Statistical Neuroimaging

Joel Winterton

November 2023

Contents

Chapter 1

Segmentation via Weighted Aggregation

1.1 Objective - Why we want to do this

The objective of using the method of Segmentation via Weighted Aggregation in this context, is to extract a range of features from each aggregate, and then train a random forest on all of the features we obtain (from multiple images), after feature extraction, the classifier is agnostic to the image.

So we are training our random forest on a dataset

$$\mathbf{f} = \{f_1, ..., f_m\}$$

where each f_i is a vector of features extracted on an aggregate indexed as aggregate i.

Any complication that arise from interpolation e.t.c. arise because we are aiming to keep linear time complexity.

Could try exploring a more intuitive, less efficient segmentation algorithm first?

1.2 Overview - How we're going to do it

This is a more in detail version of [6].

1. Make a graph of the image $G^{(1)}$, each pixel is a node and edges are between neighbouring pixels.
2. Edge weights are called an affinity, lower affinity means pixels are more "different" in some sense.

3. Aim to construct a new graph which is a coarsening of the old one, by first choosing which nodes in the graph to keep (the seeds).
4. Then decide how to interpolate the values of the non-seeds (values being for instance intensity) into the new seeds.
5. Determine what the edge weights of the new coarse graph is.
6. Calculate and store the features of each node in the new coarse graph, these new nodes are also referred to as "aggregates".
7. Repeat this coarsening process.

1.3 Nitty Gritty

1.3.1 Graph Initialisation

Covering steps 1 and 2. Let $G(V, E)$ be a graph where each vertex is a pixel in the image. In implementation it is easiest to do this by denoting each node by it's position in the image with a "nice" origin (a top corner for instance). Two nodes are adjacent in this graph if they are neighbours (horizontally, vertically or diagonally) in the image.

Then for any edge $(u, v) \in E$ define the weight function as an affinity:

$$w_{uv} = \exp\left(-d(u, v; \alpha)\right)$$

where d is some distance metric of the nodes. [3] uses the Euclidean distance of intensities of the pixels as the distance metric:

$$d(u, v) = \alpha |I_u - I_v|_2$$

This graph doesn't have to represent a singular image, and this distance metric can be used to include multiple images (referred to as multi-channel) [1]. This is useful as MR scans can have several channels (T-1, T-2 and FLAIR being the common channels).

The next few steps only require the previous graph, so assume that we have a graph $G^{(i-1)}(V^{(i-1)}, E^{(i-1)})$, where $n^{(i-1)} = |V^{(i-1)}|$, and we are aiming to construct a coarser graph $G^{(i)}(V^{(i)}, E^{(i)})$.

1.3.2 Seed selection

Covering step 3-5 The idea here is to group up nodes of the previous graph in a way that maintains some sort of strong coupling between the nodes being merged together, this group of nodes will be referred to as a segment. This can be thought of in terms of graph cuts. We are aiming to find cuts of a graph that maximise some similarity between nodes inside cliques, while keeping the difference between nodes adjacent across a cut high. For any segment $S \subset V^{(i-1)}$ let $\mathbf{u} = (u_1, ..., u_{n^{(i-1)}})$ denote the segment, where

$$u_i = \begin{cases} 1 & \text{if } v_i \in S \\ 0 & \text{else} \end{cases}$$

We sum the weights of all the edges on the boundary of the segment using the fact that $(u_i - u_j)^2 = 1$ if nodes are on the boundary of the segment, and everywhere else $(u_i - u_j)^2 = 0$ everywhere else. Hence the sum of weights on the boundary is given by:

$$\sum_{i>j} w_{ij}(u_i - u_j)^2$$

We then sum over all edge weights inside segment, using that $u_i u_j = 1$ if both nodes are in segment and $u_i u_j = 0$ otherwise. Hence the "weight volume" is given by:

$$\sum_{i>j} w_{ij} u_i u_j$$

We can then define the so-called saliency of a segment $\Gamma(\mathbf{u})$, which is the sum of all boundary weights normalised by the weight volume:

$$\Gamma(\mathbf{u}) = \frac{\sum_{i>j} w_{ij}(u_i - u_j)^2}{\sum_{i>j} w_{ij} u_i u_j}$$

This can be expressed more compactly using the Graph Laplacian matrix L with entries

$$L_{ij} = \begin{cases} \sum_{k:k \neq i} w_{ik} & \text{if } i = j \\ w_{ij} & \text{else} \end{cases}$$

along with the weight matrix W with entries

$$W_{ij} = \begin{cases} w_{ij} & \text{if } (v_i, v_j) \in E \\ 0 & \text{else} \end{cases}$$

Hence

$$\Gamma(\mathbf{u}) = \frac{\mathbf{u}^T L \mathbf{u}}{\frac{1}{2}\mathbf{u}^T W \mathbf{u}}$$

This saliency can then be used to find segments, since low values of this saliency function represent strongly coupled nodes. To determine low values of Γ we let the entries of $\mathbf{u}$ take real values instead of binary values. To minimise this, we use the Rayleigh quotient [4]. Define

$$R(\mathbf{x}, A) = \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}$$

Note that if we let

$$\mathbf{y} = \frac{\mathbf{x}}{||\mathbf{x}||_2}$$

so that $||\mathbf{y}||_2^2 = 1$, then

$$R(\mathbf{x}, A) = \frac{\frac{1}{||\mathbf{x}||_2^2}\mathbf{x}^T A\mathbf{x}}{\frac{1}{||\mathbf{x}||_2^2}\mathbf{x}^T\mathbf{x}} = R(\mathbf{y}, A)$$

so when minimising this quantity over $\mathbf{x}$, we can add this constraint:

