Evaluation and Training Metrics for Medical Image Segmentation

by

Maddalena Andreoli Andreoni
Bachelor of Engineering in Computer Engineering

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Chicago, 2020

Chicago, Illinois

Defense Committee:
Prof. Piotr Gmytrasiewicz, Chair and Advisor
Prof. Natalie Parde
Prof. Pier Luca Lanzi, Politecnico di Milano

TABLE OF CONTENTS

CHAPTER **PAGE**

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

MRI	Magnetic Resonance Imaging
HCA	Hierarchical Cluster Analysis
FCN	Fully Convolutional Network
CNN	Convolutional Neural Network
BRATS	Brain Tumor Image Segmentation Benchmark
GAN	Generative Adversarial Network
T1	T1-weighted Magnetic Resonance Image
T1c	T1-weighted, contrast-enhanced Magnetic Resonance Image
T2	T2-weighted Magnetic Resonance Image
FLAIR	Fluid-Attenuated Inversion Recovery Magnetic Resonance Image
DBSCAN	Density-based spatial clustering of applications with noise
GLCM	Gray-Level Co-occurrence Matrix
PCA	Principal Component Analysis

SUMMARY

This thesis will define and develop twenty metrics that are computed over tumor segmentation in medical MRI images and that aim to describe such segmentation in an objective and quantitative way. It will proceed to compute each metric over the Brain Tumor Image Segmentation Benchmark dataset to show their descriptiveness over real data. Subsequently it will attempt classification of tumor grade label using the defined metrics.

CHAPTER 1

INTRODUCTION

In recent years, medical image analysis has undergone considerable increase in importance in the diagnosis process. The huge amount of gathered imaging data and the advancements in automatic image processing have radically transformed how medical images are used for both research and diagnostic purposes: with the increase in computational capabilities of computers, able to process large amount of data in a very short period of time, it has been made clear that a personalisation of medical treatments, tailored to the individual, is possible with the aid of automated tools. Simultaneously, this amount of new information highlights the need for such automated tools to support and work side by side with medical professionals during the clinical process. Indeed, many clinical applications nowadays require computer-aided diagnosis systems, such as diabetes inspection, surgical planning and cancer diagnosis [2].

Expert systems have been used since their conception for image processing, medical image processing included [3]. Edge and line detection filters, region growing and similar low-level pixel processing methods were used since the mid-Seventies to construct rule-based systems that could highlight sections of image for the purpose of human inspection. More interestingly, supervised machine learning algorithms have been used for various medical imaging tasks since their development in the 1990s. Exam and object/lesion classification, organ or lesion detection, object segmentation, content-based image retrieval and image enhancement are all medical tasks that have been undertaken successfully by various machine learning algorithms in recent years [3].

The segmentation of organ and lesion areas in particular has become a field widely studied in computer science, as manual segmentation annotation is a tedious and subjective process when carried out by human experts. Segmentation, defined as the task of labeling only and all the pixels belonging to the desired object, requires both a rough localisation of the object, finding its bounding box within the whole picture, and a precise detection of its boundaries. This second task is particularly critical for medical image segmentation: the segmented objects are often extremely variable in size, shape and location from patient to patient, even healthy ones (such is the case, for example, for pancreas segmentation [4]). A correct segmentation becomes especially critical when it's used for surgical planning.

For these reasons, the development of automatic medical image segmentation will most likely support the diagnosis process with increasing frequency in the near future [5]. In this context, it's becoming more and more important to define quantitative and rigorous approaches to evaluate the performances of such algorithms, as well as to provide medical professionals, with no knowledge of machine learning, understandable metrics to describe their results.

In this thesis, we develop a set of metrics apt to describe quantitatively aspects of a segmented lesion that were before only qualitatively described. Such metrics aim to describe both the distribution of intensity within a segmentation using various definitions of homogeneity, and the shape of the segmentation by defining various measures that quantify object symmetry and regularity.

Furthermore, we perform an experimental analysis of the effectiveness of the developed metrics in correctly classifying tumor grade by studying their behaviour and distribution, and by training an out-of-the-box decision tree classification algorithm to correctly predict new data.

This thesis will be divided as follows: Chapter 2 will briefly review the current state of the art in clustering and image segmentation algorithms. Chapter 3 will define each developed metric in detail; chapter 4 extensively describes the utilised dataset and explains our implementation choices. Finally, Chapter 5 showcases our experimental results.

CHAPTER 2

STATE OF THE ART

In this chapter, we will present an overview of the most common algorithms and methods to perform clustering and image segmentation. Specifically, Section 2.1 will cover the three most common clustering algorithms, i.e. hierarchical clustering, k-means and density-based clustering. Section 2.2, on the other hand, will introduce first some non-machine learning approaches to image segmentation, focusing in particular on medical image segmentation; then, we will bring attention in Section 2.3 to the current state of the art in medical image segmentation, which is obtained through machine learning algorithms (specifically fully convolutional neural networks).

2.1 Cluster Analysis

In its most intuitive definition, cluster analysis (or clustering) is the unsupervised task of finding a set of groups (or clusters) in a dataset, so that objects belonging to the same group are similar and objects belonging to different groups are different according to some similarity measure.

The definitions of this similarity measure, as well as those of what constitutes a cluster, are many and have given rise to numerous algorithms. Here we'll present an overview of some that we've taken into consideration as preprocessing step for our task.

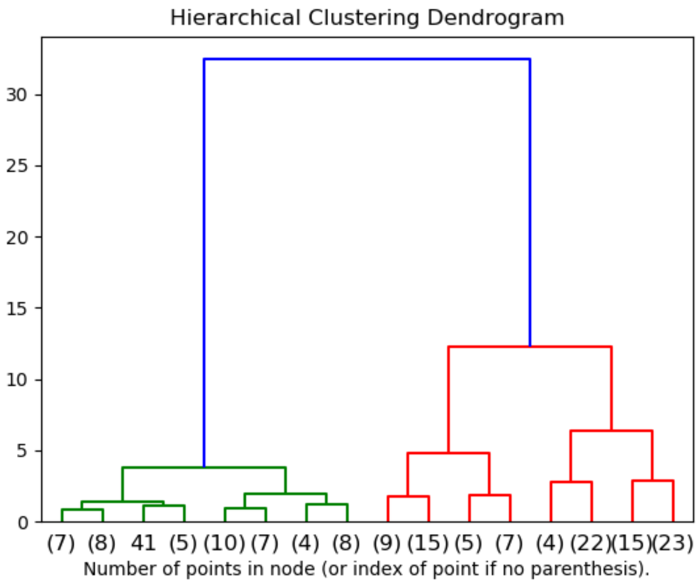


Figure 1: Example of the Dendrogram of a Hierarchical Clustering Algorithm [1].

2.1.0.1 Hierarchical Clustering

Hierarchical Cluster Analysis (HCA) [6] is a greedy approach to clustering based on the idea that observation points spatially closer are more likely related than points spatially farther away. A distance matrix between each point in the dataset is computed, based on a chosen distance metric (the most common are Euclidean and Manhattan distance, Maximum distance, L_r norm).

Then, HCA computes a dendrogram that is a hierarchy of nested clusters, where the leaves are single-point clusters, and the root is a cluster of all points in the dataset, as seen in Figure Figure 1. A linkage criterion determines arcs between the nodes of the dendrogram, defining a distance between two clusters and determining whether they should be merged together. HCA algorithms can be divided in two rough sub-categories depending on the linkage computation: methods which do not require a cluster center to be specified (i.e. single link or nearest neighbor, complete link, weighted and unweighted average), and methods that make use of a cluster center computation (such as centroid, median and minimum variance) [7]. The main disadvantage of HCA is its complexity, which is $\mathcal{O}(n^3)$ time-wise and $\mathcal{O}(n^2)$ memory-wise, with n being the number of samples in the dataset.

2.1.0.2 Centroid-based Clustering

Centroid-based Clustering represents clusters as a central vector. The most common centroid-based algorithm is the k-means clustering, where the number of clusters k is a parameter of the algorithm. The algorithm requires an initialization step, which can use either the Forgy method (k data points from the dataset are chosen at random as initial means) or the Random Partition method (each data point is assigned to a random mean) [8]. Then, it iteratively assigns each point of the dataset to a cluster, based on Manhattan distance between the point and the mean [9], and recomputes the means. There are two main limitations to k-means clustering: the first is that as an algorithm it requires to know a priori the number of clusters k. The second is that the cluster model requires spherical, separable clusters to be efficient.

2.1.0.3 Density-Based Clustering

Density-based clustering is based on the assumption that the considered dataset is a sample from an unknown probability density [10]. Clusters are then defined as high-density areas, which are computed by defining a local density estimate (usually nearest neighbors) and a distance

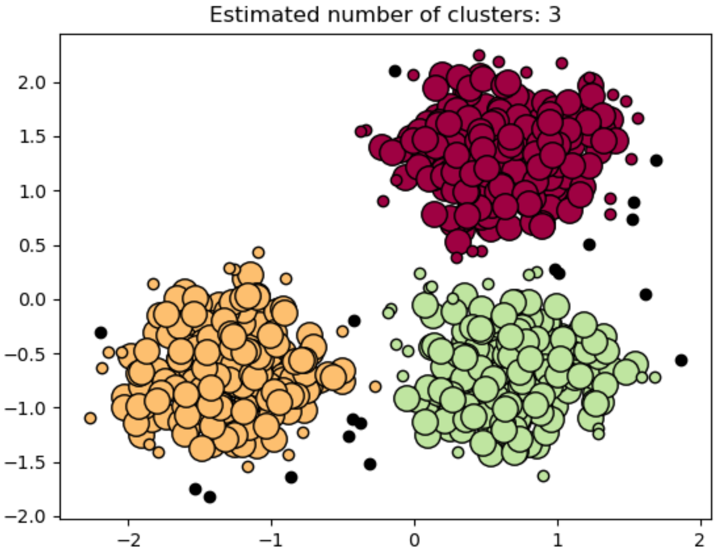


Figure 2: Density-Based Clustering using DBSCAN algorithm [1].

metric between points. Figure 2 shows an example using the common DBSCAN algorithm [11], where large circles identify core samples of a cluster and black points are noise points.

2.2 Medical Image Segmentation

As a computer vision task, semantic image segmentation is defined as the process of predicting a class label at each pixel in an input image [12]. This process aims to define specific objects within the image, or object boundaries. In medical applications, image segmentation is a fundamental part of medical image analysis (together with classification and abnormality detection) and is used to aid radiologists and clinicians in the diagnosis process [5].

While the generalized semantic segmentation task may apply to innumerable kinds of images, medical image analysis focuses on a few imaging modalities (CT, ultrasound, MRI, X-ray) [13] that medical segmentation algorithms need to specialize on.

This section will briefly expound on non-machine learning methods for semantic segmentation applicable to a medical domain; later on, we will go into details of the current Deep Learning State of the Art for image segmentation.

2.2.1 Otsu's Thresholding

Otsu's Thresholding Method [14] is an automatic process that finds the optimal single threshold that assigns the pixels in the image to either one of two classes: Foreground or Background.

At its core, Otsu's method consists in computing the intra-class variance for each possible threshold (i.e. every intensity of the image), and then choosing the threshold that minimizes this variance:

$$\arg\max_{t}(\sigma_w^2(t) = w_0(t)\sigma_0^2(t) + w_1(t)\sigma_1^2(t)) \tag{2.1}$$

The drawback of this computation is that it ensures good performances only if the histogram of intensities of the image has a bimodal distribution; that is, only if there is a deep valley between two peaks, as seen in Figure 3. Otsu's method has been improved by successive studies, developing 2D- and 3D-Otsu methods, which consider, together with the intra-class variance, also the pixel's spatial neighborhood information. However, naive 2D-Otsu has a time complexity of $\mathcal{O}(N^4)$ (N being the number of pixels), and 3D-Otsu a complexity of $\mathcal{O}(N^6)$.

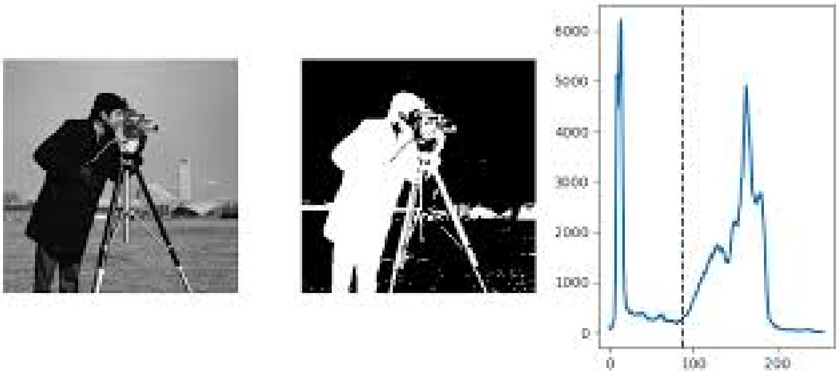


Figure 3: Example of Otsu's Thresholding method [1]

Feng et al. [15] have developed a Fast 3D-Otsu algorithm specifically for medical image segmentation that also allows for multilevel thresholding segmentation. Their method either is comparable or outperforms the most common multilevel thresholding methods, such as Particle Swarm Optimization, Bacteria Foraging Optimization, Adaptive Bacterial Foraging and Real-coded Genetic Algorithm, and can be used for automatic region of interest (ROI) extraction, and organ volume calculation.

2.2.2 Region Growing

Similar to a thresholding method, in that it requires a threshold to the average color difference between pixels [16], Region Growing (also called Region Merging) iteratively examines and merges neighboring regions whose average color difference is lower than the threshold, starting from pixels adjacent to the initial seed points.

(a) (b)

Figure 4: Canny operator applied to perform edge detection. Source: Wikipedia

2.2.3 Edge Detection

Identifying object or region boundaries is an intuitive way of performing image segmentation: boundaries, or edges, usually correspond to an abrupt change in either color, intensity and/or texture in an image. We define an edge as the location of a *rapid intensity variation* [16], which can be located through a derivative computation. Several methods of edge detection have been developed over the years using first-order and second-order operators.

