UNIVERSITY OF SOUTHAMPTON

Predicting brain age from neuroimaging using Machine Learning and Deep Learning

by

Ziyang Wei

Supervisor:Kate Farrahi

Second Examiner:Dr. Yvonne Howard

A dissertation submitted in partial fulfillment for the degree of MSc Computer Science

in the
Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

September 2019

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Master of Computer Science

by Ziyang Wei

Some related research find that aging can be considered as a process of brain shrinking, while the quantification of this change is still uncertain. This project is mainly focus on revealing the relationship between aging and the change in the brain using MRI with machine learning and deep learning technology. The content of this paper will cover the process of data cleaning, skull stripping, image resizing, n4 bias field correction, template registration, and intensity normalisation.Then I will focus on the machine learning methods that are applied in the project, then followed by the deep learning method with using 3D Resnet.Afterwards, I will display their result and compare their prediction performance from many aspects. Then I will analyse each approach's strength and weakness.Finally, I will draw a conclusion,which includes the self-evaluation of this project and difficulties encountered in the course of the project.The future work will be discussed in the end of this paper. The best performance of using machine learning method is achieved by random forest regression with applying Simon Jgou's normalisation method, which gets a 5.815 years mean absolute error in the end. Furthermore, A 3D CNN using Resnet34 architecture yield the best performance(4.19 years), which is closed to the performance of state of art for 4.16 years (Cole et al. (2017)). An online tool has also been constructed on https://wzy-codify.com/age-predictor, which demonstrates the whole predicting process including skull stripping, n4 field bias correction, template registration, normalisation and age prediction. The technical detail has been introduced in the Chapter.7.

Contents

List of Figures

List of Tables

Nomenclature

MRI	Magnetic Resonance Imaging
CSF	Cerebrospinal Fluid
CNN	Convolutional Neural Network
CONV	Convolution
TR	Repetition Time
TE	Time to Echo
SD	Standard Deviation
MAE	Mean Absolute Error
MSE	Mean Square Error
RMSE	Root Mean Square Error
GMM	Gaussian Mixture Model
NAWM	Normal Appearing White Matter
KDE	Kernel Density Estimate-based
CDF	Cumulative Distribution Function
GP	Gaussian Process
ReLU	Rectified Linear Unit
MLP	Multi-layer Perceptron

Acknowledgements

I would like to thank my supervisor Kate Farrahi who proposed this topic and gave me many useful instructions when I got stucked in building convolutional neural network.I would also want to appreciate Simon Jgou, who is the author of the blog article "How to estimate the age of your brain with MRI data and Artificial Intelligence".He answered my many questions about medical imaging and normalisation method with his professional knowledge.I would also want to appreciate my second examiner Dr. Yvonne Howard, who gave me many a lot of guidance on writing structure and encouraged me to try more experiments.Finally, I would also want to thank my classmate Yalun Hu who patiently explained the basic mechanism of Resnet to me.

Statement of Originality

1. I have read and understood the ECS Academic Integrity information and the Universitys Academic Integrity Guidance for Students
2. I am aware that failure to act in accordance with the Regulations Governing Academic Integrity may lead to the imposition of penalties which, for the most serious cases, may include termination of programme.
3. I consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to verify whether my work contains plagiarised material, and for quality assurance purposes
4. I have acknowledged all sources, and identified any content taken from elsewhere
5. I have not used any resources produced by anyone else.
6. I did all the work myself, or with my allocated group, and have not helped anyone else.
7. The material in the report is genuine, and I have included all my data/code/designs
8. I have not submitted any part of this work for another assessment.
9. My work did not involve human participants, their cells or data, or animals.

Chapter 1

Introduction

1.1 Background

Machine Learning has recently widely been used in medical research field including tissue classification, tissue segmentation(Dey and Hong (2018)) and tumor detection(Shree and Kumar (2018)), and it achieves many state-of-art performance compared with the traditional methods.While researches about predicting the brain age are still very rare since it is too hard to capture the features of the brain associated with aging.

The human 's brain changes from it born to death and this process occurs with a gradual decline in volume of ventricle.However, the quantification of the age-related change in whole brain is still uncertain and the effects of ageing on the brain vary significantly between individuals. Therefore, it is very difficult for a physician to determine the age of the subject from the brain image only.While deep learning method can capture the general feature between brain age and its neuroimaging from a large dataset and achieve the state of art performance of 4.16 years(Cole et al. (2017)).

1.2 Basic knowledge for brain and MRI

1.2.1 MRI

The full name of MRI is Magnetic Resonance Imaging, which uses a powerful magnetic field, radio waves and a computer to produce detailed pictures of internal body structures. The main magnet coils create a strong, constant and stable magnetic field to which the patient is exposed. The strength of the magnetic field is measured in units of Tesla(T). One Tesla is equal to approximately 20000 times the earths magnetic field. For most of the image are operated at 1.5 T or 3 T environment(Currie et al. (2013)).As for this project, all the neuroimaging are 1.5T. Since the ouput image may be influenced

by magnetic field more or less, N4 bias field correction is required before processing the image and this part will be introduced at Chapter 3.3.

Furthermore, the most common MRI sequences can be divided into T1-weighted and T2-weighted scans. T1-weighted images are produced by using short TE and TR times. The contrast and brightness of the image are predominately determined by T1 properties of tissue. In contrast, T2-weighted images are produced through longer TE and TR times. In these images, the contrast and brightness are predominately determined by the T2 properties of tissue.Generally speaking, T1- and T2-weighted images can be easily recognised by observing the CSF. For T1-weighted image, CSF is dark. For T2-weighted image, it is bright. Figure.1.1 has shown the difference between T1 weighted and T2 weighted image clearly.For this project, all the MRI are T1 weighted since it can display different tissues more clearly.

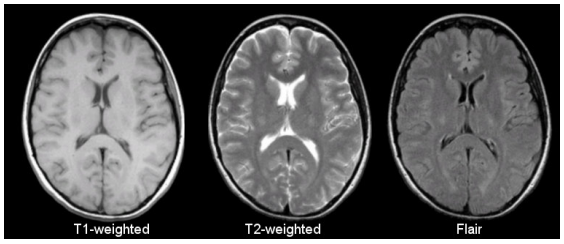


FIGURE 1.1: Comparison of T1 vs.T2 vs.Flair(Brain)

1.2.2 Brain Aging

The human's brain mainly consists of two main tissues:grey matter and white matter.The grey matter is made of neuronal and glial cells that is used to control brain activity.The white matter is made of fibre tracts located throughout the inner regions, which enables the brain cells to quickly send and receive messages. The Cerebrospinal Fluid (CSF) is secreted into the ventricles , and circulates throughout the nervous system to mitigate the brain and the spinal cord against shocks.These three components are the most important features for our research since they will change significantly with human aging. Figure.1.2 shows an anatomy of a human brain clearly.

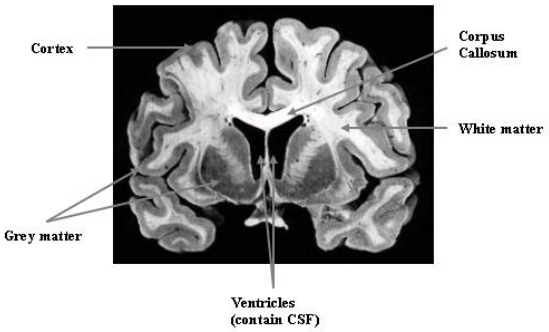


FIGURE 1.2: Anatomy of the human brain

Sowell et al. (2003) found that the volume of white matter and gray matter decrease with aging due to the shink of ventricle, while the CSF volume increase when human grows up like the Figure.1.4 shows below.Moreover,Jgou (2018) also mentioned that there are three main features associated with brain aging and they are shown in the Figure.1.3. Those pattern makes it possible that we can use machine learning technology to capture them and use it to predict the human age.

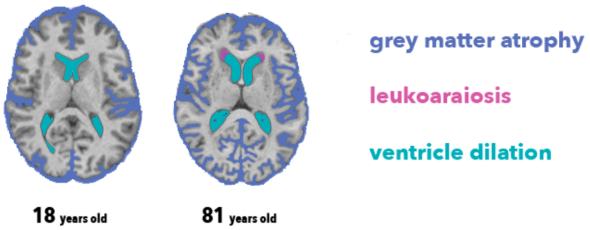


FIGURE 1.3: 18 year old brain versus 81 year old brain

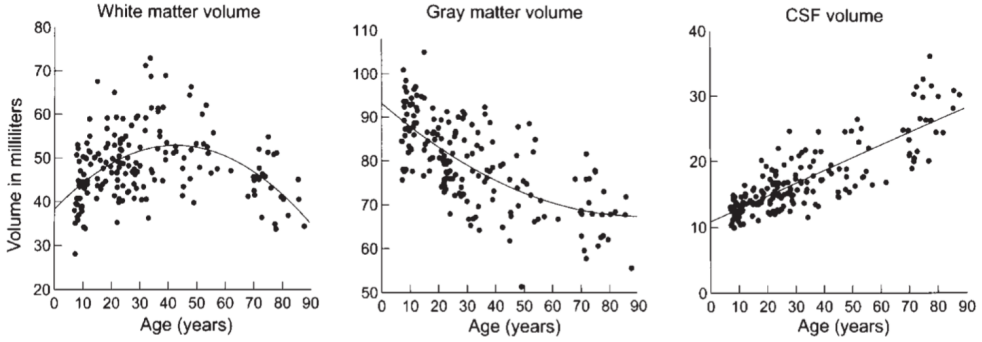


FIGURE 1.4: Scatterplots of the nonlinear effects of age on total brain white matter, total brain gray matter and total brain CSF volume.

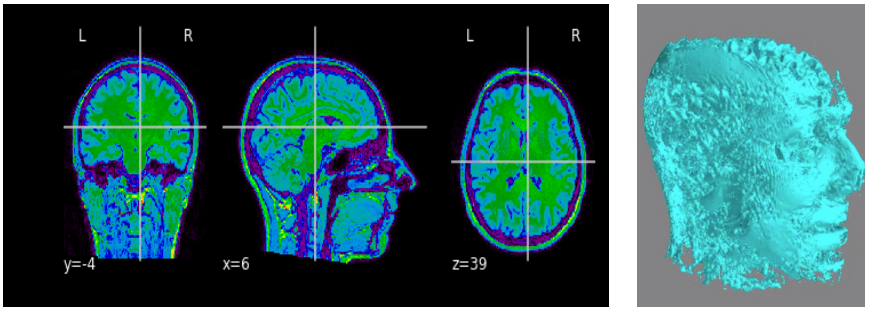
1.2.3 NIfTI-1 Data Format

The .nii file type is primarily associated with NIfTI-1 Data Format and can be read by using "nibabel" library with Python.A nibabel image contains three things:

1. The image data array with 3 dimension
2. Affine matrix reveals the position of the image array data in a reference space
3. Image metadata

1.2.3.1 Image data visualisation

The third party library 'nilearn' provides a convenient approach to view the whole brain from three views.Through this way, we can check whether the brain image is intact after applying many image preprocessing steps .Moreover, 'mayavi' makes it possible that we can view the contour of a brain(Ramachandran and Varoquaux (2011)).Figure.1.5 shows the visualisation result with different library.



(a) nilearn (b) mayavi

FIGURE 1.5: Visualize .nii file with thrid party library

1.2.3.2 Affine matrix

If we collect the anatomical image with a different field of view and orientation, their image data will be different with each other.But if we may want to relate them together so that we can recognise that they are the same brain, we need a reference space to keep track of the relationship of voxel coordinates.To solve this, An affine array is used to store the relationship between voxel coordinates in the image data array and coordinates in the reference space.Basically speaking, we can use an affine matrix to cast the existing cordinate to an universal coordinate system called "RAS+".(Matthew Brett and Chen (2018))

The affine matrix is the basic concept when we want to transform our brain image with different orientations and shapes to the image with the same size and orientation in the future work.Therefore, It is necessary to introduce them in detail.

