federated genomic datasets, and WebDISCO [186] for patient survival data.

Bogdanov *et al.* discuss how secure multi-party computation can be used as a privacy-enhancing technology, and they provide a detailed description of the solution [31]. Kamm *et al.* — in a paper that shares several authors with the previous reference — explain privacy enhancing in the context of genome data using secret values to obscure the data during processing [143]. Pihur *et al.* use their "draw and discard" method that maintains a number of versions of a model on a server, then selects one at random to update using data from an institution. They then use this to randomly replace one of the instances on the server, thus gradually progressing the models using data from multiple institutions [234]. Nishio and Yonetani develop a means of selecting institutions for carrying out model updates at using the computational resource limits of these institutions and the timeframe the update must be completed in to select the institutions that give the most efficient update [222] — they do not factor in the data domain differences between institutions when deciding

which to select however.

Data anonymisation via de-identification methods have also been put forward, such as k-anonymity [279], l-diversity [191], and differential privacy [80]; which enable easier use of the data including the possibility of transfer. However, these methods remove information that may be useful for machine learning purposes.

Smith *et al.* look at multi-task learning — where a model learns to solve multiple tasks by taking advantage of similarities between the tasks — but in a federated setting [273]. Multi-task learning is a substantial field in itself that will not be covered in detail in this thesis, but the following provide a good starting point: [8, 10, 11, 87, 166, 310]. Essentially, multi-task learning involves learning a single model that can be applied to two or more different tasks. By doing so, the learning acts as a regularisation method making the model less likely to overfit than a single-task model. This has some similarities to federated learning, but operates on the assumption that the data can be communicated and avoids model parameter averaging.

Kamp *et al.* explore how the institution models diverge over time as they fit to their local data [145]. This divergence harms the aggregation stage of federated learning. They employ a new algorithm called Dynamic Averaging [146], which modifies the federated learning algorithm by monitoring the divergence of each model over time. When a set have diverged beyond some defined threshold, model averaging occurs to pull these diverged models back together.

5.5.1 Previous Work with Medical Data

A federated learning system was tested in 2016: euroCAT by Deist *et al.* [70]. They took five radiation clinics across three countries (Belgium, Germany, and

the Netherlands) and used a support vector machine and Bayesian network as their models to federate. Their results show successful learning of the models for dyspnea (shortage of breath) [139], thus acting as a proof-of-concept that federated learning can be used in the medical domain. However, Konecny *et al.* have found that federated learning can train very slowly in the presence of a large number of institutions [156].

In 2018, Sheller *et al.* used federated learning on a U-Net model — a type of deep neural network that downsamples an input through convolutions and downsampling layers before upsampling through transpose convolutions and upsampling layers [245] — to show how federated learning can be used on the medical dataset BraTS (an MRI brain tumour dataset) [268]. The BraTS dataset is described in detail later in this chapter (Section 5.8.2) and used in one of my experiments (Section 5.12.5). Also in 2018, another research group — Brisimi *et al.* — used a support vector machine for federated learning on patient electronic health records [38].

Recently, Roy *et al.* in 2019 developed BrainTorrent, a version of federated learning aimed at medical applications [250]. It removes the central server from the federated system by enabling peer-to-peer sharing of the models. At each cycle a random institution checks the model version at all other institutions and then aggregates the models that are new locally before training the new model on its own data. This shares some similarities with part of my SFL method (see Section 5.7), but the authors do not take into account the domain shift between institutions in their calculations (Section 5.7.1). Domain shift between institutions is a significant issue for any federated learning system and may be due to different treatment guidelines [71], technological differences [192], or variations between clinician's interpretations [246].

5.6 Clinician's Opinions

To gain insight into the opinions of the target user base of a medical federated learning system and the desire to have such a system, I created a survey aimed at clinical staff of all backgrounds. Such a system could be deployed in a wide variety of situations, so I aimed for responses from a wide background, although I did not reveal the technical details of federated learning during the survey.

I received 15 responses from a variety of backgrounds and experience levels from consultant radiologists to doctors, and from professors to students. However, all responses were from the United Kingdom[1]. Therefore the applicability of these results outside of the United Kingdom is limited.

The responses showed most (60%) are not aware of a clinical decision support application at their institution that continually improves. The respondents that were aware reference a voice activation reporting software that learns continually as reports are dictated.

The respondents generally agreed that they would trust the software vendor (i.e. Canon) to validate updates. They also wanted to be able to compare, revise, or undo updates to the system if they did not believe they were beneficial, and wished to be notified when the system updates. Overall, there was a strong desire to implement continuous learning in systems currently in use.

Respondents have a varied opinion on the frequency of updates with the most populat options being weekly or quarterly (every three months). This would be frequent enough to keep the systems in the federation synchronised. As for where the updates should come from, everyone believes that their own institution and large regional institutions should be involved. Opinions fall

[1]I sent the survey to many international partners of Canon Medical Research Europe, but failed to receive any responses.

gradually from regional institutions to national to international, with only 47% wanting all institutions worldwide offering updates. This captures the concerns that data that are less local to an institution may be less useful.

Despite this risk of non-useful data, 67% of respondents are happy for all users of the system to contribute to system learning, with the remaining respondents seeking an accreditation process before a user can contribute. This could reflect the need for the system to be used in a wide variety of scenarios by a range of users, so the system should be trained to be suitable for any realistic possibility.

In the final comments at the end of the survey concerns are raised about how one person's improvement can be another person's problem and how I could measure the performance of the system over time and offer the ability for expert review of tricky cases before the system learns on them. These could be addressed by the system adapting to each user via some degree of additional fitting to the core model, and for users uncertain of how to review a tricky case to seek expert advice within their institution than attempt it themselves.

The survey questions in full along with all responses and extended analysis can be found in Appendix A.

5.7 Soft Federated Learning

SFL is a novel advancement to CFL that accounts for weaknesses in the original algorithm. It modifies CFL in the following ways:

Before starting: I calculate a novel quantity I call *influence* between every pair of institutions including pairings of institutions with themselves. The influence is a measure of similarity between two datasets that also factors in the sizes of the datasets. Section 5.7.1 explains the process in

detail, while here I provide a high level overview. Within each pair, the influence is calculated both ways as it is not a symmetric quantity, so the influence of I_A on I_B may not be the same as I_B on I_A. I refer to the two institutions as the *giving* and *receiving* institution — I explain why in the next step. The influences are normalised to sum to one on a per institution basis. Table 5.2 shows an artificial example of what a table of influences may look like for three institutions. The influence values are used during the aggregation step in a cycle of federated learning.

Influence derives from the ability of an institution's model to perform well on the data the aggregated model will be used on. The model used for the influence has the same architecture as the model used later during the federated learning to ensure influences derived from it are directly applicable.

		Giving Institution		
		I_1	I_2	I_3
Receiving Institution	I_1	0.7	0.1	0.2
	I_2	0.3	0.6	0.1
	I_3	0.0	0.1	0.9

Table 5.2: An artificial example of three institutions showing the influence the receiving institution gets from the giving institution. The influence for each receiving institution sums to one. The Receiving Institution is taking in the models from other institutions to aggregate them. The aggregation uses the influence from the Giving Institution as the weighting of that model.

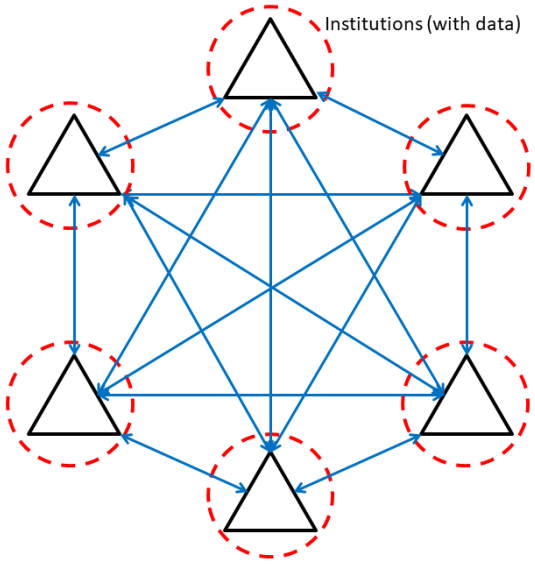


Figure 5.4: SFL removes the central institution and instead connects every institution directly to every other institution for model transfer. Additionally, an asymmetric weighting (not shown) modifies each model transfer based on the similarity between a pair of institutions. Like CFL, this requires at least two institutions. Refer to Figure 5.2 for the equivalent figure for CFL.

During cycles: Influence, explained simply, is a weighting used during model aggregation in a cycle. An institution with a higher influence receives a higher weighting on its model when aggregated, and thus influences the final model more. Referring to Table 5.2, in this case for I_1 as the receiving institution the values are 0.7, 0.2, 0.1 and so it will generate its new model from a 70% weighting of its previous model (I_1), 10% weighting of the model at I_2, and a 20% weighting of I_3's model. These three institutions are the giving institutions, which have the potential to contribute to the receiving institution's model. Each institution creates a unique model in this way. There is no central location now; the aggregation happens at each institution instead.

SFL does not explicitly factor in the amount of training data a model has seen when performing the aggregation, unlike CFL. Instead, the dataset

sizes naturally forming part of the influence calculation.

SFL's differences to CFL are summed up as follows:

1. In SFL there is no central location for aggregating the models. Instead, the aggregation takes place within each institution with the aggregation being specific to that institution. See Figure 5.4

2. The institutions do not all contribute equally to the newly aggregated models. Instead, each institution performs the aggregation with different weightings in relation to its calculated influence on the institutions involved. This weighting is derived from the performance of a test model on other institutions at the start of SFL (explained in Section 5.7.1). See Figures 5.5 and 5.6.

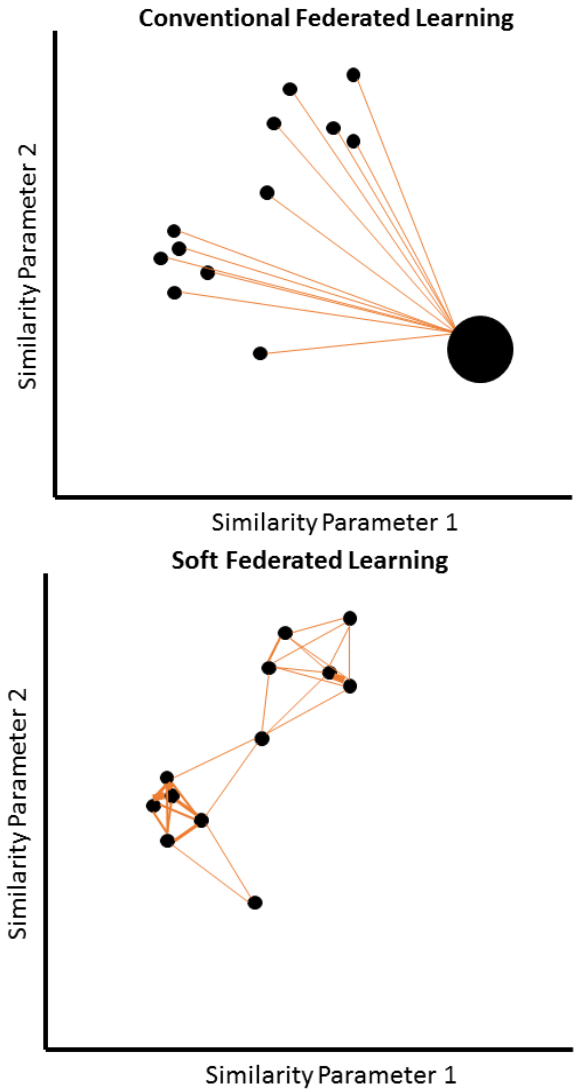


Figure 5.5: A comparison of CFL to SFL in similarity space. Each small circle represents an institution, and the axes represent some notion of similarity space onto which the institutions' data are projected. The distance between institutions represents (inversely) the degree of similarity between those institutions' data. **Top:** In CFL an extra central location is added (large circle) to which each institution is connected. The relative positions of the institutions in this space are not used. **Bottom:** In SFL there is no central location, but each institution is connected to each other by some weighting (represented by the thickness of the connecting lines). More similar (closer) institutions receive a higher weighting from each other, while dissimilar (distant) institutions can have a zero weighting (indicated by a missing connection between two institutions). The weighting between pairs is not symmetric as it factors in the size of each institution as well as their similarity, but I do not show this asymmetry on this image.

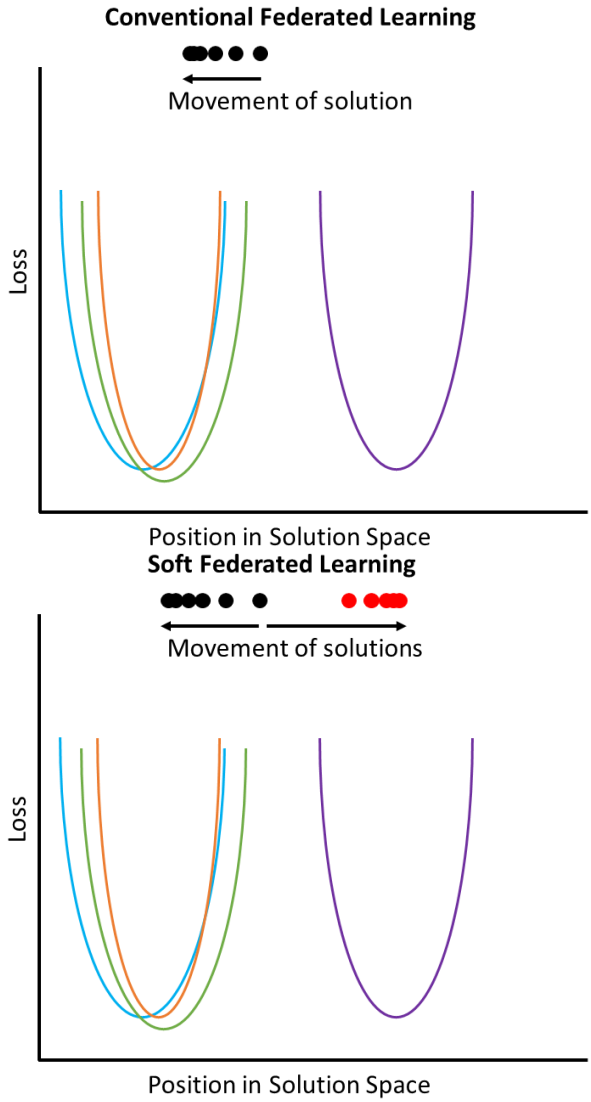


Figure 5.6: A comparison of CFL to SFL in solution space. The parabolic curves each represent the loss function for an institution's data. The lateral position represents the current solution as a projection onto a 1D axis, such that the current solution will be providing some value of loss for each institution. The aim of federated learning is to minimise the overall loss. The dots along the top represent the solution found in consecutive cycles, starting from the centre. **Top:** In CFL the solution will move to the average of all losses to minimise the average loss. **Bottom:** In SFL each institution has its own solution, which seeks to minimise its own loss function. Outlying institutions have less of an effect here.

5.7.1 Calculation of the Influence

The influence is calculated through an empirical means to provide the weighting for each model during the aggregation step of federated learning. This calculation takes a pair of institutions with one being the giving institution, I_{give}, and one being the receiving institution, $I_{receive}$. It then determines how much influence $I_{receive}$ should receive from I_{give}. That is, it is the weighting of I_{give}'s model to use when aggregating at $I_{receive}$. Both institutions may be the same to calculate self influence (i.e $I_{give} = I_{receive}$), or they may be different ($I_{give} \neq I_{receive}$). The influence calculation must be done for every permutation of every possible pairing of institutions including self-pairings. This is due to its asymmetric behaviour.

All influences are first calculated before the federated learning cycles begin. They may then be re-calculated periodically to account for data drift over time[2]. There are four stages to calculating the influence values.

The data at each institution is divided into a training set and a testing set at a 2:1 ratio. During the calculation of the influence, only the training set is used. This is to avoid biasing the influences with test data. The testing set is used only during the federated learning itself to evaluate model performance.

Stage 1 – Model Training: I train n models at each institution, where n is any integer ≥ 2. This represents the number of folds used. I use three folds in my experiments. These models use the same architecture and hyperparameters as the models to be used during the federated learning, but are entirely separate from them. The models are trained to a good fit in an n-fold manner on the training data present in the institution.

[2]Data drift over time is when the data generated by an institution changes over time. For instance, a hospital gets a new scanner, or modifies its protocols, and this slightly changes the data it produces. SFL relies on the influences being accurate enough to improve the aggregation.

I have an additional constraint that each fold must have at least one example of each class[3]. The fold left out is used in the model evaluation in Stage 2. *Only* the training data is used to ensure the test dataset remains unseen for when I perform the federated learning.

Stage 2 – Model Evaluation: I now take pairs of institutions, and these pairs may be self-pairing or different. For the pair I take the models trained at I_{give} and test them at $I_{receive}$. Each model is tested on the fold not used for training, so for a three-fold evaluation the model trained on folds 1 and 2 at I_{give} is tested on fold 3 at $I_{receive}$ for example. This involves the transfer of model parameters but not of any data samples. The accuracy, between 0 and 1 representing 0% and 100% accuracy, of the models is noted and averaged between the folds. I refer to this average value as an *accuracy value*.

If the two institutions have similar data and the data are plentiful, the models trained at one institution should perform well on the other. This is known in literature as the system having a low covariate shift [269] – the inputs (data) change little between institutions while the outputs (class definitions) remain the same. On the other hand, if the data between institutions diverge greatly, the performance of one model on another will be low (high covariate shift). A third option is if the institutions are similar (low covariate shift) but one institution has plentiful data, while the other has very limited data. In this case a model trained on the larger dataset, it should generalise well to the smaller one, but a model trained on the limited dataset may overfit[4] locally to the data and perform less

[3]This is a realistic scenario as even the rarest pathologies get several measurements taken - i.e. generate several data samples.

[4]Overfit rather than underfit as I will have a model that is relatively complex compared to the amount of data it is seeing in this data-sparse institution. This is due to it needing the same architecture as all other models, such as those used on more complex institutions.

well on the large dataset.

Refer to Figure 5.7 for a schematic of Stages 1 and 2 for two institutions with different sizes (four possible pairings), which results in four accuracy values. Self-pairings are shown as well as the training/testing sizes for each pairing. Three folds are used for training.

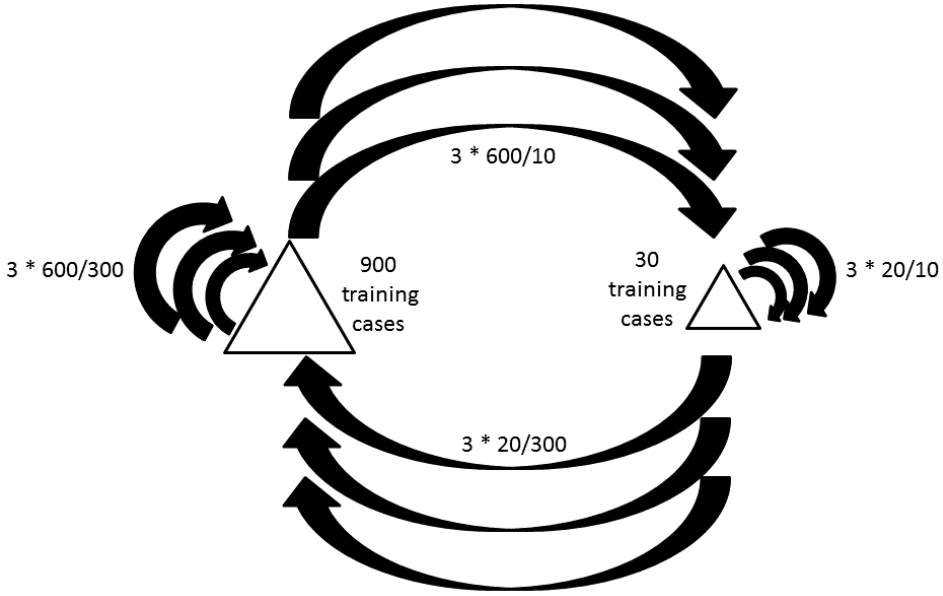


Figure 5.7: Visual representation of all four possible pairings of the influence calculation for a pair of institutions (triangles) — one with 900 training cases, and one with 30 - these numbers were chosen partially arbitraily and partially for convenience of example. Each arrow represents a model being trained and tested. The arrow's origin marks the training institution (I_{give}) and points towards the testing institution ($I_(receive)$). I perform three-fold cross-validation, which results in three train/tests for each possible pair. The numbers next to the arrows show the number of training and testing cases. E.g. 3 * 600/10 is three-fold cross-validation with each fold having 600 training examples and 10 testing examples. All data samples used in the influence calculations (training and testing) come from the institutions' training cases.

Stage 3 – Conversion to Influence: Next, I calculate a value called *trivial chance accuracy* from the class balance present in the training data of

The model is likely to fit perfectly to the training data (overfit), and so not generalise well.

$I_{receive}$, where the models were tested. I do this by assuming a "dumb" model is evaluating on the test dataset and giving the same class output for all samples where this class is the most common class. For instance, in MNIST there are ten classes with an approximately equal number of samples each. Therefore a dumb model can achieve about a 10% accuracy by predicting any of the classes for all samples, so 0.1 is the trivial chance accuracy. Likewise, in a binary class task where 70% of the class labels are the same value, a model predicting this class for all labels would achieve a 70% accuracy, so 0.7 is the trivial chance accuracy.

The trivial chance accuracy is subtracted from the accuracy values from Stage 2 to give *accuracy above trivial chance*. This may be negative in the case of a poorly performing model, in which case I clamp it at 0. These values are then normalised to be between 0 and 1 by dividing by the max range from trivial chance accuracy to 100% accuracy – see Equation 5.9 where *Influence* is the influence value, A is the accuracy value (between 0 and 1), and T is the trivial chance accuracy. $A - T$ is the accuracy above trivial chance, and $1 - T$ is the normalising factor.

$$\text{Influence} = \begin{cases} \frac{A-T}{1-T} & \textit{if } A - T > 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.9}$$

Stage 4 – Institution Normalisation: Once all possible pairings and permutations have been evaluated, I have the full set of influence values. I then take the influence values for each institution and normalise them to 1. This normalisation ensures that the total influence from all models when aggregating sums to 1, and therefore the parameters do not explode or grow slowly. Table 5.4 previously showed an example of the full set of influence value for three fictional institutions.

> The influence values are essentially a measure of how useful one institution's model is for another. It can be seen as a measure of similarity, but it also factors in the size of the dataset naturally in the way a model may overfit to a small set of data, leading to low performance, and hence reducing the influence from this institution to others.

Algorithm 6 captures these steps in pseudocode and shows how the influence ties into the aggregation step.

The influence values are re-calculated periodically to factor in drift of the institutions. This re-calculation happens on a longer time scale than a cycle of federated learning and this time scale is chosen according to the task needing to be solved. For a typical medical application, I might re-calculate the influences on a yearly basis. I do not re-calculate them for every cycle of federated learning due to the computational cost of the previous stages.

Algorithm 6 Soft Federated Learning Pseudocode

// Randomly initialise and train a model at each institution:

for n in $f = 1$ to $f =$ number of folds **do**

for i in $i = 1$ to $i = N_I$ **do**

$M_{i,f} \leftarrow \text{SGD}(M_{i,f}, L(M_{i,f}, d_{i,f}^{train}))$

// Test trained model at all institutions

// Select institution to test models at:

for j in $j = 1$ to $j = N_I$ **do**

// Select institution to bring trained model from:

for k in $k = 1$ to $k = N_I$ **do**

// Find the predictions for each class for each fold

for n in $f = 1$ to $f =$ number of folds **do**

$P(\mathbf{y}|x_j^{test}, M_{k,f}) = \{P(y_0|x_j, M_{k,f}), \ldots, P(y_{max}|x_j, M_{k,f})\}$

// Take the maximum predicted class

$y_{pred} = \text{maximizer}_c P(y_c|x)$

// Find the number of correct predictions (accuracy)

$\text{Acc}(\mathbf{y}_{pred}, \mathbf{y}_{gt}) = \frac{1}{N_d^{test}} \sum_{s=1}^{N_d^{test}} \text{correct}(y_{pred,s}, y_{gt,s})$

// Average across all trained model institutions to get final accuracy

$\text{Average Acc} \leftarrow \frac{1}{\#folds} \sum_{g=1}^{\#folds} \text{Acc}_g$

// Convert to accuracy above trivial chance (refer to Equation 5.9)

$\text{Final Acc} \leftarrow \frac{\text{Average Acc}-\text{Trivial}}{1-\text{Trivial}} if \text{ Average Acc} - \text{Trivial} > 0 \text{ else } 0$

// Normalise the Final Accuracies to 1

$\text{Summed Accuracies} \leftarrow \sum_{r=1}^{r=N_I} \text{Final Acc}_r$

// Find influence for each final accuracy value

for r in $r = 1$ to $r = N_I$ **do**

$\text{Influence}_r \leftarrow \frac{\text{Final Acc}_r}{\text{Summed Accuracies}}$

// Randomly initialise and then pre-train an initial model:

$M_{init}^{t=0} \leftarrow \text{rand init}(M_g)$

$M_{init}^{t=1} \leftarrow \text{SGD}(M_{init}^{t}, L(M_{init}^{t=0}, d_{init}^{train}))$

// For each institution

for i in $i = 1$ to $i = N_I$ **do**

// Copy the initial model to the institution to begin

$M_i^t \leftarrow M_{init}^t$

// For each cycle of federated learning

while t $\leq$ T **do**

// For each institution

for i in $i = 1$ to $i = N_I$ **do**

// Train locally

$M_i^{t+1} \leftarrow \text{SGD}(M_i^t, L(M_i^t, d_i^{train}))$

// Parameter feedback and weighted averaging.