First-order operators compute the first-grade derivative of the image and then search for the local maxima of the gradient magnitude to identify the edges. Since derivative computation accentuates high frequencies, it is very sensitive to noise, therefore first-order edge detection is usually computed over a smoothed version of the image. The best-known first-order edge detector is the Canny filter (seen in Figure 4), which computes the convolution of the image with the first-order derivative of a two-dimensional Gaussian in a direction $\boldsymbol{n}$ [17]:

$$G = exp(-\frac{x^2 + y^2}{2\sigma^2}) \tag{2.2a}$$

$$G_n = \frac{\delta G}{\delta \mathbf{n}} = \mathbf{n} \cdot \Delta G \tag{2.2b}$$

Given the Gaussian operator G_n, an edge point is a local maximum in the direction $\boldsymbol{n}$ of the operation G_n applied to an image I, i.e.

$$\frac{\delta}{\delta \mathbf{n}} G_n * I = 0 \tag{2.3a}$$

which is equivalent to

$$\frac{\delta^2}{\delta^2 \mathbf{n}} G * I = 0. \tag{2.3b}$$

Another common first-order operator is the Sobel operator [16]:

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * I \text{ and } \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I \tag{2.4a}$$

where the gradient magnitude at each point is computed as:

$$G = \sqrt{G_x^2 + G_y^2}. \tag{2.4b}$$

Second order operators, on the other hand, compute the second-order derivatives of the intensity to detect rate of change in the intensity. The most widely-used operator is the Laplacian of the Gaussian, or Marr-Hildreth algorithm, with the following convolution kernel which is applied to the image: [16]

$$\Delta^2 G_\sigma(\mathbf{x}) = \frac{1}{\sigma^3}(2 - \frac{x^2+y^2}{2\sigma^2})exp(-\frac{x^2+y^2}{2\sigma^2}). \tag{2.5}$$

2.3 Machine Learning in Medical Image Segmentation

The State of the Art for medical image segmentation in the last few years has been generated nearly exclusively by Deep Learning-based methods [2, 5, 18].

Fully Convolutional Neural Networks (FCN) are one of the most common and efficient architectures used for Semantic Segmentation, and they reach state of the art accuracy in general domain tasks [19]. They accept arbitrary-sized input, and are translation invariant. Each layer type in a FCN (convolution, pooling, ReLU) depend only on relative spatial coordinates.

We call $\mathbf{x}_{ij}$ the data feature vector at location (i, j) in a given layer, and $\mathbf{y}_{ij}$ the vector in the next layer. Then the next layer's feature vector is computed by:

$$\mathbf{y}_{ij} = f_{ks}(\{\mathbf{x}_{si+\delta i, sj+\delta j}\}_{0 \leq \delta i, \delta j \leq k}) \tag{2.6}$$

where k is the kernel size, s is the stride, and f_{ks} is the layer function.

In a fully convolutional network, all layers obey the trasformation rule

$$f_{ks} \circ g_{k's'} = (f \circ g)_{k'+(k-1)s',ss'}. \tag{2.7}$$

One of the major advantages of using a FCN, or a Deep Learning approach in general, is that it does not require hand-crafted features to be accurate. This is particularly critical in a domain such as medical image analysis where clinical experts have little to no knowledge of Deep Learning and Deep Learning experts similarly have little knowledge of the clinical domain [5]. FCNs automatically learn meaningful features directly from the raw data.

Fully convolutional networks have been introduced in Medical Image Segmentation by Ronneberger et al. in 2015 with U-Net [20]. U-Net is a 2D encoder-decoder architecture where excessive data augmentation was used to obviate the lack of annotated training data. U-Net applies elastic deformations to the available training images, in order to expand the training corpus and teach the network deformation invariance, which is especially important in medical domains as tissue deformation is one of the most common variations in medical images.

Havaei et al. [21] also use a convolutional neural network (CNN) as architecture for the segmentation of glioblastomas on the BRATS 2013 Dataset [22]. For each MRI slice, they predict tumor segmentation using two CNN of different kernel size. The so-called "local pathway" has a kernel of 7 × 7, while the "global pathway" of 13 × 13. This is so the pixel label will be predicted using both local information and regional information, i.e. where roughly the pixel is located inside the brain.

Moeskops et al. [23] attempt to generalise the Medical Image Segmentation problem by training a singular instance of a CNN architecture to recognize and segment different tissues in brain MRI images, pectoral muscle tissue in breast MRI, and coronary arteries in cardiac CTA. They first trained the same architecture separately in the three different tasks, then in two out of three tasks, and then they trained it in all three tasks at the same time. Their results show that this kind of generalized CNN has dice score comprarable to that of a specialized CNN for each of the three tasks.

Roth et al. [4] apply a CNN architecture to pancreas segmentation, which had been previously one of the hardest medical segmentation tasks due to high variability of shape, location and size from patient to patient. Their approach is coarse-to-fine, using a pruning algorithm to select a Region of Interest on which to iteratively apply the CNN architecture, and they reach State of the Art for pancreas recognition and segmentation.

All of the methods illustrated above perform the segmentation slice by slice, and then fuse the 2D segmentation result to obtain a 3D volumetric segmentation. This is computationally efficient, since 3D convolution has proven slow to draw inferences [5] and its network size and parameters number are prohibitive. However, a drawback of 2D segmentation is that it does not fully exploit the 3D context of volumetric data. Li et al. [2] formulates a coarse-to-fine pancreas segmentation framework in which a first FCN (*ResDSN Coarse*) roughly localises the organ in question, then a second FCN of smaller kernel overlap size (*ResDSN Fine*) precisely performs segmentation using the full 3D sub-region identified by *ResDSN Coarse*.

2.3.1 Generative Adversarial Networks

In 2014, Goodfellow et al. introduced a novel deep generative model called GAN [24]. GAN's framework is formulated as a minmax two-player game between a *generative network G*, tasked with capturing the data distribution, and a *discriminative network D* trained to discern whether a data sample belongs to the ground truth or is generated by G.

Formally, given $V(G, D)$ the game's value function:

$$\min_G \max_D V(G, D) = \mathbb{E}_{x \sim p_{\text{data}(x)}}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]. \tag{2.8}$$

The two networks, in the case of [24], are multi-layered perceptrons, and the results are backpropagated in the next iteration of training.

Drawing from Goodfellow's GAN framework, Luc et al. [12] apply Adversarial Networks to semantic segmentation. They use CNNs to model both the generative and the discriminative networks, and apply a large Field of View (FOV) and a small FOV for training. They find that adversarial training improves spatial accuracy for the segmentation labels.

An application of GANs to medical image segmentation has been proposed by Xue et al. with SegAN [25]. They use a FCN as the generative model and develop a multi-scale loss function used for both the generative and discriminative model. They test their framework on the BRATS 2013 dataset [22] and achieve state of the art performance.

CHAPTER 3

METRICS DEFINITION

This chapter will delve into our problem definition, first by detailing our cluster analysis preprocessing choices, then by going into the details of each metric we develop.

Gliomas are the most common primary brain tumor in adults, comprising about 30% of all brain tumors and 80% of malignant brain tumors [26]. The World Health Organization (WHO) has identified in 2016 a pathological classification criterion that divides gliomas in four different grades [27] in increasing prognosis severity: grade I are benign tumors and comparatively low risk. Grade II gliomas (also called Low Grade) have benign tendencies and generally carry a good prognosis for the patients; however overtime they are likely to exhibit an increase in grade, therefore are considered malignant. Finally, grade III and IV gliomas (High Grade) are malignant and portend a worse prognosis.

Our data will be comprised of Magnetic Resonance images (MRI) of brain volumes where the glioma has been previously segmented by annotators. An MRI volume is a three-dimensional representation of the brain, composed of typically 128 slices; each slice is an image taken on the horizontal axis. MR images have various intensity channels, called modalities; the dataset we use provides four: T1-weighted (T1), T1-weighted contrast enhanced (T1c), T2-weighted (T2) and Fluid-Attenuated Inversion Recovery (FLAIR). Each of these modalities has been captured using different magnetic resonance methods and highlights different elements in organ structure. Each slice also has a segmentation map associated to it. The segmentation map is such that an intensity of 1 corresponds to a pixel belonging to a lesion, 0 otherwise.

Our task then is to define a set of robust metrics that describe the segmentation result for each MRI volume in a quantitative and objective way and that therefore allow us to try and draw inferences from them. Some of the metrics we introduce are geometrical properties of the segmentation domain (i.e. dimension, symmetry, cluster numerousness and distance), while others describe the area of the image that the segmentation highlights (i.e. histogram, homogeneity, correlation).

3.1 Clustering

All the metrics we introduce require an additional preprocessing step to be carried out on the dataset; we have therefore fed each segmentation image to a clustering algorithm, in order to isolate and analyse separately each individual area of the lesion described by the segmentation. We have done this both on each distinct 2D MRI slice and on whole 3D MRI images.

For each MRI slice, we used the segmentation result at each pixel to reconstruct a segmentation map, as seen in Figure 6a.

We rebuilt the 3D segmentation map by ordering the slices belonging to the same MRI and assigning them a value in the third dimension, as to create a 3D matrix from them. In the BRATS dataset, each 3D MRI image has 128 slices [22], but the approach we used is agnostic with respect to the number of slices per MRI, which can vary by dataset. We then fed this image as a feature array to a Density Based Clustering algorithm (DBSCAN) [11] with Euclidean distance metric and no constraint on sample number or sample distance to define a cluster. Figure 5 shows an example of 3D clustering on an MRI volume where two separate clusters where found. We used DBSCAN as opposed to other clustering methods for three reasons:

- There is no knowledge a priori in the dataset about the number of clusters for each 2D image (and even more so for 3D images)
- The data analysed with the following metrics is the prediction of a deep network, and not an expert-annotated dataset. We needed therefore to choose a clustering algorithm robust to noise.
- The shape of the legion vary for each image and is not regular.

We found DBSCAN to be one of the simplest but most robust algorithm that could satisfy all of the constraints in our data.

3.2 Metrics

3.2.1 Metrics by MRI

We define as metrics by MRI those metrics, detailed below, that describe the relations between clusters within a 3D MRI image. For the purpose of these metrics, the noise points detected by the DBSCAN algorithm are ignored.

- ***Cluster Numerousness***, defined as the number of distinct cluster labels within the 3D segmentation image.
- ***Center of Mass Cluster Distance***. We first compute the center of mass for each cluster: since all points belonging to the segmentation clusters have weight equal to 1

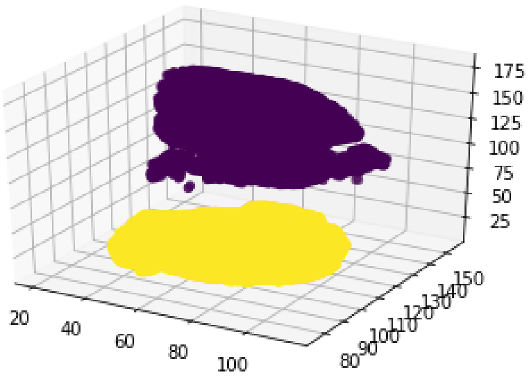
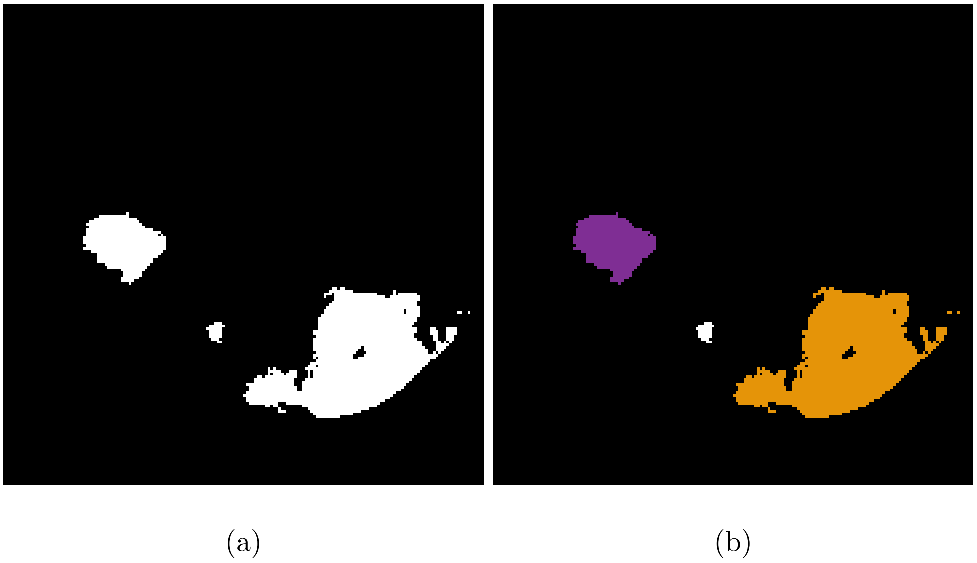


Figure 5: clustering of a 3D segmentation image. The two clusters are very clearly distinct, and they have different shape and size.


(a)

(b)

Figure 6: clustering of a 2D segmentation slice. (a) is the original segmentation, (b) is the cluster result.

(while points that are not in the segmentation have null weight), the center of mass equals the average of the coordinates $\mathbf{r}_i = (x, y, z)$ of all points in a cluster: CoM $= \frac{1}{N}\sum_{i=1}^{N}\mathbf{r}_i$.[1] For each pair of clusters we then compute the Euclidean distance between the centers of mass.

- ***Minimum Percentile Cluster Distance*** The minimum distance between two clusters C1 and C2 is defined as $\min_{p_1 \in C1, p_2 \in C2} \mathrm{dist}(p_1, p_2)$. To guarantee robustness to the metric, we compute the n-th percentile of the distances, so as to avoid unwanted noise to skew the value of the metric. Experiments will include evaluations at different percentiles.

3.2.2 Metrics by Cluster

While the metrics detailed above outline a relationship between clusters, the metrics by cluster aim to describe features of a single cluster within an MRI image.

- ***Cluster Size*** is defined as the number of pixels within a cluster.
- ***Histogram Intensity***. To compute the histogram of an individual cluster, we first have to apply the cluster as a mask onto its original MRI image, that is given $I_{ij,\mathrm{MRI}}$ the MRI image pixels, and C_{ij} the cluster pixels:

$$\mathrm{I}_{ij,\mathrm{Mask}} = \begin{cases} I_{ij,\mathrm{MRI}} & \text{if } C_{ij} = 1 \\ 0 & \text{if } C_{ij} = 0 \end{cases} \tag{3.1}$$

[1] This metric can be extended in future works to account for probability weights of the segmentation image instead of a binary classification. In that case, the center of mass would be computed as the weighted average of point coordinates.

We ignore zero-value pixels in the mask for the purpose of the histogram, since they represent pixels where there is no cluster.