Generally speaking, we have some voxel space coordinate(i, j, k),and the coordinates we generate in the reference space is (x, y, z).We define the transform function f that accepts a voxel coordinate and returns a coordinate in the reference space:

$$(x, y, z) = f(i, j, k) \tag{1.1}$$

The scaling operation in 3D space can be defined by a diagonal 3 by 3 matrix. If we want to zoom the first dimension by p, the sencond dimension by q, and the third by r,

we can compute the result like below:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} pi \\ qj \\ rk \end{bmatrix} = \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & r \end{bmatrix} \begin{bmatrix} i \\ j \\ k \end{bmatrix} \tag{1.2}$$

A rotation in three dimensions can be represented as a 3 by 3 rotation matrix.A rotation by θ around the third array axis is:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ k \end{bmatrix} \tag{1.3}$$

The rotation by ϕ around the second axis is:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ 0 & 1 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) \end{bmatrix} \begin{bmatrix} i \\ j \\ k \end{bmatrix} \tag{1.4}$$

The rotation by γ around the first axis is:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix} \begin{bmatrix} i \\ j \\ k \end{bmatrix} \tag{1.5}$$

A translation in three dimensions can be defined as a vecotr of length 3 to be added to the length 3 coordinate.A translation of a units on the first axis, b units on the second and c on the third can be represented as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} i \\ j \\ k \end{bmatrix} + \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{1.6}$$

Moreover, we can combine all the rotation and scaling operation to a single matrix M by multiplying transformations one by one. Therefore, the function f can be rewriten to:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = M \begin{bmatrix} i \\ j \\ k \end{bmatrix} + \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{1.7}$$

The affine matrix is derived from the Equation1.7.If m_i, j is the value in the column of

j and the row of i, the image affine matrix A is:

$$A = \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} & a \\ m_{2,1} & m_{2,2} & m_{2,3} & b \\ m_{3,1} & m_{3,2} & m_{3,3} & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.8}$$

The affine matrix provides the information about how to relate the existing coordinate with the reference space and it make it possible that we can map brain images with different shapes and orientations to the same size image with the same orientation through a series of transformation.

1.2.3.3 Metadata for .nii file

Apart from the affine matrix, .nii file also provides many other information that will be used in the project. Below shows some of the metadata information that can be accessed by nibabel.The *pixdim* attribute tells us the resolution of the brain image, which will be used in the process of image resampling.

1.3 Research Aim and Objectives

In order to learn the feature of brain image and predict age based on the pattern of it, the first step of it is to collect a large amount of MRI data for training the model.Then, we need to do the data cleaning work for the collected data in case of the negative effect raised by noise data.

Afterwards, I will process the collected data. The aim for this step is listed below.

1. Reiszing.The resolution of collected image should be resample to $1mm^3$.
2. Skull stripping.Some redundant tissues should be removed in this procedure.
3. N4-bias correction.The negative side-effect raised by magnetic field should be eliminated in this procedure.
4. Template registration.Images with various shapes and orientations will be registered into the same shape and the same orientation.
5. Intensity Normalisation.Intensity range varies significantly between each different hospitals and universities due to the use of different devices.Therefore, the voxel intensities should be normalised to the same range and the contrast information should be kept during this period.

Followed by the image preprocessing, I will train the model with different approaches including simple linear regression, gaussian process regression, random forest regression and 3D CNN.Then the performance of each model with vairous parameter will be compared so as to obtain the best model.

Finally, an online web tool for predicting the brain age and visualising the whole process will be constructed using Django as its framework.

1.4 Technologies and Tools

Third-party libraries used in this project:

deepbrain==0.1 nibabel==2.4.1 nilearn==0.5.2 numpy==1.16.4 pandas==0.24.2 pyecharts==1.2.1 scikit-image==0.15.0 scikit-learn==0.21.3 scipy==1.3.0 SimpleITK==1.2.0 sklearn==0.0 torch==1.1.0.post2 torchbearer==0.4.0 torchvision==0.3.0 tqdm==4.32.1

Template registration Tool:

- FSL(published by Woolrich et al. (2009), Currie et al. (2013), Wolfers et al. (2017))

Chapter 2

Data Collection

2.1 Data source overview

For this project, I collected 1635 samples from different hospitals and universities and it is accessible at https://storage.cloud.google.com/dissertation_wzy/raw.zip.Table. 2.1 has shown the data source in detail. The access to those data has been approved by the provider.

Cohort	N	Age mean(SD)	Age range	Sex(male/female)	Repository details
Beijing	198	21.16(1.82)	18.0-26.0	76/122	INDI
Atlanta	28	30.89(9.72)	22.0-57.0	13/15	INDI
NewYork	104	25.41(10.17)	7.88-49.16	51/53	INDI
Queensland	19	25.95(3.78)	20.0-34.0	11/8	INDI
ICBM	86	44.19(17.81)	19.0-85.0	41/45	INDI
Oulu	103	21.52(0.57)	20.0-23.0	37/66	INDI
Cambridge	198	21.03(2.3)	18.0-30.0	75/123	INDI
Oxford	22	29.0(3.71)	20.0-35.0	12/10	INDI
PaloAlto	17	32.47(7.87)	22.0-46.0	2/15	INDI
Dallas	24	42.62(19.65)	20.0-71.0	12/12	INDI
NewHaven	35	29.11(8.72)	18.0-48.0	18/17	INDI
Baltimore	23	29.26(5.34)	20.0-40.0	8/15	INDI
AnnArbor	60	36.22(23.97)	13.41-80.0	38/22	INDI
Leipzig	37	26.22(4.94)	20.0-42.0	16/21	INDI
SaintLouis	31	25.1(2.28)	21.0-29.0	14/17	INDI
Bangor	20	23.4(5.18)	19.0-38.0	20/0	INDI
IXI	563	48.65(16.46)	19.98-86.32	250/313	IXI database
Pittsburgh	17	37.94(8.76)	25.0-54.0	10/7	INDI
Leiden	31	22.19(2.53)	18.0-28.0	23/8	INDI
Orangeburg	20	40.65(10.75)	20.0-55.0	15/5	INDI

Table 2.1: Data source table

Figure.2.1 has shown the data source distribution through pie chart.From it we can

know that data provided from IXI takes a large proportion.While IXI did not provide the brain image, they provide a whole head instead.

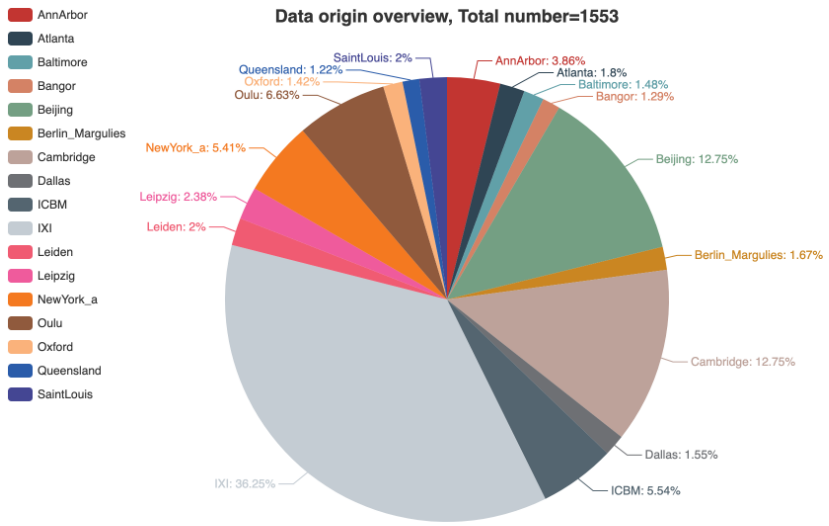


FIGURE 2.1: Data source

Figure.2.2 describes the age distribution in the training sample.It is obvious that the data sample is so unbalanced that the young people take a large percent in it, while the elders are so few.It may cause a problem that even if we predict all the test data to be 30 years old, the average absolute error still looks not so bad.The problem can be solved by introducing more metrics to evaluate the performance of our model such as Pearson correlation coefficient, R Squared and root mean squared error.

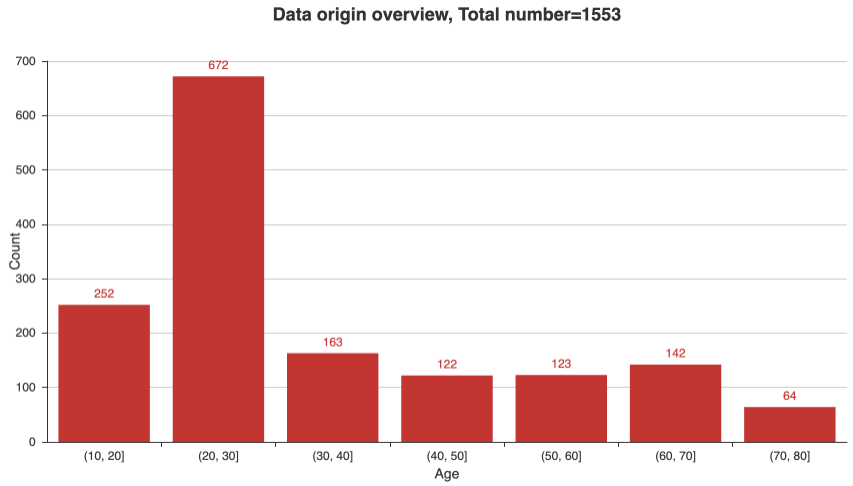


FIGURE 2.2: Age distribution

Figure.2.3 shows the sex distribution of the training sample.The distribution for sex is almost balanced as expected.

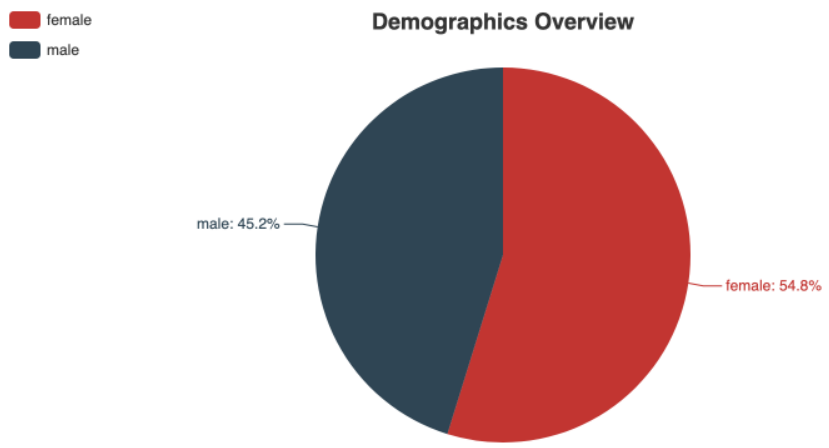


FIGURE 2.3: Sex distribution

2.2 Data Cleaning

Before the start of image preprocessing, I notice some problems among the original dataset, which will cause some severe problems for further research.The problems have been listed below:

- Some of the subjects have the .nii file but missed in the .csv file
- Some of the subjects appear in the .csv file but miss the .nii file
- Duplicated problem is very common in the dataset.For example,two subjects may have the same ID but have different age or sex.
- Some subjects' age are obviously be marked wrongly.For example, I notice some of the subjects' old are less than 0, which is obviously impossible

Those problems may raised by the transition of data through complicated data systems or unware manual modification and so on.To address those problems, I removed all the error data.

Chapter 3

Image Preprocessing

Since images are collected from different devices, their resolutions, contrasts, voxel intensity ranges, shapes, field of views, orientations, are largely different with each other, which will incur a negative impact on the predict result.Therefore, the image preprocessing is necessary for the research before we build the predict model.Figure.3.1 shows the flowchart of the image preprocessing.

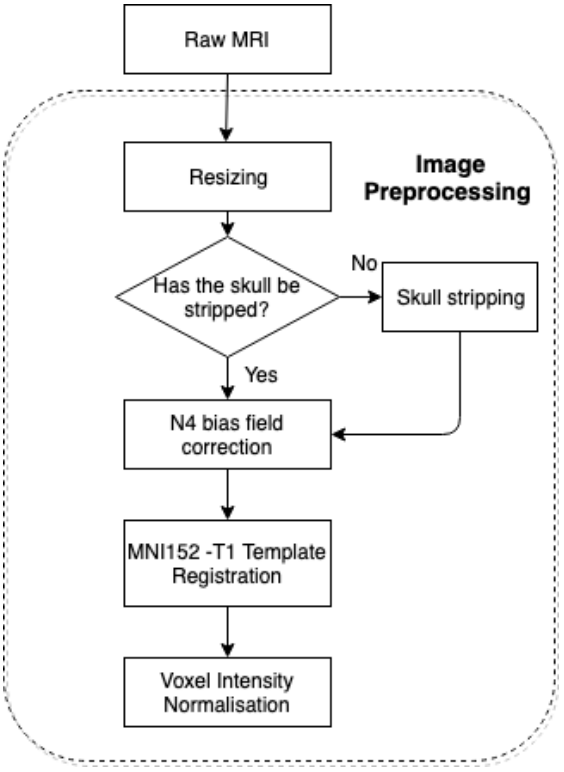


FIGURE 3.1: Overview of image preprocessing

3.1 Resizing

Due to the different scanners' parameters,the image resolutions may vary from devices to devices.Since we can obtain the voxel dimension from the header of NIfTI file easily, we can resize the images to the same dimension.The standard dimension for this project has been set to $1 * 1 * 1$ to keep the information as complete as possible.