// The models are sent to each institution and multiplied by the corresponding influence

$M_i^t(\mathbf{b}_i, \mathbf{w}_i) \leftarrow \sum_{j=1}^{j=N_I} ((\mathbf{b}_i, \mathbf{w}_i) * \text{influence}_i)$

5.7.2 Shortcomings of SFL over CFL

SFL is more computational complex than CFL largely due to the influence calculation.

SFL's influence calculation, when done for every pairing, has a computational complexity of $\mathcal{O}(N_I)$ during Stage 1 where N_I is the number of institutions due to needing to train a model at every institution, but during Stage 2 it has $\mathcal{O}(N_I^2)$ complexity because every model (N_I models) is tested at every institution (N_I institutions). However, this computational cost is spread

between N_I institutions in both cases. CFL does not feature an influence calculation phase, so this computational cost is entirely extra.

My proposal also features a $\mathcal{O}(N_I)$ cost during the aggregation part of federated learning due to the aggregation happening at every institution. However this SFL cost is again spread between all institutions, leading to an effectively $\mathcal{O}(1)$ cost in time. This matches the time cost for CFL.

The total network usage of SFL is $\mathcal{O}(N_I^2)$ because every every pair of institutions transfer a model across. This compares to $\mathcal{O}(1)$ for CFL (the model is sent back and forth along a single connection). The SFL cost is spread between all institutions, so the network usage per institution is $\mathcal{O}(N_I)$.

The running of SFL, like CFL, happens in the background using spare compute power and network bandwidth for the systems in question.

5.7.3 Benefits of Removing the Central Location

In CFL there is some central location, typically owned by the research centre or business that deployed the algorithm, which is used for model aggregation and communication. SFL does not feature any central location. This gives the benefits of control and privacy over CFL. Namely:

No risk of downtime or malfunction: CFL relies entirely on the server at this central location functioning correctly over long periods of time. This server may be owned by a third party.

Control over computation: The computation of a new model happens on-site with SFL, meaning the users themselves have more control over any issues faced than if the aggregation happened off-site at an unknown location.

One fewer exposed points: With no central server, SFL has one fewer locations at which a security breach is a concern.

5.7.4 Dealing with Compromised Institutions

Unlike CFL, SFL has an inherent resistance to compromised institutions. When calculating the influence between institutions, an institution that has been compromised and is no longer providing a correct model or whose data are no longer valid, will have very little influence from and to other institutions. This leads to it becoming effectively disconnected from the federation.

The influence calculation must be repeated on a regular basis to ensure any drift between institutions or sudden change in an institution's performance is captured readily. It might also be possible to have metrics available at each institution that could determine when an institution has changed sufficiently to warrant re-calculation of the influences. For example, a separate set of validation data chosen when the influences are calculated and on which the model must achieve a certain accuracy, otherwise it is considered to have diverged. This validation data is taken from the institution being validated.

5.7.5 Adding and Removing Institutions

The federation is a flexible structure where institutions may be added or removed at any time. Here I describe how I do these two tasks.

Adding an institution: I must first run the influence method on all connections from this new institution to every other institution and to itself, such that set of influence values now includes the new institution. I then use these influences to aggregate an initial model for this institution. I do not use the influence of the new institution to itself, as the institution

does not have a previous model to use – instead I use the normalised influences from all other institutions to this new institution. Finally, this model is trained locally for some period of time before it is used for evaluation. This new institution now acts as any other institution in the federation.

In the unlikely scenario that a new institution does not have any influence to or from other institutions due to differences in the datasets, then this institution cannot join the federation because the federation can neither help it nor receive help from it. Training an uninitialised model locally at this institution may be a suitable alternative.

Removing an institution: To remove an institution from the federation I set all influences to and from it to be zero with the exception of the influence to itself, which now becomes 1. It now has no influence to the rest of the federation and the rest of the federation has no influence to it. The model will learn locally (wholly influenced by itself) as it goes forwards.

In exceptional cases the removal of an institution may also disconnect other institutions from the federation. An institution whose only other non-zero influencing institution is the one removed now has no influence to or from the rest of the federation. This disconnected institution may reconnect later when a new institution is added that bridges the gap between it and the federation.

Removing all trace of an institution: A challenge arises if an institution desires to be *fully removed* from the federation. I define this as any trace of that institution being part of the federation being removed. An institution may wish to do this for privacy reasons. The model parameters,

the *knowledge*, learned on the data at this institution will have diffused gradually through the federation with each cycle. The interactions of knowledge transfer and mixing between institutions is complex and this learned knowledge cannot be easily separated out for any particular institution.

As a possible solution, I could keep a complete history of every model in every institution for every cycle and all data used and influences. If such historical data were available, I could recalculate every cycle of federated learning from when the removed institution joined, but exclude influences from this removed institution. This would result in new models at each institution that have no trace of the removed institution. However this is impractical over the time scales a federated model may be deployed for in terms of the storage capacity and cost needed for data that will be rarely used, and for computational cost of re-running the entire federated from cycle zero. Further, such full removal may lead to a step change in performance of a model at an institution, which when dealing with medical patient data is undesirable and has inherent risks.

5.7.6 Other Data-Comparative Techniques

I use model training/testing to measure the similarity between two datasets, however this is a rich area with many other techniques available when the datasets are modelled as probability distributions. In this section I state the more prominent methods. I have not used these methods as they require transferring the data (or some aspect of it) between institutions or only offer a symmetric relationship.

f-divergence: The f-divergence is a measure of the divergence of two probability distributions (P, Q) using the odds ratio [280] and a weighting

function, f [6, 63, 212]:

$$D_f(P||Q) = \int_{\Omega} f(\frac{dP}{dQ})dQ \tag{5.10}$$

The function (f) takes one of several forms depending on the desired properties. Specific examples can be found in Table 5.3.

Divergence	Corresponding f(x)
KL-divergence [164, 165, 96]	$x \log x$
Reverse KL-divergence	$-\log x$
Hellinger Distance [125, 221, 95]	$(\sqrt{x}-1)^2$, $2(1-\sqrt{x})$
Total Variation Distance [99]	$\frac{1}{2}\|x-1\|$
Pearson χ^2-divergence [220]	$(x-1)^2$, x^2-1, x^2-x
Neyman χ^2-divergence (reverse Pearson) [39]	$\frac{1}{x}-1$, $\frac{1}{x}-x$

Table 5.3: Some examples of the f-divergence function [177].

Jensen-Shannon divergence: A symmetrised and smoothed version of the KL-divergence. Essentially the average KL-divergence of distribution 1 given distribution 2, and distribution 2 given distribution 1 [178].

Bhattacharyya distance: A symmetric measure of the "distance" between two probability distributions via a measure of the amount of overlap [30]:

$$D_B(p,q) = -ln(BC(p,q)) \tag{5.11}$$

where D_B is the Bhattacharyya distance, p and q are the probability distributions, and BC is the Bhattacharyya coefficient, which has two closely related definitions depending on if the probability distributions are discrete or continuous.

For discrete distributions:

$$BC(p,q) = \sum_{x \in X} \sqrt{p(x)q(x)} \tag{5.12}$$

with x being a sample.

For continuous distributions I take the integral over the sample range:

$$BC(p,q) = \int \sqrt{p(x)q(x)}\, dx \tag{5.13}$$

The Bhattacharyya distance is related to the Hellinger Distance (HL):

$$HL(p,q) = \sqrt{1 - BC(p,q)} \tag{5.14}$$

Kolmogorov–Smirnov: A non-parametric test for equality of two one-dimensional probability distributions via sampling of the cumulative probability [155, 272]. The statistic is the maximum difference between the cumulative distributions.

5.8 Datasets

In my experiments in this chapter I use two datasets. For the first dataset I take the MNIST digits — 70 000 2D 28 x 28 pixel handwritten digits — introduced previously in Chapter 2 (Section 2.9) and transform these with affine transformations or noise to form the first dataset (Sections 5.8.1). For the second I use the BraTS dataset, which is a set of 3D brain volume scans (Section 5.8.2).

5.8.1 MNIST

To simulate institutions I divide the MNIST data into disjoint subsets (institutions) and apply affine transformations or add noise to the samples to represent differences between institution's. Each institution is constructed to have a random subset of the MNIST data, however I have a restriction of a minimum of three data samples for each of the ten classes for the SFL influence three-fold training.

MNIST Transformations

I have four affine transforms and four noise transforms that can be applied to each MNIST sample: rotation, translation, scaling up (magnification), shearing, salt & pepper noise, Gaussian noise, intensity inversion, and gradient noise. These are detailed in Table 5.4 and shown in Figure 5.8. For institutions with a transform applied, this transform is applied consistently to each data sample in that institution.

The transformations provide challenging datasets with some of the transformations resulting in part of the digit being outside the view (see Figure 5.8). This challenge is intentional to put the federated learning systems under stress and emphasise the difference between SFL and CFL.

Transform	Definition
Rotation	A random rotation is applied around the central point of the digit with zeros padding new pixels.
Translation	The digit is translated along both axes with zeros padding new pixels.
Scaling Up	The digit is zoomed in around its central point.
Shearing	The digit is sheared — pixels are translated as a function of their position along each axis. Zeros pad new pixels.
Salt & Pepper Noise	Random pixels are set to the minimum or maximum intensity of all samples.
Gaussian Noise	Every pixel value has a value added to or subtracted from it with this value sampled from a Gaussian distribution for every pixel.
Inversion	Every pixel value is multiplied by -1.
Intensity Gradient	An amount is added to every pixel depending on the pixel's position along each axis.

Table 5.4: Definitions of the transforms used for the MNIST data. See Figure 5.8 for visual examples.

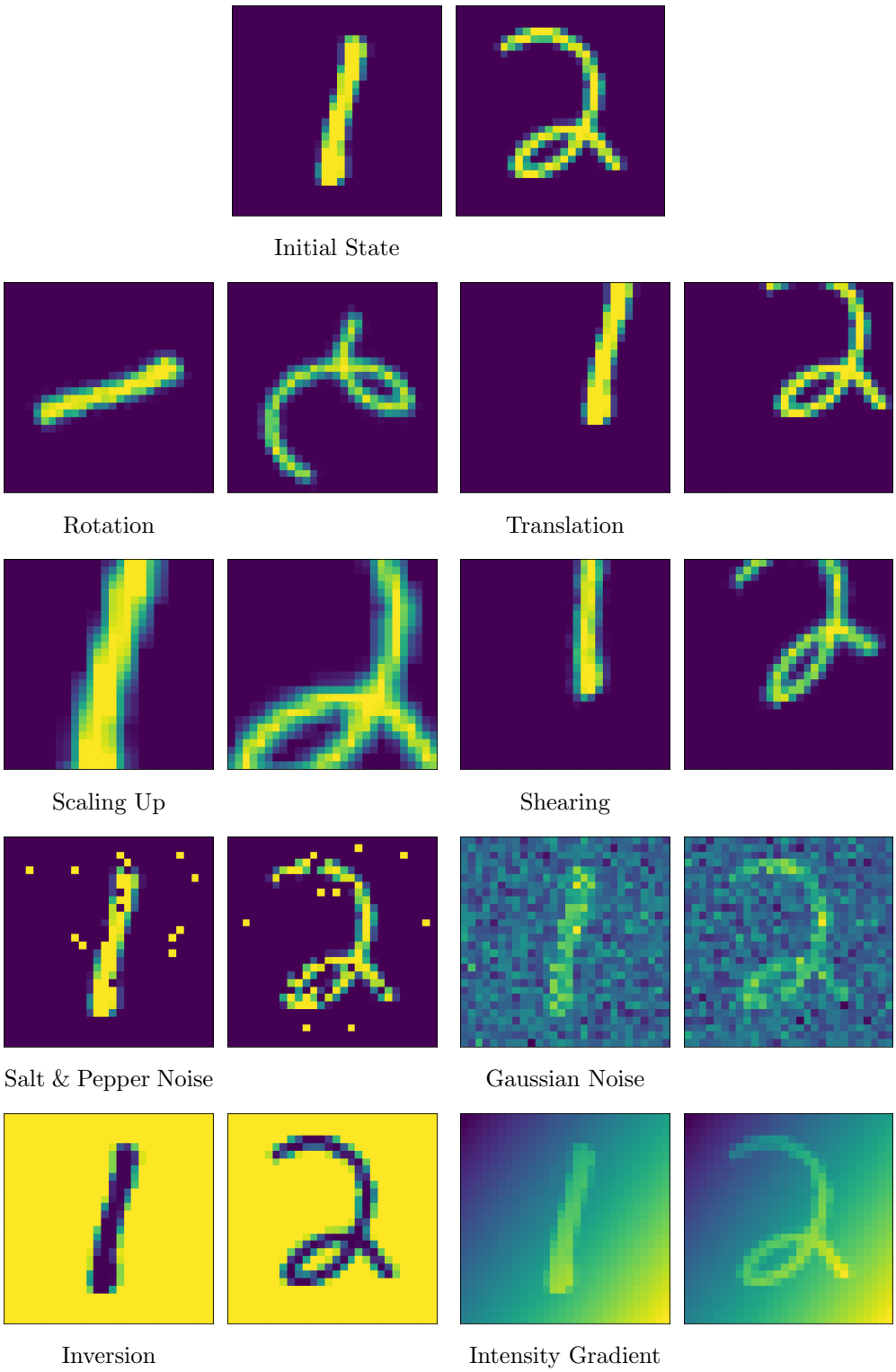


Figure 5.8: Visual examples of the transformations used. Two digits are shown: a one and a two. These are in their original form at the top, and then each pair shows a transformation from this initial state with the transformation noted below each pair. Refer to Table 5.4 for transform definitions.

5.8.2 BRaTS

I use the volumetric imaging data from the Multimodal Brain Tumor Segementation Challenge 2018 dataset (BraTS)[265] — a multi-institutional medical dataset that consist of 285 patients with brain glioma (a type of cancerous brain tumour). Each patient has four scan modalities: MRI T1, Contrast-Enhanced MRI T1 , MRI T2, and MRI Flair — these modalities were introduced previously in this thesis (Chapter 1 — Section 1.1.2), as well as ground truth segmentation of the background (BG), gadolinium-enhancing tumour (ET), peritumoral edema (ED), and the necrotic (fluid-filled) & non-ehancing (solid) tumour (under a single ground truth label — NCR/NET) [205].

Due to the four imaging modalities providing similar data for the model, in the interest of a runtime that does not slow the clinical decision process, I only use the contrast enhanced T1 which provides a good contrast between the tumour regions. Each scan consists of 155 slices of 240×240 pixels and come from one of 14 institutions. As spatially adjacent slices are similar, I subsample every tenth slice leading to 16 slices per patient. Table 5.5 details the amount of data from each institution and Figure 5.9 shows an example slice of contrast-enhanced T1 and ground truth.

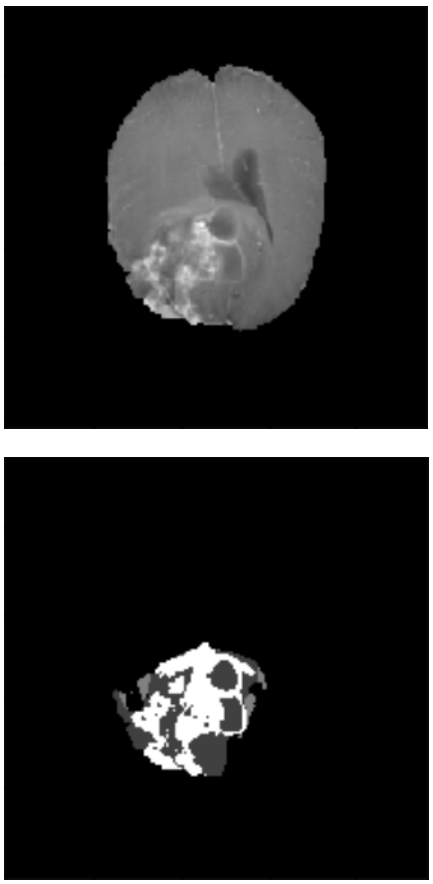


Figure 5.9: An example BraTS slice with **Top:** Contrast-enhanced T1 MRI and **Bottom:** Ground truth segmentation for the glioma.

Institution	Patients	Slices / Modality	Train/Test Size	Included?
HGG2013	20	320	224/96 (14/6)	Y
HGGCBICA	88	1408	992/416 (62/26)	Y
HGGTCIA01	22	352	240/112 (15/7)	Y
HGGTCIA02	34	544	384/160 (24/10)	Y
HGGTCIA03	12	192	128/64 (8/4)	Y
HGGTCIA04	8	128	96/32 (6/2)	N
HGGTCIA05	4	64	48/16 (3/1)	N
HGGTCIA06	8	128	96/32 (6/2)	N
HGGTCIA08	14	224	160/64 (10/4)	Y
LGG2013	10	160	112/48 (7/3)	Y
LGGTCIA09	11	176	128/48 (8/3)	Y
LGGTCIA10	35	560	384/176 (24/11)	Y
LGGTCIA12	6	96	64/32 (4/2)	N
LGGTCIA13	13	208	144/64 (9/4)	Y

Table 5.5: The number of patients, total number of slices per modality (16 slices per patient), and the size of the training and testing dataset in terms of number of slices and in brackets the number of patients these slices came from for each institution within the BraTS dataset. I also note if I include the institution in my experiments (Y=yes, N=no).

I exclude institutions with fewer than 10 patient scans as I believe the variability these small datasets contribute to the federation exceeds the benefit of including them. Some institutions are prefixed with HGG or LGG. These stand for High-Grade Glioma and Low-Grade Glioma respectively, which refers

to the type of glioma in the patient data. High grade is the faster growing and more aggressive form of glioma but can mimic LGG in MRI scans [1] (often a surgical sample is required to differentiate the two types). I divide the patients into a training and testing dataset within each institution with 70% contributing to training and 30% to test — this split is frequently used for neural network training, although I also explored similar divisions and found no significant accuracy difference. This division is done at a patient level, so all slices from a single patient appear in either the training or testing dataset. The proportion of each class in the training and testing datasets for the used institutions can be found in Table 5.6. The classes are strongly biased towards the background class.

5.9 Experiment Design

I detail five experiments chosen to highlight the behaviour of SFL and contrast it with CFL. The first four experiments use the MNIST dataset with transforms for simplicity to allow for focus on the behaviour of the algorithms. The final experiment uses the BraTS medical dataset. The experiments are run for 50 cycles with three cross-fold validation to quantify noise-induced error boundaries. I return and update all model parameters in SFL and CFL at each cycle. 50 cycles is chosen as prior experiments (not shown in this thesis) showed that around 30 cycles achieves accuracy convergence, and the additional 20 act as a buffer for experiments that converge more slowly due to uncontrollable factors.

My evaluation on MNIST occurs at the sample level — classification of whole samples to one of ten digits. For BraTS I evaluate at a pixel level and use Dice as the evaluation metric. Model training and evaluation is explained in detail in Section 5.10.

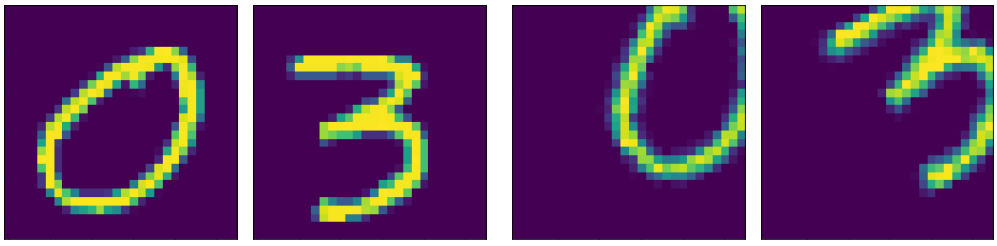
	Training Class Proportions				Testing Class Proportions			
Institution	**BG**	**NCR/NET**	**ED**	**ET**	**BG**	**NCR/NET**	**ED**	**ET**
HGG2013	98.7	0.2	0.8	0.3	98.6	0.4	0.7	0.3
HGGCBICA	99.2	0.1	0.5	0.1	99.1	0.1	0.6	0.2
HGGTCIA01	99.1	0.2	0.5	0.3	98.6	0.2	0.9	0.4
HGGTCIA02	99.0	0.2	0.6	0.3	98.9	0.1	0.7	0.3
HGGTCIA03	98.8	0.1	0.7	0.5	99.0	0.1	0.5	0.3
HGGTCIA08	98.7	0.2	0.7	0.4	98.9	0.2	0.6	0.4
LGG2013	99.2	0.3	0.5	<0.1	99.4	0.4	0.2	<0.1
LGGTCIA09	98.7	0.5	0.6	0.2	98.8	0.6	0.6	<0.1
LGGTCIA10	98.7	0.6	0.7	<0.1	98.7	0.6	0.6	0.1
LGGTCIA13	98.9	0.6	0.4	<0.1	98.1	1.0	0.9	0.1
Average:	98.9	0.3	0.6	0.2	98.8	0.4	0.6	0.2

Table 5.6: The percentage of each ground truth class in the ground truth data of each institution for the training and testing datasets for the BraTS institutions used in my experiments.

5.9.1 Experiment 1 - Knowledge Transfer Test

This experiment has three institutions. Two have normal MNIST digits (no transform), while the third uses a strong shear transform to make it different to the other institutions. Figure 5.10 demonstrates this transform. The first of the normal institutions (I_1) has a small dataset (30 training samples) while the other (I_2) is large (3000 training samples). The third institution (I_3) has 3000 training samples.

The purpose of this experiment is two-fold. First, it aims to show how the influence from the large normal institution to the small institution is larger than from the other way around due to the size of the datasets alone. Second, the third institution should then both give to and receive from the other institutions an almost-zero influence due to the data transform.



Initial State Sheared

Figure 5.10: Visual examples of the shear transformation used in Experiment 1. A 0 and a 3 are shown on the left as non-transformed MNIST digits, and on the right with the shear transform applied.

5.9.2 Experiment 2 - Noisy Institutions

I use nine institutions in this experiment. At the first institution I have 30 training samples of non-transformed MNIST digits, but at the other eight I do not use MNIST digits — I instead generate Gaussian noise samples for the data and assign a random class label to each. Figure 5.11 shows an example sample. The noise is normalised so it matches the intensity range of MNIST

digits. There are 30 samples with no signal at each of these eight institutions. The digit labels 0–9 are assigned in equal proportions. The choice of 30 samples here is a trade-off against speed and reliability of the final results. As only one model (institution) is training on MNIST data, little data are needed to train successfully. Adding further samples did not produce any increase in accuracy in my tests. The choice of 30 samples for the noisy institutions is to match the number of samples in the first institution to ensure that CFL uses an equal weighting from all institutions during model aggregation.

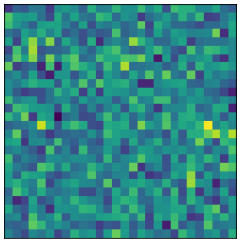


Figure 5.11: An image of MNIST size (28 x 28 pixels) consisting of Gaussian noise only (no digit present).

This experiment aims to demonstrate how SFL learns to ignore these signalless institutions and focuses on maximising accuracy at the signal institution, while CFL fails to do so and thus performs worse than SFL at the signal institution.

5.9.3 Experiment 3 - Identical Institutions

Here I use three institutions, each with 100 training samples with no transforms. The choice of 100 samples ensures a balanace against speed and accuracy. I carried out experiments not shown here showing an increase in samples does not affect the results.

I expect this to show how SFL approximates to CFL when the domains and dataset sizes are identical as the influences will be approximately identical.

5.9.4 Experiment 4 - 50 Institutions

I have 50 institutions, each with 100 training samples, and each institution has a random transform at a random intensity within a sensible range. The transforms are selected from Table 5.4. The 100 samples is to match Experiment 3 in training parameters, and the 50 institutions was chosen partially arbitraily as a large number to challenge the algorithms without being unrealistic. My experiments took around six days to finish, which I considered to be an acceptable balance between waiting and attaining results.

This experiment stresses the SFL algorithm to explore its ability to operate at scale and find strong influences between similar institutions in a very mixed scene.

5.9.5 Experiment 5 - BraTS Data

I train on ten BraTS institutions (see Table 5.5) each containing a different amount of data ranging from 112 training slices to 992. Each institution has data from a different real world institution. I run for 30 cycles to see the performance of SFL and CFL in this realistic scenario. Further details about the data are in Section 5.8.2.

5.10 Model Training and Evaluation

I use two models, one for the MNIST data (Experiments 1-4) and one for BraTS (Experiment 5).