The major challenge in the histogram computation is to decide the number of bins. This is because the number of bins must be chosen so that the histogram is meaningful for every MRI image. This means that there must be enough bins so that the curve of different histograms can be compared and made inferences upon, but not so many as to have bins with very few examples in them. We found that having between 15 and 20 bins seems to satisfy both constraints, but at the same time we've decided to keep the bin number parametrical so as to allow for more experimentation.

- ***Intensity Heterogeneity***. The most intuitive and rudimentary way to define heterogeneity in an image is to analyze exclusively its distribution of intensities [28]. That is, an image where all pixels have the same intensity (e.g. a completely white image) will be less heterogeneous (more homogeneous[1]) than an image where all pixels have different intensities. In order to quantify this measure of heterogeneity, we compute the **standard deviation**, **skewness** and **kurtosis** of the intensities within a given cluster.

  The standard deviation is a measure of how much a distribution deviates from its mean; given N pixels in a given cluster with intensity x_i, and a mean intensity $\bar{x}$, it is computed as $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}$. The standard deviation of an image with all pixels of the same intensity will be 0 (because $x_i = \bar{x}\ \forall x_i$). Therefore, we can use the standard deviation as

[1] A measure of homogeneity will be opposite to that of heterogeneity for the same intensity distribution: heterogeneity $= 1 -$ homogeneity

a primitive measure of heterogeneity.

Similarly, the kurtosis and skewness of a distribution describe its shape and can be used to define a simple measure of heterogeneity. Skewness describes the asymmetry of the distribution of intensities. The closer a distribution resembles a normal distribution, the closer its skewness will be to zero. Despite not being necessarily linked to heterogeneity, we can postulate that a homogenous cluster will have skewness closer to zero than an heterogeneous one.

Finally, Kurtosis describes how much the tails of a distribution are significant with respect to the mean. The higher the Kurtosis, the more outliers a distribution has, and the more prominent its tails are. A completely homogenous image will have kurtosis 0, as every pixel has the same intensity of the mean. Therefore we can say that the higher the kurtosis of a given cluster is, the more heterogeneous the cluster.

- ***GLCM Homogeneity*** As defined by Haralick et al. [29], a Gray-Level Co-occurrence Matrix (GLCM) represents how many pixels of gray-level value i occur at a distance d and angle θ from a pixel of gray-level value j. For formulaic ease, d and θ can be decomposed into their Cartesian projections $\Delta x = d \cos\theta$ and $\Delta y = d \sin\theta$.

  An element $P(i, j, d, \theta) = P(i, j, \Delta x, \Delta y)$ of the co-occurrence matrix for an image I of dimension $n \times m$ is then defined by Haralick as:

$$P(i, j, d, \theta) = \sum_{k=1}^{n} \sum_{l=1}^{m} \begin{cases} 1 & \text{if } I(k, l) = i \text{ and } I(k + \Delta x, l + \Delta y) = j \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

Haralick then defines 14 features that can be computed on the GLCM, among which the Angular Second Moment (ASM): $f_1 = \sum_i \sum_j \{p(i,j)\}^2$, where $p(i,j)$ is a normalized element of the matrix P. The ASM represents a measure of homogeneity in the image: an homogenous image will have few dominant gray-tone transitions, i.e. P will be sparse and its elements will have larger magnitude, while an heterogeneous image will have a lot of different tone transitions, hence P will have a large number of small elements [29]. ASM, being the sum of squares of elements of P, will then be larger for homogenous images and smaller for heterogeneous ones.

- ***Spatial Heterogeneity*** Brooks and Grisby [28] develop in their 2013 paper "Quantification of heterogeneity observed in medical images" a statistic that measures "the distance-dependent average deviation from the smoothest intensity gradation feasible" [28]. They claim that measures of heterogeneity based exclusively on intensity distribution are not meaningful enough when compared to a human expert visual inspection.

  For each non-repeating pair of object pixels m and n with intensities I_m and I_n (ignoring the zero-intensity background), they compute the Bresenham line $\mathcal{L}$, i.e. the ordered set of pixels between m and n such that these pixels form the straightest line between m and n. Then, they compute the discrete distance $L = |\mathcal{L}|$ and the average grayscale gradation between m and n:

$$\overline{\Delta I} = \frac{1}{L} \sum_{l \in \mathcal{L}} |I(r_{ml}) - I_l| \tag{3.3}$$

where I_l is the intensity of pixel $l \in \mathcal{L}$, and $I(r_{ml})$ is computed as:

$$I(r_{ml}) = I_m + \frac{I_n - I_m}{r_{mn}} r_{ml} \tag{3.4}$$

r being the Euclidean distance between the two indexed pixels.

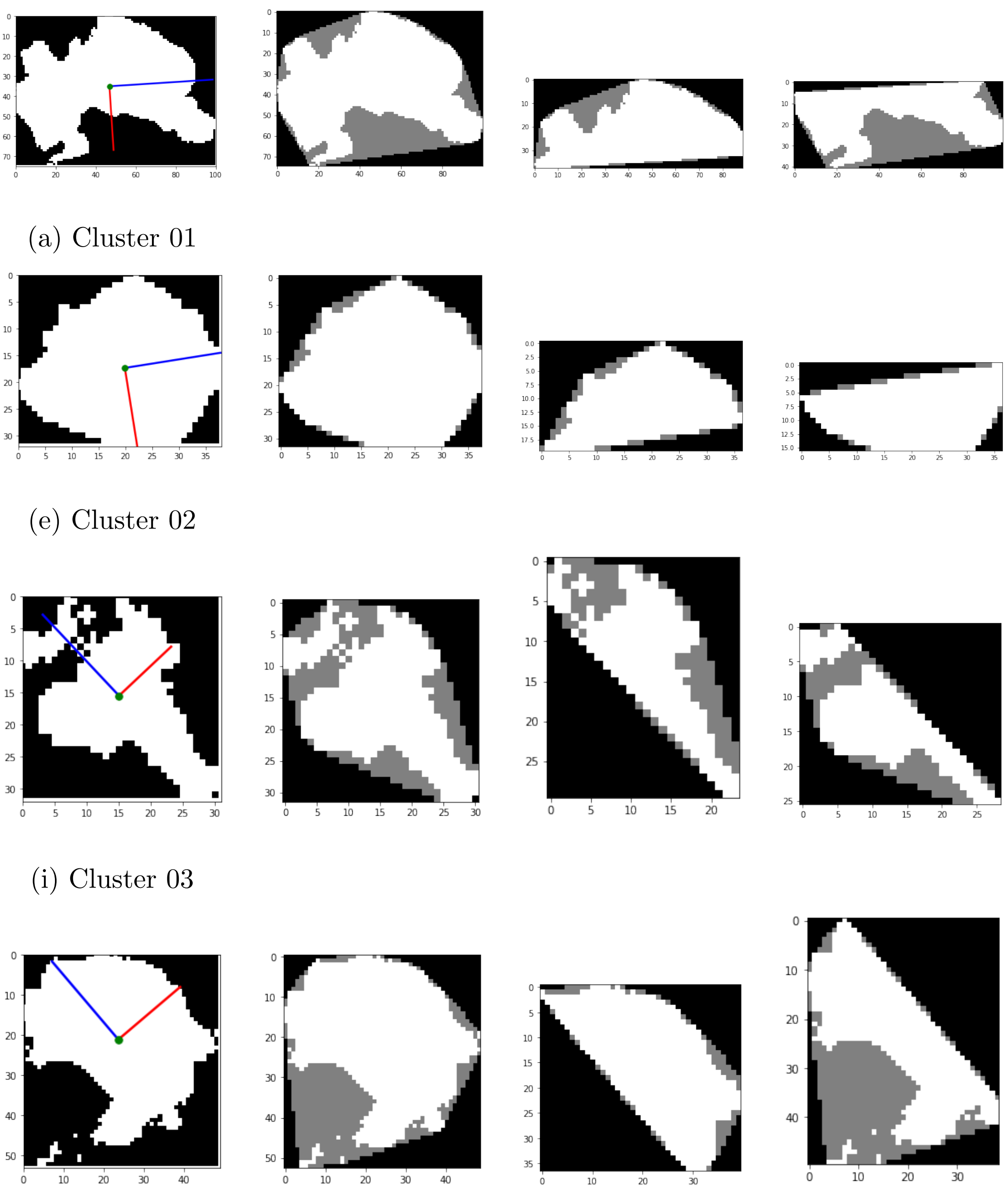
Once $\overline{\Delta I}$ and L are computed for each nonrepeating pair of pixels, $\overline{\Delta I}$ is ensemble-averaged for each L, such that every discrete separation L (that is, every possible discrete distance between two pixels) has associated a single value $\overline{\overline{\Delta I}} \equiv \langle \overline{\Delta I} \rangle_{\text{ens}}$. $\overline{\overline{\Delta I}}$ is then normalized to the largest discrete distance $\tilde{L} = max(L)$.

Finally, the heterogeneity metrics by Brooks and Grisby is then computed as:

$$\zeta \equiv \int_0^1 \overline{\overline{\Delta I}}(L/\tilde{L}) d(L/\tilde{L}). \tag{3.5}$$

- ***Symmetry*** To define a measure of symmetry, we consider each cluster as a geometrical region, not considering its intensity. Unfortunately, we could not rely on the classical definition of symmetry, as none of the clusters are technically completely symmetric. Moreover, we decided that a binary measure of "symmetric" an "not symmetric", as is the classical definition, would not be meaningful in terms of diagnostics and image analysis.

  As seen in Figure 7 and Table I, we define symmetry as a series of values computed for the whole image and for its halves. For the whole image, we computed solidity and eccentricity. Solidity is defined as the ratio of pixels in the cluster to pixels of the convex



(a) Cluster 01

(e) Cluster 02

(i) Cluster 03

(m) Cluster 04

Figure 7: Symmetry evaluation examples; the first column portrays the clusters and their two main axes; the major axis (depicted in blue) will be then used to divide the cluster. The second column shows the convex hull area, while the third and the fourth the upper and lower half of the cluster with their own convex hull area respectively.

Cluster	**Solid.**	**Ecc.**	**U Solid.**	**U Perim.**	**U Ecc.**	**L Solid.**	**L Perim.**	**L Ecc.**
Cluster 1	0.742	0.786	0.846	260.85	0.926	0.652	324.20	0.949
Cluster 2	0.947	0.553	0.882	98.04	0.876	0.947	92.90	0.908
Cluster 3	0.688	0.765	0.683	101.63	0.932	0.674	99.05	0.937
Cluster 4	0.725	0.611	0.898	127.23	0.888	0.604	168.82	0.901

TABLE I: Metrics values referring to Figure Figure 7

hull image, i.e. the smallest convex polygon enclosing the cluster (as seen in the second column of Figure 7). The more regular the cluster, the closer its solidity is to 1. Similarly, eccentricity refers to the ellipse with the same second-moments as the cluster. The closer eccentricity is to zero, the closer the cluster is to a circle. These two measures, despite not being directly related to a cluster's symmetry, can still be helpful to define how regular a cluster's shape is.

We then compute the cluster's centroid and the main axis. Using the line drawn by the main axis direction, we split the cluster in two halves and compute solidity, eccentricity and perimeter for both. It is reasonable to say that the more different the metrics of the two halves are, the more asymmetrical the cluster is.

Figure 7 showcases this reasoning by applying the metrics to four clusters with wildly different geometrical shapes. Cluster 2 is, among the four, the most symmetrical, and indeed its solidity is very high, and its eccentricity is the lowest of the four; the two halves differ minimally in perimeter, and have both high solidity and high eccentricity. Cluster 1 and 4 are asymmetrical: the two halves have very different perimeters and solidity. Finally, Cluster 3 shows the interesting property of being symmetrical along the main axis, but extremely irregular in shape. The two halves have close solidity and perimeter, but overall extremely low solidity.

CHAPTER 4

EXPERIMENTAL DESIGN

This chapter will provide a detailed description of the BRATS dataset we used to compute our metrics (Section 4.1); Section 4.2 explains the tradeoff choice between metric's meaningfulness and computational cost for Spatial Heterogeneity. Finally Section 4.3 will discuss our implementation.

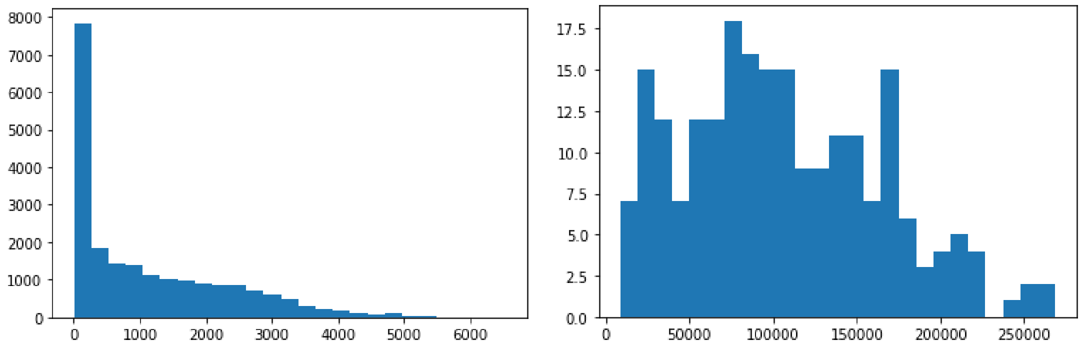
4.1 Dataset

The dataset we are using to test our metrics is the Multimodal Brain Tumor Image Segmentation Benchmark (BRATS) [22], more precisely the 2015 version. It was developed and made publicly available in 2012 in order to evaluate brain tumor segmentation algorithms and provide a benchmark for future works.

The BRATS2015 dataset consists of 65 multi-contrast MR scans from glioma patients [22], and 65 synthetic MR scans.

Of the 65 clinical MRI, 14 are low-grade (astrocytomas and oligoastrocytomas) and 51 are high-grade (anaplastic astrocytomas and glioblastomas). Each image has four different MRI contrasts, as mentioned in Chapter 3: T1, T1c, T2 and FLAIR.

The synthetic data, on the other hand, consists in simulated images of 35 high-grade and 30 low-grade gliomas. These images were generated using tumor simulation software. Both the clinical data and the synthetic data was manually annotated.