3.2 Skull stripping

Data from the source of 'IXI' does not provide the extracted brain.Instead they provide the whole head as its input like the figure.3.2 shows below.Since we only need the brain component as our training data, the unnecessary part like face and ear should be excluded automatically.

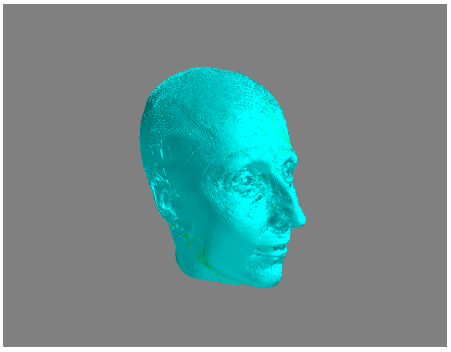


FIGURE 3.2: Image with no skull stripping

3.2.1 Pretrained model based on Unet

To address this problem, I use a pretrained model called DeepBrain created by Itzcovich (2018), which is trained based on the UNET proposed by Ronneberger et al. (2015).The author achieves state-of-the art accuracy > 0.97 dice metric on the test set that is compound with a subset of entries from the CC359 dataset, NFBS dataset and ADNI dataset. Figure. 3.3 illustrates the architecture of the UNET.Unet is an end-to-end fully convolutional network.Basically speaking,it contains two paths. The first path is the encoder that is used to capture the feature of the image. The encoder contains a stack of convolution layers and max pooling layers. The second path is the decoder which is used to enable precise localization using transposed convolutions.

The output of the pretrained model is the 3D array where each number represents the possibility that the voxel is part of brain, which ranges from 0 to 1.
The output of 'DeepBrain' has been displayed in the figure.3.4.Although the brain part

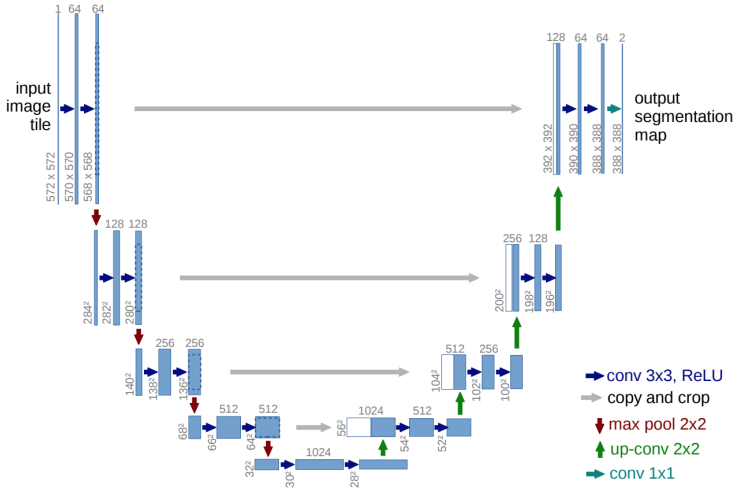


FIGURE 3.3: Unet architecture

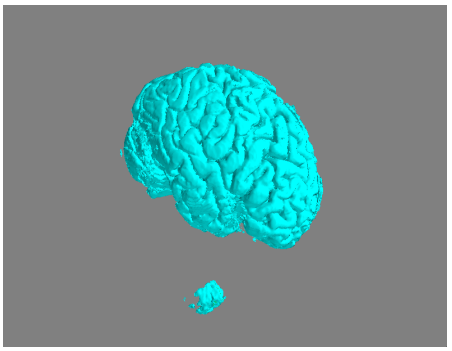


FIGURE 3.4: Output of DeepBrain

has been extracted very well, there are still some small blobs be recognizeds as the part of brain.

3.2.2 Remove small blobs with morphology method

To remove the small blobs produced from DeepBrain net, I use a simple method based on the morphology. Since the voxel of human brain are fully connected with each other and its area are the largest among all the connected region, we can divide the image into several components. Afterwards, we compute the area of each component and keep the largest one.Then, we will get the final brain like figure.3.5 shows below.
We can see the skull-stripping result more clearly using the nilearn library. Figure.3.6 shows the brain mask of the whole head and the blue part is the extracted brain and its performance is meet our research requirement.

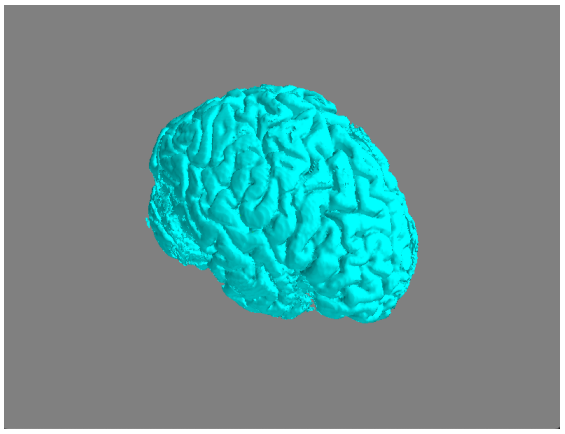


FIGURE 3.5: Final skull-stripped brain

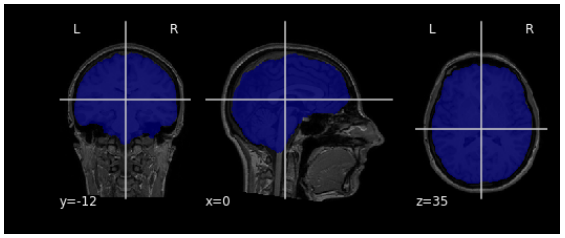
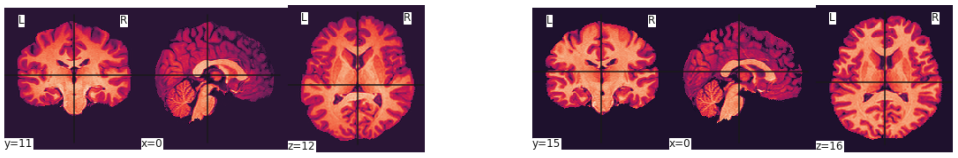


FIGURE 3.6: Brain mask of the whole head

3.3 N4 bias field correction

Bias field signal is a low-frequency and smooth signal that can corrupt MRI images, which may produce unsatisfactory results when we apply image processing algorithms such as segmentation, texture analysis or classification.Figure.3.7 shows the difference between corrected image and not corrected image clearly.For the image that has not been corrected, the intensity values for the white matter volumns does not same.The white matter near the gray matter is much darker than the other white matter.



(a) not corrected (b) corrected

FIGURE 3.7: Difference between corrected image and not corrected image

Tustison et al. (2010) proposed a method based on B-spline approximation to remove the inhomogeneity raised by bias field. A n4 bias field correction tool called 'SimpleITK' implemented this algorithm and make it accessible for public use.
Since the algorithm needs to iterate so many times, the computation time for correcting is up to 6 minute per neuroimaging.The number of our training dataset is more than

1600 and it takes nearly 5 days to correct all the images even though the parallelizing operations are applied.

3.4 Template registration

Since every neuroimagings' shape and orientations are different with each other, it is necessary to transform all the images to have the same orientation, field of view, center of mass, etc.
Jenkinson et al. (2002) proposed a method for the robust and accurate linear template registration.We denote the template image as the reference image(Y) and the original image as the floating image(X).A common way of formulating it as a mathematical problem is to construct a cost function which estimates the dissimilarity between two images and then search for the transformation ($T*$) which gives the minimum cost.

$$T^* = arg \min_{T \in S_T} C(Y, T(X)) \tag{3.1}$$

where S_T is the transformation equations that have been mentioned in the section.1.2.3.2 before.C is the cost function and $T(X)$ denotes the image X that has been registered.Y is the reference.
Therefore, the task of template registration is to search for the affine transformation matrix which can minimize the cost value. The affine matrix that have been defined in section1.2.3.2 have 12 parameters in total, which will result that the optimization takes place in a high dimensional space.The author proposes a multi-resolution technique to solve the problem and speed up the search for parameters.

The template for the registration that I used in the project is the MNI152 standard-space T1-weighted average structural template image, where MNI represents the Montreal Neurological Institute of McGill University Health Centre and 152 scans were used to locate structures(Maintz and Viergever (1998)).

Since the search for the parameters is time-consuming, it takes nearly 20 hours to finish template registration for all the images.After completing the template registration, their shapes and orientations are unified to the same scale of $182 * 218 * 182$.

3.5 Voxel Intensity Normalisation

Since our neuroimagings come from different hospitals and universities, their voxel intensity are totaly different and arbitrary(may from 0 to 1000, or from 0 to 2000).Figure.3.8 shows the slice images from different hospitals.It is obviously that there voxel intensity are various which may have an negative impact on the experiment result.Figure.3.9 shows

the intensity histogram for different images with no normalisation.The peak for white matter and the peak for grey matter are distributed in different zone, which will cause some models cannot capture the feature of image correctly. Therefore, normalisation should be applied to every images before training model.

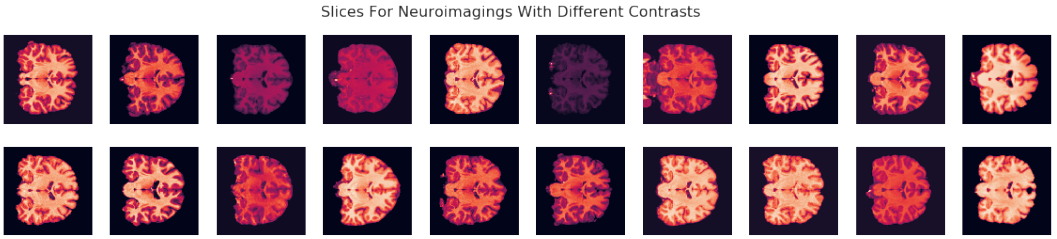


FIGURE 3.8: Slices for neuroimagings from different hospitals

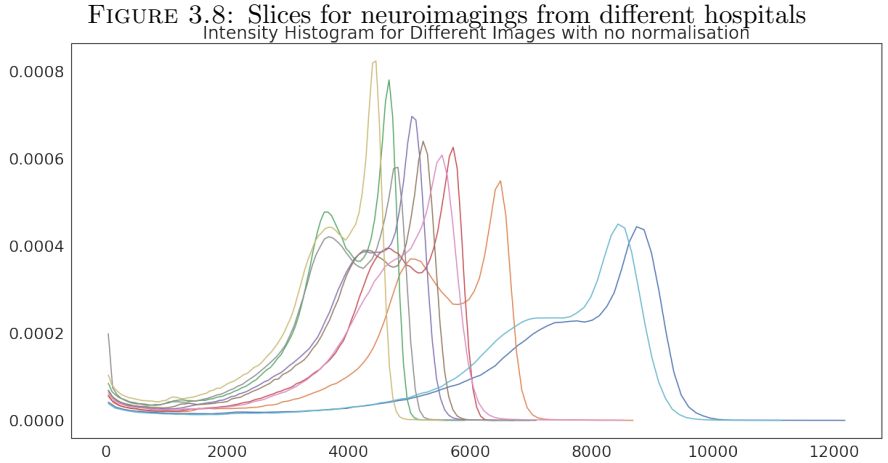


FIGURE 3.9: Intensity histogram for images with no normalisation

There are many normalisation algorithms that can be applied to normalise the intensity distribution of a sample of MR brain images.Reinhold et al. (2019) summarised the common algorithms for normalisation in their papers. For this project, I implemented Z-score normalisation, GMM based normalisation and white stripe normalisation independently and they produce different results in the following experiments, which will be discussed and compared in the Chapter.6.Finally, Jgou (2018) proposed a novel normalisation method in his blog and it produced a good result compared with other methods.

We suppose that $I(x)$ be the MR brain image and $B \in I$ is the brain mask of the image.

3.5.1 Z-score normalisation

Z-score normalisation is the most simple and the most common algorithm among all the approaches.The equation of the Z-score is:

$$I_{z-score}(x) = \frac{I(x) - \mu_{zs}}{\sigma_{zs}} \tag{3.2}$$

where μ_{zs} is the mean value of the intensities inside the brain mask and σ_{zs} is the standard deviation.This method can guarantee the standard deviation always be 1 and

the mean of value always be 0.The problem of this problem is that it cannot fix a specific tissue's intensity to a certain value.

3.5.2 GMM based normalisation

The full name of GMM is Gaussian mixture model based normalisation.It seeks for finding a mixture of three normal distributions to fit the histogram of intensities. The three gaussian distributions correspond to gray matter, white matter and CSF respectively. The equation of it is :

$$I_{gmm}(x) = \frac{c_2 * I(x)}{\mu_{gmm}} \tag{3.3}$$

where $c2$ is a constant and μ_{gmm} is the mean value of the white matter. Since the intensity of white matter is the largest one among all the tissues, the μ_{gmm} is mean of the mixture component with the maximum intensity for T1 images.