5.10.1 MNIST Model

The MNIST model (Figure 5.12) uses a series of convolutional and pooling layers, with fully-connected dense layers at the end that condense the output

down to ten classes, which are each given a probability value. The class with the highest probability is chosen as the predicted label (as per Equation 5.5). Rectified linear unit activations are used throughout the model except for the final layer, which uses Softmax to deliver probability values.

The MNIST model is pre-trained using 1000 raw MNIST digits that are randomly selected but not used in any of the institutions. There are 1000 test samples at each institution for evaluation regardless of how many training samples are used. The experiments are run for 50 cycles each. Accuracy of the class predictions vs ground truth classes for samples is used as the evaluation metric.

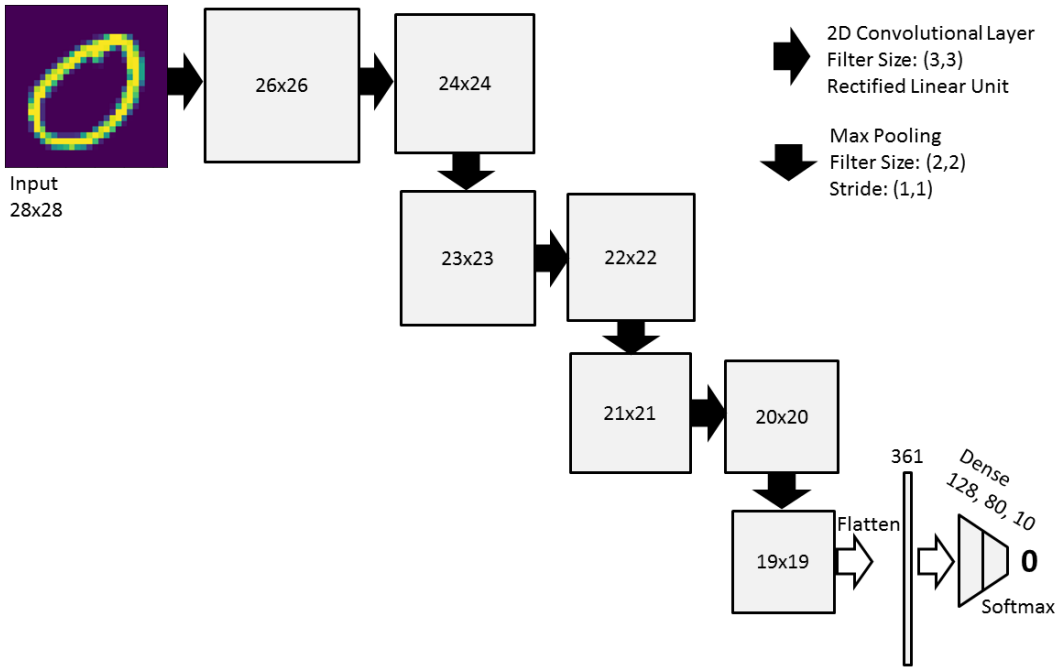


Figure 5.12: The model for the MNIST experiments. The number of kernels used for the four convolutional layers from start to end are: 32, 16, 8, 16.

5.10.2 BraTS Model

My BraTS model (Figure 5.13) is based on the U-Net by Ronneberger *et al.* [245], which has shown to be one of the most successful architectures for the BraTS data [144]. The U-Net uses a series of convolutional and pooling layers in the first half of the model to reduce the dimensionality and focus information; it then uses a form of upsampling — in my case transpose convolutions — and further convolutions to expand the data back up and continue processing it. Skip connections are used to concatenate the layers in the first half of the model to the second half.

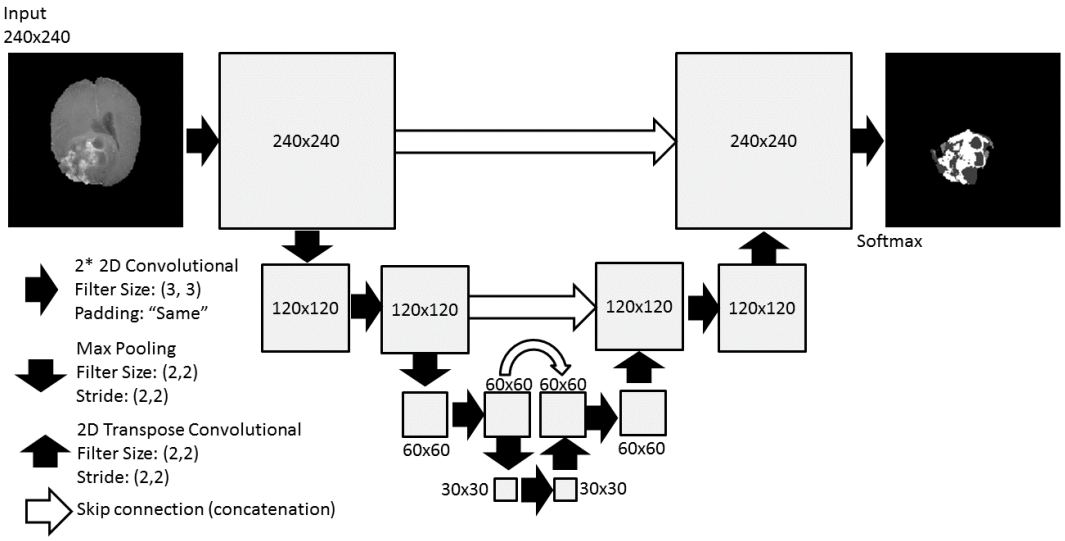


Figure 5.13: The model for the BraTS experiment. Note the convolution arrows represent two convolutional operations, while the other types are a single operation. The number of kernels is 16 for the highest level (least reduced), and doubles with every level going down: 32, 64, and then 128 for the lowest (most reduced) level. The number of kernels then halves going back up.

My model is a smaller and simpler version of the U-Net due to the computational cost of training a full U-Net. I have reduced the "depth" (number of pooling/transpose convolution layers) and reduced the number of kernels

used. This reduced model has a weaker performance than the full U-Net, yet is sufficient for my comparative purpose. I tested with a range of depths and convolutions and found this one to be both quick and capable of detecting the glioma to a sufficiently high accuracy to show the differences between the algorithms.

After the final layer, a (1,1) kernel size convolutional layer with softmax activation and four kernels is used to generate the probability maps for the four classes. Rectified linear units are used for the activation function elsewhere.

I pre-trained my model on the data not used in the selected institutions (refer back to Table 5.5).

A custom loss function adjusts the weight of the classes based on the inverse proportion of the class in the dataset is used for training. Referring to Table 5.6, the background class is weighted very lowly, while the other classes — particularly the ET class — have high weightings. This reweighting is standard practice to avoid the network finding the trivial solution by assigning every pixel to the background class. Figure 5.14 shows an example of my predictions and the ground truth for two classes in a typical sample evaluation. My experiment with this model runs for 30 cycles.

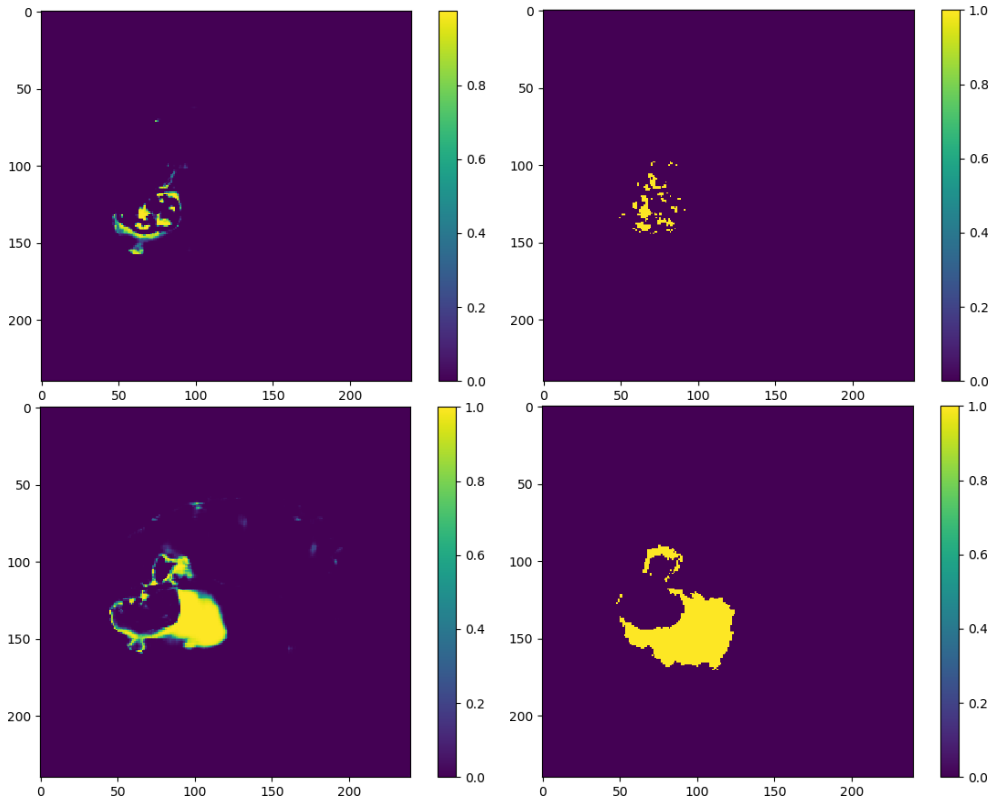


Figure 5.14: Predictions (left) and ground truth (right) segmentations for an example slice of BraTS data showing the NCR/NET class on top and the ED class beneath.

During the evaluation stage I apply a three pixel morphological closing operation to remove boundary complexity believed not to be significant. Performing the same effect on the predictions ensures a fair evaluation. This processing led to a minor improvement in my performance for all methods. The Dice score [74], on a per class basis and averaged, is used for the evaluation metric. Dice is a measure of the overlap of two regions and scores from 0 to 1 with 1 representing a perfect overlap. In terms of the number of True Positive (TP), False Positive (FP), and False Negative (FN) pixels, it is defined as:

$$\text{Dice} = \frac{2\,TP}{2\,TP + FP + FN} \tag{5.15}$$

5.11 Baseline Measures

In addition to CFL and SFL, I employ three further methods as baseline performance measures:

Global Pooling: I take the data in all institutions and pool it in a new location (separate to all institutions). A model is trained and evaluated on this pooled dataset. This non-federated baseline provides an upper-bound performance.

Local: I train and evaluate a model at each institution. There is no transfer of models or data. Clearly, federated methods must exceed this trivial baseline to warrant the added complexity.

Ensemble: This is the same as the Local method during training, however during evaluation the set of all models are evaluated at each institution and the average accuracy of the models gives the performance. This method can be seen as a halfway between Local and federated learning. In CFL and SFL I average the models from all institutions, but with the Ensemble method I do not average, I simply evaluate them all individually and average the results. The models the Ensemble method trains will naturally diverge from each other throughout the cycles since they are not tied together. This answers the question "How does model aggregation affect the performance compared to using an ensemble method?"

Formally, the ensemble and local methods have the same training function:

$$M_i^{t+1} \leftarrow \text{SGD}(M_i^t, L(M_i^t, d_i^{train}))\forall i \in \{1, \ldots, N_I\} \tag{5.16}$$

with notation defined previously in Section 5.3.

The global pooling uses a notion of some central location:

$$M_{cent}^{t+1} \leftarrow \text{SGD}(M_{cent}^{t}, L(M_{cent}^{t}, D^{train})) \tag{5.17}$$

The ensemble method also has a more advanced evaluation procedure than what was define previously (Section 5.3.2) to factor in the ensemble of models:

$$M_E = \{M_{i=1}, \ldots, M_{i=N_I}\} \tag{5.18}$$

An ensemble of models, M_E, is formed from the set of all models.

I calculate the set of class probabilities using this ensemble of models:

$$P(\mathbf{y}|x^{test}, M_E) = \{P(y_0|x, M_E), \ldots, P(y_{max}|x, M_E)\} \tag{5.19}$$

where

$$P(y|x, M_E) = \frac{1}{N_L} \sum_{i=1}^{i=N_L} P(y|x, M_i) \tag{5.20}$$

which is the average class predictions for a given class, y, by the ensemble.

5.12 Results

Here I detail the results to the experiments described in Section 5.9. Throughout this section I will show three types of graphs. The first are the *influence heatmaps*, which display the amount of influence a each institution receives from the other institutions.

The second type of graph seen in this section are the *result graphs*. These show a plot for each institution and within each plot are five lines showing the five evaluation methods: global pooling, the ensemble method, the local method, CFL, and SFL. These figures should be viewed in colour. The x-

axis shows the cycle number, which is synonymous to the number of training loops the models have gone through. The y-axis is the performance of the models on the (unseen) testing data at that institution. The global pooling method is identical for each institution as it has been calculated separately and superimposed onto each figure as a reference.

The third type of graph are the *accuracy graphs* that give the accuracy of each method at each institution at 50 cycles. They can be considered a cross-section through the corresponding result graph at cycle 50. These figures also display the average accuracy across the institutions.

The purpose of the influence heatmaps is to provide insight into the SFL method, the result graphs show training performance over time, and the accuracy graphs show the final accuracy performance.

All experiments were conducted three times with the results averaged. The shaded regions on the result graphs and the error bars on the accuracy graphs show one standard deviation from the mean.

I shorten institution names from here throughout the remainder of the chapter from, for example, "Institution 1" to "I_1".

5.12.1 Experiment 1 - Knowledge Transfer Test

I have three institutions. Two (I_1 and I_2) from the same domain while the third is from a very different domain, and two (I_2 and I_3) with plenty of data while I_1 have few.

In Figure 5.15 I observe that I_1 receives a higher influence from I_2 than from itself. It also receives no influence from the third institution. I_2 is influenced predominantly by itself with a small influence from I_1 and a minimal amount from I_3. I_3 on the other hand is almost entirely influenced by itself and recognises the other two institutions as outliers relative to itself.

The evaluation in Figure 5.16 show that for I_1 the local accuracy is the weakest method, with CFL and the ensemble method next, SFL closely after, and the global pooling method on top. In I_2 CFL outperforms the ensemble method, followed by SFL, local, and pooling. The ensemble method has a similar performance in I_1 and I_2. Finally, in I_3, the pooling, local, and SFL methods have the highest accuracy, but CFL approaches their performance in later cycles. The ensemble method is 66% accurate.

Overall, from Figure 5.17, SFL, CFL, and pooling are the highest accuracy methods at around 96 – 97% average accuracy, local is next at 92%, and ensemble is the poorest with 85%.

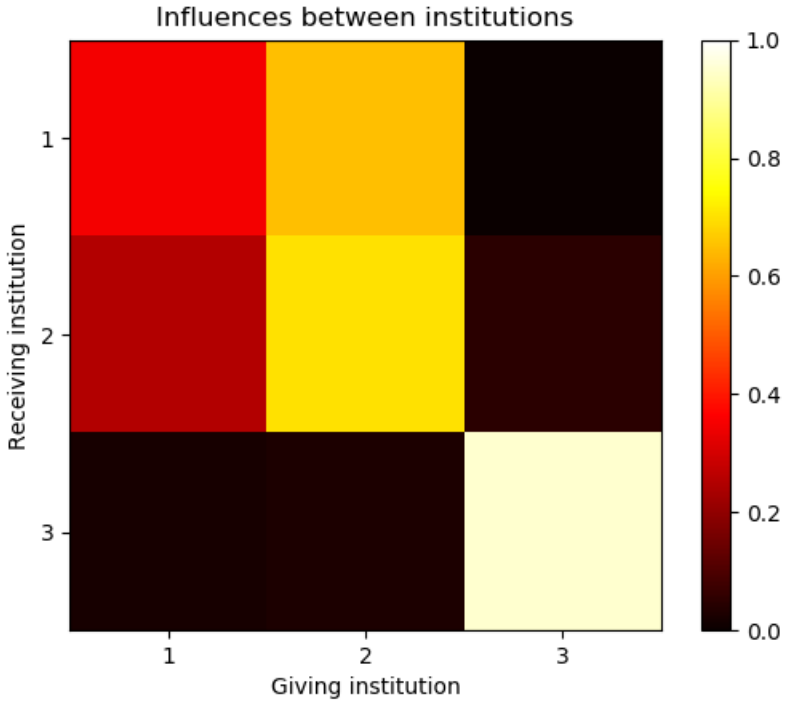


Figure 5.15: The influence heatmap for Experiment 1 (Knowledge Transfer).

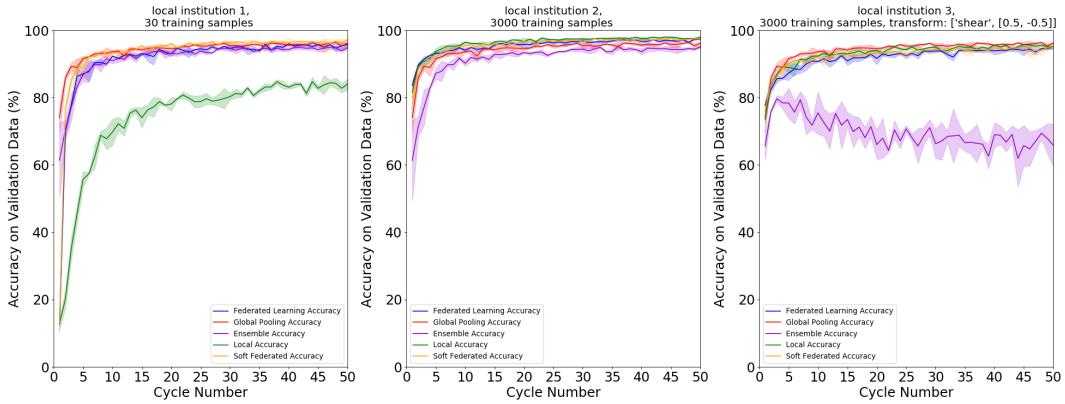


Figure 5.16: The result graphs for Experiment 1 (Knowledge Transfer). Each plot represents an institution showing the performance of the five evaluation metrics (y-axis) across 50 cycles (x-axis). The colors are: Yellow - SFL, Blue - CFL, Purple - Ensemble, Green - Local, and Red - Global Pooling. The shaded region is the error at one standard deviation from the mean for the three-fold experiments. The institutions are numbered from the left starting at I_1.

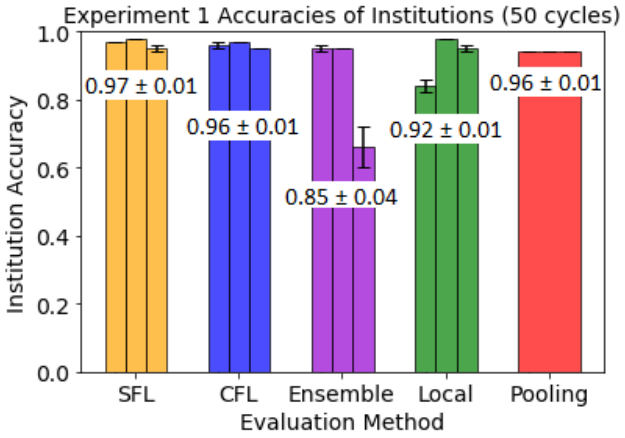


Figure 5.17: The accuracy of each of the three institutions in Experiment 1 (Knowledge Transfer) at 50 cycles for the five evaluation methods. The error bars are one standard deviation from the mean for the three-fold experiments The institutions are displayed starting from I_1 on the left. The number on each set of bars is the average accuracy for that evaluation method.

5.12.2 Experiment 2 - Noisy Institutions

I have one institution (I_1) with normal MNIST digits, and eight institutions (I_2 – I_9) with no signal, but Gaussian noise in the shape of an MNIST digit instead. In Figure 5.19 the pooling method performs weakly on this data, but does perform above random chance accuracy (20%). The evaluation methods on I_2 – I_9 show no learning, with all performances at the random chance level (about 10% as there are ten classes).

I_1 has a good performance with all methods. The ensemble method performs around 40% with substantial errors (10-15% absolute error each way), CFL reaches 95% but does not appear to have plateaued by 50 cycles. The local method and SFL are at 98% (Figure 5.20). SFL shows I_1 receives almost zero influence from the other institutions (Figure 5.18) and the other institutions have a pattern of influences that is noisy. The number of training samples in each fold of the influence function is 10 (30 samples between three folds).

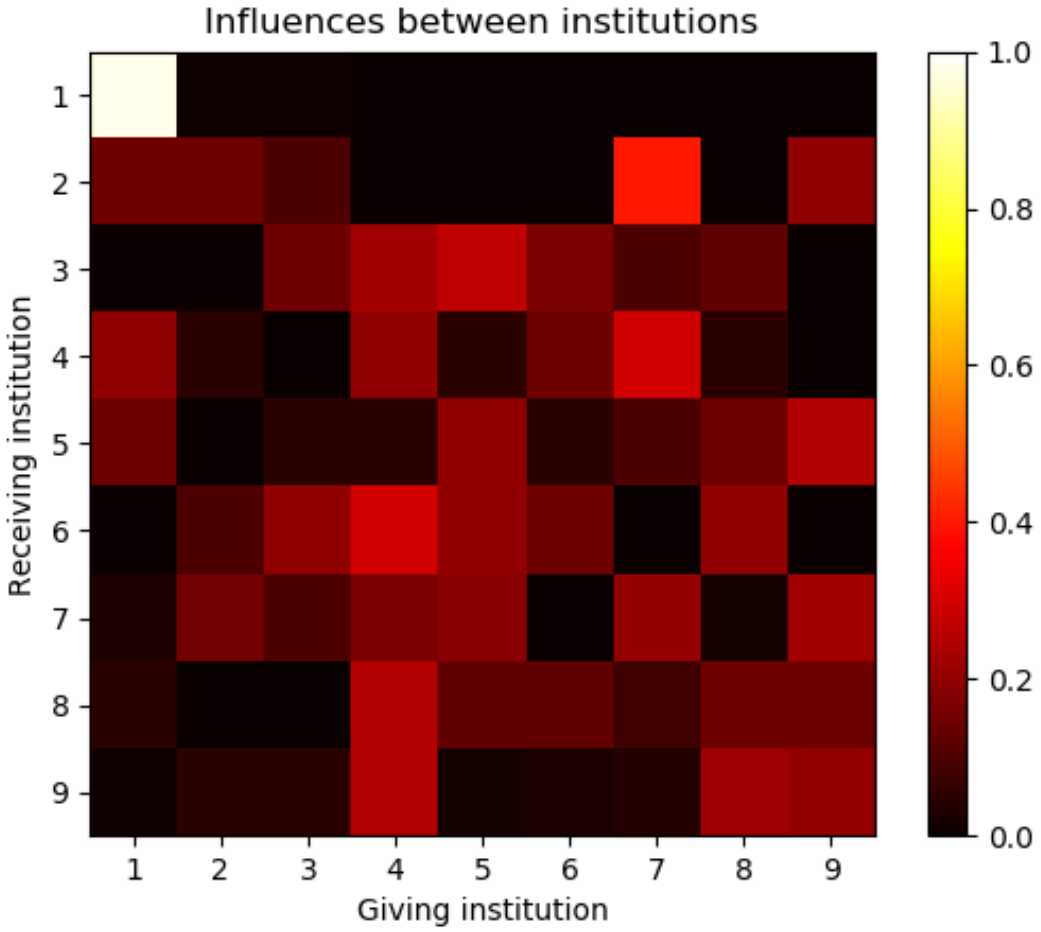


Figure 5.18: The influence heatmap for Experiment 2 (Noisy Institutions).