(a)

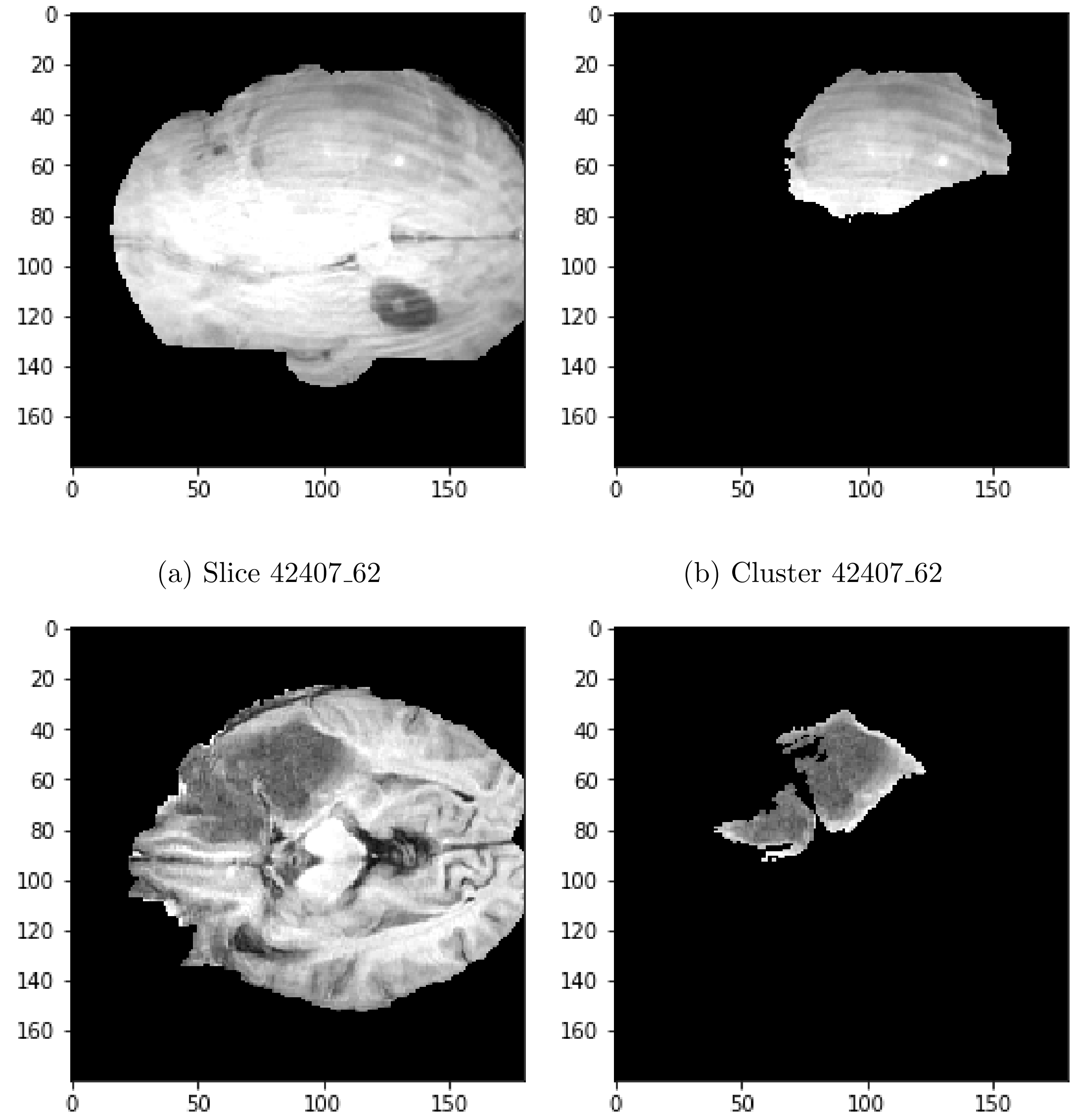
(b)

Figure 8: Cluster size distribution within the dataset. Figure (a) shows the distribution of sizes in 2d slices, while (b) shows size distribution in the 3D volume clusters.

4.2 Random Pixel Choice in Spatial Heterogeneity

The computation of Spatial Heterogeneity as defined by Brooks and Grisby [28] presents one major flaw when applied to clusters with a great number of pixels: it has exponential time complexity, more precisely of $\mathcal{O}(\overline{L}N^2)$ where N is the number of pixels in the given cluster and $\overline{L}$ the average distance between two pixels in the cluster [28]. We found therefore that a complete computation of the metric was excessively expensive.

The average size of a single 2D cluster in our dataset is 1109 pixels, and that of a 3D cluster is 106 thousand pixels, with a maximum of 268741. Figure 8 shows dimension distribution within the dataset.



(a) Slice 42407_62

(b) Cluster 42407_62

(c) Slice 42403_48

(d) Cluster 42403_48

Figure 9: Slices used for the statistical evaluation of random pixel selection during spatial heterogeneity computation and respective clusters.

		200 pixels, 30 runs		**400 pixels, 30 runs**		**1000 pixels, 30 runs**		**2000 pixels, 15 runs**	
id	**total pixels**	**mean**	**deviation**	**mean**	**deviation**	**mean**	**deviation**	**mean**	**deviation**
42407_62	3872	0.0889	0.0027	0.0883	0.0019	0.0890	0.0012	0.0886	0.0009
42403_48	2002	0.2045	0.0126	0.2087	0.0072	0.2094	0.0050	0.2100	$2.7755e^{-17}$

TABLE II: **Statistical evaluation of random pixel selection impact on spatial heterogeneity**

In their paper, Brooks and Grisby suggest taking "as large a random subset of all possible pixel pairings as is computationally accessible" [28]. We wanted to establish whether decreasing the pixels of the clusters to a smaller, random subset could still yield a meaningful metric; it is clear that there is a trade off between the metric's validity and its computational feasibility. Thus we conducted a statistical analysis over two randomly chosen slices, shown in Figure 9, over which we computed spatial heterogeneity for 200, 400, 1000 and 2000 random pixels.

As can be seen in Table II, we ran the algorithm 30 times with subsets of 200, 400 and 1000 pixels, while for the 2000 pixels subset we made only 15 runs due to time complexity. We then computed the mean and standard deviation of the runs' results for each subset. As can be seen, the average results between subsets are very similar, and even within a subset the standard deviation never exceeds 1% of the metric's range (and that only in the case of 200 pixels, which is 10% of the total number of pixels of that particular slice).

We've decided thus to run the algorithm on a subset of 200 pixels, as even 400 pixels per slice proved to be computationally challenging.

4.3 Implementation

The code for this thesis was written in Python 3.6.9. We exploited pre-existing libraries where possible, while a few metrics we implemented ourselves.

We implemented Clustering on both 2d slices and 3d volumes, using the DBSCAN implementation provided by scikit-learn [1] for both. 2D slices have shape 180 × 180 × 4, where the third dimension represents the four modality channels. Since our data is exclusively two-dimensional, we re-created each volume by stacking MRI slices on the z axis, thus creating a 4 dimensional matrix of 128 × 180 × 180 × 4. Our code however is independent on number of slices and number of pixels, therefore could theoretically be applied to any MRI volume.

We saved each volume's clusters in a separate npy file for easier manageability; each of these files contains a matrix where each row $[l, z, x, y]$ contains the spatial information and label of each pixel belonging to a cluster. l is a discrete label assigned by the algorithm to each pixel belonging in a cluster, therefore two pixels a and b belong to the same cluster if $l_a = l_b$. $l = -1$ designates noise points; these noise points will be ignored for further computations.

For the metrics, we mainly used NumPy [30], SciPy [31] and Scikit-Image [32]. Cluster numerousness was simply computed as the cardinality of the set of labels of a volume, excluding $l = -1$. Center of Mass of each cluster is calculated as the average of cluster pixel coordinates, and then their distance is the simple Euclidean distance between two points. As for the percentile distance, for each pair of clusters we had to compute pairwise distance for all pixels, then find the desired percentile.

Cluster histogram, standard deviation, skew and kurtosis are all computed directly using NumPy, whereas for GLCM homogeneity we used the `feature` module on scikit-image. In order to do this, we had to normalize the greyscale values of our slices to a $[0, 255]$ integer interval. We computed the grey co-occurrence matrix for an offset distance of 5 pixels and angles $(0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3}{4}\pi)$. Then, for each of these four GLCMs, we computed homogeneity. This wields four homgeneity values per cluster, which are then averaged.

Following Brooks and Grisby [28], we implemented spatial heterogeneity ourselves. The process used to compute it is shown in Algorithm 1.

Finally, the symmetry metrics were computed using the `measure` module of scikit-image.

GLCM homogeneity, spatial heterogeneity and symmetry measures all require a 2d image to be computed. We therefore computed them over each slice of an MRI volume, and then computed the average of their results weighted by the number of pixels in the cluster in the slice. Given a set of slices S and a metric m:

$$\overline{m}_{\text{total}} = \frac{\sum_{s \in S} m_s p_s}{\sum_{s \in S} p_s} \tag{4.1}$$

where m_s is the metric calculated for slice $s \in S$ and p_s the number of pixels of s. Since these metrics are all very sensitive to the number of pixels, we've decided to exclude the slices

Procedure 1: Spatial Heterogeneity Algorithm

```
input  : slice $i$ where all pixels not belonging to a cluster are set to zero.
output: Heterogeneity metric $\zeta$ computed on $i$

pixel_list ← $[x, y]$ if $i[x, y] \neq 0$
if |pixel_list| > 200 then
    sampled_pixel_list ← 200 random samples ∈ pixel_list
    pixel_list ← sampled_pixel_list
end
pair_elements ← ∅
for m in pixel_list do
    remove m from pixel_list
    if pixel_list not empty then
        for n in pixel_list do
            BresenhamLine = Bresenham(m,n)
            L = |BresenhamLine|
            $r_{mn}$ = dist(m,n)
            $I_n \leftarrow i[x_n, y_n]$
            $I_m \leftarrow i[x_m, y_m]$
            $I_{mn} = I_n - I_m$
            $I_{\text{vals}} \leftarrow \emptyset$
        end
    end
end
```

Procedure 2: Spatial Heterogeneity Algorithm, cont.

```
            $I_{mn} = I_n - I_m$
            $I_{\text{vals}} \leftarrow \emptyset$
            for $l$ in BresenhamLine do
                $r_{ml} = \texttt{dist}(m, l)$
                $I_{rml} = I_m + \frac{I_{mn}}{r_{mn}} r_{ml}$
                $I_l \leftarrow i[x_l, y_l]$
                $I_{\text{vals}} \leftarrow \texttt{abs}(I_{rml} - Il)$
            $\Delta I_{nm} = \frac{\sum I_{\text{vals}}}{L}$
            pair_elements $\leftarrow [L, \Delta I_{nm}]$
$L_{\max} = \max(L \in \text{pair\_elements})$
for unique $l$ in pair_elements do
    $\overline{\overline{\Delta I}} = \frac{\sum_{\Delta I \in l} \Delta I}{|l|}$
return $\zeta \equiv \int_0^1 \overline{\overline{\Delta I}}(L/\tilde{L}) d(L/\tilde{L})$
```

where a cluster has less than 4 pixels. These slices would, in fact, have very skewed results as their sample would be too small to be meaningful.

We computed all metrics for each volume in the dataset, with the exception of the minimum percentile distance, due to complexity constraints. For those metrics that require pixel intensity to be computed (deviation, skewness and kurtosis, GLCM homogeneity and spatial heterogeneity), we computed them for every one of the four modalities. Center of Mass cluster distance was set to NaN wherever the volume had only one cluster (making a distance between clusters not defined). All metrics except cluster number and cluster distance are computed for each cluster in the volume, then averaged among its clusters in order to give us an aggregated measure per volume. The exception to this are the symmetry metrics, which have not been averaged.

CHAPTER 5

RESULTS

This chapter will explain the general results of our analysis, displaying scatter plots for each pair of metrics and explaining their relationship and descriptiveness (Section 5.1). Section 5.2 will then test the meaningfulness of our metrics with respect to the tumor grade label.

5.1 Metrics descriptiveness

For gliomas, both high grade and low grade, neuroimaging protocols such as computed tomography (CT) and MRI are extensively used both before, during and after treatment to evaluate the progression of the tumor and the effectiveness of a given therapy [22]. However, these images are evaluated by doctors with qualitative criteria (for example the presence of hyper-intense tissue in a T1c scan) or with rudimentary quantitative measures, such as the diameter of the lesion [33, 34]. The primary goal of our research was to develop quantitative metrics that could possibly be applied to an MRI brain volume as additional diagnostic measures.

Figure 10, Figure 11, Figure 12, Figure 13 and Figure 14 show scatter plot matrices for the metrics referring to each modality, and for the symmetry metrics. The classification labels, that is the tumor grade, is shown in different colors within the plots. These matrices help us define the relationships, if any, between our metrics.