This method can guarantee the intensity value of white matter always be 1.Figure .3.10 shows that all the brain's white matter has been fixed to 1.The most advantage of this method is it that it can limit various intensity ranges to the same range between 0 to 1.5.

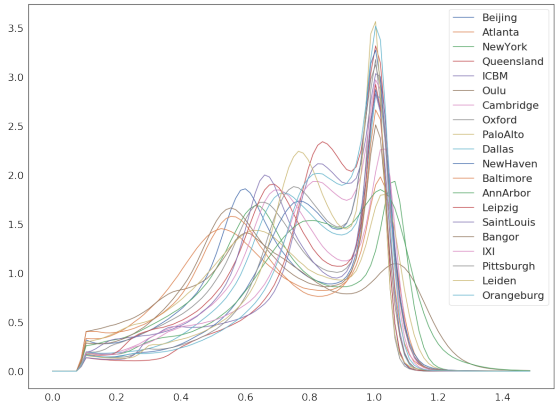


FIGURE 3.10: Average per-hospital intensity histogram with GMM normalisation

3.5.3 White-stripe normalisation

Shinohara et al. (2014) performs a Z-score normalisation based on the intensity values of normal appearing white matter(NAWM). The NAWM can found by selecting the highest intensity peak with KDE(Kernel Density Estimate-based) method.The kernel function we use is a Gaussian kernel and the bandwidth can be obtained by calculating $max(I)/80$.The white strip normalized image can be defined as:

$$I_{ws}(x) = \frac{I(x) - \mu_{ws}}{\sigma_{ws}} \tag{3.4}$$

where μws is the intensity related with the peak of white matter and σ_{ws} is the sample standard deviation associated with the white stripe Ω_τ.The white strip is defined as the 10% segment of intensity values around μ_{ws}. We denote $F(x)$ be the cdf of the specific MR image $I(x)$ inside its brain mask B and define and define $\tau = 5\%$.Then, the white stripe Ω_τ can be defined as:

$$\Omega_\tau = \{I(x)|F^{-1}(F(\mu_{ws}) - \tau) < I(x) < F^{-1}(F(\mu_{ws}) + \tau)\} \tag{3.5}$$

This method can make the intensity of white matter always be 0.Figure.3.11 shows the difference between the histogram for raw image and the histogram for image with white stripe normalisation.The voxel intensity is roughly been scaled between -50 to 50. But it is also obviously that some images are normalised wrongly such as the one from Atlanta:it fix the intensity of grey matter to 0 instead of white matter.The problem is raised by the arbitrarily thought that the white matter is the most part of the brain.Therefore, the performance of following experiment based on this normalisation method is not good as expected, and it will be discussed in the Section.6.

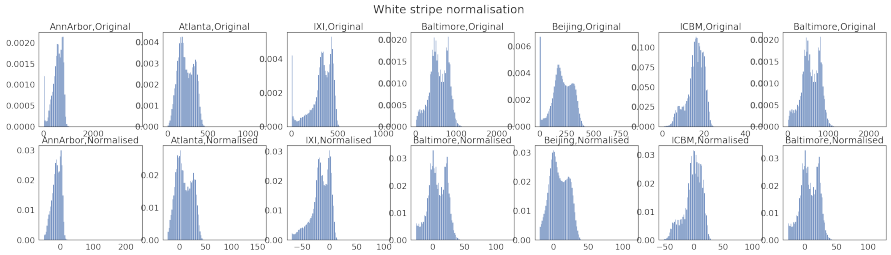


FIGURE 3.11: Voxel intensity histogram for images with white stripe normalisation

3.5.4 Simon Jgou's normalisation

Although GMM method can fix the grey matter to 1 and white strip normalisation could fix it to 0, the intensity for grey matter is still unfixed matterFigure3.10.Jgou (2018) proposed a novel normalisation method that can additionally fix the grey matter to a certain value.The normalisation equation is:

$$\begin{aligned} c &= \frac{\mu_{gm}}{\mu_{wm}} \\ a &= \frac{c_4 - c^2}{c - c^2} \\ I_{sj}(x) &= a * I(x) + (a-1) * I(x)^2 \end{aligned} \tag{3.6}$$

where μ_{gm} is the mean value of the intensity inside the grey matter and μ_{wm} is the mean value of the intensity inside the white matter.c_4 is a constant number that denotes the intensity value of the normalised grey matter.For this project, I let c_4 be 0.75

Figure.3.12 shows the average intensity histogram with the homemade normalisation.The

grey matter has been fixed in around 7.5 and the white matter has been fixed in 1.

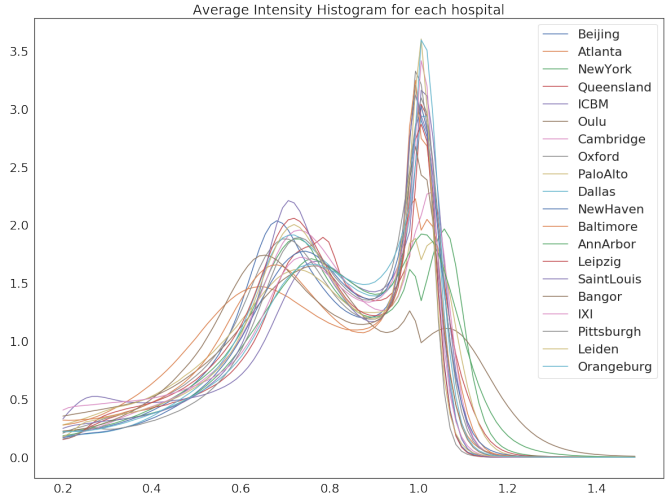


FIGURE 3.12: Average per-hospital intensity histogram with the homemade normalisation

3.5.5 Normalisation Result

As the Figure.3.13 shown below, the top row is the cross-section of the raw brain images, the bottom row is the images normalised with GMM method.It is clearly that after applying GMM normalisation method, the white matter parts of the brain showed the same intensity for all images.

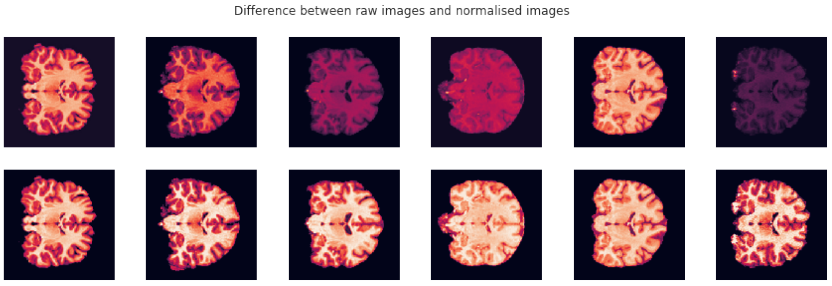


FIGURE 3.13: Difference between raw images and normalised images

Chapter 4

Machine Learning Method and Implementation

After processing the collected image, we have enough training data with the same shape, same orientation and same contrast. This chapter will mainly introduce how I convert those data into feature vector and how I build a model to train those data.

4.1 Image representation

In the section of 3.5, the histogram was used to show the range of voxel intensity.Moreover, the histogram is also suitable to represent a brain.Figure.4.1 shows a typical histogram image with the bin set to 300.

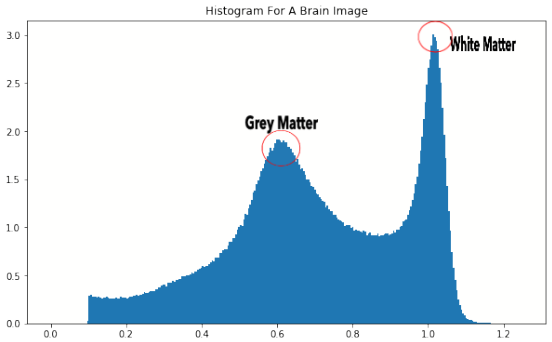


FIGURE 4.1: Intensity Histogram for a brain

We suppose that the first peak of the histogram represents the grey matter and the second peak of the it is the white matter.We can check this assumption easily by looking at the figure 4.2.For the grey matter, I keep the voxels with intensity ranging from 0.5 to 0.7 and set other voxels to 0.For the white matter, I keep the voxels with intensity ranging from 0.9 to 1.1 and set other voxel to 0.Afterwards, we visualise the two images

respectively.According to the Figure.1.2 that is given before, our assumption proves to be true.

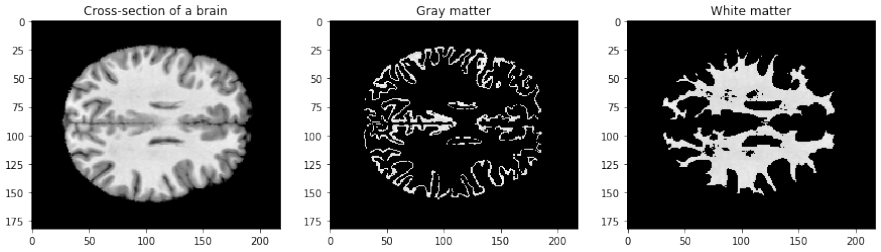


FIGURE 4.2: The cross-section image for grey matter and white matter

Based on this fact, the feature vector for a subject can be represent by the intensity histogram.

4.2 Linear Regression

Linear regression was developed in the field of statistics and is studied as a model for understanding the relationship between input and output numerical variables. We define the X is the explanatory variable and y is label.For every sample x_i,$f(x_i)$ is the predict value.

$$f(x_i) = \sum_{m=1}^{p} w_m x_{im} + w_0 = w^T x_i \tag{4.1}$$

The goal of linear regression is to fit the target label as best as possible with the predicted results, which means we need to minimise the loss function:

$$J(w) = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2 = \frac{1}{n}\|y - Xw\|^2 \tag{4.2}$$

Moreover, we can also obtain the w by computing the normal equation:

$$\hat{w} = (X^T X)^{-1} X^T y \tag{4.3}$$

4.3 Gaussian Process Regression

The function of Gaussian processes (GP) is to extend multivariate Gaussian distributions to infinite dimensionality.Formally, a Gaussian process generates data located throughout some domain such that any finite subset of the range follows a multivariate Gaussian distribution(Ebden (2015)).

We expect that if our two varibale is close, their corresponding outputs should also have similar values either. Suppose we have an N dimensional Gaussian modeling $y_0 ... y_N$, its covariance matrix is Σ.To achieve this, we can choose a squared exponential kernel as the covariance function:

$$\kappa(x, x') = \exp\left(-\frac{(x - x')^2}{2}\right) \tag{4.4}$$

As the Figure.4.3 shows below, the output of this function is 1 when $x = x'$ and tends to be 0 when its arguments drift apart. Based on the function's property, we can build

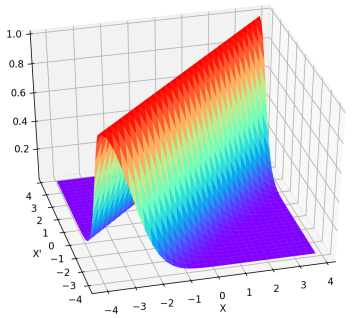


FIGURE 4.3: Image for Kernel function $\kappa(x, x')$

a model $p(y|x)$ using a multivariate normal:

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|m(\mathbf{x}), \mathbf{K}) \tag{4.5}$$

where $K = \kappa(x, x)$ and $m(x) = 0$.Now we denote our training data as x and its output is $y = f(x)$, and our test set is x_* and the predict data is $y_* = f(x_*)$.The key assumption in GP is that our data can be represented as a sample from a multivariate Gaussian distribution, we have

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{y}_* \end{pmatrix} \sim \mathcal{N}\begin{pmatrix} m(\mathbf{x}) \\ m(\mathbf{x}_*) \end{pmatrix}, \begin{pmatrix} \mathbf{K} & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{pmatrix} \tag{4.6}$$

where $\mathbf{K} = \kappa(\mathbf{x}, \mathbf{x})$,$\mathbf{K}_* = \kappa(\mathbf{x}, \mathbf{x}_*)$ and $\mathbf{K}_{**} = \kappa(\mathbf{x}_*, \mathbf{x}_*)$.As we mentioned before, the $m(x)$ and $m(x_*)$ are 0.According to the property of multivariate normal distribution, we have:

$$p(\mathbf{y}_*|\mathbf{x}_*, \mathbf{x}, \mathbf{y}) = \mathcal{N}(\mathbf{y}_*|\mu_*, \Sigma_*) \tag{4.7}$$

$$\mu_* = m(\mathbf{x}_*) + \mathbf{K}_*^T\mathbf{K}^{-1}(\mathbf{y} - m(\mathbf{x})) \tag{4.8}$$

$$\Sigma_* = \mathbf{K}_{**} - \mathbf{K}_*^T\mathbf{K}^{-1}\mathbf{K}* \tag{4.9}$$

where μ_* is the predict value.