$$\min_{\mathbf{x}} \; R(\mathbf{x}, A) \equiv \min_{\mathbf{x}:\, ||\mathbf{x}||_2^2=1} \; R(\mathbf{x}, A)$$

then plugging in this constraint into R to simplify the denominator gives the optimisation problem:

$$\min_{\mathbf{x}:\, \mathbf{x}^T\mathbf{x}=1} \; \mathbf{x}^T A\mathbf{x}$$

Then using the Lagrangian method of multipliers [2] we have Lagrangian:

$$\mathcal{L} = \mathbf{x}^T A\mathbf{x} - \lambda(\mathbf{x}^T\mathbf{x} - 1) \implies \nabla_{\mathbf{x}}\mathcal{L} = 2A\mathbf{x} - 2\lambda\mathbf{x}$$

and so the minimisation problem is equivalent to the eigen-problem:

$$A\mathbf{x} = \lambda\mathbf{x}$$

for the smallest eigenvalue λ of A. Now consider the generalised Rayleigh quotient:

$$R_G(\mathbf{x}, A_G, B_G) = \frac{\mathbf{x}^T A_G\mathbf{x}}{\mathbf{x}^T B_G\mathbf{x}} = R(\frac{\mathbf{x}}{||\mathbf{x}||_2})$$

Assuming B is symmetric, positive-definite it has a unique Cholesky decomposition [5] $B_G = LL^T$ where L is an upper triangular matrix. So we can write:

$$R_G(\mathbf{x}, A_G, B_G) = \frac{\mathbf{x}^T A_G\mathbf{x}}{(L^T\mathbf{x})^T(L^T\mathbf{x})}$$

And so now we can use a substitution of $A_G = L^{-1}A(L^T)^{-1}$ then:

$$R_G(\mathbf{x}, A_G, B_G) = \frac{(L^T\mathbf{x})^T A(L^T\mathbf{x})}{(L^T\mathbf{x})^T(L^T\mathbf{x})} = R(L^T\mathbf{x}, L^{-1}A(L^T)^{-1})$$

And so to minimise the generalised Rayleigh quotient we can use:

$$\min_{\mathbf{x}} R_G(\mathbf{x}, A_G, B_G) \equiv \min_{\mathbf{x}:\, \mathbf{x}^T\mathbf{x}=1} R_G(\mathbf{x}, A_G, B_G)$$

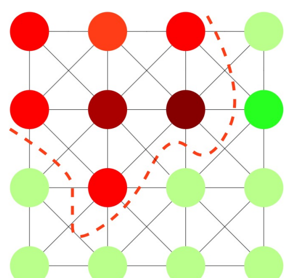

Figure 1.1: Example of a graph cut with high saliency.

Chapter 2

Introduction

The current general aim of the project is to explore and implement methods that identify brain lesions from an MRI scan. The dataset that we have is 36 people, and several other datasets are similarly sized.

2.1 Random Forests

The primary method to be explored is using Random Forests.

- There will need to be significant changes to learning process so as to be applicable to 2D images (or indeed 3D images).
- Another complication is that the images are anisotropic, and so either pre-processing correction is needed, or the learning process will need to take this into account.

Chapter 3

Rough notes

3.1 General resources

3.1.1 General Imaging and MRI Pipeline

Need to construct a general overview medical imaging, maybe going into slightly more detail about MRI scan pipeline.

3.1.2 MS Lesion Segmentation Pipeline

For MS lesion segmentation pipeline, can use elpful overview from Survey of automated multiple sclerosis lesion segmentation techniques on magnetic resonance imaging

3.2 Isolated Concepts

3.2.1 Random Forests

Need to understand what "Random forests require heirarchy" means and how this can be applied to images (what does it mean to extract heirarchy from an image)

3.2.2 Segmentation and Heirarchy

Explore Segmentation by Weighted Aggregation method. Explore how heirarchy is obtained from an image.

3.3 Scale Space / Smoothing

Scale space seems to be deeply connected to the smoothing of an image using a Gaussian Kernel. **Todo**

- Explore and understand image scale space, along with the motivating idea behind segmentation.
- Explore scale space segmentation Wikipedia page.
- Explore Segmentation by Weighted Aggregation method.

3.4 Todo

Then reattempt to understand some of Automatic Segmentation and Classification of Multiple Sclerosis in Multichannel MRI. The main heuristic of this paper is to do MS lesion segmentation in MRI scans by training Random Forests at a multiscale level. Then explore the more advanced version of this: Spatially Adaptive Random Forests Spatially Adaptive Random Forests .

Chapter 4

Question

Bibliography

[1] A. Akselrod-Ballin, M. Galun, J.M. Gomori, M. Filippi, P. Valsasina, R. Basri, and A. Brandt. Automatic segmentation and classification of multiple sclerosis in multichannel mri. *IEEE Transactions on Biomedical Engineering*, 56(10):2461–2469, Oct 2009.

[2] Yadolah Dodge. Lagrange multiplier. *Springer eBooks*, page 292–294, Feb 2008.

[3] Galun, Eitan Sharon, Ronen Basri, and Laura Brandt. Texture segmentation by multiscale aggregation of filter responses and shape elements. *CiteSeer X (The Pennsylvania State University)*, Jan 2003.

[4] Benyamin Ghojogh, Bghojogh@uwaterloo Ca, Fakhri Karray, Karray@uwaterloo Ca, Mark Crowley, and Mcrowley@uwaterloo Ca. *Eigenvalue and Generalized Eigenvalue Problems: Tutorial.* 2022.

[5] Nicholas J. Higham. Cholesky factorization. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(2):251–254, Jun 2009.

[6] Eitan Sharon, Meirav Galun, Dahlia Sharon, Ronen Basri, and Achi Brandt. Hierarchy and adaptivity in segmenting visual scenes. *Nature*, 442(7104):810–813, Jun 2006.