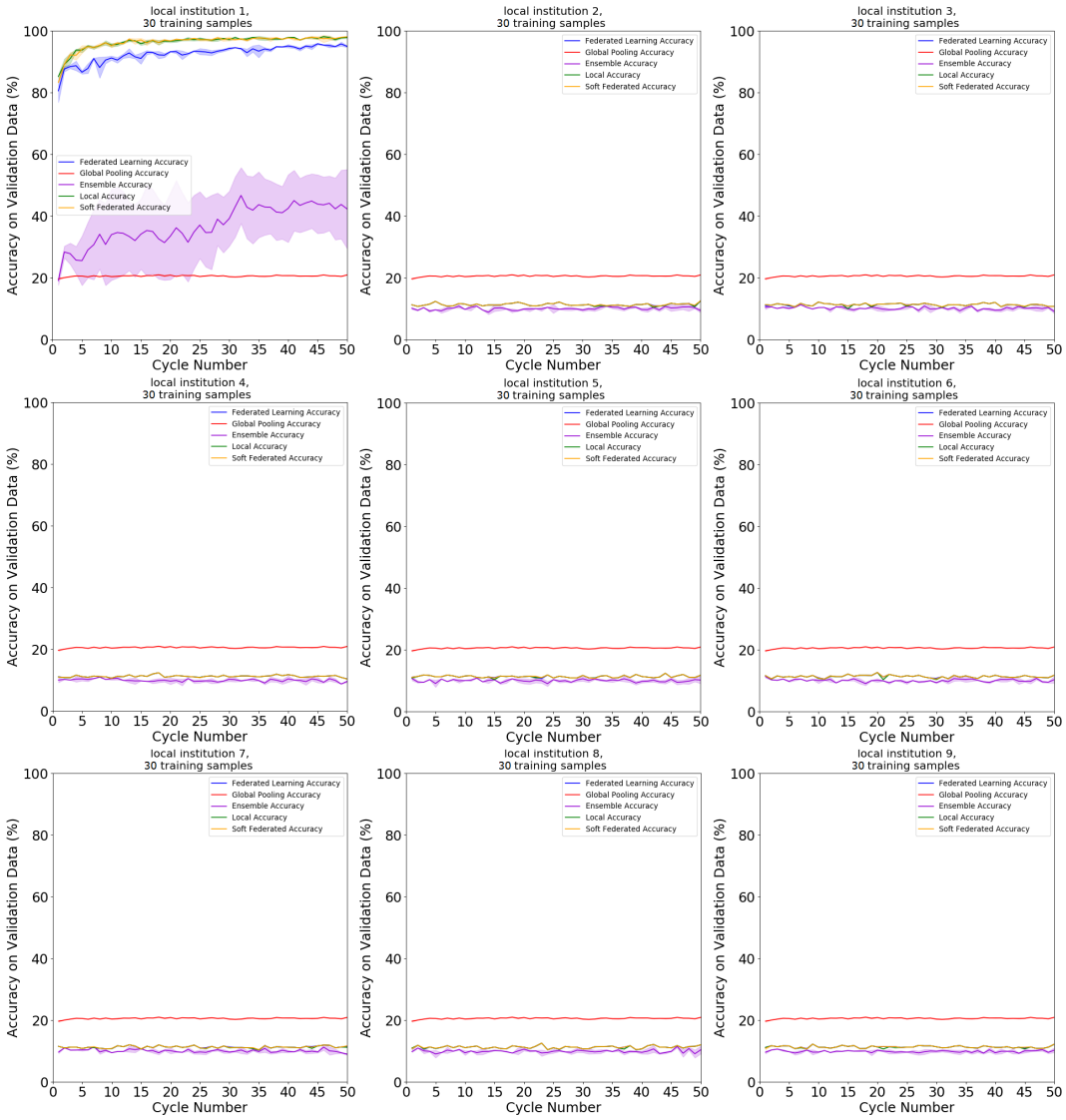


Figure 5.19: The result graphs for Experiment 2 (Noisy Institutions). Each plot represents an institution showing the performance of the five evaluation metrics (y-axis) across 50 cycles (x-axis). The colors are: Yellow - SFL, Blue - CFL, Purple - Ensemble, Green - Local, and Red - Global Pooling. The shaded regions are the error at one standard deviation from the mean for the three-fold experiments. The institutions are numbered from the left along then down starting at I_1.

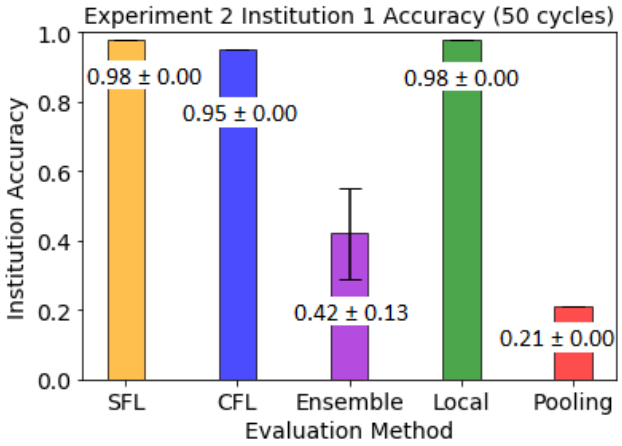


Figure 5.20: The accuracy of I_1 (normal MNIST data) in Experiment 2 (Noisy Institutions) at 50 cycles for the five evaluation methods. The error bars are one standard deviation from the mean for the three-fold experiments. The number on each bar is the accuracy for that evaluation method.

5.12.3 Experiment 3 - Identical Institutions

This experiment uses three identical institutions with limited training data (30 samples). The evaluation methods have comparable accuracies between institutions (Figure 5.22) with the local method having the lowest overall accuracy, and pooling and the ensemble method having the highest performance. SFL and CFL are the next highest (Figure 5.23). The influences between institutions are approximately equal (Figure 5.21)

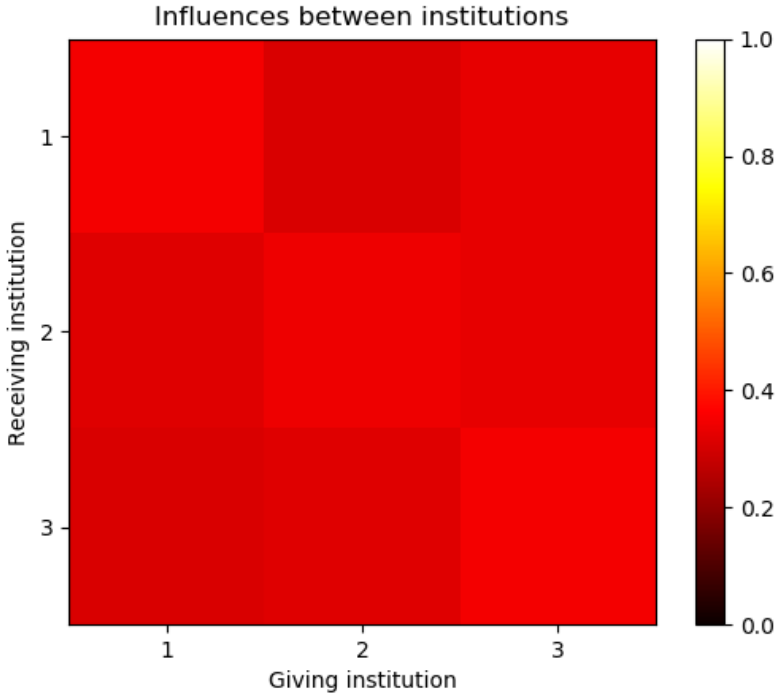


Figure 5.21: The influence heatmap for Experiment 3 (Identical Institutions).

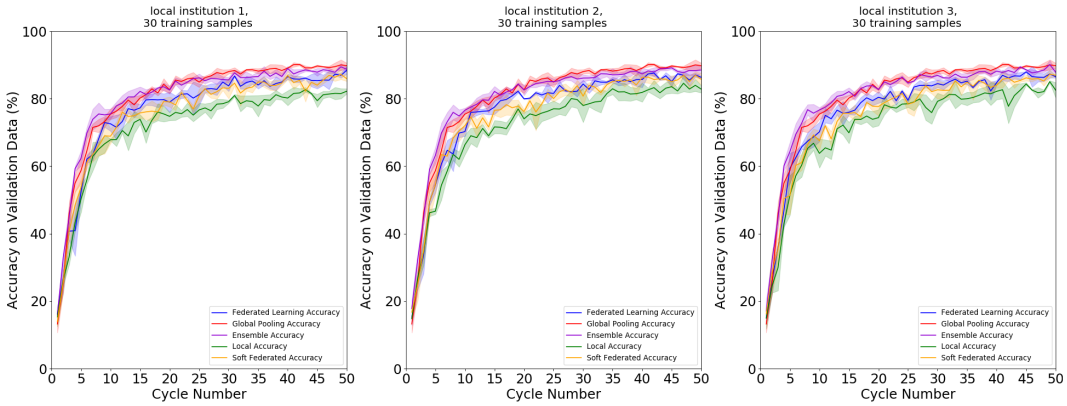


Figure 5.22: The result graphs for Experiment 3 (Identical Institutions). Each plot represents an institution showing the performance of the five evaluation metrics (y-axis) across 50 cycles (x-axis). The colors are: Yellow - SFL, Blue - CFL, Purple - Ensemble, Green - Local, and Red - Global Pooling. The shaded regions are the error at one standard deviation from the mean for the three-fold experiments. The institutions are numbered from the left starting at I_1.

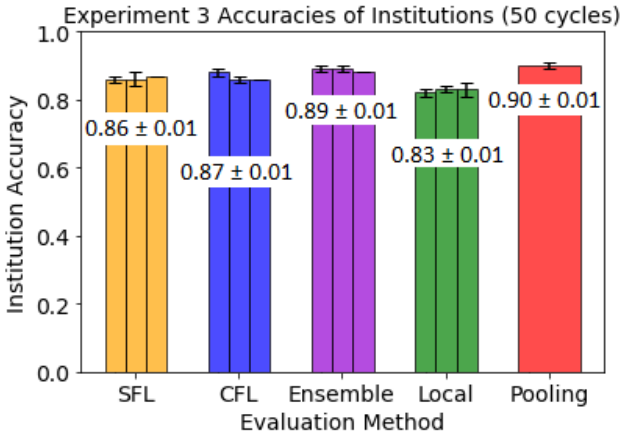


Figure 5.23: The accuracy of each of the three institutions in Experiment 3 (Identical Institutions) at 50 cycles for the five evaluation methods. The error bars are one standard deviation from the mean for the three-fold experiments. The institutions are displayed starting from I_1 on the left. The number on each set of bars is the average accuracy for that evaluation method.

5.12.4 Experiment 4 - 50 Institutions

This experiment has 50 unique institutions each with 100 training samples. The institutions have a data domain sampled randomly from Table 5.4 with random parameter settings in a sensible range. The results show that institutions with the inverse transform, such as I_3 in Figure 5.24, get their influences from other inverse institutions, with a minor level of noise from the other institutions. The transforms rotation, Gaussian noise, salt & pepper noise, and intensity gradient in particular receive a high level of influence from non-rotation or non-Gaussian noise institutions respectively. Often they receive the highest influence from their own institution.

Figure 5.25 shows these institution's result graphs and Figure 5.26 the overall accuracies but divided into the different transforms.

The overall accuracy figure for each method when averaged across the transform sets (i.e. taking the values in Figure 5.26 and averaging) and averaged

across individual institutions is displayed in Table 5.7.

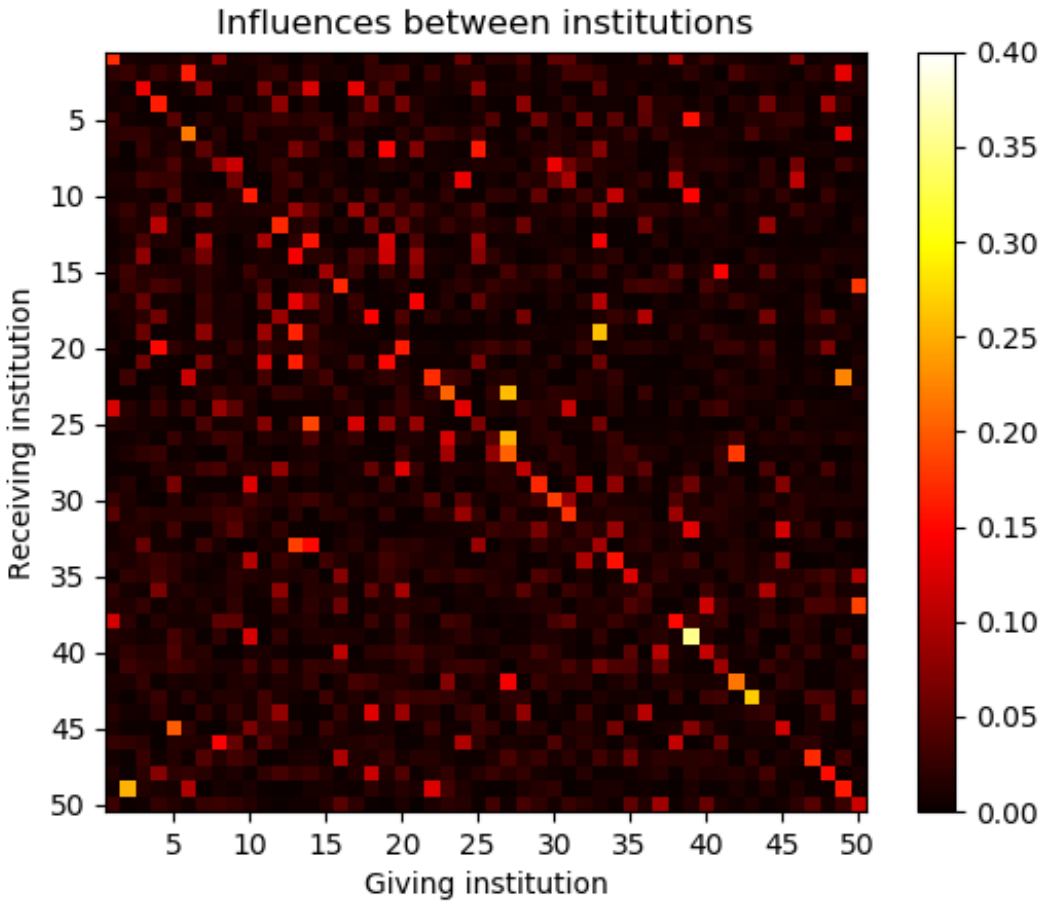


Figure 5.24: The influence heatmap for Experiment 4. Note the scale has been adjusted to improve visualisation.

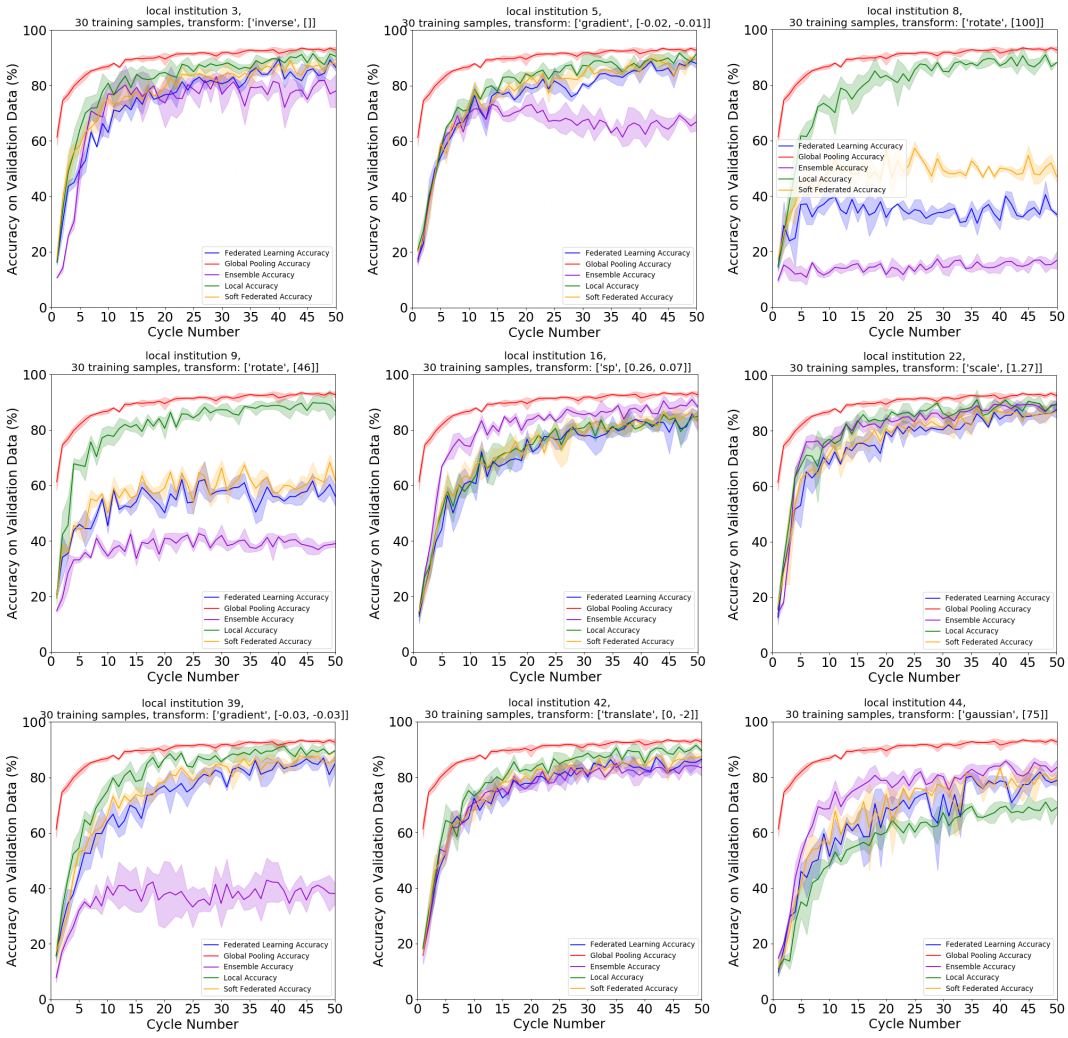


Figure 5.25: A selection of result graphs for Experiment 4. Each plot represents an institution showing the performance of the five evaluation metrics (y-axis) across 50 cycles (x-axis). The colors are: Yellow - SFL, Blue - CFL, Purple - Ensemble, Green - Local, and Red - Global Pooling. The shaded regions are the error at one standard deviation from the mean for the three-fold experiments. The institutions are numbered from the left along then down as institutions 3 (inversion), 5 (intensity gradient), 8 (rotation), 9 (rotation), 16 (salt & pepper noise), 22 (scaling up), 39 (intensity gradient), 42 (translation), and 44 (Gaussian noise).

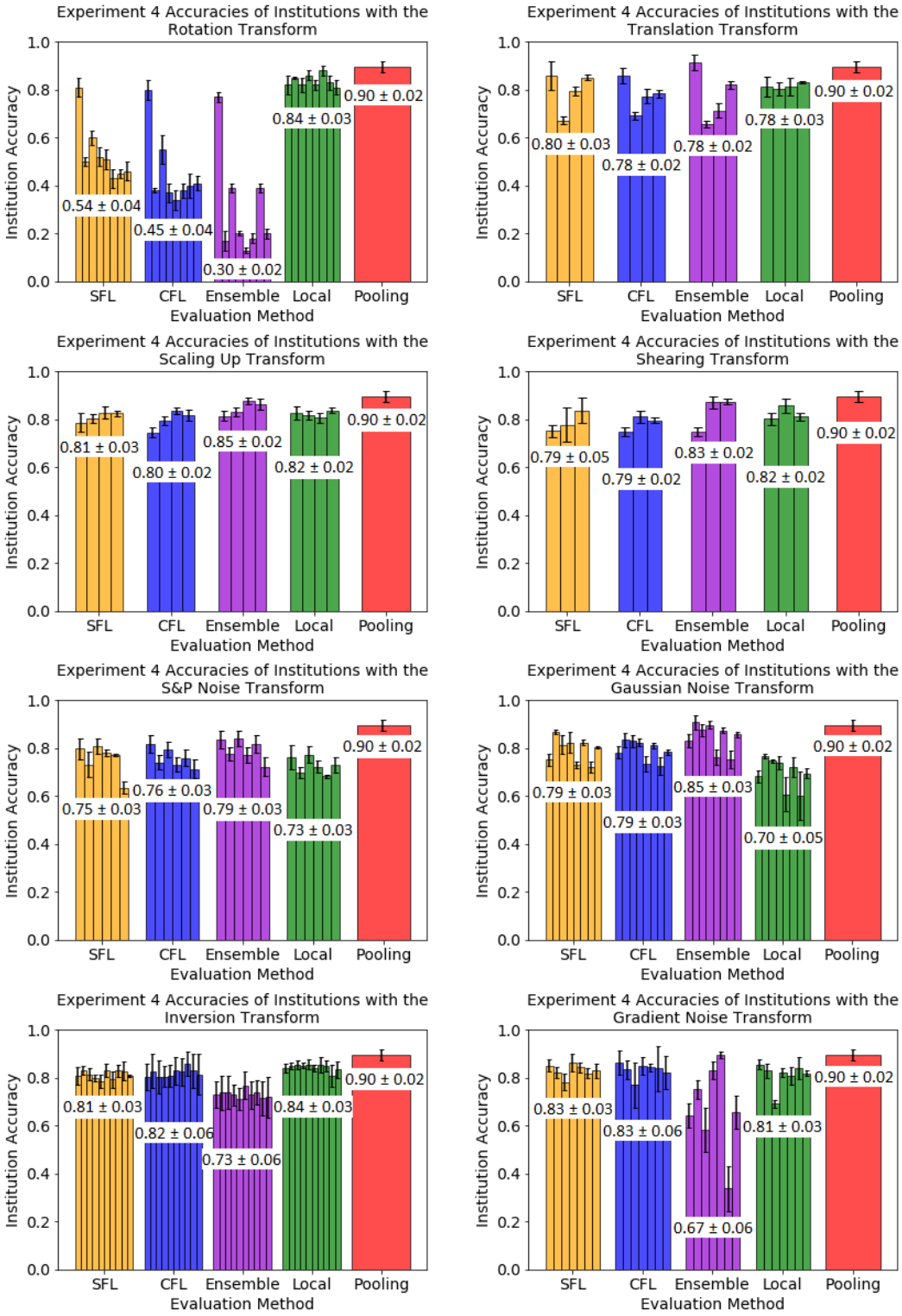


Figure 5.26: The accuracy of each of the 50 institutions at 50 cycles in Experiment 4 divided into figures according to the transform used. The error bars are one standard deviation from the mean for the three-fold experiments. The number on each set of bars is the average accuracy.

Method	Transform sets average	Institutions average
SFL	0.77±0.03	0.76±0.03
CFL	0.75±0.04	0.74±0.04
Ensemble	0.73±0.04	0.70±0.04
Local	0.79±0.03	0.79±0.03
Pooling	0.90±0.02	0.90±0.02

Table 5.7: The average accuracy for Experiment 4 for each method when averaged across transform sets and across institutions.

5.12.5 Experiment 5 - BraTS Data

I train for 30 cycles at ten institutions each with their own varying quantities of MRI medical data for brain glioma. The influence graphs (Figure 5.27) show some institutions (1, 2, 4, 5, 9, 10) receiving the most influence from themselves, but overall the influences are variable. The results and accuracies of the methods in Figures 5.28 and 5.29 show the ensemble method performing the poorest at 0.12 Dice at 30 cycles, the local method next at 0.27 although with great variation between the institutions, SFL and CFL are next around 0.45, and global pooling achieves 0.54. In some cases the SFL, CFL, and local methods exceed the pooling method by a small margin.

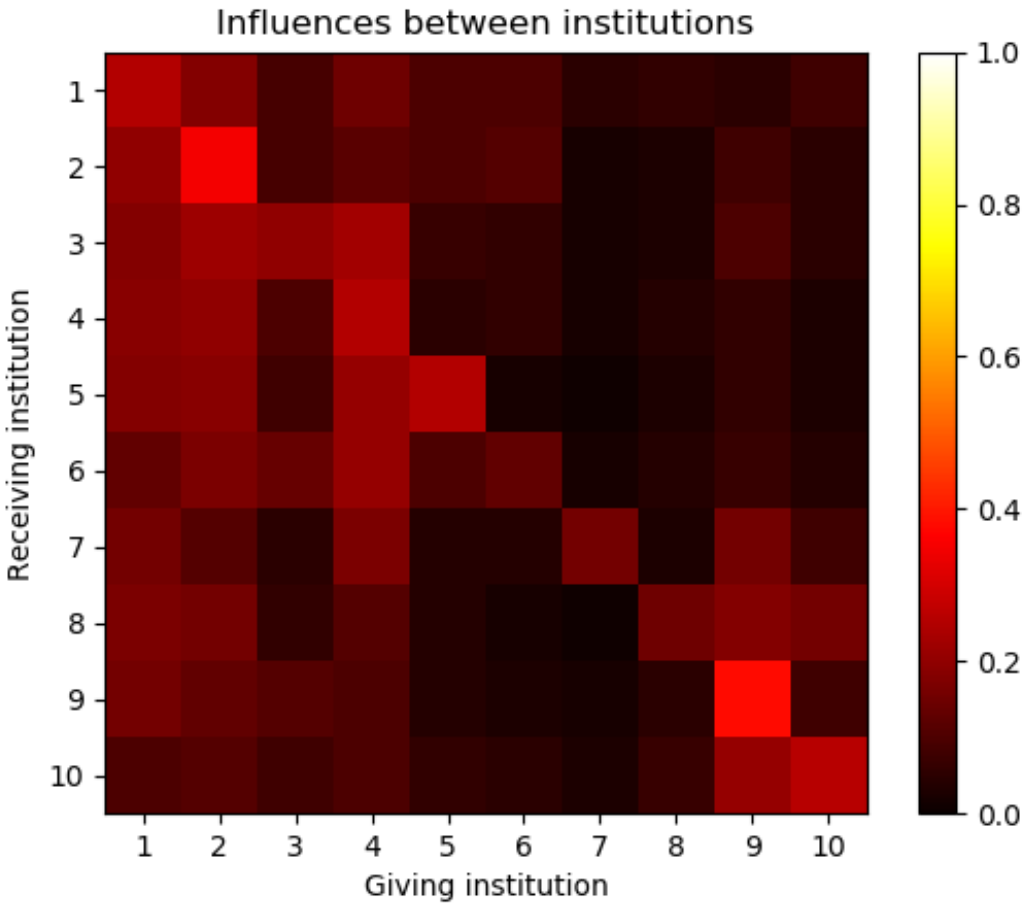


Figure 5.27: The influence heatmap for Experiment 5 (BraTS Data).