As the multivariate gaussians can reflect local patterns of covariance between individual points, the combination of mulitiple Gaussians can represent non-linear relationships and is more flexible than the traditional parametric models.The gaussian process re-

gression works well on small datasets and can provide uncertainty measurements on the predictions.Since our dataset has only 1636 samples, it is suitable to use this method to predict the brain age.

4.4 Random Forest Regression

The Random Forest is one of the most effective machine learning models for predicting value.The basic idea of it is that it make predictions by combining different decision tree.So that each decision tree provide different insights into the relationship of the data.Then, I take the average of those decision trees' output as the final result.This process effectively bringing in the insights from all of them.The idea of averaging models is called ensembling.

First, we collect 1/5 training samples randomly to build a single decision tree.The selection of feature is also in random, which means we only choose k features to represent a sample. For any of a single decision tree,we find a partition point s of a feature A, and divide the dataset into $D1$ and $D2$ respectively to minimize the mean variance of each set D1 and D2.The equation has been shown below:

$$\underbrace{min}_{A,s}\left[\underbrace{min}_{c_1}\sum_{x_i\in D_1(A,s)}(y_i-c_1)^2+\underbrace{min}_{c_2}\sum_{x_i\in D_2(A,s)}(y_i-c_2)^2\right] \tag{4.10}$$

where $c1$ is the average label value for $D1$ and $c2$ is the average label value for $D2$.(Breiman (2001))

It is obvious that each tree would probably overfit terribly since their training sample is limited by random selection and their features are also sorted out randomly. But since they are all using different random samples, they all overfit in different ways on different things.Since the errors they get are not correlated with each other, they can still yield a good result at last.

The random forest regression yield the best performance among all the machine learning methods because it has many advantages below:

- Overcome overfitting.If we only build a single regression tree and do the prediction, it is extremely easy to overfitting since our model may learn some noise inside the data.
- Measuring the relative importance of each feature on the prediction is very easy with using random forest regression.
- It has few statistical assumptions.Unlike many other methods that assume the data obey gaussian distribution, random forest regression do not do any assumptions

Chapter 5

Deep Learning Method and Implementation

The machine learning method that mentioned in the chapter 4 has a large problem that the feature based on the intensity histogram overlooks the space information in the neuroimaging.However,it is extremely difficult to extract the space information by using the traditional feature engineering method.To tackle this problem, convolutional neural network(CNN) would provide a suitable architecture to inter image features automatically. Deep learning methods have recently made a good progress and have been applied in many fields like speech recognition, machine translation and automatic driving.Several different network architectures have been proposed, which have enabled to reach the state-of-art performance in many task like image classificationDeng et al. (2009). When properly trained, CNN have been proved to be invariant to several variability sources, such as contrast or rotation(Krizhevsky et al. (2012)).For this project, I modified a network architecture that is based on the Resnet(He et al. (2016)) to make it more applicable to regression problems.The implementation of the neural network is referenced by the work of Chen et al. (2019), who trained a 3D CNN to solve the tissue segmentation task and get a state-of-art performance by using it.

5.1 3D Convolutional Neural Network

A CNN is a special kind of artificial neural work that can preserve the spatial relationships in the image, with using a few connections between the layers.Because of it, CNN was proposed specifically to solve image related problems and get a good performance in many tasks.However, most of the CNN are applied in 2D images,and 2D CNN only can capture 2-dimensional spatial information, and neglect the information along the third dimension.To address this problem, 3D CNN is first proposed by Ji et al. (2012) to solve the human action recognition task.They extended the 2D CNN to a 3D CNN in both

space (2D) and time for video classification.Similar to video data (x, y, z), the extracted low level feature for MRI also has three dimensions$(182 * 218 * 182)$.The input will be arranged into a grid structure and be fed through multiple layers,each layers operate on a small region of the last layer.A common CNN is usually consisted of convolution layers, activation layers, Max pooling layers, batch normalisation layers and fully connected layers.(Figure.5.1)

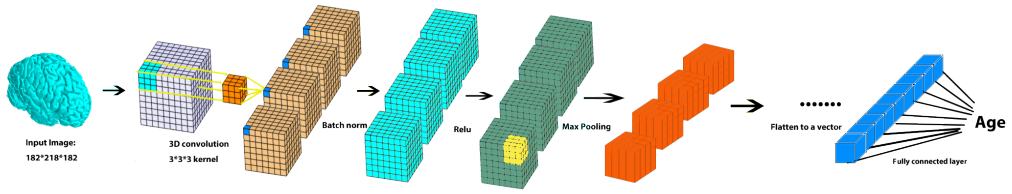


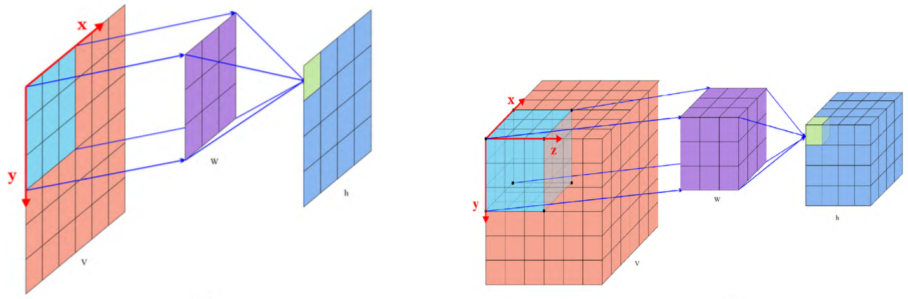
FIGURE 5.1: Overview of the 3D CNN architecture

5.1.1 3D Conv Layer

For convolutional layers, the feature map from the previous layers will be convolved with a set of filters called kernel to produce a new feature map.As for the 2D convolutional layer, the convolutional filter moves in 2-directions (x, y) to calculate low dimentional features from the image data.Therefore, the output feature map is still a 2D matrix.While for the 3D convolutional layer,3D kernels are convolved over 3D feature cubes in 3 directions (x, y, z) to get the feature maps.Figure.5.2has shown the difference between 2D convolution layer and 3D convolution layer clearly.The value of position (x, y, z) on the j^{th} feature map in the i^{th} layer can be computed from:

$$h_{x,y,z}^{i,j} = f((W_{i,j} * V^{i-1})_{x,y,z} + b_{i,j}) \tag{5.1}$$

where $W_{i,j}$ and $b_{i,j}$is the weights and the bias for j^{th} feature map respectively.V^{i-1} is the previous feature maps.$*$ denotes the convolution operation and f is the non-linear function.The weights of W and b can be optimised by a given loss function through backward propagation.



(a) 2D Conv (b) 3D Conv

FIGURE 5.2: Difference between 2D convolution and 3D convolution(Zou et al. (2017)

5.1.2 ReLU

Rectified Linear Unit(ReLU) a typical non-linear function f in the equation.5.1.It makes it possible for the neural network to approximate nonlinear function.From the point of mathematics, it defined as

$$ReLU(z) = max(0, z) \tag{5.2}$$

Compared with other activation functions like sigmoid function, deep convolutional neural networks with ReLUs can train several times faster than others. Furthermore, since ReLU is zero for all negative inputs, it is possible for any given unit to not activate, which means it is sparsely activated and more in line with human's neuro.

5.1.3 3D Batch Normalisation

Since the inputs to each layer are influenced significantly by the parameters of previous layers, small changes to the network parameters may amplify because the neural network is deep.It cause the problem that he layers must keep adapting to the new distribution.The input distribution to a learning system changes is called Internal covariate shift(Shimodaira (2000)).Ioffe and Szegedy (2015)proposed an algorithm called BN Transform to handle the problem. The batch normalisation layer can force the network to periodically change the previous output to zero mean and unit standard deviation, which can speed up the training and reduce the dependence on hyper parameters.

5.1.4 Max Pooling

After applying multi convolutional layers, each feature map is then feed into the pooling layer.The pooling operation takes small grid regions as its input and compute a number for each region like the convolution operation did before.The computed number is computed by using the max function.

5.1.5 Fully Connected Layer

As the flowchart shown in the Figure.5.1, after going through several convolutional layers, we will flatten the previous result to a long vector. The fully connected layer followed by it can yield the final scalar.It is the same as of the traditional multi-layer perceptron neural network(MLP).

5.2 ResNet

In order to make the neural network capture more features and give more accuracy, we often add more layers to the neural network. However, with the network depth increasing, accuracy gets saturated and then degrades rapidly due to the notorious problem of vanishing/exploding gradients(Bengio et al. (1994)).He et al. (2016) introduced the residual mechanism to solve the problem.Like the Figure.5.3 shown below, the author recast the original map into $F(x)+x$ rather than $F(x)$, where $F(x)$ is the residual function.The residual block can easily be implemented by introducing shortcut connections.

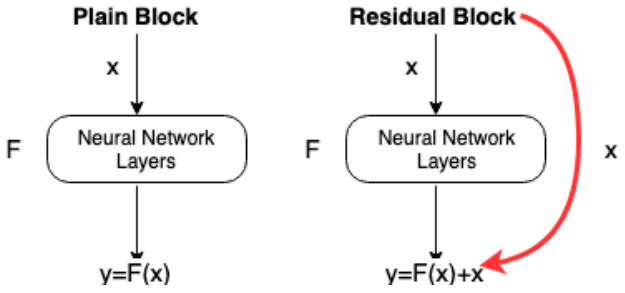


FIGURE 5.3: Residual blocks

The implementation of Resnet is inspired by the philosophy of VGG nets(Simonyan and Zisserman (2014)).Most of the convolutional kernel size are $3*3$.Besides, it obeys two design rules:for the same output feature map size, the layers have the same number of filters,and if the feature map size is halved, the number of filters will be doubled so that the time comlexity of each layer can be preserved.The neural network perform downsampling by convolutional layers with the stride of 2.

For the residual blocks of the network,if the input and output are of the same dimensions, the identity shortcuts (x) can be directly used and the residual block function should be

$$y = F(x, \{W_i\}) + x \tag{5.3}$$

While if their dimensions are different with each other,the shortcut still performs identity mapping, with extra zero entries padded with the increased dimension.A linear projection W_s should be added to match the dimensions:

$$y = F(x, \{W_i\}) + W_s x \tag{5.4}$$

Each ResNet block is either 2 layer deep or 3 layer deep.The 2-depth layer is called Basicblock, while the 3-depth layer is called BottleNeck. Figure.5.4 shows the two types of Resnet blocks respectively.

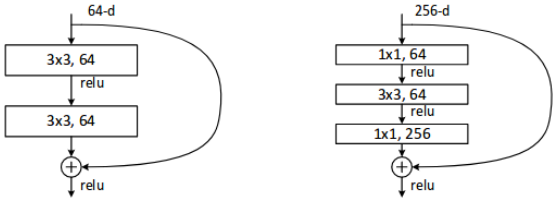


FIGURE 5.4: ResNet 2 layer and 3 layer Block

5.3 Architecture

5.3.1 Loss Function

Since the problem we have is the regression task,the most suitable loss function for it is the Mean Squared Error(MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{5.5}$$

where n is the sample number, Y_i is the predict value and $\hat{Y}_i$ is the ground truth value.

5.3.2 Optimiser

The optimiser used in the neural network is Adam proposed by Kingma and Ba (2014),which is designed to combine the advantages of Adagrad and RMSprop.The main function of it is to speed up the gradient descent.The learning rate of this model is set to 0.0001 in this project.

5.3.3 3D-Resnet

The Resnet introduced in Section.5.2 is mainly used for 2D images. While for the neuroimaing, the Resnet2D should be modified to make it to adapt to the current task.Figure.5.5 shows the whole architecuture of the 3D Resnet 18.The original shape of the input is the $182 * 218 * 182$.At first, we apply a convolutional operation on the raw image with the kernel size be set to $7 * 7 * 7$(with stride of 2 and padding of 3).Then the $3 * 3 * 3$ max pooling layers with the stride of 2 is followed by it.Then we go deepen into the residual blocks.It consists of four main layers and each layers contains two basic blocks.The channel number doubles when it go through a main layers and the image shape will get smaller correspondingly to infer a sufficiently rich representation of the brain.The basic block is mainly consists of a $3 * 3 * 3$ convolutional layer, a 3D batch normalisation layer, a rectified linear unit.With the shortcut connection given in the

basic block, the input is then be added to the output to do the residual map.I added an average pooling layer to the original architecture to make the shape be resized to $1 * 1 * 1$.Then we flatten it into a $512d$ vector. The final age prediction is computed by using a fully connected layer, which maps the output of the 512 vector to a single scalar.