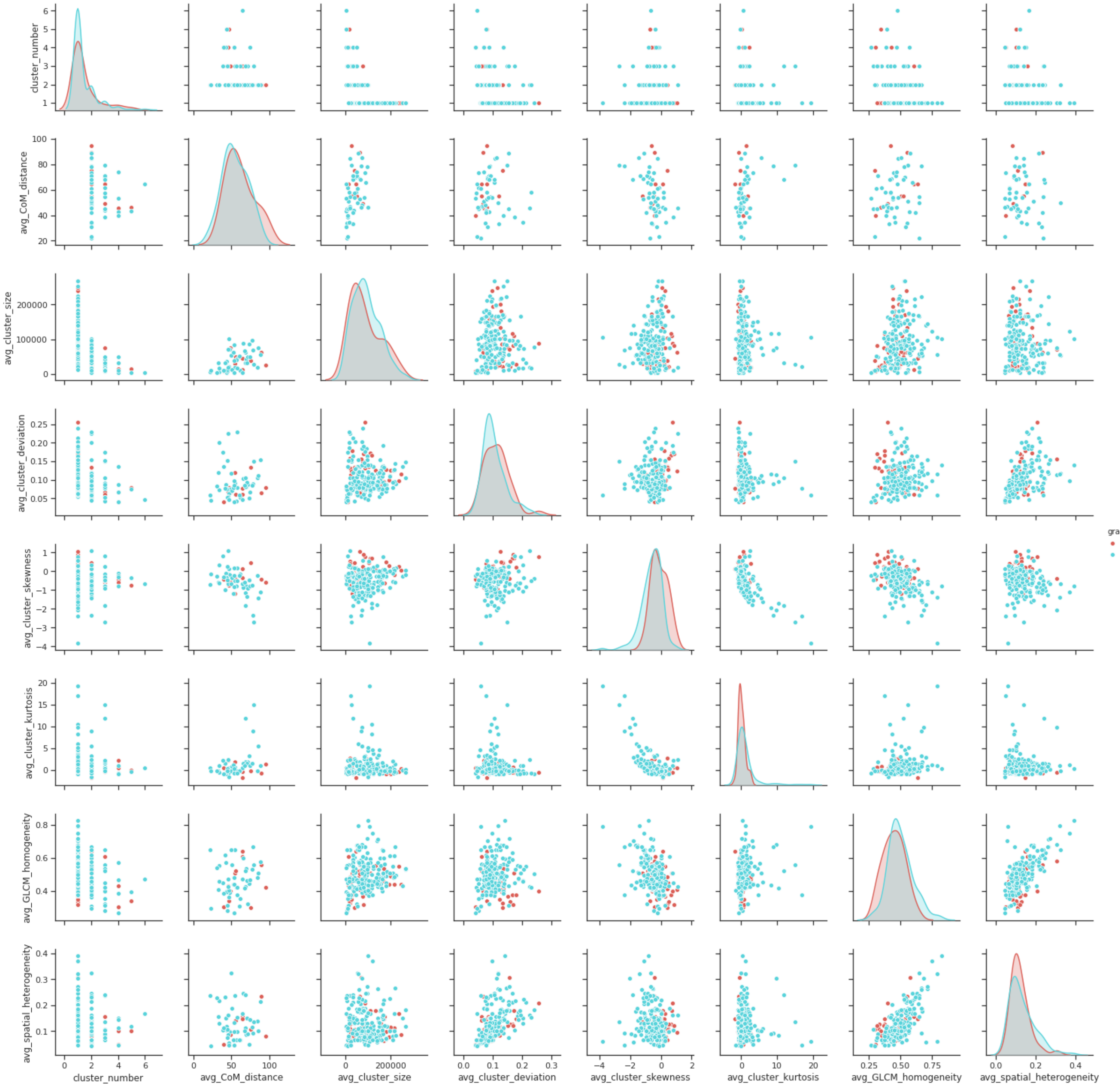


Figure 10: Scatter plots for the metrics calculated on the T1 modality

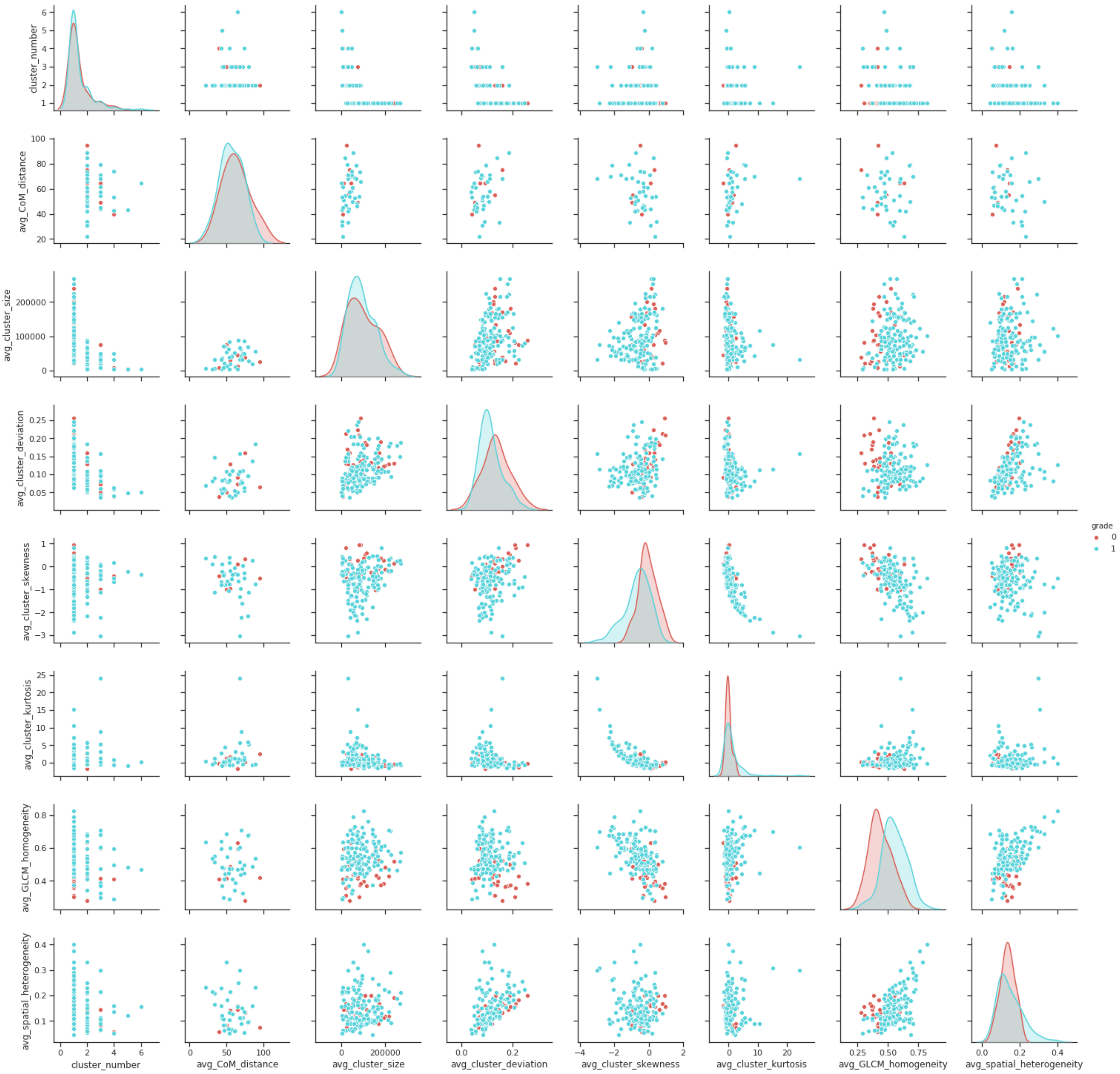


Figure 11: Scatter plots for the metrics calculated on the T1c modality

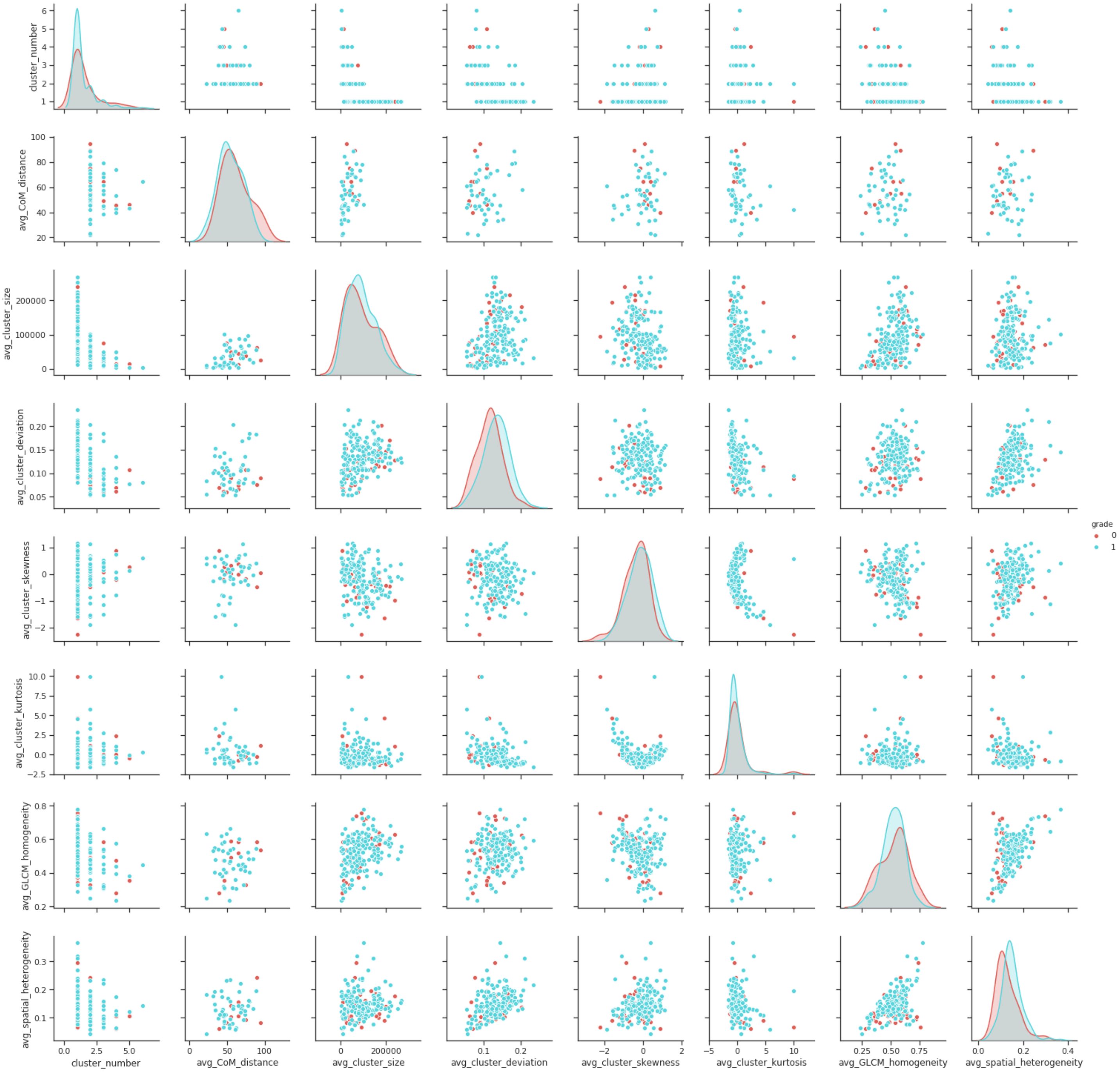


Figure 12: Scatter plots for the metrics calculated on the T2 modality

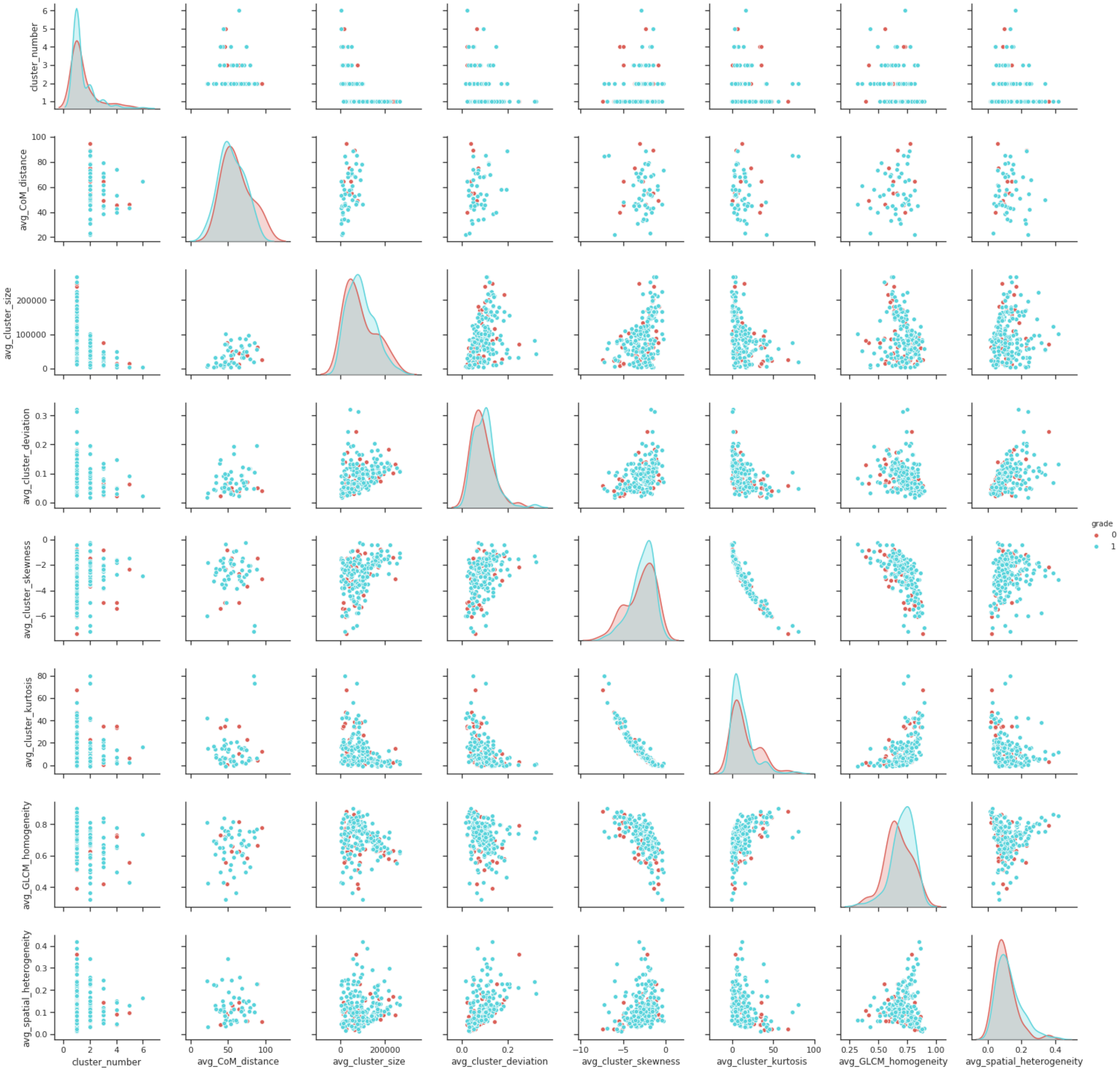


Figure 13: Scatter plots for the metrics calculated on the FLAIR modality

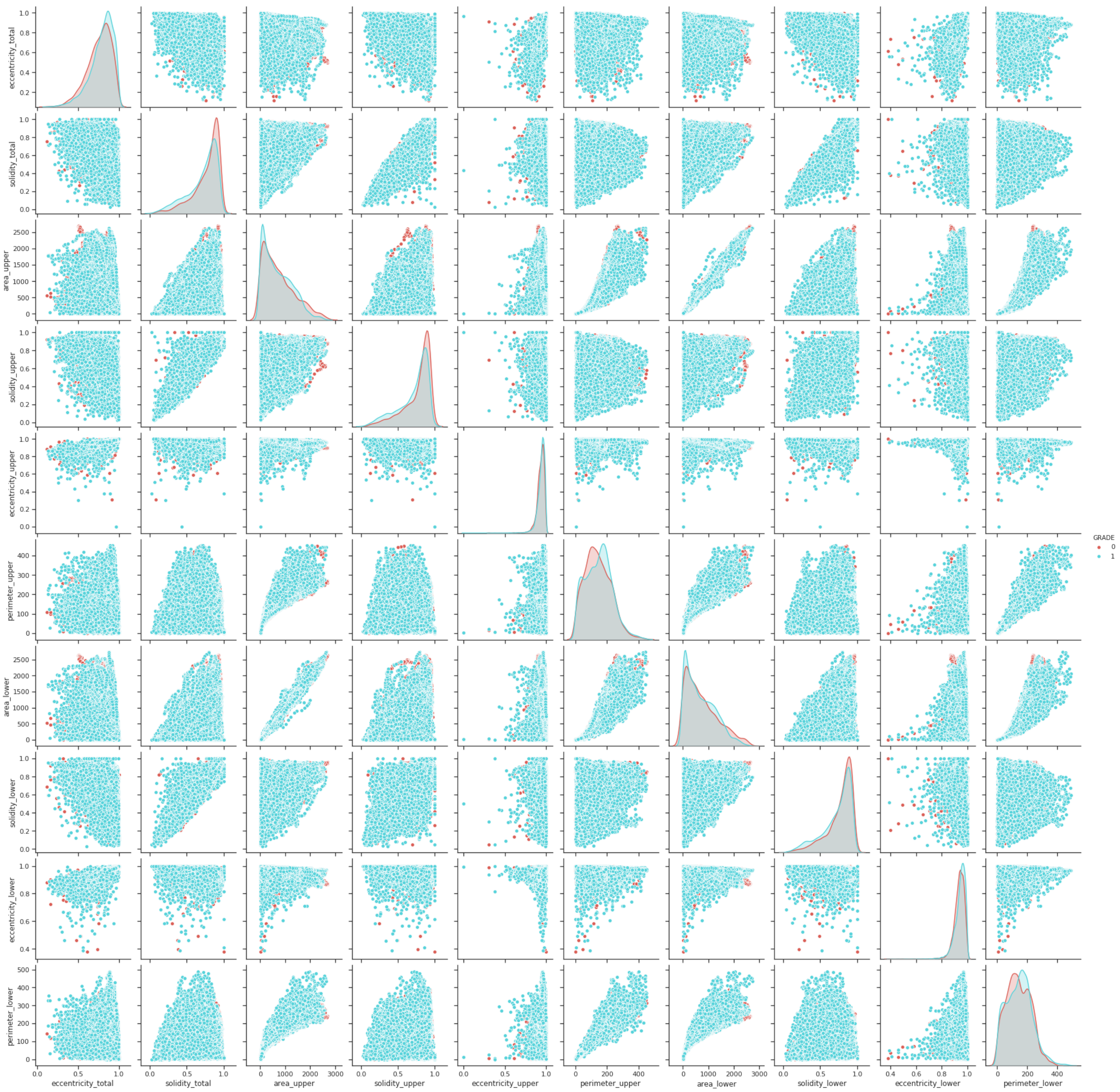


Figure 14: Scatter plots for the symmetry metrics

First of all, we want to highlight how the differences in distribution between modalities are little, and all metrics show a tendency to behave in a similar way in all modalities. This homogeneity in results attests to a robustness of the developed metrics to different imaging modalities they can be applied to.

It is particularly evident in FLAIR, but percievable in all other modalities also, how kurtosis and skewness are not independent (confirmed in Table IV, Table V, Table VI and Table VII) but have a very clearly defined polynomial relationship.

Looking especially at T1 and T1c, it is interesting to note how plotting sample points in the GLCM homogeneity and spatial heterogeneity axis, there seems to be a linear tendency in the samples. This will also be confirmed in Table V in the following section, suggesting the two metrics are not independent. Similarly, spatial heterogeneity seems to have a somewhat linear relationship with standard deviation, although not as strong as the one with GLCM homogeneity.

As for the symmetry metrics, there are strong linear relationships between perimeters and areas of the two halves of a cluster (perimeter_lower and _upper and area_lower and _upper in Figure 14). Perimeter and area are also related, as shown by their relative scatter plots, which have a very definite shape.

Table III shows a comparison between two slices from two different volumes. Qualitatively, the first lesion has more homogeneous intensity, and its shape is more symmetrical and regular than the second one, as can be seen in Figure 15. The metrics computed encase this visual qualities in quantitative values that can be compared. Spatial heterogeneity is very low for

Spatial Heterogeneity	**Spatial Heterogeneity**
0.0467	0.2532
GLCM Homogeneity	**GLCM Homogeneity**
0.5816	0.5577
Symmetry Metrics	**Symmetry Metrics**
solidity = 0.8801	solidity = 0.6430
eccentricity = 0.7643	eccentricity = 0.8697
upper eccentricity = 0.9588	upper eccentricity = 0.9588
upper solidity = 0.8361	upper solidity = 0.8400
upper perimeter = 165.40	upper perimeter = 222.02
upper area = 990	upper area = 1250
lower eccentricity = 0.9327	lower eccentricity = 0.9701
lower solidity = 0.9056	lower solidity = 0.5132
lower perimeter = 161.98	lower perimeter = 249.78
lower area = 970	lower area = 1123

TABLE III: Comparison of metrics for two very different slices

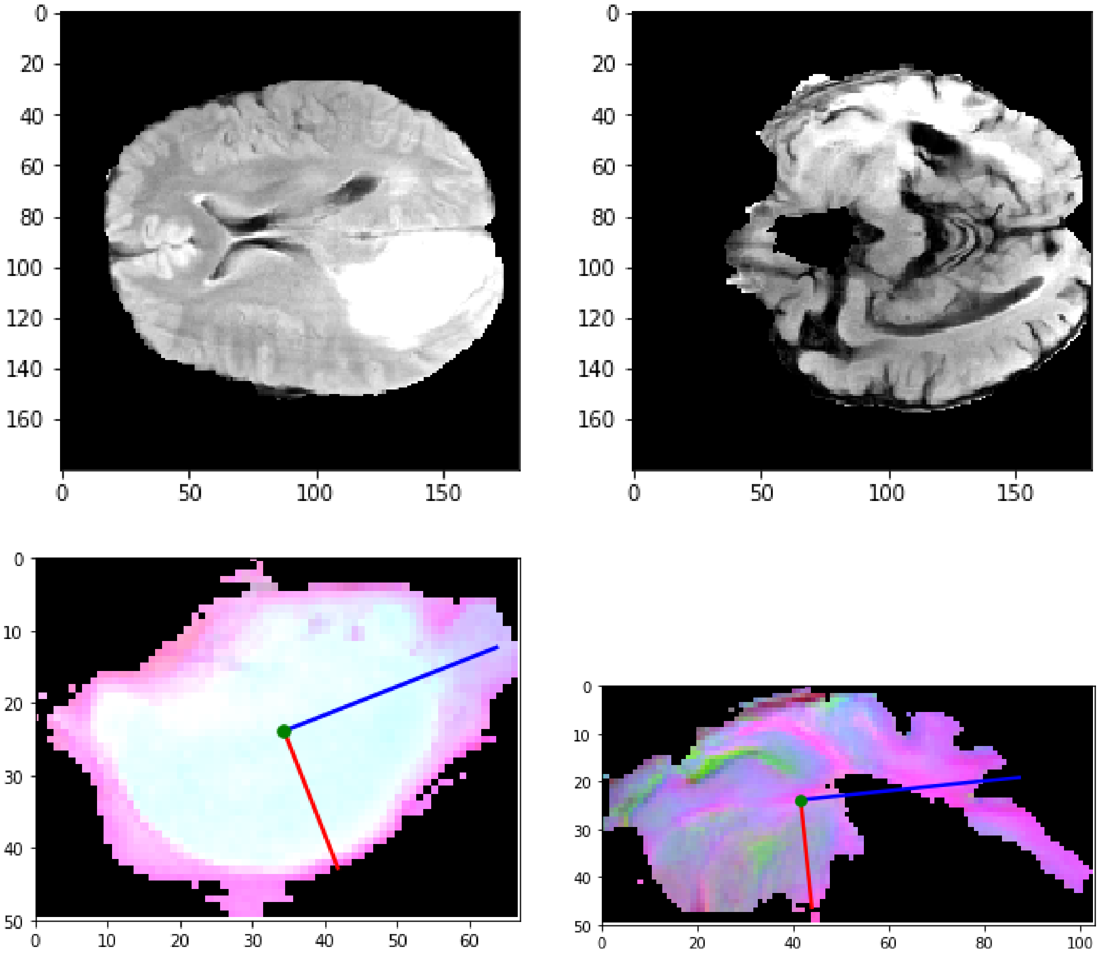


Figure 15: Slices used for descriptive comparison and relative clusters

the first slice and very high for the second one; GLCM homogeneity, although with a subtler difference, is higher in the first case, demonstrating higher homogeneity of the first slice with respect to the second one.

Similarly, the very high solidity of the first image compared to the second one suggests a more regular shape. The difference between upper and lower perimeter in the first case is only 4 pixels, while in the second one it's 26 pixels; the difference in solidities (0.07 and 0.33) is also a very strong indicator that the first lesion is more symmetrical than the second one.

5.2 Inference of tumor grade from image properties

Clinically, the grade of a glioma depends on its histology and pathology [27]. From the histological point of view, once a glioma's phenotype and genotype have been identified, then the grade follows the World Health Organization categorisation. Conversely, the grade aims to capture a pathological notion (that is, the higher the grade, the worse the prognosis).