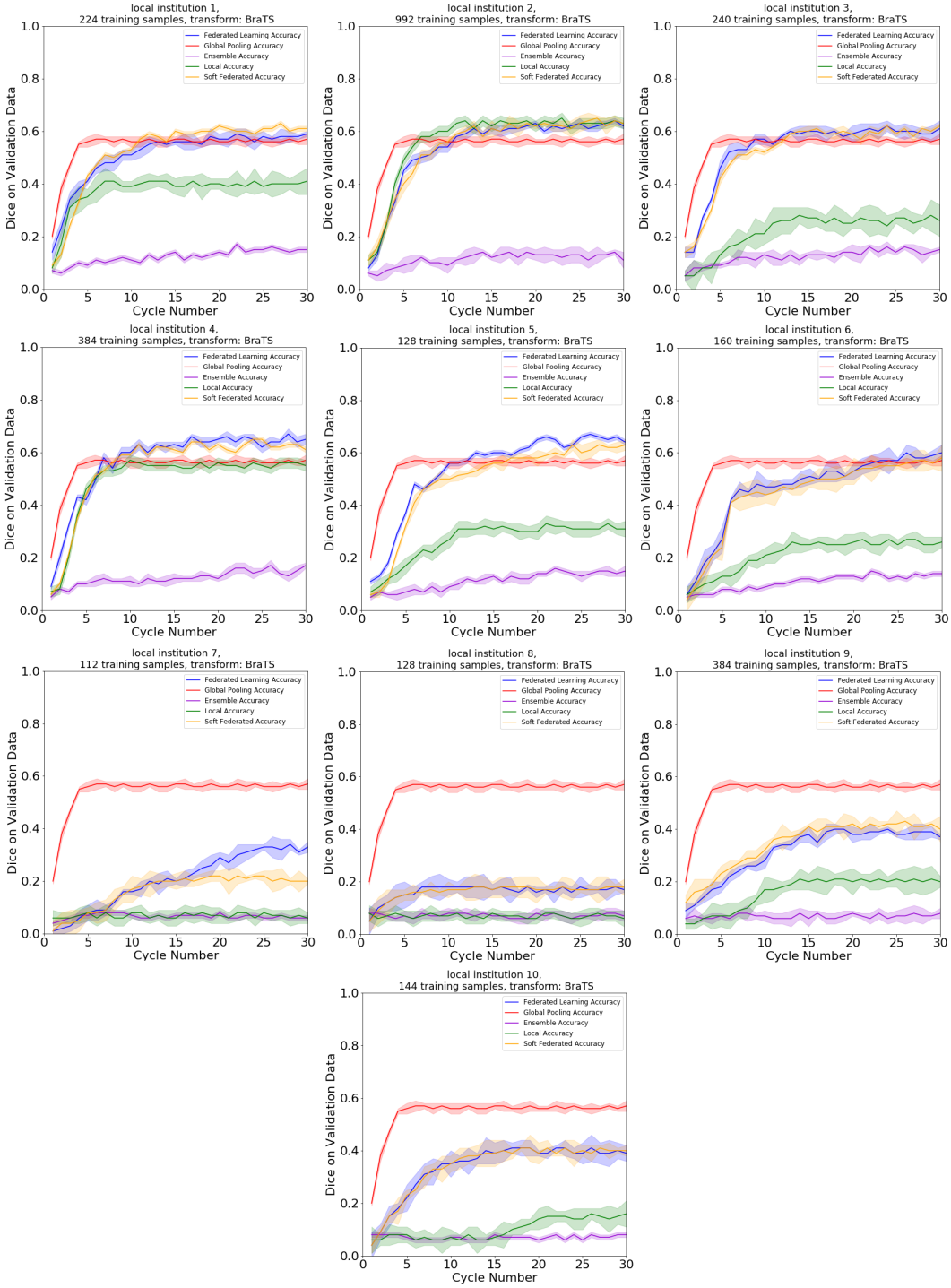


Figure 5.28: The result graphs for Experiment 5 (BraTS Data). Each plot represents an institution showing the performance of the five evaluation metrics (y-axis) across 30 cycles (x-axis). The colors are: Yellow - SFL, Blue - CFL, Purple - Ensemble, Green - Local, and Red - Global Pooling. The shaded regions are the error at one standard deviation from the mean for the three-fold experiments. The institutions are numbered from the left starting at I_1.

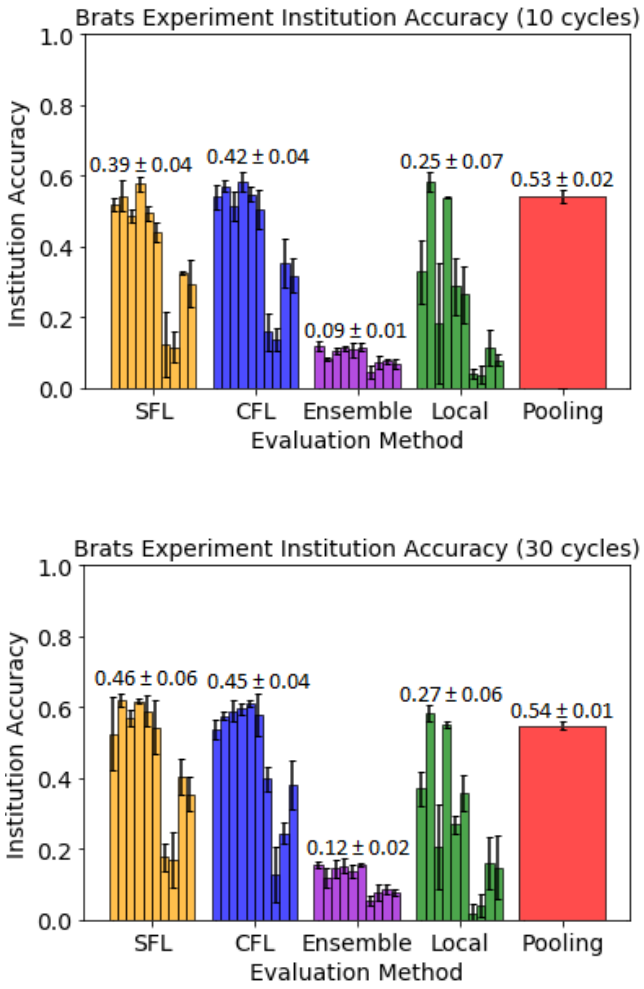


Figure 5.29: The accuracy (Dice coefficient) of each of the ten institutions in Experiment 5 (BraTS Data) at 10 cycles (**Top**) or 30 cycles (**Bottom**) for the five evaluation methods. The error bars are one standard deviation from the mean for the three-fold experiments. The institutions are displayed starting from I_1 on the left. The number on each set of bars is the average accuracy for that evaluation method.

5.13 Discussion

5.13.1 Experiment 1 - Knowledge Transfer Test

This experiment demonstrates a number of key principles surrounding the functioning of federated learning.

1. A lack of training data in I_1 leads to a lower local accuracy here with all other methods outperforming by a margin of around 10-15%.

2. The similarity of the data in I_1 and I_2 leads the ensemble method to perform similarly between these institutions; but the uniqueness of I_3 causes difficulty for this method. This is because two of the ensemble models are evaluating I_3's data using I_1's or I_2's training data, which is different, and this is out-voting the ensemble model from I_3. The ensemble method does not factor in the relative usefulness of each model for the institution I am evaluating at.

3. Pooling performance is weakened by the variety of data, although the model is still able to learn to classify the mixed data well. In I_2 all methods apart from the ensemble method are more accurate than pooling.

4. The CFL method is able to operate well on mixed institutions despite a small decrease in accuracy in I_3 due to the different data present. This shows that the models at each institution are learning features that are useful at all institutions, which may be edges and shapes. When the models are averaged they remain useful.

5. The SFL method is able to learn faster than CFL as it discovers a useful influence weighting for aggregation. This effect is most noticeable in I_1

around cycles 5 – 10 (Figure 5.16) and is caused by the large influence received from I_2 effectively accelerating the training by providing additional (useful) training samples. The accuracy of CFL approaches that of SFL at a high number of cycles, but remains hindered by I_3.

5.13.2 Experiment 2 - Noisy Institutions

As expected, the global pooling method fails here as much of the data has no signal — only noise. The 0.21 accuracy figure comes about from it achieving a high accuracy on the samples with signal, which make up 11% of the total data, and achieving a random chance accuracy of around 10% (one in ten classes) on the other 89% of the samples. Overall this should provide an accuracy of around 0.20 and Figure 5.19 shows the global pooling method near this accuracy across the cycles.

The ensemble method performs around 40%. This is because eight institutions ($I_2 - I_9$) are voting for random classes due to their random training, and so the deciding vote goes to I_1 if these eight decisions are all different. From the local method I know the I_1 model will make a correct guess about 80% of the time. However sometimes by random chance the eight noise institutions may vote together for an incorrect class in a way that outweighs I_1. This is why the performance of the ensemble method fails to reach that of the local method. The random chance element in the voting leads to a variable performance, hence the large error.

I_1 receives minimal influences from $I_2 - I_9$. The models learned at the noise institutions are random, but this leads to these models sometimes performing *slightly* above trivial chance when tested on I_1. This leads to the small influences seen in Figure 5.18. These weak influences do not appear to have an impact on SFL performance, which outperforms the other methods.

CFL on the other hand receives noise from these models during model aggregation, leading to a significant reduction in performance. The average of the noisy updates from $I_2 - I_8$ will be approximately a model with zero-value parameters. Therefore when aggregating with I_1 the new model will be I_1's model with its parameter values shifted by 8/9ths towards zero. Due to the way neural networks work, if all parameters within the network are multiplied by the same amount, the performance of the network will not change as it is the relative differences between parameters that define a network's ability.

The observed drop in performance is due to the average of $I_2 - I_8$ being a model that is not zero everywhere. Some parameters will be slightly above zero, and some slightly below. This may lead to the relative differences in I_1's new model changing, resulting in a small fall in accuracy.

5.13.3 Experiment 3 - Identical Institutions

As I expect, the institutions all perform similarly to each other. Local accuracy is the lowest due to a lack of training data available locally. The pooling and ensemble methods have the strongest accuracy because they operate on the assumption that all data is sampled from the same distribution, which is a valid assumption here. The small reduction in performance for SFL and CFL relative to pooling and ensemble could be due to the nature of the model aggregation. Useful parameters may average out in this stage, which slows the training speed of these methods compared to non-model averaging techniques such as the pooling and ensemble methods.

The almost identical influences between the three institutions show that SFL approximates CFL in cases where there are no distribution shifts between institution. CFL is identical to SFL when all influences are equal and all training set sizes are the same.

The general conclusion here is that if the insitutions' data domains in a federation are identical (or highly similar), then there is no need to carry out the relatively expensive SFL algorithm since CFL will perform fine in this case.

5.13.4 Experiment 4 - 50 Institutions

This experiment shows the advantages of SFL on a highly mixed set of institutions. In some cases, such as with institutions with the rotation transform, SFL performs significantly better than CFL due to the influences focusing on their own institution here. This is due to each rotation providing a largely unique data domain. The local performance here however is much stronger than SFL as SFL has been unable to recognise that isolating the institutions from the federation provide the best outcome — instead the small amount of influence noise from all other institutions has greatly weakened the useful signal (see Figure 5.24). For all other transforms SFL achieves comparable performance to CFL.

The global pooling method features the highest accuracy, which means that pooling data from these different domains and training on the whole set provides the best model for generalising. The performance being higher than the local method indicates two things. First the amount of training data (100 samples) is insufficient to train the best models — this is especially apparent in the salt & pepper and Gaussian noise institutions where the performance of the local method is 0.73 ± 0.03 and 0.70 ± 0.05 respectively — the lowest of the transform sets. The random noise component of these two transforms lead to more training data being needed. Second, the model itself has sufficient capacity when working on the pooled data to train on the varied training data and fit well. The institutions must have some useful general knowledge that can be pooled (the higher CFL performance confirms this), and the global

pooling model has been able to generalise to all transforms.

This experiment also shows the strengths and weaknesses of the ensemble method, which performs the second best (behind global pooling) on the salt & pepper and Gaussian noise institutions due to its ability to average out the noise with the ensemble's models, but it finds in particular the rotation transform challenging as the parameters learned on other transforms cannot be easily applied here.

5.13.5 Experiment 5 - BraTS Data

This successful experiment showed that federated learning can be used on medical data and that SFL is able to recognise (to limited extent) the differences between institutions that see high-grade glioma and ones that see low-grade. The first six institutions have high-grade brain glioma, while the remaining have low-grade. This is reflected in the influences (Figure 5.27) where the first six receive considerably more influence from themselves than from the final four. Likewise, in these low-grade institutions, the influences from the high-grade appear subdued.

However, there is little difference between SFL and CFL at each institution (Figure 5.29). The influences are similar across all institutions, which indicates that there is not enough domain shift in this dataset for SFL to improve the performance.

The ensemble method is weak at all institutions, showing a slow rate of improvement at some over time. The local performances are also weak, with the exception of I_2 which has much more training data than the other institutions.

I_8 is an interesting case as no method is able to perform particularly well (see Figure 5.28). The data at that institution appears similar to the other institutions at first glance, however the influence for I_8 from itself is one of the

lowest influences it receives, implying that a local model is unable to perform well on its own data. Upon inspection of its data, I see that it features many narrow pathology regions. These are not as common in the other datasets. Due to the small size of I_8 (11 patient datasets), this will be affecting training. The local method will be struggling to learn, while the models in the ensemble are trained on more rounded regions. The convolutional and downsampling layers will further shrink the data, potentially reducing the narrow regions further.

5.14 Conclusion

I have shown that SFL has several potential use cases where it can outperform other algorithms. In this section I capture key insights into federated learning and the SFL algorithm.

5.14.1 Soft Influences

SFL and CFL represent different ends of a spectrum. At one end, SFL can isolate an institution from the federation if their data domain is unique, and modify the influences in other cases, while at the other end CFL does not perform any localisation. SFL acts like CFL when all the influences are equal. Between these two extremes there is a middle ground where the influences are calculated as in the SFL case, but then they are weighted by some amount towards all being equal ($=\frac{1}{N_I}$), which is the CFL case. I will refer to this CFL case as the *equality level*. This softens the influence values. While it is not clear that using this weighting will improve model accuracy, it does enable the federation to become better connected relative to the SFL case and this may be useful when adding or removing institutions (see Section 5.7.5) to provide

more models for aggregation at a new institution or keeping the institutions connected when removing an institution.

An example scenario where this may be useful is when there is a federation consisting of a set of hospitals and SFL has been used to calculate the influences on some set of recent cases in the hospitals. Now there are two obvious cases which could invalidate the influence values. The first is the event of a disease outbreak — the hospitals start seeing more of one type of pathology (one class). As the data, which will be used for model training, have now changed, the influence values are no longer accurate and without softening them, they may damage the federation.

The second case is when a hospital changes its data acquisition protocol, for example changing the way data are collected or upgrading a scanner. This shifts the data domain and leaves the hospital out of sync with the original influence values. As re-calculating the influences can be costly (Section 5.7.2), I do not want to be doing this frequently. With softer influences, the impacts of this scenario are reduced.

5.14.2 Drifting Influences

An alternative to using fixed influences is to have influences change over time. For instance, after they are initially calculated using SFL, they begin to soften over time, drifting towards the equality level, reducing at a rate proportional to their strength. This works under the assumption that the data domains of the institutions will drift over time, thereby invalidating the influences, but since I may not be able to afford to re-calculate them often due to computational resource constraints, softening them over time works as an alternative to minimise the impact of this drift. A softer set of influences leads to increased coherence between the models, which in turn leads to more stability since each

institution is less isolated than in the case of the initial influences.

I can also switch this around and instead of softening the influences, I proportionally strengthen influence signals to reduce the noise from weaker institutions. Such a technique may have been useful in Experiment 4 (Section 5.12.4) to help isolate the different transforms. This is discussed below in Further Work.

5.14.3 Sparse Influences

For large federations where calculating the influences is prohibitive, it may be possible to calculate them sparsely — on a subset of pairs of institutions — to reduce the time and computational cost. For pairs where it is not calculated, it defaults to the CFL influence level (equality level) and then is normalised to 1. The aim here is to shift the CFL performance towards SFL at a lower time and computational cost than running SFL fully. An important note here is every institution must have the influence calculated to itself (self-pairing) as part of the pairings, as this is usually the strongest influence value. If this is not done, institutions will lack the localisation to themselves and so will fail to perform well in institutions with relatively unique data domains.

5.14.4 Further Work

I would like to explore the SFL algorithm on further medical datasets, particularly ones featuring domains more varied than seen in the BraTS data. I believe SFL will perform well in these cases as my earlier experiments showed it works for varied MNIST institutions.

There are also several routes available for further development of the SFL algorithm. Ideas that I would like to trial in the future include:

Soft influences: As described above. The influence values could be softened either initially or over time to better connect the federation and reduce the impact of domain drift.

Favouring self influences: I give a slight boost to self influence values and reduce the other influences. This helps to favour institutions' own model when there are a large number of other institutions factored in, thereby helping the model stay focused on its own institution. The amount of boost/reduction that works best may be case dependent and is an area for further research.

Specialised model heads: This idea derives from the field of multi-task learning — refer back to Section 5.5 for an overview of relevant literature. I take a base model and train it using CFL. The base model is designed to be capable of learning generally useful features. Then, within each institution, I remove the final classification layer of the base model, add a few additional model layers to the end of the model, such as the addition of convolutional and upsampling layers, as well as a new classification layer. I call these layers the model head. The base model's parameters are then frozen (or otherwise have inertia added, which reduces learning rate) and the entire model is trained locally on the institution's data. This will focus training on the model head, aiming to specialise it by taking the embeddings from the base model and using these as the inputs to the specialised head, which then seeks to fit to the local nuances of the data.

This should improve the performance within an institution. The base model learns generally useful features, while the specialised model heads learn how to convert these features into a high-accuracy prediction.

It is not clear how the useful parameters learned within one head can be transferred to another model's head (at a different institution). Using the notion of influence described in Section 5.7.1 I may be able to use the SFL algorithm to transfer only the model heads between institutions and use these to form the new model heads with each cycle.

Reducing influences that are below the average influence: One idea to improve the signal achieved from strong influence links is to set all influences that are below the equality level to zero (before normalising the influences to one). This means a more sparsely connected federation. This sparsity would also reduce computation time.

The issue with this sparsity is now each model is aggregated from fewer models, which increases overfitting risks. This may only be suitable for federations of many institutions.

A different threshold could be used rather than the equality level, but this choice would be arbitrary.

Another way is to emphasise the differences between institutions. One way of doing this is to square the influences (and then normalise). This increases the influence at high levels while reducing it at low levels. This may have been beneficial in Experiment 4 for the Rotation transform institutions (Section 5.12.4). However the choice of how to emphasise the differences is again arbitrary and a single choice may not be suitable for all tasks or federations.

To scale with the number of institutions, I could also divide the SFL influences by the equality level to provide a final quantity that is normalised against the number of institutions.

5.14.5 An (almost) Infinite Federation

An interesting thought experiment arises when I take soft federated learning to the extreme. What would happen if I took every dataset in existence (regardless of the data within it) and calculated the influences? This is of course impractical, but offers interesting outcomes.

It is common knowledge that different tasks require different model architectures to achieve high performance. For example, the type of model used for imaging data is very different to that used for time series data. While data embedding techniques can (theoretically[5]) be used to reduce all data into a format suitable for a single model architecture, or additional specialised layers or pre-processing at the start of a model could make the data consistent, it remains unlikely that a single model architecture would be adequate for all datasets. Concepts from the field of multi-task learning could be extended here. But let us assume some solution to this problem exists and I can feed very different data into a single learning model.

It is likely that most, if not all, of the datasets will be linked to one another in this extremely large federation. There will be clusters within the federation of closely linked datasets that share similar data — for instance hospitals may form a cluster — and these clusters will be linked to other related clusters. There may also be weak links to a number of other datasets as Experiment 4 showed that quite different datasets can still feature small influences between them (Section 5.12.4). Within each cluster there are likely subclusters for more specific types of data, and these themselves may also have more subclusters, and so on going down into smaller and smaller subclusters.

Now imagine a randomly initialised model being copied to every dataset

[5]In practice this would be very difficult as trying to capture the information within datasets of different features into some common size for a neural network (for instance) is not a straightforward task.

and being trained and then aggregated using the SFL algorithm. The models for very different datasets will diverge across cycles as there are no (or little) influence connections between them, while for similar datasets, and even between similar clusters, the models stay reasonably tied together. All models are tied to each other by the influences somehow, but ones connected through a long series of other institutions will have more flexibility to diverge.

These models are now effectively learning on all data in existence with a focus on similar data. They are localising to a specific dataset (and specific task) while using context from similar tasks to improve their performance. The need to pool the data into a central location is gone too. It also does not require an (extremely) high capacity model to capture all the information in the data, while if the data were pooled like in the global pooling method case, a high capacity model would be needed.

It is unclear if SFL would be able to train the models in practice, or if the task performances would be good. There are also practicality issues such as gaining access to all of the data and having sufficient computational resources. Still, in theory SFL or some variant of it can be used within a highly diverse and large federation.

Chapter 6

Conclusions

6.1 Looking Back

As the global population ages and the shortage of clinicians worsen, it becomes ever more urgent for AI to aid the roles of human clinicians. The topics studied in this thesis — novelty, distillation, and federation — are three important areas in machine learning for medical imaging that bridge the gap between where we are today and where we need to be. The next generation of intelligent healthcare methods must be able to detect diseases and anomalies never seen during training (novelty), must be understandable at the patient level (distilled), and will need to be trained on large amounts of data without transferring sensitive data (federated), especially in the face of ever-tightening data laws.

Chapter 2 outlined the novelty forest, an efficient abnormality detection method that aims to operate at a lower computational cost to existing abnormality methods while offering admissible accuracy performance. An abnormality detection method is one that seeks to classify outliers through the analysis of inliers in the raw data or an embedding of it. Efficiency is especially

important in high dimensional datasets, such as in the medical imaging domain, where the number of dimensions often slows other methods, even after applying embedding techniques.

The novelty forest's performance in terms of speed and accuracy was contrasted with a number of other simple novelty methods on interstitial lung disease data under a number of embeddings in Chapter 3. The results showed that the novelty forest is fast with good general accuracy, but another method — the one-class support vector machine — gave the highest accuracy in a quicker time by using principal component analysis to embed the data.

Next I examined distillation for the analysis of ischemic stroke patients (Chapter 4). I took non-contrast CT data and used a bespoke neural network to analyse and contrast the brain hemispheres to provide a voxel-level probability map of early ischemic change. The challenge here was to translate this 3D volume probability map into a clinically meaningful, easily interpretable, and reliable quantity that could aid clinical decision making.

I achieved this by using an established scoring method — ASPECTS — which scores ischemic stroke severity by the number of affected regions in the most affected hemisphere. I crafted an ASPECTS atlas — a volumetric map of brain regions supplied by the middle cerebral artery — and aligned this to the probability maps. Then, using rules that converted the probabilities into ischemic presence/absence in each ASPECTS region and dichotomising the final score, I delivered a single value that aids the clinician in the choice of patient treatment. It is clinically accepted and easily interpretable, and I demonstrated it had good reliability with other means of scoring ASPECTS.

To finish I looked at federated learning in Chapter 5, which I demonstrated on the task of brain tumour segmentation to prove its feasibility within the medical field. I introduced a novel evolution called soft federated learning that

performed favourably to its conventional form in a series of carefully designed experiments. This soft version allows for a wider and more varied federation of institutions, which enables more hospitals to join the federation without damaging the average accuracy performance.

My results showed that while soft federated learning can be used to recognise and adapt to different domains, it weakens when there are a large number of institutions due to the each institution taking a small amount of the influence. I was unable to show a significant difference with conventional federated learning on medical data, but no negative performance was noted. Further research is needed.

6.2 The Present

The field of machine learning for medical imaging is advancing rapidly. The number of research publications have grown by an order of magnitude every decade over the past 30 years. The use of machine learning has been welcomed in the field, although regulatory restraints have slowed the progress. The new field of federated learning, introduced in its current form in 2017, has opened new opportunities in the data-restrictive world of medical image analysis. Federated learning is generally accepted by clinicians, but there is an inherent risk with a system that changes its behaviour over time, especially when dealing with human life and wellbeing. We must take care to ensure our new systems are reliable, consistent, and able to compete with existing talent.

This research in this thesis helps in many ways to deal with these risks by showing how my machine learning methods compare to other methods out there and can be used for abnormality detection (novelty), can be interpreted easily and reliably despite their complexity (distillation), and can be used despite the transfer restrictions faced by data today (federation).

6.3 Going Forward

Detailed suggestions for future work are given at the end of each chapter, but here I capture the most promising ideas and summarise what I have learned.

6.3.1 Novelty

The novelty forest offered a means of abnormality detection with minimal time cost, especially when dealing with a large number of dimensions. I showed it performed reasonably against other simple abnormality detection methods, but also that it has a weakness in the way it utilises a minority of features and thus may miss features that are important in the test data but not present in the training data. In this way it could struggle to build a successful model of normality because the abnormal data may be different in ways not checked by the novelty forest.

To improve the novelty forest's accuracy without reducing speed I could explore new information functions for node divisions, and employ more intelligent searching over the features to try to maximise the variety of features checked. Exactly how this would be done remains a research area for the future.