All the results are presented in the Section 6.The best result was achieved by using a learning rate of 0.0001 on the Resnet34 with the input images are normalised in GMM method, which achieves the mean absolute error of 4.14 years eventually.The training process took 10 hours using four GPUs provided by University of Southampton.The implementation code is referenced from the work of Chen et al. (2019), who implemented a 3D Resnet neural network to do the tissue segmentation work.

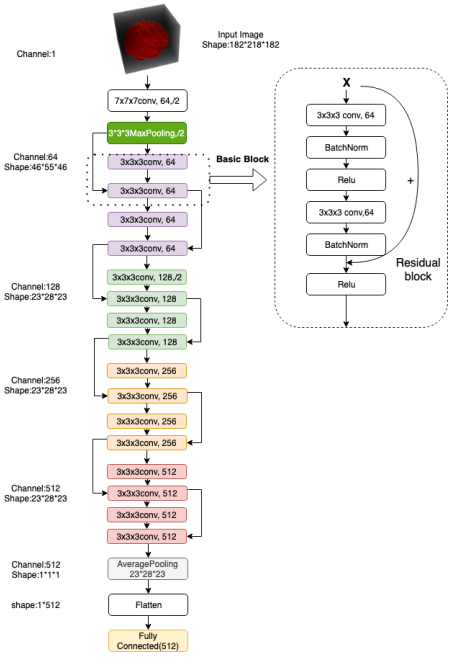


FIGURE 5.5: Architecture of 3D-Resnet18

layer	output size	10-layer	18-layer	34-layer	50-layer
conv1	91*109*91	7*7*7,64,stride2			
conv2	46*55*46	$3 \times 3 \times 3$ $maxpooling, stride2$			
		$\begin{bmatrix} 3*3*3, 64 \\ 3*3*3, 64 \end{bmatrix} \times 1$	$\begin{bmatrix} 3*3*3, 64 \\ 3*3*3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3*3*3, 64 \\ 3*3*3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1*1*1, 64 \\ 3*3*3, 64 \\ 1*1*1, 256 \end{bmatrix} \times 3$
conv3	23*28*23	$\begin{bmatrix} 3*3*3, 128 \\ 3*3*3, 128 \end{bmatrix} \times 1$	$\begin{bmatrix} 3*3*3, 128 \\ 3*3*3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3*3*3, 128 \\ 3*3*3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1*1*1, 128 \\ 3*3*3, 128 \\ 1*1*1, 512 \end{bmatrix} \times 4$
conv4	23*28*23	$\begin{bmatrix} 3*3*3, 256 \\ 3*3*3, 256 \end{bmatrix} \times 1$	$\begin{bmatrix} 3*3*3, 256 \\ 3*3*3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3*3*3, 256 \\ 3*3*3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1*1*1, 256 \\ 3*3*3, 256 \\ 1*1*1, 1024 \end{bmatrix} \times 6$
conv5	23*28*23	$\begin{bmatrix} 3*3*3, 512 \\ 3*3*3, 512 \end{bmatrix} \times 1$	$\begin{bmatrix} 3*3*3, 512 \\ 3*3*3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3*3*3, 512 \\ 3*3*3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1*1*1, 512 \\ 3*3*3, 512 \\ 1*1*1, 2048 \end{bmatrix} \times 6$
	1*1*1	average pooling 512d, fc			
Parameters number		14356033	33160513	63470145	46157121

TABLE 5.1: Detail architecture of 3D Resnet

Chapter 6

Result

In this chapter, I will display the performance of different models including linear regression, Gaussian regression, CNN and compare their performance by calculating their mean absolute errors.

6.1 Metrics

6.1.1 Mean Absolute Error

The mean absolute error can be defined as:

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j| \tag{6.1}$$

It is the most common method to measure difference between the predicted value and the ground truth value,where all individual differences have equal weight.

6.1.2 Pearson Correlation Coefficient

Pearson correlation coefficient represents the strength of the association between two variables.The value ranges from -1 to 1 ,which indicates the extent to which two variables are linearly related.The formula can be defined below:

$$PR_{XY} = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}} \tag{6.2}$$

6.1.3 R Squared

The pearson correlation coefficient explains the strength of the relationship between an independent and dependent variable, while R-squared explains to what extent the variance of one variable explains the variance of the second variable. The value of R-squared ranges from 0 to 1.

6.1.4 Root Mean Square Error(RMSE)

The formula of the RMSE is:

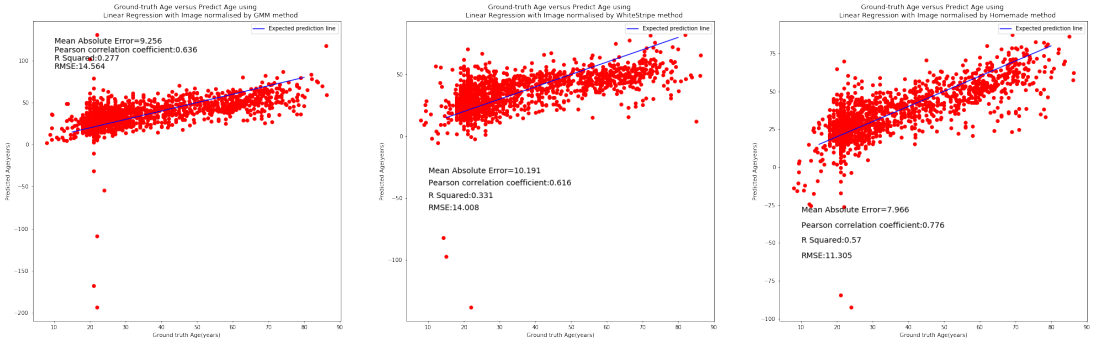
$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2} \tag{6.3}$$

RMSE is a quadratic scoring rule that also measures the average magnitude of the error.It is computed from the square root of the average of squared differences between prediction and actual observation. Compare with the mean absolute error, the RMSE gives a high weight to large errors,which means it is a better performance method when large errors are particularly undesirable.

6.2 Linear Regression Method Result

Linear regression is the first method I use to predict the individual's age.Figure.6.1shows the final distribution between predicted age and ground-truth age and table.6.1 compares the result of different normalisation method.It is obviously that simple linear regression cannot capture the feature very well and it may even yield some abnormal values such as -200 when predicting the age of young people.As a result of the bad model, the predicted value deviates greatly from the true value.The reason why linear regression fails is that the intensity histogram of the brain and the true age is not correlated linearly.

As for the normalisation methods applied in the brain image, homemade method proposed by Simon Jgous yield the best performance among all the method. Since the normalisation method fix the grey matter's intensity and white matter's intensity to a certain value at the expense of white and grey matter contrast, it is more easy for linear regression model to capture the relationship between age and grey matter and white matter.



(a) GMM (b) WhiteStripe (c) Homemade

FIGURE 6.1: Accuracy of Linear Regression Method with various normalisation method

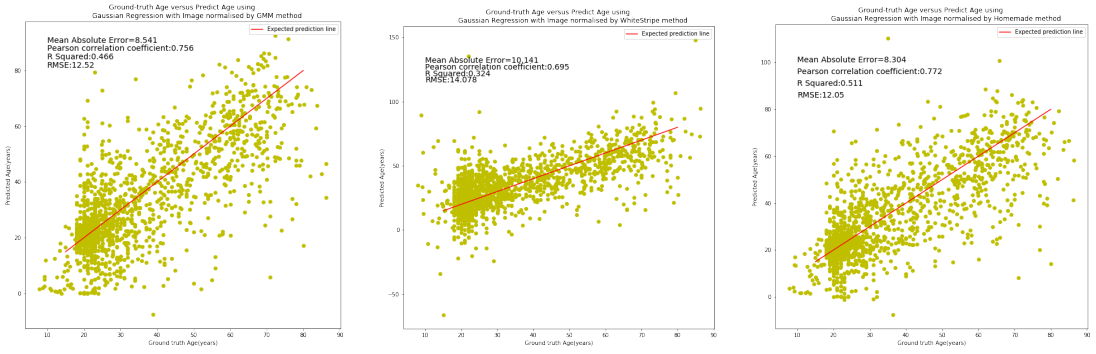
Linear Regression Performance				
Normalisation Method	MAE(years)	Pearson Correlation Coefficient	R^2	RMSE
GMM method	9.256	0.636	0.277	14.564
WhiteStripe method	10.191	0.616	0.331	14.008
Homemade method	7.966	0.776	0.57	11.305

TABLE 6.1: Performance of Linear Regression with different normalisation method

6.3 Gaussian Process Regression Method Result

The result of the Gaussian process regression method has been shown in the Figure.6.2 and Table.6.2.We can see that although Gaussian Process Regression did not improve the performance significantly compared to the Linear Regression method,it prevents the occurrence of abnormal values.

As for the normalisation method applied in the image preprocessing,the homemade method still has advantages compared with other two methods.The failure of the white stripe method is because it does not normalise the values to the same range.



(a) GMM (b) WhiteStripe (c) Homemade

FIGURE 6.2: Accuracy of Gaussian Regression Method with various normalisation method

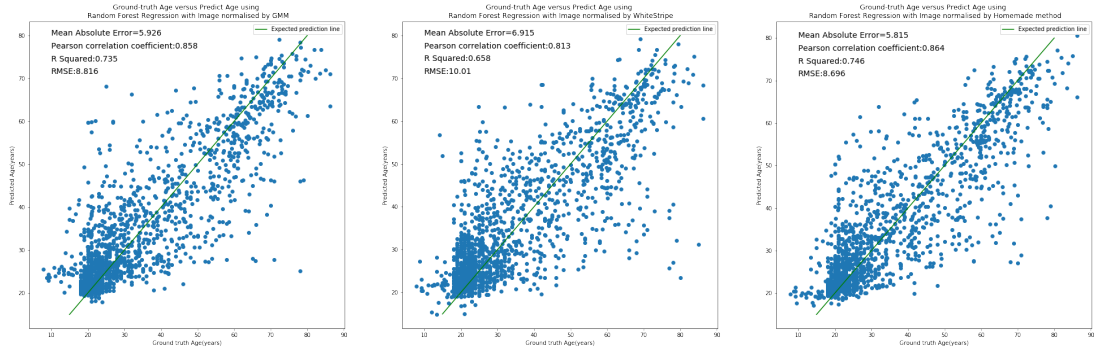
Gaussian Regression Performance				
Normalisation Method	MAE(years)	Pearson Correlation Coefficient	R^2	RMSE
GMM method	8.541	0.756	0.466	12.52
WhiteStripe method	10.141	0.695	0.324	14.078
Homemade method	8.304	0.772	0.511	12.05

TABLE 6.2: Performance of Gaussian Regression with different normalisation method

6.4 Random Forest Tree Result

Figure.6.3 and 6.3 displays the result of using random forest tree method.The performance of random forest tree method far exceeds the other two methods.The mean absolute error drop to only 5.8 years, which is only little worse than the performance of the state of art(4.16 years).Moreover, it can be trained in few seconds, while the deep learning methods often take one or two days to train a model with using four or more GPU. It can get a nice performance since it introduce a random process that can avoid overfitting sufficiently.But the problem of it is also very obvious that the RMSE is still so high, which means it often yield some large errors, especially for the old people.The phenomenon can be observed from the figure,where the dots are distributed very sprasely.

As for the normalisation methods,the performance of homemade method still surpass the two other methods due to the same reason that has been mentioned before.



(a) GMM (b) WhiteStripe (c) Homemade

FIGURE 6.3: Accuracy of Random Forest Regression Method with various normalisation method

Random Forest Regression Performance				
Normalisation Method	MAE(years)	Pearson Correlation Coefficient	R^2	RMSE
GMM method	5.926	0.858	0.735	8.816
WhiteStripe method	8.915	0.813	0.685	10.01
Homemade method	5.815	0.864	0.745	8.896

TABLE 6.3: Performance of Random Forest with different normalisation method

6.4.1 Feature Importance Measurement

The random forest tree method can also measure the importance of each feature by computing the mean decrease impurity.Since our features are represented by intensity histogram with bins are set to 100, the features' importance figure can tell us which is the most important component related with aging.