Most gliomas characteristics, especially the histological ones, cannot be analyzed through an MRI scan, but need mircoscopic, cellular analysis. MRI scans aid in the diagnosis process on the macroscopic scale, identifying the shape and location of the tumor [27]. The macroscopic information however does not have a definitive correlation to the tumor grade.

With the following analysis we wanted to examine whether the metrics we developed could potentially carry meaning with respect to the grade of the tumor.

First of all, we computed the correlation and mutual information scores for each singular metric with respect to the grade. The correlation we computed also for each pair of metrics. Table IV displays the correlation for the T1 metrics; similarly Table V for the T1c metrics, Table VI for the T2 metrics and Table VII for the FLAIR metrics. Additionally, Table VIII shows correlation among the symmetry metrics, and Table IX among the clustering metrics. Table X and Table XI show the mutual information score between each individual metric and the tumor grade.

The results suggest, as expected, a strong correlation between deviation, skewness and kurtosis, particularly when computed on the T1c and Flair modalities. This makes sense as skewness and kurtosis are not independent. Similarly, deviation shows a moderate correlation to the heterogeneity measures, and the correlation between GLCM homogeneity and spatial heterogeneity is above 0.5 for three out of four modalities. Indeed, as discussed in Chapter 3, deviation, GLCM homogeneity and spatial heterogeneity should all have a similar meaning.

T1	**deviation**	**skewness**	**kurtosis**	**GLCM h.**	**spatial h.**	**grade**
deviation	1.0000	0.1849	-0.1199	0.0726	**0.4151**	-0.0790
skewness	0.1849	1.0000	**-0.6962**	-0.4496	-0.1585	-0.2995
kurtosis	-0.1199	**-0.6962**	1.0000	0.1995	-0.0574	0.1496
GLCM homogeneity	0.0726	-0.4496	0.1995	1.0000	**0.6724**	0.2113
spatial heterogeneity	0.4151	-0.1585	-0.0574	0.6724	1.0000	0.0734
grade	-0.0790	**-0.2995**	0.1496	0.2113	0.0734	1.0000

TABLE IV: Correlation table for T1 metrics

T1c	deviation	skewness	kurtosis	GLCM h.	spatial h.	grade
deviation	1.0000	0.2815	-0.1966	-0.0608	**0.4149**	-0.1924
skewness	0.2815	1.0000	**-0.7377**	-0.5364	-0.1121	-0.3162
kurtosis	-0.1966	**-0.7377**	1.0000	0.2653	0.1597	0.1336
GLCM homogeneity	-0.0608	**-0.5364**	0.2653	1.0000	0.5138	0.4142
spatial heterogeneity	0.4149	-0.1121	0.1597	**0.5138**	1.0000	0.0907
grade	-0.1924	-0.3162	0.1336	**0.4142**	0.0907	1.0000

TABLE V: Correlation table for T1c metrics

The correlation between each singular metric and the tumor grade is on average very weak. As seen in Table Table V, the modality that seems to carry slightly more meaning with regards to tumor grade is T1c, with a -0.31 correlation for skewness and 0.41 for GLCM homogeneity. In general however, no metric seems to show a strong connection with the tumor grade. The same can be said for the modality-independent metrics, as can be seen in Tables Table VIII and Table IX.

Furthermore, we tried to compute the mutual information score between singular metrics and the grade, on the assumption that their relationship may not be linear. Results are shown in Table Table X. It is however confirmed also by the mutual information score that no singular metric has any strong relationship with the grade in and on itself.

T2	**deviation**	**skewness**	**kurtosis**	**GLCM h.**	**spatial h.**	**grade**
deviation	1.0000	-0.0781	-0.3453	0.2204	**0.3962**	0.2054
skewness	-0.0781	1.0000	**-0.3246**	-0.2085	0.2623	0.1247
kurtosis	**-0.3453**	-0.3246	1.0000	0.1049	-0.1625	-0.0408
GLCM homogeneity	0.2204	-0.2085	0.1049	1.0000	**0.5206**	-0.0294
spatial heterogeneity	0.3962	0.2623	-0.1625	**0.5206**	1.0000	0.1752
grade	**0.2054**	0.1247	-0.0408	-0.0294	0.1752	1.0000

TABLE VI: Correlation table for T2 metrics

Once established that there was no obvious direct relationship between any single metric and tumor grade, we wanted to test whether a) a linear transformation over the metrics could showcase a correlation with the grade label and b) a simple, out-of-the-box classification method could correctly predict tumor label given our metrics. For a), we applied Principal Component Analysis over a number of combinations of metrics, while for b) we performed Decision Tree Classification.