6.3.2 Distillation

In my distillation work, I learned that it is possible to convert a neural network's probabilistic voxel output to a single meaningful value, however this value is only as reliable as its ground truth. The conversion rules, use of the atlas, and dichotomisation threshold were logical choices to distill the data, but the weakness of the ground truthing limited the final performance.

Future work takes two routes. I could improve the ground truth in the

context of how clinicians use the CT scan to better match my output to theirs to develop a highly reliable and thus deployable commercial tool for this use case. The second route could be to use distillation for a wider range of medical problems to show how it could be used in very different contexts. The distillation method I developed is specific to ischemic stroke in CT brain and cannot be directly applied to any other form of data or pathology.

6.3.3 Federation

With federated learning I described where it is useful and demonstrated its weakness on carefully crafted data. My soft federated learning algorithm aims to overcome these weaknesses, although it also featured weaknesses of its own. For example, federations of many institutions diluted the influence values that provide the power behind soft federated learning. This is an issue when soft federated learning is designed to work with large and varied federations.

Despite this, I believe federated learning will be a key technique for learning on medical data in the future. My soft federated learning provides insight into how this area could be developed going forward. If the influence function was more resilient at telling apart useful (similar) and not useful (dissimilar) institutions, especially when the federation is large, or the way similarity is calculated is improved, then soft federated learning or a technique derived from it could exceed the accuracy of conventional federated learning.

6.4 Closing Remarks

Medical image analysis with machine learning is an extensive, varied, and challenging area. Pathology is greatly varied and in some cases very rare, the machine learning algorithms are complicated with complex outputs while

clinicians need understandable and interpretable outputs, and medical data is difficult to collate in a single location.

This thesis showed a variety of techniques and experiments to aid in these challenges. However, the use of novelty, distillation, and federation extends beyond medical image analysis and are each applicable in many cases in the field of machine learning. I hope the new concepts and methods introduced in this thesis are built upon and inspire research in the future.

Appendices

Appendix A

Opinions About Federated Learning — Full Analysis

To understand the opinions of the potential user base of a federated learning system, I created a survey aimed at clinical staff of all backgrounds to gather their insights into a clinical decision support application that improves during use. Such a system could be deployed in a wide variety of situations, so I desired to receive responses from a wide background. I did not reveal the technical details of federated learning during the survey.

A total of 15 responses were received.

The questions and the raw responses follow. Analysis and discussion of these responses are in Section A.2.

A.1 Survey Questions and Responses

1. **What is your current job title and how many years of experience do you have in this role?**

   Respondent #1: Consultant Neuroradiologist, 4 years

Respondent #2: Consultant Radiologist, 11 years

Respondent #3: Consultant Radiologist, 4 years

Respondent #4: Consultant Radiologist, (did not state years of experience)

Respondent #5: Cardiothoracic Radiologist, 19 years

Respondent #6: Forensic Pathologist, 11 years

Respondent #7: Clinical Director, (did not state years of experience)

Respondent #8: Clinical Lead (Electronic Patient Records), 7 years

Respondent #9: Doctor, 23 years

Respondent #10: Professor of Radiology, 6 years; (previously: radiologist, 23 years)

Respondent #11: Clinical Scientist, 20 years

Respondent #12: Clinical Research Fellow, 3.5 years

Respondent #13: Clinical Research Fellow, 2 years

Respondent #14: Specialist Trainee in Radiology, (did not state years of experience)

Respondent #15: PhD Student, 1.5 years

2. **Which country do you work in?**

   United Kingdom - 15 (100%) (5 stated Scotland (33%), 10 stated United Kingdom (67%))

3. **Do any clinical decision support applications at your institution continually improve? If yes, please describe briefly the system(s) you are aware of.**

Yes - 1 (7%)

No - 9 (60%)

Not Sure - 5 (33%)

Additional comment by the Yes respondent: "We use dragon voice recognition software. It supposedly continually learns from the reports being dictated. It does not do this well - in fact I would suggest the way it has been set up means it does not do it at all."

Additional comment by a Not Sure respondent: "Our voice activation reporting software learns continually but is not a clinical decision support application"

4. **Please indicate your level of agreement with each statement:**

a. I trust the software vendor to validate any updates.

Strongly Disagree - 1 (7%)

Disagree - 1 (7%)

Undecided - 3 (20%)

Agree - 9 (60%)

Strongly Agree - 1 (7%)

b. I must be able to compare the performance of a new system with the previous version.

Strongly Disagree - 0 (0%)

Disagree - 1 (7%)

Undecided - 5 (33%)

Agree - 7 (46%)

Strongly Agree - 2 (13%)

c. I must be notified when my system updates.

Strongly Disagree - 0 (0%)

Disagree - 2 (13%)

Undecided - 3 (20%)

Agree - 4 (27%)

Strongly Agree - 6 (40%)

d. I must be able to undo a system update if I deem it to have made the system worse.

Strongly Disagree - 0 (0%)

Disagree - 2 (13%)

Undecided - 0 (0%)

Agree - 9 (60%)

Strongly Agree - 4 (27%)

e. I am willing to record discrepancies as I use a system to help improve it.

Strongly Disagree - 0 (0%)

Disagree - 1 (7%)

Undecided - 1 (7%)

Agree - 9 (60%)

Strongly Agree - 4 (27%)

f. I want systems I use to continually improve.

Strongly Disagree - 1 (7%)

Disagree - 0 (0%)

Undecided - 0 (0%)

Agree - 8 (53%)

Strongly Agree - 6 (40%)

5. **How regularly would you like your system to update? (i.e. to receive improvements).**

   Daily - 0 (0%)

   Weekly - 5 (33%)

   Monthly - 5 (33%)

   Quarterly - 4 (27%)

   Annual - 1 (7%)

   Never - 0 (0%)

6. **From where would you like contributions to the system improvements to come from? (Check all that apply).**

   My Institution - 15 (100%)

   Large Regional Institutions - 15 (100%)

   All Regional Institutions - 11 (73%)

   Large National Institutions - 12 (80%)

   All National Institutions - 8 (53%)

   Renowned International Institutions - 9 (60%)

   All Institutions Worldwide - 7 (47%)

   I do not want any system improvements - 0 (0%)

7. **Whose input would you like to contribute to system improvements?**

   Only mine - 0 (0%)

All users above a particular grade - 0 (0%)

All users who have completed an accreditation process - 5 (33%)

All users of the system - 10 (67%)

8. **Do you have any final comments, ideas, or concerns about anything in this survey?**

   Response #1 - "For me much would depend on the way the software worked. In terms of global reach, this should be tempered by regional variations in risk and healthcare provision, but it should be possible to identify relevant data to inform particular 'improvements'. Also, one person's improvement is another person's problem. I would like to see improvements assessed in terms of a measurable output, which could be an audited comparison of diagnostic performance against another standard, or might be an ongoing survey of ease of use."

   Response #2 - "Channels should be adapted where users can give feedback and suggestions which can be reviewed by an expert group and then prioritised as a change. Change and improvements would need dedicated testing and validation which would need to be funded as part of the existing license. Users should get feedback with regards changes."

A.2 Survey Analysis and Discussion

From Q1 I observe a wide range of professions and levels of experience within the respondents, however as Q2 reveals, despite my best efforts to reach an international community, all respondents were from the UK with many being

from Scotland. This limits the applicability of my results to locations outside of the UK.

Q3 revealed that most (60%) of respondents were not aware of any continually improving clinical decision support applications at their institution, but 33% were not sure. Both additional comments refer to voice recognition software, although one of them says that the software is not learning. Voice recognition software pools the data it learns on to enhance development of a suitable model. This is possible because voice data are not as sensitive as medical data and can be sent off-site. This is different to federated learning where data cannot be sent off-site.

For Q4 I compute an average score for each question for better analysis. This score is calculated as

$$\text{Average Score} = \frac{0 * \text{SD} + 0.25 * \text{D} + 0.5 * \text{U} + 0.75 * \text{A} + 1 * \text{SA}}{n} \tag{A.1}$$

with SD, D, U, A, SA referring to the number of respondents for each answer category: Strongly Disagree, Disagree, Undecided, Agree, and Strongly Agree respectively; and n being the number of respondents (15). This score will be between 0 and 1 with 0 representing complete strong disagreement, 1 being full strong agreement, and 0.5 being an undecided or balanced opinion response.

Using Equation A.1 I get the following scores:

4a. 0.63 (weak agreement with "I trust the software vendor to validate any updates").

4b. 0.67 (weak agreement with "I must be able to compare the performance of a new system with the previous version").

4c. 0.73 (general agreement with "I must be notified when my system updates").

4d. 0.75 (general agreement with "I must be able to undo a system update if I deem it to have made the system worse").

4e. 0.77 (general agreement with "I am willing to record discrepancies as I use a system to help improve it").

4f. 0.80 (strong agreement with "I want systems I use to continually improve.")

As most responses were either in agreement or undecided, I highlight the respondents who disagreed for each question:

4a. Strong Disagreement by Respondent 7 (Clinical Director), and Disagreement by Respondent 8 (Clinical Lead).

4b. Disagreement by Respondent 10 (Professor of Radiology).

4c. Disagreement by Respondent 10 (Professor of Radiology) and Respondent 2 (Consultant Radiologist).

4d. Disagreement by Respondent 10 (Professor of Radiology) and Respondent 12 (Clinical Research Fellow).

4e. Disagreement by Respondent 11 (Clinical Scientist)

4f. Strong Disagreement by Respondent 11 (Clinical Scientist)

I note that two senior roles (Clinical Director and Clinical Lead) do not believe the software vendor will validate updates correctly. Most other roles did believe in this. The Professor of Radiology disagrees with 4b to 4d, indicating

that they did not desire to compare the performance of system updates, be notified when an update happens, or undo updates. This could be because they trust the software vendor to get it right, or they could be too busy to wish to be notified of these items or have to act on them. The Clinical Research Fellow also disagreed with being able to undo system updates. Only one of the three consultant radiologists did not want to be notified when the system updates.

The Clinical Scientist is the only respondent to disagree with the last two statements, and they disagreed strongly with the final one. While it is promising to see most respondents agreeable to these systems and willing to donate their time to recording discrepancies to help the system improve, the Clinical Scientist did not want to help the system learn and is strongly against working with a system that continually improves. This may be due to the job of a scientist where repeatability is of critical importance. Working with a system whose output is constantly in flux is not compatible with this goal. Alternatively the Clinical Scientist may not feel medically qualified to provide corrections for the system.

Q5 asked how regularly the respondent would like their system to update. Despite the Strong Disagreement by the Clinical Scientist in the previous question, they and everyone else were open to updates: no respondent selected Never. The responses are largely divided between Weekly, Monthly, and Quarterly. As federated learning requires frequent updates to keep the models converged, Weekly would be the best option for a deployed system.

In Q6 every respondent desired for their own institutions and large regional institutions to be included in the training. This could be due to the similarities between these institutions, so the training data is less diverse and so catalyses a better model. There is a drop of 27% when asked about All Regional In-

stitutions, indicating that small regional institutions aren't as desirable in the federation, possibly because their data are of lower quality than larger institutions. A similar drop to 80% is seen for Large National Institutions, again possibly due to differences at the regional and national level. This is an effect seen in the UK where different nations may have different protocols for data collection. Around half of respondents wanted All National Institutions. 60% wanted Renowned International Institutions. This could be because of the high standards at these institutions and a desire to learn from them for one's own institution. Support for using all institutions worldwide was under half (47%). This is a split opinion with some respondents perhaps believing that using more data would mean better local performance, while others believe more diverse data would harm the local performance.

Finally, in Q7, most respondents were willing to let all users of the system contribute to learning (67%) while the remainder desired an accreditation process (33%). No respondents selected the Users Above a Particular Grade option, indicating that grade is not as relevant as specific accreditation. The openness to allowing all respondents to contribute to system learning could be due to the high level of training respondents have before they enter a hospital environment to use a machine like this, and thus mistakes will be rare and damage to the learning because of these mistakes will be low and outweighed by more positive responses. Someone lacking confidence is also unlikely to mark discrepancies between their work and the model's output, so this reduces the risk of poor training data further.

At the end of the survey two further comments were made. Response #1 mentioned the variation between regions and healthcare provisions, which is a problem I am aware of. They also mention how improvements should be measurable against some audited task. However, there is no way to know

if the performance against this task is applicable to every institution in the federation. An audited task for each institution would be a suitable alternative.

Response #2 discusses modifying the importance of the training data, such that some data are more important and prioritised above other data. The response suggests an expert group can do this prioritisation. This comment is useful, as some training data is more valuable to a model that others (e.g. edge cases or difficult cases), but as processing is cheap (i.e. happens in the background), I have no concerns about using all of the data present and not prioritising changes. Also, what one institution decides is important, may not be what another institution declares important, so there is an added complication if this was to go ahead.

A.3 Additional Discussion Topics

This section details the common questions I have received while presenting about federated learning at technical talks. I detail my response to each question.

1. **How much of a footprint in hospitals will this take? Does it need a server room, or can it run in the background on the tasks that come in? How about for the network usage?**

   I hope that the final system will train silently in the background of an installed system at a hospital, such as on a CT scanner. This is in-line with other deployed or soon-to-be-deployed AI systems in hospitals. The training may need to operate on a low number of resources, and as such may be slow, but this is acceptable as we run the federated system over a long period of time (months, years). Data pre-processing in particular may slow down training and require a substantial use of resources. Hence

this should be kept to a minimum. A federated learning system should, as much as possible, be able to work on raw input data, or data that has undergone a low amount of pre-processing or utilises only quick pre-processing methods.

Network usage will be low. As we will only be sending model parameters (megabytes worth of data) and this will not be continuous but rather infrequent, in the region of once a week or month, and certainly no more often than once a day. This leads to network usage being insignificant. Even at the central institution's end, where a large number (tens or hundreds or potentially more) could arrive and require processing, this is not expected to be difficult, especially if time is not a critical factor in this situation.

2. **Will patients need to consent to have a model trained on their data sent off-site and used within a federated system?**

   This is a grey area that it beyond a technical exercise. In the UK it isn't clear if patients need to consent to having their data used by an AI that is an integral part of a piece of hardware. Having a model learned from their data transmitted around the world is a different category of problem and consent may be required in this case. When working with a federated system that could span many countries and continents, guidelines and laws surrounding this novel technology become increasingly vague. Further guidance must be sought from legal professionals before a federated learning system can be deployed in practice.

3. **How can hospital staff, especially those without technical knowledge on the workings of the system in question, know that no sensitive data is being sent off-site? How can they trust the**

system?

It would be possible to describe the data being sent off-site to the clinical staff before it is allowed to leave, such that is can be reviewed to ensure no sensitive information is leaving. This is possible if the dataset is small and sent off-site infrequently. Exactly how this is done is a topic for future research as unfortunately the network's parameters are themselves a series of incoherent numbers and arrays that are not human readable. Finding ways to display this information graphically or in some way that can be understood will help. Other than these factors, it may be down to an issue of trust with the clinical staff and the AI.

4. **Data is constantly being pulled in and sent away using the system. Is there some way to be certain of the quality of the incoming or outgoing data?**

   The models being sent away should be validated to ensure they aren't corrupt or haven't trained on incorrect data (either data that is corrupt, or incorrectly labelled, or simply irrelevant – for example, from a different system). Likewise, models being sent into a hospital should be validated, either at the central institution or at the hospital, so that its ability shows no significant failures. A significant failure in this case is defined as when a human could notice a mistake in the output of the AI model that was not there previously. Validating a model could consist of running test cases through it with results compared to the previous cycle of the model to ensure no major (negative) differences have taken place.

5. **Will the system be compatible with the current software used within hospitals around the world and is it future-proof, such that it can always be used – at least until a superior technology**

renders it obsolete? Would it be easy to replace in this case?

An AI system is a special case. It naturally forgets old cases as new cases come in, leading to adaption over time. To assist in this, data should not be retained for training/evaluation for a long period of time. It is assumed that improvements in hardware will lead to improvements in the data quality. If data quality was to suddenly change, the model could perform poorly on the new data for a while until it learns how to accurately process this change. This is especially true if no other institution within the federation is using this style of data. In this case, it may be beneficial detecting the drop in performance and taking this hospital out of the federation to prevent it affecting the performance at other institutions. The model at this hospital can train locally until such a time that performance has returned, and then it can re-join the federation. When a new federated model is returned to that institution, its model parameters could be shifted towards its previous iteration to prevent issues around needing to relearn the new style of data again.

6. **Training the model may require data to be kept for an extended period of time for repeated training (appearing in multiple epochs or across multiple cycles) or as an example to use for future reference (i.e. a test sample). How long can data be retained for?**

   The amount of time we can store data for is bounded by the data laws in the regions we are operating the federation in, and by storage allowances for the data, but where possible data should be kept until the point where it is no longer useful. Exactly when this is is an area for future research.

7. **How will clinical staff know the system is getting better with**

each update? What if the system becomes better than they are, such that they claim the system is getting worse because it is making claims that increasingly diverge from their own opinion, even if the system is more accurate?

Model validation, as discussed in point 4, should allow clinical staff to know a model is not getting worse. If the test performance improves, it is likely to be the case that the model has improved, especially if the test cases represent the full range of cases the algorithm will see.

In response to the second part of this question, this is moving towards a new question of trust. How can I trust an AI model to make the right decisions when its decisions differ from our own and yet are believed to be better than our own?

Part of the answer may be blind trust. This is like how we trust a calculator to provide the correct answer to some complicated sum that would take a human some time to work out. AI is more complicated than a mathematical calculation, although it can be seen as a very advanced calculator and treated as one. I know and accept that a regular calculator is faster than us and makes fewer mistakes. Could I also accept the same from an AI?

The other part of the answer goes down the path of explainability. An AI model should be able to explain why it came to a decision on a particular task in a way that a human can understand and can even learn from. This way, the AI is seen as a tool to aid in the understanding of a task rather than a mysterious black box system that "just gives answers". There is active research into AI explainability [2, 77, 114, 129].

Bibliography

[1] Jill M Abrigo, Daniel M Fountain, James M Provenzale, Eric K Law, Joey SW Kwong, Michael G Hart, and Wilson Wai San Tam. Magnetic resonance perfusion for differentiating low-grade from high-grade gliomas at first presentation. *Cochrane Database of Systematic Reviews*, (1), 2018.

[2] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI). *IEEE Access*, 6:52138–52160, 2018.

[3] Robert J Adams, Kimford J Meador, Kapil D Sethi, James C Grotta, and David S Thomson. Graded neurologic scale for use in acute hemispheric stroke treatment protocols. *Stroke*, 18(3):665–669, 1987.

[4] Malik Agyemang. Local sparsity coefficient-based mining of outliers. 2002.

[5] Gregory W Albers, Louis R Caplan, J Donald Easton, Pierre B Fayad, JP Mohr, Jeffrey L Saver, and David G Sherman. Transient ischemic attack — proposal for a new definition, 2002.

[6] Syed Mumtaz Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1):131–142, 1966.

[7] Naomi S Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185, 1992.

[8] Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853, 2005.

[9] Peter Argentiero, Roland Chin, and Paul Beaudet. An automated approach to the design of decision tree classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1):51–57, 1982.

[10] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. *Advances in neural information processing systems*, pages 41–48, 2007.

[11] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

[12] Sushanth Aroor, Rajpreet Singh, and Larry B Goldstein. BE-FAST (Balance, Eyes, Face, Arm, Speech, Time) Reducing the Proportion of Strokes Missed Using the FAST Mnemonic. *Stroke*, 48(2):479–481, 2017.

[13] Eirini Arvaniti, Kim S Fricker, Michael Moret, Niels Rupp, Thomas Hermanns, Christian Fankhauser, Norbert Wey, Peter J Wild, Jan H Rueschoff, and Manfred Claassen. Automated gleason grading of prostate cancer tissue microarrays via deep learning. *Scientific reports*, 8, 2018.

[14] National Stroke Association. What is TIA, 2018. https://www.stroke.org/understand-stroke/what-is-stroke/what-is-tia/, visited 2019-03-22.

[15] Stroke Association. Treatments, 2019. https://www.stroke.org.uk/what-is-stroke/diagnosis-to-discharge/treatment, visited 2018-02-09.

[16] RI Aviv, J Mandelcorn, S Chakraborty, D Gladstone, S Malham, G Tomlinson, AJ Fox, and S Symons. Alberta Stroke Program Early CT Scoring of CT perfusion in early stroke visualization and assessment. *American Journal of Neuroradiology*, 28(10):1975–1980, 2007.

[17] B, McMahan and D, Ramage. Federated learning: Collaborative machine learning without centralized training data, 2017. https://ai.googleblog.com/2017/04/federated-learning-collaborative.html, 2018-02-09.

[18] John Bamford, P Sandercock, Martin Dennis, C Warlow, and J Burn. Classification and natural history of clinically identifiable subtypes of cerebral infarction. *The Lancet*, 337(8756):1521–1526, 1991.

[19] Jamie L Banks and Charles A Marotta. Outcomes validity and reliability of the modified rankin scale: implications for stroke clinical trials: a literature review and synthesis. *Stroke*, 38(3):1091–1096, 2007.

[20] Iñigo Barandiaran. The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal. Mach. Intell*, 20(8), 1998.

[21] Daniel Barbará, Yi Li, Julia Couto, Jia-Ling Lin, and Sushil Jajodia. Bootstrapping a data mining intrusion detection system. In *Proceedings of the 2003 ACM symposium on Applied computing*, pages 421–425. ACM, 2003.

[22] Philip A Barber, Andrew M Demchuk, Jinjin Zhang, Alastair M Buchan, ASPECTS Study Group, et al. Validity and reliability of a quantitative computed tomography score in predicting outcome of hyperacute stroke before thrombolytic therapy. *The Lancet*, 355(9216):1670–1674, 2000.

[23] V Barnett and T Lewis. Outliers in statistical data. *New York*, 1994.

[24] BBC. 10 charts that show why the NHS is in trouble, 2018. https://www.bbc.co.uk/news/health-42572110, visited 2019-07-06.

[25] BBC. NHS vacancies a 'national emergency', 2018. https://www.bbc.co.uk/news/health-45485814, visited 2019-07-06.

[26] Katherine Berg, Sharon Wood-Dauphine, JI Williams, and David Gayton. Measuring balance in the elderly: preliminary development of an instrument. *Physiotherapy Canada*, 41(6):304–311, 1989.

[27] Katherine O Berg, Sharon L Wood-Dauphinee, J Ivan Williams, and Brian Maki. Measuring balance in the elderly: validation of an instrument. *Canadian journal of public health / Revue canadienne de sante publique*, 83:S7–11, 1992.

[28] K Berger, B Weltermann, P Kolominsky-Rabas, S Meves, P Heuschmann, J Böhner, B Neundörfer, HW Hense, and T Büttner. The reliability of stroke scales. The german version of NIHSS, ESS and Rankin scales. *Fortschritte der Neurologie-Psychiatrie*, 67(2):81–93, 1999.

[29] Dimitris Bertsimas and Jack Dunn. Optimal classification trees. *Machine Learning*, 106(7):1039–1082, 2017.

[30] Anil Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Calcutta Math. Soc.*, 35:99–109, 1943.

[31] Dan Bogdanov, Liina Kamm, Sven Laur, Pille Pruulmann-Vengerfeldt, Riivo Talviste, and Jan Willemson. Privacy-preserving statistical data analysis on federated databases. In *Annual Privacy Forum*, pages 30–55. Springer, 2014.

[32] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konecny, Stefano Mazzocchi, H Brendan McMahan, et al. Towards federated learning at scale: System design. *arXiv preprint arXiv:1902.01046*, 2019.

[33] Andrew P Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[34] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[35] Leo Breiman. *Classification and regression trees*. Routledge, 2017.

[36] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Optics-of: Identifying local outliers. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 262–270. Springer, 1999.

[37] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. LOF: identifying density-based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM, 2000.

[38] Theodora S Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International journal of medical informatics*, 112:59–67, 2018.

[39] M Broniatowski and Samantha Leorato. An estimation method for the Neyman chi-square divergence with application to test of hypotheses. *Journal of multivariate analysis*, 97(6):1409–1436, 2006.

[40] Thomas G Brott and HP Adams. National Institutes of Health Stroke Scale (NIHSS). *PROQOLID Patient-Reported Outcome and Quality Of Life Instruments Database*, 6, 1989.

[41] Askiel Bruno, Neel Shah, Chen Lin, Brian Close, David C Hess, Kristin Davis, Vanessa Baute, Jeffrey A Switzer, Jennifer L Waller, and Fenwick T Nichols. Improving modified Rankin Scale assessment with a simplified questionnaire. *Stroke*, 41(5):1048–1050, 2010.

[42] Cheryl D Bushnell, Dean CC Johnston, and Larry B Goldstein. Retrospective assessment of initial stroke severity: comparison of the NIH Stroke Scale and the Canadian Neurological Scale. *Stroke*, 32(3):656–660, 2001.

[43] Lianfang Cai, Nina F Thornhill, Stefanie Kuenzel, and Bikash C Pal. Real-time detection of power system disturbances based on k-nearest neighbor analysis. *IEEE Access*, 5:5631–5639, 2017.