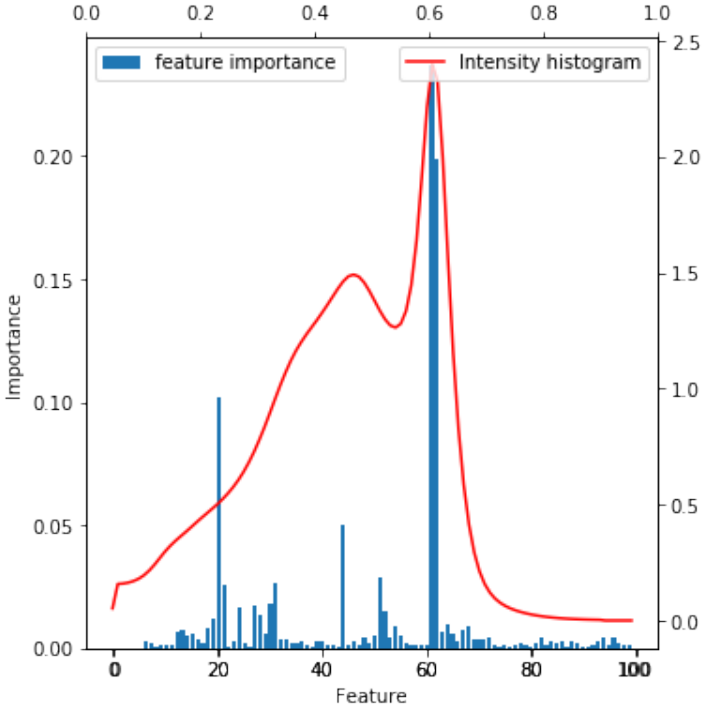
Figure .6.4 shows the feature importance and average of the intensity histogram from all training samples.The figure reveals that the intensity of the sixtieth voxel is the most important one among all the features.Coincidentally , the $60th$ feature represents the white matters in our brain like the Figure.6.4(b) shown below.The second important feature is the $20th$ feature, which corresponds to the intensity ranging from 0.285 to 0.315.Figure.6.4(d) shows the corresponding tissue of its cross-section image, which includes the boundary of the skull and CSF(see Figure.1.2). The third vital feature is the $42th$ feature, which corresponding to the intensity of the grey matter, which is also the first peak of the intensity histogram.Figure.6.4(c) proves the assumption with the cross-section image.

As the Section.1.2.2 mentioned before, the most significant change in our brain is that our white-matter and grey matter will shrink with ageing, while the CSF will dilate in the ventricle.This experiment further proves that this assumption is correct.

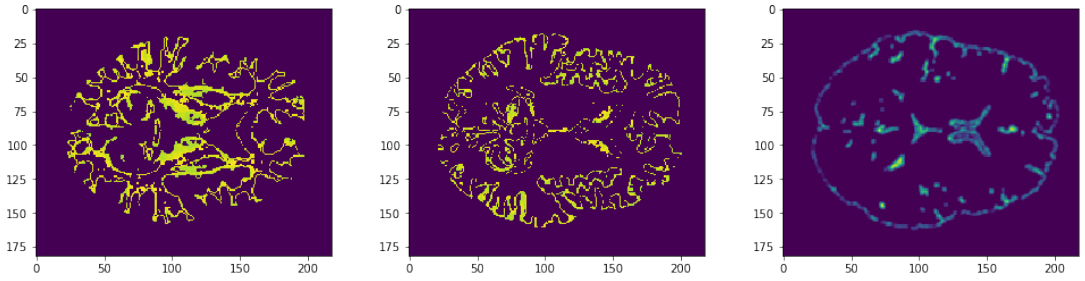
6.5 3D Resnet result

All of the machine learning methods overlook the space information contained in the neuroimaging although they may produce a descent result with only using intensity histogram information.The deep learning method compensates for this deficiency by applying convolution operation. Therefore, the best result in our experiment is achieved by using neural network method.

The performance of state-of-art is achieved by Cole et al. (2017), who used a VGG-like neural network on single tissues(i.e. GM or WM) to train a model to predict the brain age.The benchmark of state of art is given in the Table.6.4.The lowest MAE achieved was using single GM data (MAE=4.16 years) and the predictions based on raw data can achieve 4.65 years. However, the author did not publish his test dataset in his paper, where he choose 200 random samples from his data sets of 2001 samples. All of my datasets used in the project are included in the author's dataset(See Appendix in the paper of Cole et al. (2017)),while there are nearly 300 samples I have no right to access.Since my test set is not exactly the same as the author's test set, the final data given in this chapter may not accurately represent the real performance of the model.



(a) Feature Importance versus Average Intensity Histogram



(b) Image for Feature 60 (c) Image for Feature 42 (d) Image for Feature 20

FIGURE 6.4: Accuracy of Random Forest Regression Method with various normalisation method

Input data	MAE(years)	Pearson Correlation Coefficient	R^2	RMSE
GM	4.16	0.96	0.92	5.31
WM	5.14	0.94	0.88	6.54
GM+WM	4.34	0.96	0.91	5.67
Raw	4.65	0.94	0.88	6.46

TABLE 6.4: State of art Performance

6.5.1 Performance on Different Network Depth

Table.6.5 shows the performance on different depth of 3D Resnet with applying GMM normalisation method.It reveals that Resnet34 can achive the best performance among all the models since the network is deep enough to capture the aging-related features in the brain.However, if we use 3D Resnet50, the performance drops significantly to 14.509

due to the overfitting raised by too many parameters contained in the 3D Resnet50. Figure.6.5 compares their loss curve.It is obviously that the training loss of Resnet50 is lower than the Resnet34, while the test loss is much higher than the Resnet34.The phenomenon shown in the Figure.6.5 is a clear sign of overfitting.

Benchmark for 3D-Resnet with Different Depth				
Depth of Neural Network	MAE(years)	Pearson Correlation Coefficient	R^2	RMSE
3D Resnet10	4.895	0.913	0.83	7.23
3D Resnet18	4.387	0.93	0.865	6.311
3D Resnet34	4.193	0.949	0.897	5.841
3D Resnet50	14.509	0.212	-0.22	20.187

TABLE 6.5: Performance of 3D Resnet with different depth

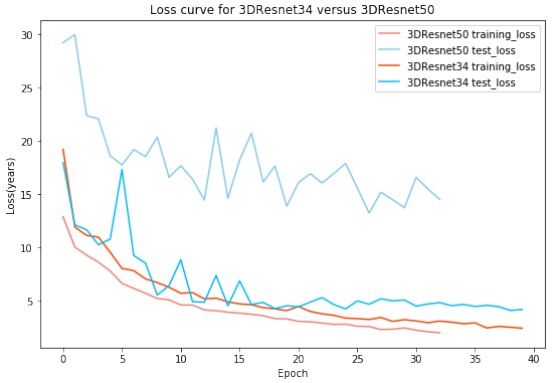


FIGURE 6.5: Loss Curve for resnet34 and resnet50

6.5.2 Performance on Different Normalisation Methods

Table.6.6 shows the performance on different normalisation methods with using Resnet34.Unlike the machine learning method result, the best normalisation method for neural network approach is the GMM method,which get 4.193 years error compared with the other methods. The value of R^2,RMSE and Pearson Correlation Coefficient also proves that the 3D Resnet34 model with using GMM method is robust and reliable.

The failure of the homemade method proposed by Jgou (2018) is mainly because the contrast information is compromised by the non-linear operation(See Equation.3.6),which was mainly used to fix the intensity of grey matter to 7.5.For machine learning methods, fixing the intensity value of both gray matter and white matter can effectively help us measure the amount of them.While for deep learning method, the internal contrast information is lost during this process.

The Zscore method failed mainly because there is no reference tissue in this method, which means the same tissue may have different intensity in different images.This adds extra difficulty to the neural network to identify tissues(See Figure.6.6).While the GMM method solve this problem very well by only fixing the white matter's intensity.

Benchmark for Neural Network with Using Different Normalisation Methods				
Normalisation Method	MAE(years)	Pearson Correlation Coefficient	R^2	RMSE
GMM method	4.193	0.949	0.897	5.841
Zscore method	7.794	0.93	0.727	9.243
Homemade method	5.11	0.92	0.839	6.957

TABLE 6.6: Performance of the Resnet34 with different normalisation methods

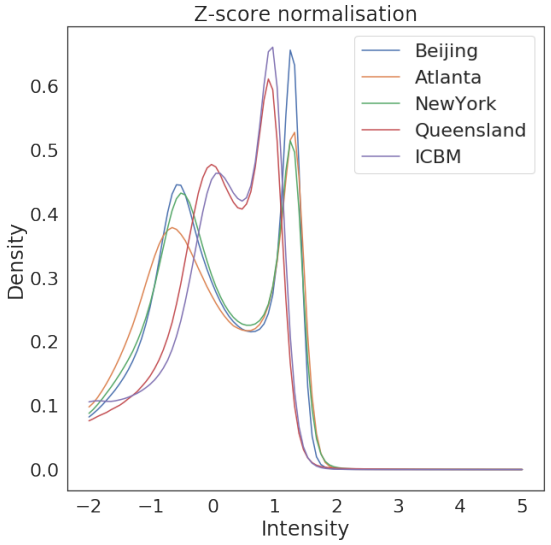


FIGURE 6.6: Result of Zscore Normalisation Method

6.5.3 3D Feature map visualisation

The neural network method often be regarded as a black box since features in the neural network are extracted automatically and often cannot be understood by human.To understand the mechanism of the neural network better, I visualise the neural network through by propagating the given input through the network to the required point and use a hook to intercept the feature maps as they are created.We plot the feature maps every time when we do a convolution operation.Figure.6.7 shows an example of the final feature map visualisation result.It is very clearly that when images at the shadow layer, we can recognize the brain easily.But with the layer going deepen, the extracted features become more and more abstract and obscure.

6.6 Summary

To sum up,the performance of machine learning is significantly lower than that of deep learning since machine learning methods cannot utilise the space information. Among all the methods of linear regression, gaussian regression and random forest regression, the linear regression yield the worst result due to the simplicity of the model.The gaussian regression did not improve the performance significantly as well. However, the random

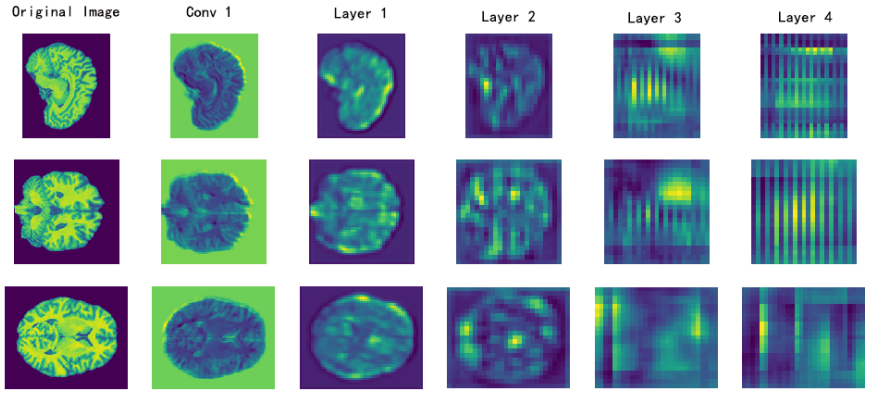


FIGURE 6.7: Feature map visualisation for 3D Resnet34

forest regression yield the best performance among all the adapted machine learning approaches.It solved the overfitting problem by introducing random process and obtained a good result.Furthermore, the random forest regression still provides the importance of each features, which fits our hypothesis very closely that the most important change related to aging in our brain is the volumn of white matter, that of grey matter and CSF.

As for the deep learning method, the result shown in the table above proves that our 3D Resnet outperform the traditional machine learning approach.The model performance improves with the depth of neural network until it reaches 50, where it starts to have very severe overfitting problem.

Furthermore, the normalisation method also plays an important role in the final performance of our model. The homemade method proposed by Jgou (2018) is more suitable for applying in the machine learning method, while the GMM method is a preference choice for the neural network method.

Finally, our model is outperform the state-of-art performance slightly with only using raw image data.However, since the test datasets we used are not exactly the same, there is not enough evidence to support this conclusion.

Chapter 7

Deployment

In this chapter, I will introduce the process of deploying the whole service,which can accept the uploaded neuroimaging and preprocess them automatically followed by feeding the image to neural network to get the final predicted age.The online tool has been deployed on the website `www.wzy-codify.com/age-predictor` and it accept *.nii* file with skull not been stripped.

7.1 Technology Stack

The frontend and backend of this service are completely separated.The communication between them is rely on the Ajax technique, which can make the service more flexible.

The frontend is build mainly with those techniques:

1. Vue.js.It is an frontend framework that features an incrementally adoptable architecture that focuses on declarative rendering and component composition.For this project, the webpage often need to be rendered according to the server-returned data, and Vue.js provides a mechanism for two-way binding, which makes the front-end development much easier.
2. ElementUI.It is an Vue 2.0 based UI component library.Using it will alleviate writing CSS style work at large.