5.2.1 Principal Component Analysis

We performed Principal Component Analysis (PCA), keeping the first two components for visualization ease. Table XII shows the components' plot, their mutual information with respect to the grade, and their explained variance. The plots of Table XII show evident overlap of the two grade labels for all the PCA results; this is further confirmed by the low mutual information

FLAIR	deviation	skewness	kurtosis	GLCM h.	spatial h.	grade
deviation	1.0000	0.4325	**-0.4544**	-0.1841	0.4160	0.0758
skewness	0.4325	1.0000	**-0.9520**	-0.6546	0.2322	0.0496
kurtosis	-0.4544	**-0.9520**	1.0000	0.5280	-0.2662	-0.0759
GLCM homogeneity	-0.1841	**-0.6546**	0.5280	1.0000	0.0743	0.1518
spatial heterogeneity	**0.4160**	0.2322	-0.2662	0.0743	1.0000	0.1341
grade	0.0758	0.0496	-0.0759	**0.1518**	0.1341	1.0000

TABLE VII: Correlation table for FLAIR metrics

between each component and the classification label. In confirmation of the intuition of the previous section, the higher information is carried by the T1c modality.

The explained variance of the first two components computed by the PCA is very low, with a maximum of 0.87 total explained variance for the modality-independent metrics. The loss of information resulting from PCA makes its usefulness in a classification context extremely limited; it can be, however, used for visualization purposes.

5.2.1.1 Decision Tree Classification

We performed classification applying a decision tree classifier to the metrics for the four modalities, plus separately for the symmetry metrics.

To exploit as best as we could the limited dataset, we performed Leave-One-Out cross-validation. To avoid overfitting we imposed 40 minimum samples per tree split and a minimum

symmetry	e_tot	s_tot	area_u	s_u	e_u	p_u	area_l	s_l	e_l	p_l	grade
eccentricity total	1.0000	-0.3634	-0.1820	-0.3680	0.6765	-0.0256	-0.1799	-0.3717	**0.7034**	-0.0203	0.0902
solidity total	-0.3634	1.0000	0.3255	**0.8922**	-0.1673	0.0835	0.3120	0.8684	-0.2289	0.0990	-0.1136
area upper	-0.1820	0.3255	1.0000	0.3272	-0.0898	0.8623	**0.9903**	0.3163	-0.1378	0.8659	-0.0314
solidity upper	-0.3680	**0.8922**	0.3272	1.0000	-0.1944	0.0504	0.2979	0.6138	-0.2475	0.1561	-0.1102
eccentricity upper	**0.6765**	-0.1673	-0.0898	-0.1944	1.0000	-0.0082	-0.1307	-0.2006	0.2464	-0.0512	0.0458
perimeter upper	-0.0256	0.0835	0.8623	0.0504	-0.0082	1.0000	0.8699	0.1572	-0.0270	**0.9137**	0.0298
area lower	-0.1799	0.3120	**0.9903**	0.2979	-0.1307	0.8699	1.0000	0.3239	-0.0958	0.8721	-0.0309
solidity lower	-0.3717	**0.8684**	0.3163	0.6138	-0.2006	0.1572	0.3239	1.0000	-0.2459	0.0712	-0.0817
eccentricity lower	**0.7034**	-0.2289	-0.1378	-0.2475	0.2464	-0.0270	-0.0958	-0.2459	1.0000	0.0270	0.0576
perimeter lower	-0.0203	0.0990	0.8659	0.1561	-0.0512	**0.9137**	0.8721	0.0712	0.0270	1.0000	0.0238
grade	0.0902	**-0.1136**	-0.0314	-0.1102	0.0458	0.0298	-0.0309	-0.0817	0.0576	0.0238	1.0000

TABLE VIII: Correlation table for symmetry metrics

of 15 samples in the leaves. The labels of our dataset are highly unbalanced: as seen in Section 4.1, only 44 of all the volumes are low grade, which is 30% of the whole dataset. To obviate this, in addition to computing the decision trees baselinestretchas is, for each cross-validation training we downsampled the high grades to 60 volumes, and upsampled the low grades by repeating 30% of the low grade volumes. This way, we obtained a balanced distribution of the two labels, with 60 high grade and 57 low grade, without having to repeat all the low

clustering	cluster#	CoM distance	cluster size
cluster#	1.0000	-0.1276	**-0.5056**
CoM distance	-0.1276	1.0000	**0.4231**
cluster size	**-0.5056**	0.4231	1.0000
grade	-0.0223	**-0.1491**	0.0258

TABLE IX: Correlation table for clustering metrics

grade data. Furthermore, applying this step during cross-validation ensures there will not be a repeated sample in both training and test sets.

Performance of the different decision trees can be seen in Tables Table XIII and Table XIV. When the decision tree is balanced as explained above, these results (Table Table XIII) show that even with the complete set of metrics, T1, T2 and FLAIR modalities are barely above random guessing with respect to the tumor grade. On the other hand, T1c metrics seem to perform well in the classification task, as do symmetry metrics. Figure Figure 16 shows the tree trained on T1c metrics.

On the other hand, training a decision tree with the dataset as is, without any label balancing, produces a very tall tree, disproportionately skewed towards the label 1, that is high grade, as seen in Figure Figure 17. As shown in the second table of Table Table XIV, the performance scores of such trees are very high, but only due to the high ratio of high grade tumors in the

	deviation	**skewness**	**kurtosis**	**GLCM h.**	**spatial h.**
T1	0.0304	**0.0607**	0.0099	0.0074	0
T1c	0	0.0701	0	**0.1165**	0.0912
T2	0.0281	0.0274	**0.0771**	0.0162	0.0469
FLAIR	0	**0.0351**	0	0	0

TABLE X: Mutual Information Score between single modality-dependent metric (indicated in the columns) and the tumor grade.

dataset. As can be seen in the confusion matrices, the correctly labeled Low grades are very few (in the case of T2 even none) when compared to the balanced cases.

metric	**mutual information score**
cluster number	0.0054
CoM distance	0
cluster size	0
eccentricity total	0.0037
solidity total	0.0087
upper area	0.0047
upper solidity	0.013
upper eccentricity	0.0078
upper perimeter	0.3467
lower area	0.0081
lower solidity	0.0107
lower eccentricity	0.0060
lower perimeter	**0.3470**

TABLE XI: Mutual information score between modality-independent metrics and tumor grade.

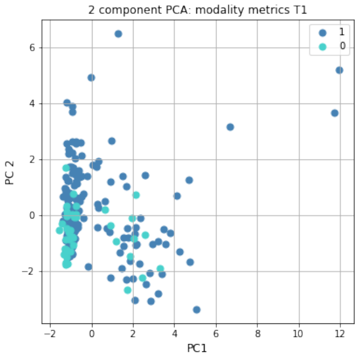
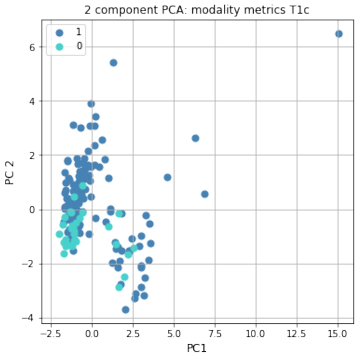
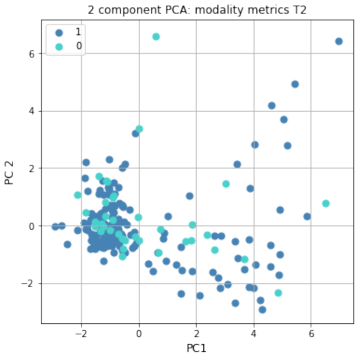
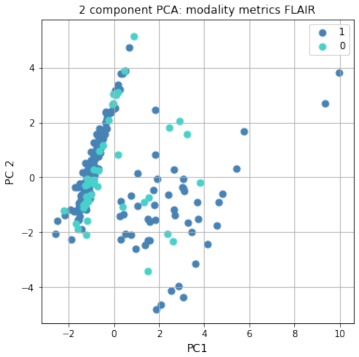
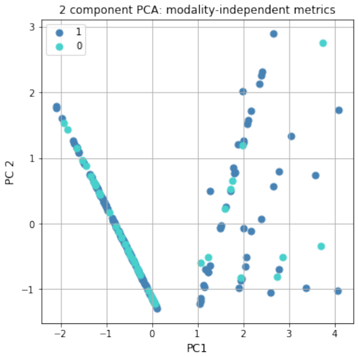
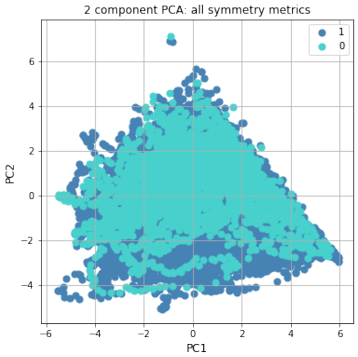
T1	T1c	T2
		
Mutual Information Measure	**Mutual Information Measure**	**Mutual Information Measure**
$\mathrm{MI}_{\mathrm{PC1}} = 0.0534$	$\mathrm{MI}_{\mathrm{PC1}} = 0.0814$	$\mathrm{MI}_{\mathrm{PC1}} = 0.0118$
$\mathrm{MI}_{\mathrm{PC2}} = 0.0120$	$\mathrm{MI}_{\mathrm{PC2}} = 0.1108$	$\mathrm{MI}_{\mathrm{PC2}} = 0.0153$
Explained Variance Per Component	**Explained Variance Per Component**	**Explained Variance Per Component**
$\sigma^2_{\mathrm{PC1}} = 0.3714$	$\sigma^2_{\mathrm{PC1}} = 0.4011$	$\sigma^2_{\mathrm{PC1}} = 0.4256$
$\sigma^2_{\mathrm{PC2}} = 0.2202$	$\sigma^2_{\mathrm{PC2}} = 0.2328$	$\sigma^2_{\mathrm{PC2}} = 0.1736$
$\sigma^2_{\mathrm{total}} = 0.5916$	$\sigma^2_{\mathrm{total}} = 0.6339$	$\sigma^2_{\mathrm{total}} = 0.5992$
FLAIR	**Modality-independent**	**Symmetry**
		
Mutual Information Measure	**Mutual Information Measure**	**Mutual Information Measure**
$\mathrm{MI}_{\mathrm{PC1}} = 0$	$\mathrm{MI}_{\mathrm{PC1}} = 0$	$\mathrm{MI}_{\mathrm{PC1}} = 0.0046$
$\mathrm{MI}_{\mathrm{PC2}} = 0.0025$	$\mathrm{MI}_{\mathrm{PC2}} = 0$	$\mathrm{MI}_{\mathrm{PC2}} = 0.0029$
Explained Variance Per Component	**Explained Variance Per Component**	**Explained Variance Per Component**
$\sigma^2_{\mathrm{PC1}} = 0.3589$	$\sigma^2_{\mathrm{PC1}} = 0.6332$	$\sigma^2_{\mathrm{PC1}} = 0.4259$
$\sigma^2_{\mathrm{PC2}} = 0.2901$	$\sigma^2_{\mathrm{PC2}} = 0.2384$	$\sigma^2_{\mathrm{PC2}} = 0.2713$
$\sigma^2_{\mathrm{total}} = 0.6490$	$\sigma^2_{\mathrm{total}} = 0.8716$	$\sigma^2_{\mathrm{total}} = 0.6972$

TABLE XII: PCA results

balanced	**TH**	**FL**	**FH**	**TL**	**precision**	**recall**	**f1**
T1	115	82	19	29	0.59	0.59	0.55
T1c	146	51	13	35	0.74	0.74	**0.72**
T2	106	91	30	18	0.47	0.51	0.46
FLAIR	104	93	18	30	0.58	0.55	0.50
symmetry	5011	9787	1284	2060	0.64	0.65	0.62

TABLE XIII: Performance measures for balanced Decision Trees classifiers for the different modalities. The left half of the table shows the confusion matrix for each modality, with the number of True High grades (TH), False High grades (FH), False Low grades (FL) and True Low Grades (TL), while the right half displays each decision tree's performance.

unbalanced	**TH**	**FL**	**FH**	**TL**	**precision**	**recall**	**f1**
T1	187	10	34	14	0.88	0.82	0.85
T1c	192	5	20	28	0.92	0.90	0.91
T2	174	23	48	0	0.80	0.71	0.75
FLAIR	188	9	41	7	0.90	0.80	0.84
symmetry	13626	1172	2315	1029	0.85	0.81	0.82

TABLE XIV: Performance measures for unbalanced Decision Trees classifiers for the different modalities.

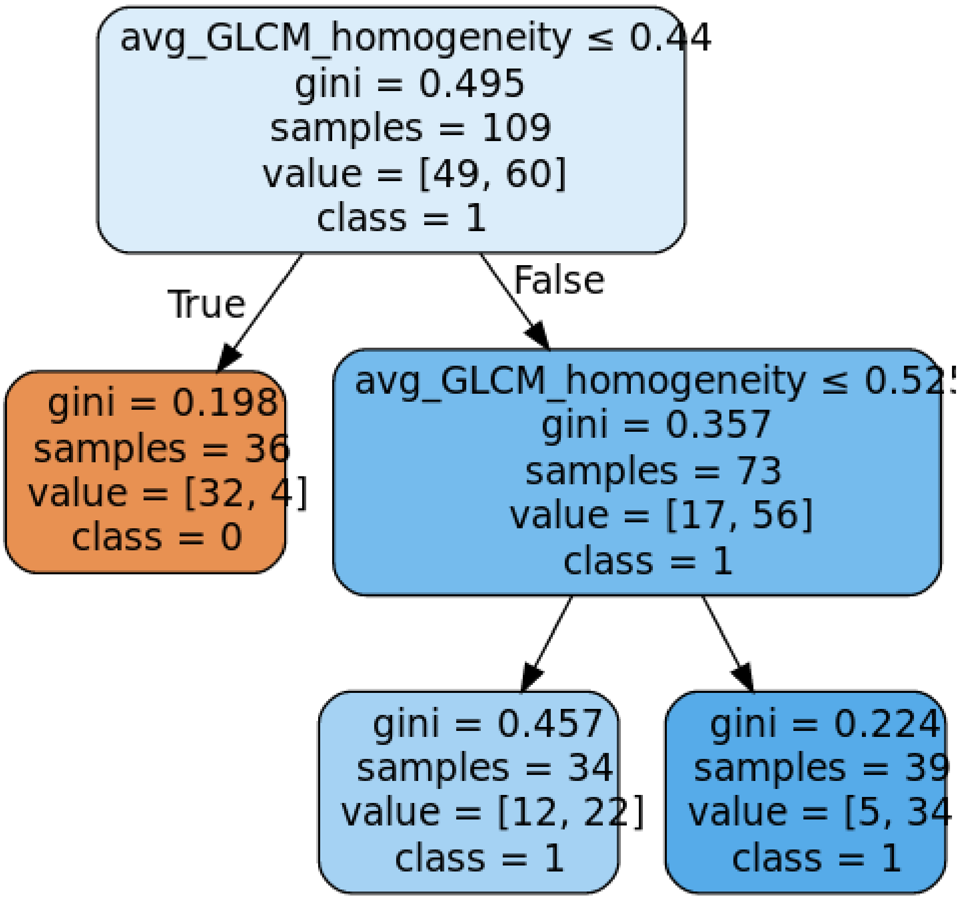


Figure 16: Balanced decision tree computed on the T1c modality. Each node of the tree displays the decision cut, if any, then the gini index for the node. The 'sample' attribute shows the number of samples of that particular node, with class distribution described by 'value' in the form [class = 0, class = 1]. Finally, 'class' is the class of the node, with 0 meaning low grade and 1 high grade. Each node is filled in different colors that display the class (orange for low, blue for high), with intensity in saturation specifying the labeling confidence of that particular node.