[44] Marco Aurélio Gralha de Caneda, Jefferson Gomes Fernandes, Andrea Garcia de Almeida, and Fabiana Eloisa Mugnol. Reliability of neurological assessment scales in patients with stroke. *Arquivos de neuro-psiquiatria*, 64(3A):690–697, 2006.

[45] Douglas Carroll. A quantitative test of upper extremity function. *Journal of Chronic Diseases*, 18(5):479–491, 1965.

[46] R Casey and George Nagy. Decision tree design using a probabilistic model (corresp.). *IEEE Transactions on Information Theory*, 30(1):93–99, 1984.

[47] The Internet Stroke Center. Stroke Statistics, 2019. http://www.strokecenter.org/patients/about-stroke/stroke-statistics/, visited 2019-03-22.

[48] Shengyun Chen, Haixin Sun, Yanni Lei, Ding Gao, Yan Wang, Yilong Wang, Yong Zhou, Anxin Wang, Wenzhi Wang, and Xingquan Zhao. Validation of the Los Angeles pre-hospital stroke screen (LAPSS) in a Chinese urban emergency medical service population. *PloS one*, 8(8):e70742, 2013.

[49] Wo-Ruo Chen, Yong-Huan Yun, Ming Wen, Hong-Mei Lu, Zhi-Min Zhang, and Yi-Zeng Liang. Representative subset selection and outlier detection via isolation forest. *Analytical Methods*, 8(39):7225–7231, 2016.

[50] François Chollet et al. Keras, 2015. https://github.com/keras-team/keras, visited 2018-02-09.

[51] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[52] C Collin, DT Wade, S Davies, and V Horne. The Barthel ADL Index: a reliability study. *International disability studies*, 10(2):61–63, 1988.

[53] Scott D Constable, Yuzhe Tang, Shuang Wang, Xiaoqian Jiang, and Steve Chapin. Privacy-preserving GWAS analysis on federated genomic datasets. In *BMC medical informatics and decision making*, volume 15, page S2. 2015.

[54] Intersoft Consulting. General Data Protection Regulation (GDPR), 2019. https://gdpr-info.eu/, visited 2019-05-31.

[55] R Cote, RN Battista, C Wolfson, J Boucher, J Adam, and V Hachinski. The canadian neurological scale: validation and reliability assessment. *Neurology*, 39(5):638–638, 1989.

[56] Robert Côté, Vladimir C Hachinski, Bette L Shurvell, John W Norris, and Christina Wolfson. The canadian neurological scale: a preliminary study in acute stroke. *Stroke*, 17(4):731–737, 1986.

[57] Shelagh B Coutts, Andrew M Demchuk, Philip A Barber, William Y Hu, Jessica E Simon, Alastair M Buchan, and Michael D Hill. Interobserver variation of ASPECTS in real time. *Stroke*, 35(5):e103–e105, 2004.

[58] Thomas M Cover, Peter E Hart, et al. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[59] A. Criminisi and J. Shotton. *Decision Forests for Computer Vision and Medical Image Analysis*. Springer, 2013. Pages 59 – 75.

[60] A Criminisi, J Shotton, and E Konukoglu. Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. *Microsoft Research technical report 114*, 2011.

[61] Antonio Criminisi, Jamie Shotton, Ender Konukoglu, et al. Decision forests: A unified framework for classification, regression, density esti-

mation, manifold learning and semi-supervised learning. *Foundations and Trends in Computer Graphics and Vision*, 7(2–3):81–227, 2012.

[62] JL Crow, NB Lincoln, FM Nouri, and W de Weerdt. The effectiveness of EMG biofeedback in the treatment of arm function after stroke. *International disability studies*, 11(4):155–160, 1989.

[63] Imre Csiszár. Eine informationstheoretische ungleichung und ihre anwendung auf beweis der ergodizitaet von markoffschen ketten. *Magyer Tud. Akad. Mat. Kutato Int. Koezl.*, 8:85–108, 1964.

[64] Neşat Çullu, Serdar Kalemci, Ömer Karakaş, İrfan Eser, Funda Yalçın, Fatıma Nurefşan Boyacı, and Ekrem Karakaş. Efficacy of CT in diagnosis of transudates and exudates in patients with pleural effusion. *Diagnostic and Interventional Radiology*, 20(2):116, 2014.

[65] Mohammad A Dabbah, Sean Murphy, Hippolyte Pello, Romain Courbon, Erin Beveridge, Stewart Wiseman, Daniel Wyeth, and Ian Poole. Detection and location of 127 anatomical landmarks in diverse CT datasets. In *Medical Imaging 2014: Image Processing*, volume 9034, page 903415. International Society for Optics and Photonics, 2014.

[66] Business News Daily. Artificial intelligence will change healthcare as we know it, 2019. https://www.businessnewsdaily.com/15096-artificial-intelligence-in-healthcare.html, visited 2019-07-06.

[67] Matt Daykin and Ian Poole. Soft federated learning, 2019. Patent in process of being granted.

[68] WJG De Weerdt and MA Harrison. Measuring recovery of arm-hand function in stroke patients: a comparison of the Brunnstrom-Fugl-Meyer

test and the Action Research Arm test. *Physiotherapy Canada*, 37(2):65–70, 1985.

[69] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. Large scale distributed deep networks. In *Advances in neural information processing systems*, pages 1223–1231. 2012.

[70] Timo M Deist, Arthur Jochems, Johan van Soest, Georgi Nalbantov, Cary Oberije, Seán Walsh, Michael Eble, Paul Bulens, Philippe Coucke, Wim Dries, et al. Infrastructure and distributed learning methodology for privacy-preserving multi-centric rapid learning health care: euroCAT. *Clinical and translational radiation oncology*, 4:24–31, 2017.

[71] Andre Dekker, Shalini Vinod, Lois Holloway, Cary Oberije, Armia George, Gary Goozee, Geoff P Delaney, Philippe Lambin, and David Thwaites. Rapid learning in practice: A lung cancer survival decision support system in routine patient care data. *Radiotherapy and Oncology*, 113(1):47–53, 2014.

[72] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE, 2009.

[73] Adrien Depeursinge, Alejandro Vargas, Alexandra Platon, Antoine Geissbuhler, Pierre-Alexandre Poletti, and Henning Müller. Building a reference multimedia database for interstitial lung diseases. *Computerized medical imaging and graphics*, 36(3):227–238, 2012.

[74] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[75] Edwin Diday and Jean-Vincent Moreau. Learning hierarchical clustering from examples—application to the adaptive construction of dissimilarity indices. *Pattern Recognition Letters*, 2(6):365–378, 1984.

[76] Youcef Djenouri, Asma Belhadi, Jerry Chun-Wei Lin, and Alberto Cano. Adapted k nearest neighbors for detecting anomalies on spatio-temporal traffic flow. *IEEE Access*, 2019.

[77] Derek Doran, Sarah Schulz, and Tarek R Besold. What does explainable AI really mean? A new conceptualization of perspectives. *arXiv preprint arXiv:1710.00794*, 2017.

[78] Dr Jeremy Jones Dr. Faniel J Bell. Larmor frequency, 2019. https://radiopaedia.org/articles/larmor-frequency, visited 2018-02-09.

[79] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[80] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[81] Imanuel Dzialowski, Michael D Hill, Shelagh B Coutts, Andrew M Demchuk, David M Kent, Olaf Wunderlich, and Ruüdiger von Kummer. Extent of early ischemic changes on computed tomography (CT) before thrombolysis: prognostic value of the Alberta Stroke Program Early CT Score in ECASS II. *Stroke*, 37(4):973–978, 2006.

[82] Hanadi El Achi, Tatiana Belousova, Lei Chen, Amer Wahed, Iris Wang, Zhihong Hu, Zeyad Kanaan, Adan Rios, and Andy ND Nguyen. Auto-

mated diagnosis of lymphoma with digital pathology images using deep learning. *Annals of Clinical & Laboratory Science*, 49(2):153–160, 2019.

[83] Salwa El Tawil and Keith W Muir. Thrombolysis and thrombectomy for acute ischaemic stroke. *Clinical Medicine*, 17(2):161–165, 2017.

[84] Levent Ertöz, Michael Steinbach, and Vipin Kumar. Finding topics in collections of documents: A shared nearest neighbor approach. In *Clustering and Information Retrieval*, pages 83–103. Springer, 2004.

[85] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231. 1996.

[86] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115, 2017.

[87] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. ACM, 2004.

[88] Avital Fast and Dorith Goldsher. Navigating the adult spine: Bridging clinical practice and neuroradiology. In *Navigating the Adult Spine: Bridging Clinical Practice and Neuroradiology*, page 17. Demos Medical Publishing, 2006.

[89] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, et al. 3D Slicer as an

image computing platform for the Quantitative Imaging Network. *Magnetic resonance imaging*, 30(9):1323–1341, 2012.

[90] Valery L Feigin, Mohammad H Forouzanfar, Rita Krishnamurthi, George A Mensah, Myles Connor, Derrick A Bennett, Andrew E Moran, Ralph L Sacco, Laurie Anderson, Thomas Truelsen, et al. Global and regional burden of stroke during 1990–2010: findings from the global burden of disease study 2010. *The Lancet*, 383(9913):245–255, 2014.

[91] R Fosbinder and D Orth. Essentials of radiologic science. In *Essentials of Radiologic Science*, page 263. Lippincott Williams & Wilkins, 2011.

[92] Python Software Foundation. Python 2.7, 2010. http://devdocs.io/python 2.7/, visited 2018-02-09.

[93] Python Software Foundation. Python 3.6, 2017. http://devdocs.io/python 3.6/, 2018-02-09.

[94] Wikimedia Foundation. Euclidean distance, 2020. https://en.wikipedia.org/wiki/Euclidean_distance, visited 2020-04-07.

[95] Wikimedia Foundation. Hellinger distance, 2020. https://en.wikipedia.org/wiki/Hellinger_distance, visited 2020-04-09.

[96] Wikimedia Foundation. Kullback–leibler divergence, 2020. https://en.wikipedia.org/wiki/information_gain, visited 2020-04-05.

[97] Wikimedia Foundation. Machine learning, 2020. https://en.wikipedia.org/wiki/Machine_learning, visited 2020-04-12.

[98] Wikimedia Foundation. Receiver operating characteristic, 2020. https://en.wikipedia.org/wiki/Receiver_operating_characteristic, visited 2020-04-07.

[99] Wikimedia Foundation. Total variation distance of probability measures, 2020. https://en.wikipedia.org/wiki/Total_variation_distance_of_probability_measures, visited 2020-04-09.

[100] KC Fu. *Sequential methods in pattern recognition and machine learning*, volume 52. Academic press, 1968.

[101] Jennifer E Fugate and Alejandro A Rabinstein. Absolute and relative contraindications to IV rt-PA for acute ischemic stroke. *The Neurohospitalist*, 5(3):110–121, 2015.

[102] The King's Fund. The NHS budget and how it has changed, 2018. https://www.kingsfund.org.uk/projects/nhs-in-a-nutshell/nhs-budget, visited 2019-07-06.

[103] F Galton. Finger prints macmillan, 1892.

[104] Google. Google scholar, 2019. https://scholar.google.co.uk, visited 2019-05-30.

[105] Google. Tensorflow federated: Machine learning on decentralized data, 2019.

[106] Open Access Government. The future of artificial intelligence in health, 2019. https://www.openaccessgovernment.org/artificial-intelligence-in-health/67967/, visited 2019-07-06.

[107] Hayit Greenspan, Bram Van Ginneken, and Ronald M Summers. Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5):1153–1159, 2016.

[108] GE Gresham, TF Phillips, and ML Labi. ADL status in stroke: relative merits of three standard indexes. *Archives of Physical Medicine and Rehabilitation*, 61(8):355–358, 1980.

[109] James C Grotta, David Chiu, Mei Lu, Suresh Patel, Steven R Levine, Barbara C Tilley, Thomas G Brott, E Clarke Haley Jr, Patrick D Lyden, Rashmi Kothari, et al. Agreement and variability in the interpretation of early CT changes in stroke patients qualifying for intravenous rtPA therapy. *Stroke*, 30(8):1528–1533, 1999.

[110] Scandinavian Stroke Study Group et al. Multicenter trial of hemodilution in ischemic stroke: background and study protocol. *Stroke*, 16:885–890, 1985.

[111] Iris Grunwald and Wolfgang Reith. Non-traumatic neurological emergencies: imaging of cerebral ischemia. *European radiology*, 12(7):1632–1647, 2002.

[112] YX Gu, Qing Ren Wang, and Ching Y Suen. Application of a multilayer decision tree in computer recognition of chinese characters. *IEEE transactions on pattern analysis and machine intelligence*, (1):83–89, 1983.

[113] The Guardian. The robot will see you now: how AI could revolutionise NHS, 2018. https://www.theguardian.com/society/2018/jun/11/the-robot-will-see-you-now-how-ai-could-revolutionise-nhs, visited 2019-07-06.

[114] David Gunning. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web*, 2, 2017.

[115] Alexander Y Gur, Yair Lampl, Bella Gross, Vladimir Royter, Ludmila Shopin, and Natan M Bornstein. A new scale for assessing patients with vertebrobasilar stroke—the Israeli Vertebrobasilar Stroke Scale (IVBSS): Inter-rater reliability and concurrent validity. *Clinical neurology and neurosurgery*, 109(4):317–322, 2007.

[116] Vladimir C Hachinski, Linnette D Iliff, Elias Zilhka, George H Du Boulay, Victor L McAllister, John Marshall, Ralph W Ross Russell, and Lindsay Symon. Cerebral blood flow in dementia. *Archives of neurology*, 32(9):632–637, 1975.

[117] L Hantson, Willy De Weerdt, J De Keyser, HC Diener, C Franke, R Palm, M Van Orshoven, H Schoonderwalt, N De Klippel, and L Herroelen. The European Stroke Scale. *Stroke*, 25(11):2215–2219, 1994.

[118] Yoshiyuki Harada, Yoriyuki Yamagata, Osamu Mizuno, and Eun-Hye Choi. Log-based anomaly detection of CPS using a statistical method. In *2017 8th International Workshop on Empirical Software Engineering in Practice (IWESEP)*, pages 1–6. IEEE, 2017.

[119] Robert M Haralick. The table look-up rule. *Communications in Statistics-Theory and Methods*, 5(12):1163–1191, 1976.

[120] Sahand Hariri, Matias Carrasco Kind, and Robert J Brunner. Extended isolation forest. *arXiv preprint arXiv:1811.02141*, 2018.

[121] C Hartmann, Pramod Varshney, Kishan Mehrotra, and C Gerberich. Application of information theory to the construction of efficient decision trees. *IEEE Transactions on Information Theory*, 28(4):565–577, 1982.

[122] DM Hawkins. Identification of outliers, london: Chap-man & hall. *Identification of Outliers*, 1980.

[123] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. 2016.

[124] Zengyou He, Xiaofei Xu, and Shengchun Deng. Discovering cluster-based local outliers. *Pattern Recognition Letters*, 24(9-10):1641–1650, 2003.

[125] Ernst Hellinger. Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. *Journal für die reine und angewandte Mathematik*, 136:210–271, 1909.

[126] Michael D Hill, Alastair M Buchan, et al. Thrombolysis for acute ischemic stroke: results of the canadian alteplase for stroke effectiveness study. *Canadian Medical Association Journal*, 172(10):1307–1312, 2005.

[127] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[128] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

[129] Andreas Holzinger, Chris Biemann, Constantinos S Pattichis, and Douglas B Kell. What do we need to build explainable AI systems for the medical domain? *arXiv preprint arXiv:1712.09923*, 2017.

[130] George Howard and David C Goff. Population shifts and the future of stroke: forecasts of the future burden of stroke. *Annals of the New York Academy of Sciences*, 1268(1):14–20, 2012.

[131] Xuya Huang, Bharath Kumar Cheripelli, Suzanne M Lloyd, Dheeraj Kalladka, Fiona Catherine Moreton, Aslam Siddiqui, Ian Ford, and Keith W Muir. Alteplase versus tenecteplase for thrombolysis after ischaemic stroke (ATTEST): a phase 2, randomised, open-label, blinded endpoint study. *The Lancet Neurology*, 14(4):368–376, 2015.

[132] WE Hunt, JN Meagher, and RM Hess. Intracranial aneurysm. a nine-year study. *The Ohio State medical journal*, 62(11):1168, 1966.

[133] William E Hunt and Robert M Hess. Surgical risk as related to time of intervention in the repair of intracranial aneurysms. *Journal of neurosurgery*, 28(1):14–20, 1968.

[134] Rights Info. NHS Staff Shortages Could Double Without 'Radical Action', 2019. https://rightsinfo.org/nhs-staff-shortages-could-double-without-radical-action/, visited 2019-07-06.

[135] Intel. Federated learning for medical imaging, 2019. https://www.intel.ai/federated-learning-for-medical-imaging, 2018-02-09.

[136] Bryan Jennett and Michael Bond. Assessment of outcome after severe brain damage: a practical scale. *The Lancet*, 305(7905):480–484, 1975.

[137] Wenchao Jiang, Pinghao Li, Shuang Wang, Yuan Wu, Meng Xue, Lucila Ohno-Machado, and Xiaoqian Jiang. WebGLORE: a web service for Grid LOgistic REgression. *Bioinformatics*, 29(24):3238–3240, 2013.

[138] Wen Jin, Anthony KH Tung, Jiawei Han, and Wei Wang. Ranking outliers using symmetric neighborhood relationship. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 577–593. Springer, 2006.

[139] Arthur Jochems, Timo M Deist, Johan Van Soest, Michael Eble, Paul Bulens, Philippe Coucke, Wim Dries, Philippe Lambin, and Andre Dekker. Distributed learning: developing a predictive model based on data from multiple hospitals without data leaving the hospital–a real life proof of concept. *Radiotherapy and Oncology*, 121(3):459–467, 2016.

[140] Ian T Jolliffe. Principal component analysis and factor analysis. In *Principal component analysis*, pages 115–128. Springer, 1986.

[141] K. Afifi-Sabet, IT PRO. NHS to reward hospitals for replacing clinicians with AI, 2019. https://www.itpro.co.uk/digital-transformation/33796/nhs-to-reward-hospitals-for-replacing-clinicians-with-ai, visited 2019-06-13.

[142] Mary A Kalafut, David L Schriger, Jeffrey L Saver, and Sidney Starkman. Detection of early CT signs of >1/3 middle cerebral artery infarctions: interrater reliability and sensitivity of CT interpretation by physicians involved in acute stroke care. *Stroke*, 31(7):1667–1671, 2000.

[143] Liina Kamm, Dan Bogdanov, Sven Laur, and Jaak Vilo. A new way to protect privacy in large-scale genome-wide association studies. *Bioinformatics*, 29(7):886–893, 2013.

[144] Konstantinos Kamnitsas, Wenjia Bai, Enzo Ferrante, Steven McDonagh, Matthew Sinclair, Nick Pawlowski, Martin Rajchl, Matthew Lee, Bernhard Kainz, Daniel Rueckert, et al. Ensembles of multiple models and

architectures for robust brain tumour segmentation. In *International MICCAI Brainlesion Workshop*, pages 450–462. Springer, 2017.

[145] Michael Kamp, Linara Adilova, Joachim Sicking, Fabian Hüger, Peter Schlicht, Tim Wirtz, and Stefan Wrobel. Efficient decentralized deep learning by dynamic model averaging. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 393–409. Springer, 2018.

[146] Michael Kamp, Sebastian Bothe, Mario Boley, and Michael Mock. Communication-efficient distributed online learning with kernels. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 805–819. Springer, 2016.

[147] S Karkanis, George D Magoulas, and N Theofanous. Image recognition and neuronal networks: Intelligent systems for the improvement of imaging information. *Minimally Invasive Therapy & Allied Technologies*, 9(3-4):225–230, 2000.

[148] Ella Kazerooni and H Barry. *Cardiopulmonary Imaging*. Lippincott Williams & Wilkins, 2004.

[149] Chelsea S Kidwell, Sidney Starkman, Marc Eckstein, Kimberly Weems, and Jeffrey L Saver. Identifying stroke in the field: prospective validation of the Los Angeles Prehospital Stroke Screen (LAPSS). *Stroke*, 31(1):71–76, 2000.

[150] Y-J Kim, BFM Romeike, J Uszkoreit, and W Feiden. Automated nuclear segmentation in the determination of the Ki-67 labeling index in meningiomas. *Clinical neuropathology*, 25(2), 2006.

[151] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[152] Donald E Knuth. Optimum binary search trees. *Acta informatica*, 1(1):14–25, 1971.

[153] Chai Kobkitsuksakul, Oranan Tritanon, and Vichan Suraratdecha. Interobserver agreement between senior radiology resident, neuroradiology fellow, and experienced neuroradiologist in the rating of Alberta Stroke Program Early Computed Tomography Score (ASPECTS). *Diagnostic and Interventional Radiology*, 24(2):104, 2018.

[154] Teuvo Kohonen, MR Schroeder, and TS Huang. Self-oragnizing maps. *Inc., Secaucus, NJ*, 43(2), 2001.

[155] Andrey Kolmogorov. Sulla determinazione empirica di una lgge di distribuzione. *Inst. Ital. Attuari, Giorn.*, 4:83–91, 1933.

[156] Jakub Konečnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, 2016.

[157] Robert K Kosior, M Louis Lauzon, Nikolai Steffenhagen, Jayme C Kosior, Andrew Demchuk, and Richard Frayne. Atlas-based topographical scoring for magnetic resonance imaging of acute stroke. *Stroke*, 41(3):455–460, 2010.

[158] Rashmi Kothari, Kent Hall, Thomas Brott, and Joseph Broderick. Early stroke recognition: developing an out-of-hospital NIH Stroke Scale. *Academic Emergency Medicine*, 4(10):986–990, 1997.

[159] Rashmi U Kothari, Arthur Pancioli, Tiepu Liu, Thomas Brott, and Joseph Broderick. Cincinnati prehospital stroke scale: reproducibility and validity. *Annals of emergency medicine*, 33(4):373–378, 1999.

[160] V Kovalev, A Kalinovsky, and V Liauchuk. Deep learning in big image data: Histology image classification for breast cancer diagnosis. In *Big Data and Advanced Analytics, Proc. 2nd International Conference, BSUIR, Minsk*, pages 44–53. 2016.

[161] E. Kreyszig. *Advanced Engineering Mathematics*. Wiley, fourth edition, 1979. Equation 5.

[162] Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. Loop: local outlier probabilities. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1649–1652. ACM, 2009.

[163] Hans-Peter Kriegel, Arthur Zimek, et al. Angle-based outlier detection in high-dimensional data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 444–452. ACM, 2008.

[164] Solomon Kullback. Information theory and statistics. john riley and sons. *Inc. New York*, 1959.

[165] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[166] Abhishek Kumar and Hal Daume III. Learning task grouping and overlap in multi-task learning. *arXiv preprint arXiv:1206.6417*, 2012.

[167] Mara M Kunst and Pamela W Schaefer. Ischemic stroke. *Radiologic Clinics*, 49(1):1–26, 2011.

[168] Erwin Kuntz and Hans-Dieter Kuntz. *Hepatology, Principles and Practice: History, Morphology, Biochemistry, Diagnostics, Clinic, Therapy.* Springer Science & Business Media, 2006.

[169] The Telegraph L. Lillywhite. We face a worldwide shortage of medics. Relaxing visa caps for the NHS is just a sticking plaster, 2018. https://www.telegraph.co.uk/global-health/climate-and-people/face-worldwide-shortage-medics-relaxing-visa-caps-nhs-just-sticking/, visited 2019-06-13.

[170] GH Landeweerd, Teun Timmers, Edzard S Gelsema, M Bins, and MR Halie. Binary tree versus single level tree classification of white blood cells. *Pattern Recognition*, 16(6):571–577, 1983.

[171] Vincent Larrue, Ruüdiger von Kummer, Achim Muüller, and Erich Bluhmki. Risk factors for severe hemorrhagic transformation in ischemic stroke patients treated with recombinant tissue plasminogen activator: a secondary analysis of the European-Australasian Acute Stroke Study (ECASS II). *Stroke*, 32(2):438–441, 2001.

[172] Longin Jan Latecki, Aleksandar Lazarevic, and Dragoljub Pokrajac. Outlier detection with kernel density functions. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pages 61–75. Springer, 2007.

[173] Nada Lavrač. Machine learning for data mining in medicine. In *Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making*, pages 47–62. Springer, 1999.

[174] M Powell Lawton and Elaine M Brody. Assessment of older people: self-maintaining and instrumental activities of daily living. *The gerontologist*, 9(3_Part_1):179–186, 1969.

[175] Breiman Leo, Jerome H Friedman, Richard A Olshen, and Charles J Stone. Classification and regression trees. *Wadsworth International Group*, 1984.

[176] H. Lepor. Prostatic diseases. In *Prostatic Diseases*, page 83. Saunders, 1999.

[177] Friedrich Liese and Igor Vajda. On divergences and informations in statistics and information theory. *IEEE Transactions on Information Theory*, 52(10):4394–4412, 2006.

[178] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

[179] Aneta Lisowska, Erin Beveridge, Keith Muir, and Ian Poole. Thrombus Detection in CT Brain Scans using a Convolutional Neural Network. In *BIOIMAGING*, pages 24–33. 2017.