The backend is build mainly build with techniques in the below:

1. Django Rest Framework.An open-source backend framework written in Python language.It provides a simple approach for building Web APIs.
2. Celery.It is a simple and reliable distributed system for scheduling tasks .

3. Redis.Redis is an in-memory data structure store, often used as a noSQL database, cache and message broker.For this project, it plays an role of message broker to distribute task.

4. MySQL.An open source database used to store records permanently.

7.2 Architecture

Figure.7.1 shows the architecture of the full service.The service is based on the website, where users can upload their neuroimaging.As for the server side, the whole process from image processing to age prediction is a very time-consuming operation(approximately 5 minute).Therefore, it is impossible for client to wait for the long time that may even exceed the maximum response time limit.Therefore, the server need to distribute the task to a worker and response the client's response at the same time.To achieve this, celery can play a role of coordinator to distribute tasks, while redis can act as a broker.When image preprocessing work and prediction work be completed in the background, the related record stored in the database will be changed at the same time.When they are working on the task, the client keep sending requests to server to check if the status is changed or not and change the client data synchronously with rendering the webpage according to the returned data.When the client end receive the final age from the server, it stop polling and render the corresponding age on the webpage.

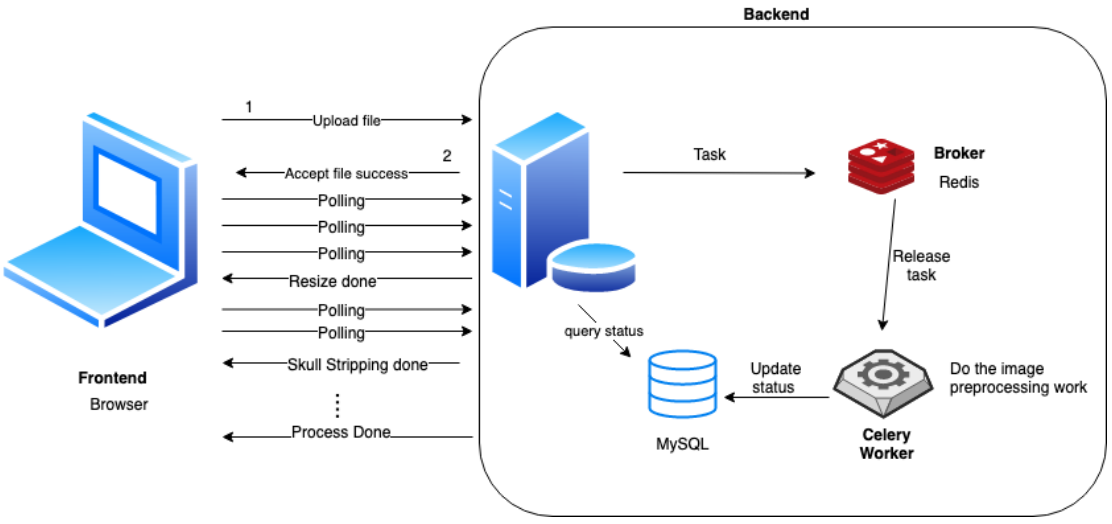


FIGURE 7.1: Architecture of the service

Chapter 8

Conclusion

In this chapter, I will give a conclusion to the whole project,discuss the main contribution I made and analyse the difficulties in this project.Then I will discuss the drawbacks of this project and propose the future work in the end.

8.1 Contribution

1. First, I reproduce the image preprocessing pipieline including image resesizing, skull stripping,n4 bias correction, template registration, and intensity normalisation.
2. Reimplement four kinds of normalisation and analyse their performance in detail.
3. Tried different machine learning methods for predicting the brain age and compared their performance
4. Revealed the most important factor asscoiated with aging using random forest regression
5. Modify and build a 3D Resnet to make it adaptable in our project
6. Train the neural network model from scratch and get the state-of-art performance
7. Deploy our project on the website to make it accessible to public

8.2 Difficulty

1. **Data Source Difficulty.** The amount of public MRI training set is very limitted and they are spread around different hospitals and universities.The process of

collecting dataset is not easy easy like the public competition like Imagenet. Morover, some of the collected data are "dirty"(missing value and abnormal value), and they need to be cleaned at first.

2. **Difficulty in choosing Image Preprocessing Tool.** The image preprocessing is very sophisticated and has 5 steps intotal.For each step,there are also a large number of open source tools available for doing it.Different tools may yield different result, which may influence the subsequent process significantly.Therefore, choosing a suitable tool or algorithm is a vital and difficult task.To solve it,I compared different tools one by one and test their performance to sort out a suitable tool for it.

3. **Preprocessing Time Difficulty.** Some of the steps such as n4 bias correction and template registration are very time-consuming and space-consuming.It may take more than 5 minutes to process a single file.Therefore, processing all of neuroimaging in one machine may take more than 7 days.How to process them parallel on several servers is a big challenge.

4. **Normalisation Implementation Difficulty.** I introduced five normalisation methods in the paper and I implement them all from scratch, which means I need to understand them fully at first. Some parameters are also need to consider carefully such as the bandwidth used in kernel density estimation of white-stripe normalisation.

5. **Neural Network Selection Difficulty.** The state-of-art neural network is based on the VGG architecture.While how to choose and implement my own neural network is a difficult problem for me.The work of Chen et al. (2019) inspired me a lot, who used Resnet on the 3D image to solve the tissue segmentation task very well. I modified its architecture a lot to make it suitable for the prediction task.

6. **Training neural network Difficulty.** Training a neural network from scratch is very time-consuming and the performance of the result is very rely on the selection of the hyper-parameters.For example, to decide the learning rate, I tried different values from 0.01 to 0.00001 and observed their performance from loss curve.Then, I choose 0.0001 as the best learning rate. However, as I mentioned before, the training data are very large and it may take more than 1 day to train a model. Therefore, the best results are not easy to obtain.

7. **Deployment Difficulty.** There are many unexpected things happened when I deploy my solution on the server.It often occurs that codes running on the local development enviorment very well but crashed on the production server. Unlike the universities' computing engine, the memory of the web server are often memory leaks due to the shortage of the memory.It is may even hard for the web server to load the state dict of neural network's t to the memory. Apart from extending

the server's RAM, setting swap space is also an effective solution, which utilise the server's disk as its virtual memory.

8.3 Drawback

1. **Datasets Problem.** It is a pity that I cannot have the same datasets with the state of art's work(Cole et al. (2017), although most of his training data are covered in my dataset.Moreover, the author did not publish his code either, which cause that I cannot even use his model to evaluate its performance based on my test dataset.

2. **Simplicity in the manual features for machine learning** For this project, I only use the intensity histogram as the input feature.The features are too simple that it cannot reveal any space information in the brain.

3. **Did not segment WM, GM and CSF separately** In the state-of-art work, the author segmented the brain into white matter and grey matter and feed them separately into the neural network and get the best performance on the white matter case, which is an efficient way to get rid of the influence raised by irrelevant tissues.

4. **Did not try more Neural Network Visualisation methods** There are many approaches for visualising the convolutional nerual network including layer activations visualisation, conv/fc filters visualisation, and heatmaps of class activations.However, I only tried the layer activation method in this project and did not get any useful information from the image of deep features .

5. **The service often crash due to the shortage of memory** Since 3D convolution operation need to take a large memory, the server often crash when it step into the final prediction stage.

8.4 Future work

For the drawbacks mentioned above, I want to introduce the future work if I have more time on the project

1. **Introduce more data to the training samples.** I found more accessible data source such as ANDI dataset(http://adni.loni.usc.edu/data-samples/access-data/)when the project almost completed.So, the next step is to absorb those data into my training dataset.Furthermore, I will apply for more neuroimaging from institutes to rich our datasets since the robustness of model is largely depends on the amount of training dataset.

2. **Construct more manual features for neuroimaging** For machine learning method, We can utilise the shape of white matter ,grey matter and CSF as its important features since we already know that the brain age is strong related with the grey matter and white matter.With the shape of them, we can try Hu moments(Hu (1962)) or Fourier descriptor to represent its shape due to their good property of translation invariants, scale invariants and rotation invariants.

3. **Segment WM, GM and CSF separately** Since the state-of-art get its best performance with only using white matter, the priority thing for me is to segment those tissues and feed them into the convolutional neural network to check their performance. The tissue segmentation task can be achieved by training a neural network like we introduced in the Section.3.2 before.

Bibliography

Yoshua Bengio, Patrice Simard, Paolo Frasconi, et al. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Sihong Chen, Kai Ma, and Yefeng Zheng. Med3d: Transfer learning for 3d medical image analysis. *arXiv preprint arXiv:1904.00625*, 2019.

James H Cole, Rudra PK Poudel, Dimosthenis Tsagkrasoulis, Matthan WA Caan, Claire Steves, Tim D Spector, and Giovanni Montana. Predicting brain age with deep learning from raw imaging data results in a reliable and heritable biomarker. *NeuroImage*, 163:115–124, 2017.

Stuart Currie, Nigel Hoggard, Ian J Craven, Marios Hadjivassiliou, and Iain D Wilkinson. Understanding mri: basic mr physics for physicians. *Postgraduate medical journal*, 89(1050):209–223, 2013.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Raunak Dey and Yi Hong. Compnet: Complementary segmentation network for brain mri extraction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 628–636. Springer, 2018.

Mark Ebden. Gaussian processes: A quick introduction. *arXiv preprint arXiv:1505.02965*, 2015.

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.

Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE transactions on information theory*, 8(2):179–187, 1962.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Ivan Itzcovich. Deepbrain. `https://github.com/iitzco/deepbrain`, 2018.

Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved optimization for the robust and accurate linear registration and motion correction of brain images. *Neuroimage*, 17(2):825–841, 2002.

Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012.

Simon Jgou. A machine learning survival kit for doctors. `https://medium.com/owkin/a-machine-learning-survival-kit-for-doctors-97982d69a375`, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

JB Antoine Maintz and Max A Viergever. A survey of medical image registration. *Medical image analysis*, 2(1):1–36, 1998.

Michael Hanke Marc-Alexandre Ct Ben Cipollini Paul McCarthy Matthew Brett, Chris Markiewicz and Chris Chen. Coordinate systems and affines. `https://nipy.org/nibabel/coordinate_systems.html#the-affine-as-a-series-of-transformations`, 2018.

Prabhu Ramachandran and Gaël Varoquaux. Mayavi: 3d visualization of scientific data. *Computing in Science & Engineering*, 13(2):40–51, 2011.

Jacob C Reinhold, Blake E Dewey, Aaron Carass, and Jerry L Prince. Evaluating the impact of intensity normalization on mr image synthesis. In *Medical Imaging 2019: Image Processing*, volume 10949, page 109493H. International Society for Optics and Photonics, 2019.

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.

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.

Russell T Shinohara, Elizabeth M Sweeney, Jeff Goldsmith, Navid Shiee, Farrah J Mateen, Peter A Calabresi, Samson Jarso, Dzung L Pham, Daniel S Reich, Ciprian M Crainiceanu, et al. Statistical normalization techniques for magnetic resonance imaging. *NeuroImage: Clinical*, 6:9–19, 2014.

N Varuna Shree and TNR Kumar. Identification and classification of brain tumor mri images with feature extraction using dwt and probabilistic neural network. *Brain informatics*, 5(1):23–30, 2018.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Elizabeth R Sowell, Bradley S Peterson, Paul M Thompson, Suzanne E Welcome, Amy L Henkenius, and Arthur W Toga. Mapping cortical change across the human life span. *Nature neuroscience*, 6(3):309, 2003.

Nicholas J Tustison, Brian B Avants, Philip A Cook, Yuanjie Zheng, Alexander Egan, Paul A Yushkevich, and James C Gee. N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6):1310, 2010.

Thomas Wolfers, Alberto Llera Arenas, A Marten H Onnink, Janneke Dammers, Martine Hoogman, Marcel P Zwiers, Jan K Buitelaar, Barbara Franke, Andre F Marquand, and Christian F Beckmann. Refinement by integration: aggregated effects of multi-modal imaging markers on adult adhd. *Journal of psychiatry & neuroscience: JPN*, 42(6):386, 2017.

Mark W Woolrich, Saad Jbabdi, Brian Patenaude, Michael Chappell, Salima Makni, Timothy Behrens, Christian Beckmann, Mark Jenkinson, and Stephen M Smith. Bayesian analysis of neuroimaging data in fsl. *Neuroimage*, 45(1):S173–S186, 2009.

Liang Zou, Jiannan Zheng, Chunyan Miao, Martin J Mckeown, and Z Jane Wang. 3d cnn based automatic diagnosis of attention deficit hyperactivity disorder using functional and structural mri. *IEEE Access*, 5:23626–23636, 2017.