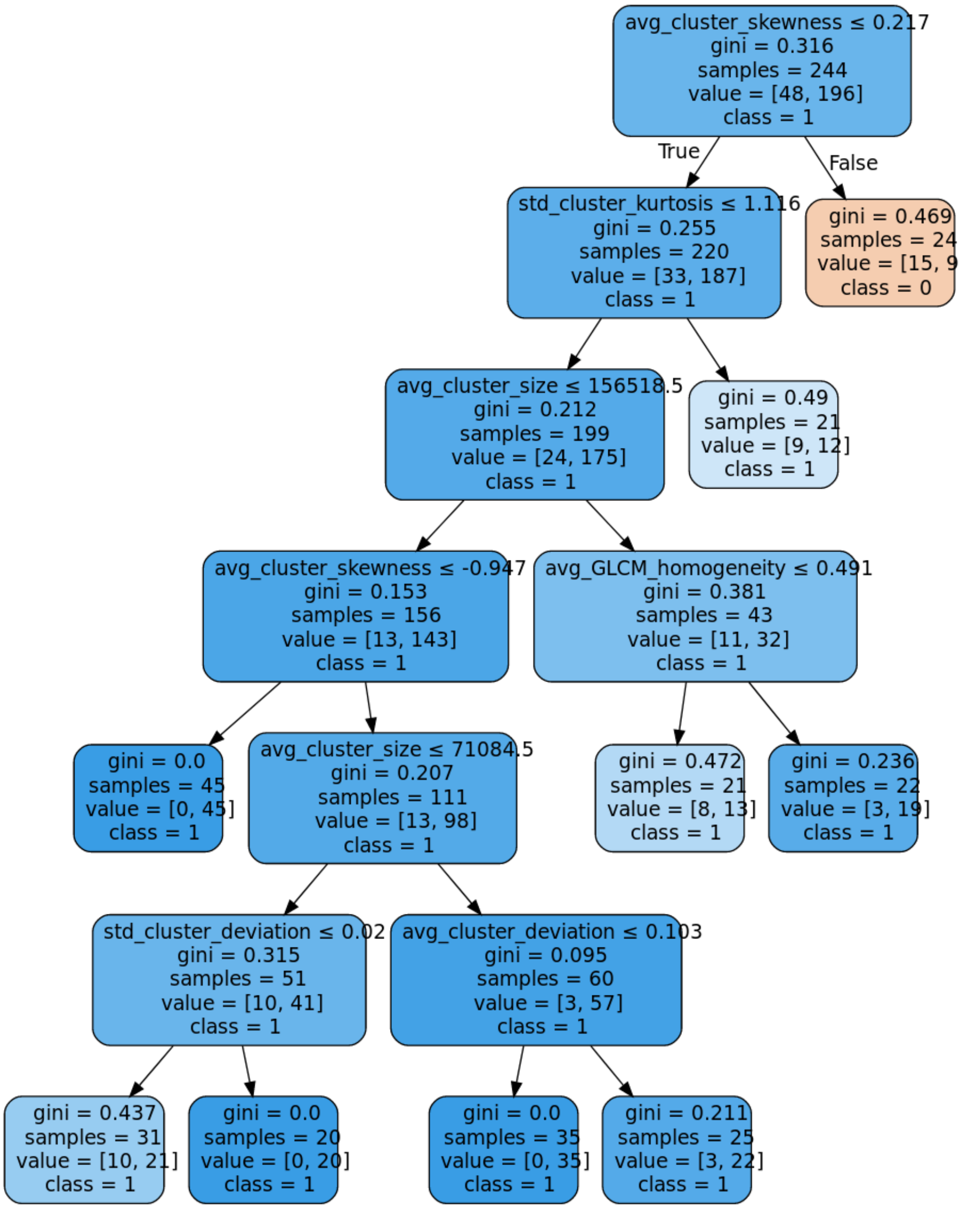


Figure 17: Unbalanced decision tree computed on the T1c modality.

CHAPTER 6

CONCLUSION

In this thesis, we have defined and developed twenty metrics that provide a quantitative description of previously segmented tumor regions within brain MR images. We have first applied DBSCAN to whole MRI volumes, in order to separate the lesion areas into different clusters, then computed the metrics on each cluster, or between the clusters, as appropriate for each metric. Some of the metrics, such as cluster number, cluster size, and symmetry metrics are independent from pixel intensity. The other metrics, namely Standard Deviation, Skewness, Kurtosis, GLCM Homogeneity and Spatial Heterogeneity were computed for each of the four MR modalities provided by the dataset: T1, T1c, T2 and FLAIR.

The goal of these metrics is to provide a quantitative, mathematically sound framework to describe and compare different tumor lesions. We have analysed each metric's relationship with the others to establish which ones were independent, and we have demonstrated their descriptiveness by showcasing meaningful examples.

Furthermore, to deepen our understanding of these metrics' possible future applications, we have investigated whether they can be meaningful with respect to the tumor grade label assigned to each MRI volume. With this purpose in mind, we have computed correlation and mutual information between each individual metric, which did not show any evident relationship between metrics and tumor grade with the exception of metrics computed on the T1c modality.

We also attempted dimensionality reduction by principal component analysis, which did not improve on the mutual information score.

Finally, we trained five Decision Tree classification models with metrics for each modality, as well as modality-independent metrics. In order to perform classification in the most efficient way, we re-balanced the dataset and added a Leave-one-out cross-validation step. We discovered that, while T1, T2 and FLAIR Decision Trees do not perform well, the T1c Decision Tree has a f1 score of 72% on testing data. This suggests that the features computed on the T1c modality could be used, in the future, for tumor label prediction.

6.1 Future Works

The current analysis explained in this thesis can be extended in future works in the following directions:

- **Extend 2d measures to 3d**. As of now spatial heterogeneity, GLCM homogeneity and symmetry metrics are calculated per slice, and then aggregated by weighted average. Future development could try and extend these three metrics to be computed directly on the 3d volume.

- **Compute 2d metrics on different axis**. In this work we computed spatial heterogeneity, GLCM homogeneity and symmetry metrics on the provided z axis slices, which scan a brain volume vertically. It would be interesting, as an added venue of study, to try and compute them on artificial slices on the x or y axis, to see whether the spatial information on those axes could be meaningful.

- **Aggregate symmetry measure**. Our development of symmetry requires a high number of metrics to be computed (10 per slice). We tried aggregating them with PCA, with no useful gain in information. Furthermore, PCA is a black box method that is not useful for human understanding of the metric. It could therefore be useful to develop an aggregate metric that encloses all ten measures in a single, understandable value.

- **Further classification on different datasets**. BRATS2018[1] provides information on overall survivability of the patient together with the brain volume and segmentation. It would be interesting to compute our metrics on such dataset and verify whether they are meaningful with respect to the survivability rate, not only on the tumor grade.

[1] https://www.med.upenn.edu/sbia/brats2018/data.html

CITED LITERATURE

1. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* , 12:2825–2830, 2011.

2. Li, Y., Zhu, Z., Zhou, Y., Xia, Y., Shen, W., Fishman, E. K., and Yuille, A. L.: Volumetric medical image segmentation: A 3d deep coarse-to-fine framework and its adversarial examples. In *Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical Informatics* , pages 69–91. Springer International Publishing, 2019.

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

4. Roth, H. R., Lu, L., Farag, A., Shin, H.-C., Liu, J., Turkbey, E. B., and Summers, R. M.: DeepOrgan: Multi-level deep convolutional networks for automated pancreas segmentation. In *Lecture Notes in Computer Science* , pages 556–564. Springer International Publishing, 2015.

5. Anwar, S. M., Majid, M., Qayyum, A., Awais, M., Alnowami, M., and Khan, M. K.: Medical image analysis using convolutional neural networks: A review. *Journal of Medical Systems* , 42(11), oct 2018.

6. Rokach, L. and Maimon, O.: Clustering methods. In *Data Mining and Knowledge Discovery Handbook* , pages 321–352. Springer-Verlag.

7. Murtagh, F.: Hierarchical clustering. In *International Encyclopedia of Statistical Science* , pages 633–635. Springer Berlin Heidelberg, 2011.

8. Hamerly, G. and Elkan, C.: Alternatives to the k-means algorithm that find better clusterings. In *Proceedings of the eleventh international conference on Information and knowledge management - CIKM '02* , pages 600 – 607. ACM Press, 2002.

9. Macqueen, J.: Some methods for classification and analysis of multivariate observations. *5-TH BERKELEY SYMPOSIUM ON MATHEMATICAL STATISTICS AND PROBABILITY* , pages 281–297, 1967.

10. Kriegel, H.-P., Kröger, P., Sander, J., and Zimek, A.: Density-based clustering. *WIREs Data Mining and Knowledge Discovery* , 1(3):231–240, apr 2011.

11. Ram, A., Jalal, S., Jalal, A. S., and Kumar, M.: A density based algorithm for discovering density varied clusters in large spatial databases. *International Journal of Computer Applications* , 3(6):1–4, jun 2010.

12. Luc, P., Couprie, C., Chintala, S., and Verbeek, J.: Semantic segmentation using adversarial networks.

13. Dar, A. S. and Padha, D.: Medical image segmentation a review of recent techniques, advancements and a comprehensive comparison. *International Journal of Computer Sciences and Engineering* , 7(7):114–124, jul 2019.

14. Otsu, N.: A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics* , 9(1):62–66, jan 1979.

15. Feng, Y., Zhao, H., Li, X., Zhang, X., and Li, H.: A multi-scale 3d otsu thresholding algorithm for medical image segmentation. *Digital Signal Processing* , 60:186–199, jan 2017.

16. Szeliski, R.: *Computer Vision* . Springer-Verlag GmbH, 2010.

17. Canny, J.: A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , PAMI-8(6):679–698, nov 1986.

18. Hesamian, M. H., Jia, W., He, X., and Kennedy, P.: Deep learning techniques for medical image segmentation: Achievements and challenges. *Journal of Digital Imaging* , 32(4):582–596, may 2019.

19. Long, J., Shelhamer, E., and Darrell, T.: Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , pages 640 – 651. IEEE, jun 2015.

20. Ronneberger, O., Fischer, P., and Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In *Lecture Notes in Computer Science* , pages 234–241. Springer International Publishing, 2015.

21. Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., and Larochelle, H.: Brain tumor segmentation with deep neural networks. *Medical Image Analysis* , 35:18–31, jan 2017.

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

23. Moeskops, P., Wolterink, J. M., van der Velden, B. H. M., Gilhuijs, K. G. A., Leiner, T., Viergever, M. A., and Išgum, I.: Deep learning for multi-task medical image segmentation in multiple modalities. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016* , pages 478–486. Springer International Publishing, 2016.

24. Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y.: Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2* , NIPS'14, page 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

25. Xue, Y., Xu, T., Zhang, H., Long, L. R., and Huang, X.: SegAN: Adversarial network with multi-scale l1 loss for medical image segmentation. *Neuroinformatics* , 16(3-4):383–392, may 2018.

26. Goodenberger, M. L. and Jenkins, R. B.: Genetics of adult glioma. *Cancer Genetics* , 205(12):613–621, dec 2012.

27. Louis, D. N., Perry, A., Reifenberger, G., von Deimling, A., Figarella-Branger, D., Cavenee, W. K., Ohgaki, H., Wiestler, O. D., Kleihues, P., and Ellison, D. W.: The 2016 world health organization classification of tumors of the central nervous system: a summary. *Acta Neuropathologica* , 131(6):803–820, may 2016.

28. Brooks, F. J. and Grigsby, P. W.: Quantification of heterogeneity observed in medical images. *BMC Medical Imaging* , 13(1):7, mar 2013.

29. Haralick, R. M., Shanmugam, K., and Dinstein, I.: Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics* , SMC-3(6):610–621, nov 1973.

30. van der Walt, S., Colbert, S. C., and Varoquaux, G.: The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering* , 13(2):22–30, mar 2011.

31. Virtanen, P., , Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., and van Mulbregt, P.: SciPy 1.0: fundamental algorithms for scientific computing in python. *Nature Methods* , 17(3):261–272, feb 2020.

32. van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T.: scikit-image: image processing in python. *PeerJ* , 2:e453, jun 2014.

33. Eisenhauer, E., Therasse, P., Bogaerts, J., Schwartz, L., Sargent, D., Ford, R., Dancey, J., Arbuck, S., Gwyther, S., Mooney, M., Rubinstein, L., Shankar, L., Dodd, L., Kaplan, R., Lacombe, D., and Verweij, J.: New response evaluation criteria in solid tumours: Revised RECIST guideline (version 1.1). *European Journal of Cancer* , 45(2):228–247, jan 2009.

34. Wen, P. Y., Macdonald, D. R., Reardon, D. A., Cloughesy, T. F., Sorensen, A. G., Galanis, E., DeGroot, J., Wick, W., Gilbert, M. R., Lassman, A. B., Tsien, C., Mikkelsen, T., Wong, E. T., Chamberlain, M. C., Stupp, R., Lamborn, K. R., Vogelbaum, M. A., van den Bent, M. J., and Chang, S. M.: Updated response assessment criteria for high-grade gliomas: Response assessment in neuro-oncology working group. *Journal of Clinical Oncology* , 28(11):1963–1972, apr 2010.

VITA

NAME:	MADDALENA ANDREOLI ANDREONI
EDUCATION:	B.Eng., Computer Engineering, Politecnico di Milano, Italy, 2016.