[180] Aneta Lisowska, Alison O'Neil, Vismantas Dilys, Matthew Daykin, Erin Beveridge, Keith Muir, Stephen Mclaughlin, and Ian Poole. Context-aware convolutional neural networks for stroke sign detection in non-contrast CT scans. In *Annual Conference on Medical Image Understanding and Analysis*, pages 494–505. Springer, 2017.

[181] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez.

A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[182] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *ICDM'08. Eighth IEEE International Conference on Data Mining*, pages 413–422. IEEE, 2008.

[183] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. On detecting clustered anomalies using sciforest. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 274–290. Springer, 2010.

[184] Sandy C Loewen and Brian A Anderson. Predictors of stroke outcome using objective measurement scales. *Stroke*, 21(1):78–81, 1990.

[185] University College London. Blog: Machine learning is transforming healthcare - but there's a long way to go, 2018. https://www.ucl.ac.uk/healthcare-engineering/news/2018/dec/blog-machine-learning-transforming-healthcare-theres-long-way-go, visited 2019-07-06.

[186] Chia-Lun Lu, Shuang Wang, Zhanglong Ji, Yuan Wu, Li Xiong, Xiaoqian Jiang, and Lucila Ohno-Machado. WebDISCO: a web service for distributed cox model learning without patient-level data sharing. *Journal of the American Medical Informatics Association*, 22(6):1212–1219, 2015.

[187] Ronald C Lyle. A performance test for assessment of upper limb function in physical rehabilitation treatment and research. *International journal of rehabilitation research*, 4(4):483–492, 1981.

[188] Gertrude Maatman. *High-Resolution Computed Tomography of the Paranasal Sinuses and Pharynx and Related Regions: Impact of CT identification on diagnosis and patient management*, volume 12. Springer Science & Business Media, 2012.

[189] Niall John James MacDougall. *Pathophysiology of post-stroke hyperglycaemia and brain arterial patency*. PhD thesis, University of Glasgow, 2013.

[190] NJJ MacDougall, F McVerry, X Huang, A Welch, R Fulton, and KW Muir. Post-stroke hyperglycaemia is associated with adverse evolution of acute ischaemic injury. In *Cerebrovascular Diseases*, volume 37, pages 267–267. Karger Publishers, 2014.

[191] Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *22nd International Conference on Data Engineering (ICDE'06)*, pages 24–24. IEEE, 2006.

[192] Dennis Mackin, Xenia Fave, Lifei Zhang, David Fried, Jinzhong Yang, Brian Taylor, Edgardo Rodriguez-Rivera, Cristina Dodge, A Kyle Jones, and Laurence Court. Measuring CT scanner variability of radiomics features. *Investigative radiology*, 50(11):757, 2015.

[193] George D Magoulas and Andriana Prentza. Machine learning in medical applications. In *Advanced Course on Artificial Intelligence*, pages 300–307. Springer, 1999.

[194] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. 1936.

[195] Florence I Mahoney and Dorothea W Barthel. Functional evaluation: the Barthel Index: a simple index of independence useful in scoring improvement in the rehabilitation of the chronically ill. *Maryland state medical journal*, 1965.

[196] Azeem Majeed. *Shortage of general practitioners in the NHS*. BMJ Publishing Group Ltd, 2017.

[197] Henry KF Mak, Kelvin KW Yau, Pek-Lan Khong, Alex SC Ching, Pui-Wai Cheng, Paul KM Au-Yeung, Peter KM Pang, Kenny CW Wong, and Bernard PL Chan. Hypodensity of >1/3 middle cerebral artery territory versus Alberta stroke programme early CT score (ASPECTS) comparison of two methods of quantitative evaluation of early CT changes in hyperacute ischemic stroke in the community setting. *Stroke*, 34(5):1194–1196, 2003.

[198] Ninan Mathew, Victor Rivera, John Meyer, Jonathan Charney, and Alexander Hartmann. Double-blind evaluation of glycerol therapy in acute cerebral infarction. *The Lancet*, 300(7791):1327–1329, 1972.

[199] mc.ai. What's new in deep learning research: Understanding federated learning, 2018. https://mc.ai/whats-new-in-deep-learning-research-understanding-federated-learning/, visited 2018-02-09.

[200] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629*, 2016.

[201] GE Mead, SC Lewis, JM Wardlaw, MS Dennis, and CP Warlow. How well does the Oxfordshire Community Stroke Project classification pre-

dict the site and size of the infarct on brain imaging? *Journal of Neurology, Neurosurgery & Psychiatry*, 68(5):558–562, 2000.

[202] Johns Hopkins Medicine. Surgical thrombectomy, 2019. https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/surgical-thrombectomy, visited 2018-02-09.

[203] Vasileios Megalooikonomou, James Ford, Li Shen, Fillia Makedon, and Andrew Saykin. Data mining in brain imaging. *Statistical Methods in Medical Research*, 9(4):359–394, 2000.

[204] William S Meisel and Demetrios A Michalopoulos. A partitioning algorithm with application in pattern classification and the optimization of decision trees. *IEEE Transactions on Computers*, 100(1):93–103, 1973.

[205] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.

[206] Brett C Meyer, Thomas M Hemmen, Christy M Jackson, and Patrick D Lyden. Modified national institutes of health stroke scale for use in stroke clinical trials: prospective reliability and validity. *Stroke*, 33(5):1261–1266, 2002.

[207] Brett C Meyer, Rema Raman, Marcus R Chacon, Matt Jensen, and Janet D Werner. Reliability of site-independent telemedicine when assessed by telemedicine-naive stroke practitioners. *Journal of Stroke and Cerebrovascular Diseases*, 17(4):181–186, 2008.

[208] Douglas M Minot, Benjamin R Kipp, Renee M Root, Reid G Meyer, Carol A Reynolds, Aziza Nassar, Michael R Henry, and Amy C Clayton. Automated cellular imaging system III for assessing HER2 status in breast cancer specimens: development of a standardized scoring method that correlates with FISH. *American journal of clinical pathology*, 132(1):133–138, 2009.

[209] Pekka K Mölsä, Leo Paljärvi, Juha O Rinne, Urpo K Rinne, and Erkki Säkö. Validity of clinical diagnosis in dementia: a prospective clinicopathological study. *Journal of Neurology, Neurosurgery & Psychiatry*, 48(11):1085–1090, 1985.

[210] John Moody and Christian J Darken. Fast learning in networks of locally-tuned processing units. *Neural computation*, 1(2):281–294, 1989.

[211] Todd K Moon. The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60, 1996.

[212] Tetsuzo Morimoto. Markov processes and the H-theorem. *Journal of the Physical Society of Japan*, 18(3):328–331, 1963.

[213] Emmanuel Müller, Matthias Schiffer, and Thomas Seidl. Statistical selection of relevant subspace projections for outlier ranking. In *2011 IEEE 27th international conference on data engineering*, pages 434–445. IEEE, 2011.

[214] National Institute of Neurological Disorders and Stroke rt-PA Stroke Study Group. Tissue plasminogen activator for acute ischemic stroke. *New England Journal of Medicine*, 333(24):1581–1588, 1995.

[215] Nancy J Newcommon, Teri L Green, Eryka Haley, Timothy Cooke, Michael D Hill, JTL Wilson, A Hareendran, T Baird, KW Muir, and

I Bone. Improving the assessment of outcomes in stroke: use of a structured interview to assign grades on the modified rankin scale. *Stroke*, 34(2):377–378, 2003.

[216] Sky News. Staff shortages the 'single biggest risk' facing NHS, 2017. https://news.sky.com/story/staff-shortages-the-single-biggest-risk-facing-nhs-11116975, visited 2019-07-06.

[217] NHS. NHS, 2019. https://www.nhs.uk/, visited 2019-06-13.

[218] NHS. Symptoms Stroke, 2019. https://www.nhs.uk/conditions/stroke/symptoms/, visited 2019-03-22.

[219] NHS England Specialised Services Clinical Reference Group for Neurosciences. Clinical commissioning policy: Mechanical thrombectomy for acute ischaemic stroke (all ages), 2019. https://www.england.nhs.uk/wp-content/uploads/2019/05/Mechanical-thrombectomy-for-acute-ischaemic-stroke-ERRATA-29-05-19.pdf, visited 2018-02-09.

[220] Frank Nielsen and Richard Nock. On the chi square and higher-order chi distances for approximating f-divergences. *IEEE Signal Processing Letters*, 21(1):10–13, 2013.

[221] MS Nikulin. Hellinger distance. hazewinkel, michiel, encyclopedia of mathematics. *Springer, Berlin. doi*, 10:1361684–1361686, 2001.

[222] Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2019.

[223] A Mohd Nor, C McAllister, SJ Louw, AG Dyker, M Davis, D Jenkinson, and GA Ford. Agreement between ambulance paramedic-and physician-recorded neurological signs with Face Arm Speech Test (FAST) in acute stroke patients. *Stroke*, 35(6):1355–1359, 2004.

[224] W O'Brien, Dennis Crimmins, W Donaldson, R Risti, TA Clarke, Scott Whyte, and Jonathan Sturm. FASTER (Face, Arm, Speech, Time, Emergency Response): experience of Central Coast Stroke Services implementation of a pre-hospital notification system for expedient management of acute stroke. *Journal of Clinical Neuroscience*, 19(2):241–245, 2012.

[225] United States Department of Health & Human Services Office for Civil Rights Headquarters. Health insurance portability and accountability act, 1996. https://www.hhs.gov/hipaa/index.html, visited 2019-05-31.

[226] OpenMined. Building safe artificial intelligence, 2019. https://www.openmined.org/, visited 2018-02-09.

[227] JM Orgogozo, R Capildeo, CN Anagnostou, O Juge, JJ Pere, JF Dartigues, TJ Steiner, A Yotis, and FC Rose. Development of a neurological score for the clinical evaluation of sylvian infarctions. *Presse medicale (Paris, France: 1983)*, 12(48):3039–3044, 1983.

[228] Sanjana Patrick, N Praveen Birur, Keerthi Gurushanth, A Shubhasini Raghavan, Shubha Gurudath, et al. Comparison of gray values of cone-beam computed tomography with hounsfield units of multislice computed tomography: An in vitro study. *Indian Journal of Dental Research*, 28(1):66, 2017.

[229] Heiko Paulheim and Robert Meusel. A decomposition of the outlier detection problem into a set of supervised learning problems. *Machine Learning*, 100(2-3):509–531, 2015.

[230] Harold J. Payne and William S. Meisel. An algorithm for constructing optimal binary decision trees. *IEEE Transactions on Computers*, (9):905–916, 1977.

[231] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[232] Kabir Peerbhay, Onisimo Mutanga, Romano Lottering, and Riyad Ismail. Mapping Solanum mauritianum plant invasions using WorldView-2 imagery and unsupervised random forests. *Remote Sensing of Environment*, 182:39–48, 2016.

[233] JH Warwick Pexman, Philip A Barber, Michael D Hill, Robert J Sevick, Andrew M Demchuk, Mark E Hudon, William Y Hu, and Alastair M Buchan. Use of the Alberta Stroke Program Early CT Score (ASPECTS) for assessing CT scans in patients with acute stroke. *American Journal of Neuroradiology*, 22(8):1534–1542, 2001.

[234] Vasyl Pihur, Aleksandra Korolova, Frederick Liu, Subhash Sankuratripati, Moti Yung, Dachuan Huang, and Ruogu Zeng. Differentially-private "draw and discard" machine learning. *arXiv preprint arXiv:1807.04369*, 2018.

[235] John C Platt, John Shawe-Taylor, Alex J Smola, Robert C Williamson, et al. Estimating the support of a high-dimensional distribution. *Technical Report MSR-T R-99–87, Microsoft Research (MSR)*, 1999.

[236] I. Poole and M. Daykin. *Novelty Forests: A reworking of probability density forests, fit for anomaly detection*. Internal Publication, revised edition, 2015.

[237] World Health publisher. Ageing and health, 2018. https://www.who.int/news-room/fact-sheets/detail/ageing-and-health, visited 2019-07-06.

[238] V Puetz, I Dzialowski, MD Hill, and AM Demchuk. The Alberta Stroke Program Early CT Score in clinical practice: what have we learned? *International journal of stroke*, 4(5):354–364, 2009.

[239] Terence J Quinn, Jesse Dawson, Matthew Walters, and Kennedy R Lees. Reliability of the modified rankin scale: a systematic review. *Stroke*, 40(10):3393–3395, 2009.

[240] Juan Manual Racosta, Federico Di Guglielmo, Francisco Ricardo Klein, Patricia Mariana Riccio, Francisco Muñoz Giacomelli, María Eugenia González Toledo, Fátima Pagani Cassará, Agustina Tamargo, Matías Delfitto, and Luciano Alberto Sposato. Stroke Severity Score based on Six Signs and Symptoms the 6S Score: a simple tool for assessing stroke severity and in-hospital mortality. *Journal of stroke*, 16(3):178, 2014.

[241] Radiopaedia. Windowing (CT), 2018. https://radiopaedia.org/articles/windowing-ct, visited 2018-02-09.

[242] Johann Radon. On the determination of functions from their integral values along certain manifolds. *IEEE transactions on medical imaging*, 5(4):170–176, 1986.

[243] Johann Radon. 1.1 über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. *Classic papers in modern diagnostic radiology*, 5:21, 2005.

[244] T Riall. Dental recruitment: Acute shortage of clinicians. *British dental journal*, 225(1):2, 2018.

[245] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[246] Tara Rosewall, Jing Yan, Andrew J Bayley, Valerie Kelly, Alana Pellizzari, Peter Chung, and Charles N Catton. Inter-professional variability in the assignment and recording of acute toxicity grade using the RTOG system during prostate radiotherapy. *Radiotherapy and Oncology*, 90(3):395–399, 2009.

[247] PM Rothwell, MF Giles, E Flossmann, CE Lovelock, JNE Redgrave, CP Warlow, and Z Mehta. A simple score (ABCD) to identify individuals at high early risk of stroke after transient ischaemic attack. *The Lancet*, 366(9479):29–36, 2005.

[248] EM Rounds. A combined nonparametric approach to feature selection and binary decision tree design. *Pattern Recognition*, 12(5):313–317, 1980.

[249] Peter J Rousseeuw and Katrien Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

[250] Abhijit Guha Roy, Shayan Siddiqui, Sebastian Pölsterl, Nassir Navab, and Christian Wachinger. Braintorrent: A peer-to-peer environment for decentralized federated learning. *arXiv preprint arXiv:1905.06731*, 2019.

[251] S Rasoul Safavian and David Landgrebe. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3):660–674, 1991.

[252] Jeffrey L Saver. Time is brain — quantified. *Stroke*, 37(1):263–266, 2006.

[253] Richard M Scheffler, Jenny X Liu, Yohannes Kinfu, and Mario R Dal Poz. Forecasting the global shortage of physicians: an economic- and needs-based approach. *Bulletin of the World Health publisher*, 86:516–523B, 2008.

[254] Daniel Schlegel, Stephen J Kolb, Jean M Luciano, Jennifer M Tovar, Brett L Cucchiara, David S Liebeskind, and Scott E Kasner. Utility of the NIH Stroke Scale as a predictor of hospital disposition. *Stroke*, 34(1):134–137, 2003.

[255] Markus Schneider, Wolfgang Ertel, and Fabio Ramos. Expected similarity estimation for large-scale batch and streaming anomaly detection. *Machine Learning*, 105(3):305–333, 2016.

[256] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[257] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *International Conference on Artificial Neural Networks*, pages 583–588. Springer, 1997.

[258] Case Western Reserve University School of Medicine. MRI Basics, 2006. https://casemed.case.edu/clerkships/neurology/Web%20Neurorad/MRI%20Basics.htm, visited 2019-06-17.

[259] Juergen Schuermann and Wolfgang Doster. A decision theoretic approach to hierarchical classifier design. *Pattern Recognition*, 17(3):359–369, 1984.

[260] Scikit-learn. sklearn.covariance.EllipticEnvelope, 2018. https://scikit-learn.org/stable/modules/generated/sklearn.covariance.EllipticEnvelope.html, visited 2018-02-09.

[261] Scikit-learn. sklearn.neighbors.IsolationForest, 2018. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html, visited 2018-02-09.

[262] Scikit-learn. sklearn.neighbors.LocalOutlierFactor, 2018. https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html, visited 2018-02-09.

[263] Scikit-learn. sklearn.svm.OneClassSVM, 2018. https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html, visited 2018-02-09.

[264] Scipy. numpy.linalg.pinv, 2009. https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.linalg.pinv.html, visited 2017-11-30.

[265] Section for Biomedical Image Analysis (SBIA) participating with the Center for Biomedical Image Computing & Analytics (CBICA). Multimodal brain tumor segmentation challenge 2018, 2018. https://www.med.upenn.edu/sbia/brats2018/data.html, visited 2018-02-09.

[266] Ishwar K Sethi. Layered neural net design through decision trees. In *IEEE International Symposium on Circuits and Systems*, pages 1082–1085. IEEE, 1990.

[267] Ishwar Krishnan Sethi and GPR Sarvarayudu. Hierarchical classifier design using mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, (4):441–445, 1982.

[268] Micah J Sheller, G Anthony Reina, Brandon Edwards, Jason Martin, and Spyridon Bakas. Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In *International MICCAI Brainlesion Workshop*, pages 92–104. Springer, 2018.

[269] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

[270] Brian Silver, Bart Demaerschalk, José G Merino, Edward Wong, Arturo Tamayo, Ashok Devasenapathy, Christina O'Callaghan, Andrew Kertesz, G Bryan Young, Allan J Fox, et al. Improved outcomes in stroke thrombolysis with pre-specified imaging criteria. *Canadian journal of neurological sciences*, 28(2):113–119, 2001.

[271] Nigel C Smeeton. Early history of the kappa statistic. *Biometrics*, 41:795, 1985.

[272] Nickolay Smirnov. Table for estimating the goodness of fit of empirical distributions. *The annals of mathematical statistics*, 19(2):279–281, 1948.

[273] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4424–4434. 2017.

[274] Lauge Sorensen, Saher B Shaker, and Marleen De Bruijne. Quantitative analysis of pulmonary emphysema using local binary patterns. *IEEE transactions on medical imaging*, 29(2):559–569, 2010.

[275] Gilbert Strang. Linear algebra and its applications. *Thomson Brooks/Cole*, 2005.

[276] Shubin Su, Limin Xiao, Li Ruan, Fei Gu, Shupan Li, Zhaokai Wang, and Rongbin Xu. An efficient density-based local outlier detection approach for scattered data. *IEEE Access*, 7:1006–1020, 2019.

[277] Geert Sulter, Christel Steen, and Jacques De Keyser. Use of the barthel index and modified rankin scale in acute stroke trials. *Stroke*, 30(8):1538–1541, 1999.

[278] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852. 2017.

[279] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[280] Magdalena Szumilas. Explaining odds ratios. *Journal of the Canadian academy of child and adolescent psychiatry*, 19(3):227, 2010.

[281] David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 54(1):45–66, 2004.

[282] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern recognition.* Academic Press, 2009.

[283] Kai Ming Ting, Guang-Tong Zhou, Fei Tony Liu, and Swee Chuan Tan. Mass estimation. *Machine learning*, 90(1):127–160, 2013.

[284] UK Government. The data protection act 2018, 2018. http://www.legislation.gov.uk/ukpga/2018/12/enacted, visited 2019-05-31.

[285] JC Van Swieten, PJ Koudstaal, MC Visser, HJA Schouten, and J Van Gijn. Interobserver agreement for the assessment of handicap in stroke patients. *Stroke*, 19(5):604–607, 1988.

[286] K Veropoulos, N Cristianini, and C Campbell. The application of support vector machines to medical decision support: a case study. *Advanced Course in Artificial Intelligence*, pages 1–6, 1999.

[287] Rüdiger Von Kummer, Kathryn L Allen, Rolf Holle, Luigi Bozzao, Stefano Bastianello, Claude Manelfe, Erich Bluhmki, Peter Ringleb, Dieter H Meier, and Werner Hacke. Acute stroke: usefulness of early CT findings before thrombolytic therapy. *Radiology*, 205(2):327–333, 1997.

[288] Abraham Wald. Sequential analysis. *John Wiley & Sons*, 1947.

[289] Qing Ren Wang and Ching Y Suen. Analysis and design of a decision tree based on entropy reduction and its application to large character set recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (4):406–417, 1984.

[290] Shuang Wang, Xiaoqian Jiang, Yuan Wu, Lijuan Cui, Samuel Cheng, and Lucila Ohno-Machado. EXpectation Propagation LOgistic REgRession (EXPLORER): distributed privacy-preserving online model learning. *Journal of biomedical informatics*, 46(3):480–496, 2013.

[291] JM Wardlaw, PJ Dorman, SC Lewis, and PAG Sandercock. Can stroke physicians and neuroradiologists identify signs of early cerebral infarction on CT? *Journal of Neurology, Neurosurgery & Psychiatry*, 67(5):651–653, 1999.

[292] Joanna M Wardlaw, Keith W Muir, Mary-Joan Macleod, Christopher Weir, Ferghal McVerry, Trevor Carpenter, Kirsten Shuler, Ralph Thomas, Paul Acheampong, Krishna Dani, et al. Clinical relevance and practical implications of trials of perfusion and angiographic imaging in patients with acute ischaemic stroke: a multicentre cohort imaging study. *Journal of Neurology, Neurosurgery & Psychiatry*, 84(9):1001–1007, 2013.

[293] Joanna M Wardlaw, Rüdiger Von Kummer, Andrew J Farrall, Francesca M Chappell, Michael Hill, and David Perry. A large web-based observer reliability study of early ischaemic signs on computed tomography. The Acute Cerebral CT Evaluation of Stroke Study (ACCESS). *PLoS One*, 5(12):e15757, 2010.

[294] Simon K Warfield, Kelly H Zou, and William M Wells. Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation. *IEEE transactions on medical imaging*, 23(7):903–921, 2004.

[295] The Next Web. How artificial intelligence is revolutionizing healthcare, 2017. https://thenextweb.com/artificial-intelligence/2017/04/13/artificial-intelligence-revolutionizing-healthcare/, visited 2019-07-06.

[296] KMA Welch, BC Tilley, JR Marler, T Brott, P Lyden, JC Grotta, et al. Tissue plasminogen activator for acute ischemic stroke. The National Institute of Neurological Disorders and Stroke rt-PA stroke study group. *N Engl J Med*, 333:1581–7, 1995.

[297] TA Whitelaw. *Introduction to Linear Algebra*. CRC Press, 1991.

[298] David C Wilbur, Elena F Brachtel, John R Gilbertson, Nicholas C Jones, John G Vallone, and Savitra Krishnamurthy. Whole slide imaging for human epidermal growth factor receptor 2 immunohistochemistry interpretation: Accuracy, precision, and reproducibility studies for digital manual and paired glass slide manual interpretation. *Journal of pathology informatics*, 6, 2015.

[299] Linda S Williams, Morris Weinberger, Lisa E Harris, and José Biller. Measuring quality of life in a way that is meaningful to stroke patients. *Neurology*, 53(8):1839–1839, 1999.

[300] Linda S Williams, Morris Weinberger, Lisa E Harris, Daniel O Clark, and José Biller. Development of a stroke-specific quality of life scale. *Stroke*, 30(7):1362–1369, 1999.

[301] JT Lindsay Wilson, Asha Hareendran, Marie Grant, Tracey Baird, Ursula GR Schulz, Keith W Muir, and Ian Bone. Improving the assessment of outcomes in stroke: use of a structured interview to assign grades on the modified rankin scale. *Stroke*, 33(9):2243–2246, 2002.

[302] JT Lindsay Wilson, Asha Hareendran, Anne Hendry, Jan Potter, Ian Bone, and Keith W Muir. Reliability of the modified rankin scale across multiple raters: benefits of a structured interview. *Stroke*, 36(4):777–781, 2005.

[303] Patrick Winston. A heuristic program that constructs decision trees. 1969.

[304] Charles D Wolfe, Nick A Taub, EJ Woodrow, and PGJ Burney. Assessment of scales of disability and handicap for stroke patients. *Stroke*, 22(10):1242–1244, 1991.

[305] Fred Wynn Wright. *Radiology of the chest and related conditions*. CRC Press, 2001.

[306] C. Burges Y. LeCun, C. Cortes. The MNIST Database of handwritten digits, 1998. http://yann.lecun.com/exdb/mnist/, visited 2018-02-09.

[307] A. Yin. Introducing Open Mined: Decentralised AI, 2017. https://becominghuman.ai/introducing-open-mined-decentralised-ai-18017f634a3f, visited 2018-02-09.

[308] KC You and King-Sun Fu. An approach to the design of a linear binary tree classifier. 1976.

[309] Ke Zhang, Marcus Hutter, and Huidong Jin. A new local distance-based outlier detection approach for scattered real-world data. In *Pacific-Asia*

Conference on Knowledge Discovery and Data Mining, pages 813–822. Springer, 2009.

[310] Yu Zhang and Dit-Yan Yeung. A convex formulation for learning task relationships in multi-task learning. *arXiv preprint arXiv:1203.3536*, 2012.

[311] Guang-Tong Zhou, Kai Ming Ting, Fei Tony Liu, and Yilong Yin. Relevance feature mapping for content-based multimedia information retrieval. *Pattern Recognition*, 45(4):1707–1720, 2012.