Medical Image Segmentation and Applications - Lab 2

Medical Image Segmentation and Applications: EM Algorithm Implementation Using Python

Mahdi Islam, Musarrat Tabassum

1. Introduction and problem definition

This report focuses on the implementation of an Expectation-Maximization (EM) algorithm to segment brain tissues into Grey Matter (GM), White Matter (WM), and Cerebrospinal Fluid (CSF) from MRI images. The goal is to preprocess brain MRI scans using T1 and T2_FLAIR modalities, apply clustering, and evaluate the segmentation results using Dice similarity coefficients.

2. Algorithm analysis / Design and implementation of the proposed solution

To achieve brain tissue segmentation, the EM algorithm is used after clustering the intensity values from the T1 and T2_FLAIR images. The key steps in the algorithm are:

1. **Preprocessing**:

   - **Normalization**: Normalizing both the T1 and FLAIR intensities ensures consistent intensity ranges across different scans.

2. **Clustering and EM Algorithm**:

   - **KMeans Initialization**: KMeans is applied to initialize the cluster centers for Grey Matter (GM), White Matter (WM), and Cerebrospinal Fluid (CSF).

   - **EM Algorithm**: The EM algorithm refines the cluster assignments and estimates the parameters (mean, variance, mixing coefficients) of each tissue type based on the voxel intensities.

   - **Cluster Sorting**: The clusters are sorted by their mean intensities to assign labels (1: CSF, 2: GM, 3: WM) consistently across different scans.

3. **Evaluation**:

   - **Dice Similarity Coefficient**: The performance of the segmentation is evaluated by comparing the predicted labels to ground truth masks, using the Dice similarity coefficient.

Code Snippet:

Main EM Algorithm

```python
1  # Expectation-Maximization (EM) algorithm
2  def em_algorithm(data, mu, sigma, pi, max_iter=100, tol=1e-6):
3
4      n_samples, n_features = data.shape
5      n_clusters = mu.shape[0]
6
7      # Initialize the responsibilities matrix (E-step)
8      responsibilities = np.zeros((n_samples, n_clusters))
9
10     log_likelihoods = []
11
12     for iter in range(max_iter):
13       ### E-Step: Compute the responsibilities
14       for i in range(n_clusters):
15           responsibilities[:, i] = pi[i] * gaussian_pdf(data, mu[i], np.diag(sigma[i]**2))
16
17       # Normalize responsibilities for each data point
18       responsibilities /= np.sum(responsibilities, axis=1, keepdims=True)
19
20       ### M-Step: Update parameters (mu, sigma, pi)
21       N_k = responsibilities.sum(axis=0)  # Total responsibility assigned to each cluster
22
23       # Update mu (means)
24       for i in range(n_clusters):
25           mu[i] = (responsibilities[:, i].reshape(-1, 1) * data).sum(axis=0) / N_k[i]
26
27       # Update sigma (standard deviations)
28       for i in range(n_clusters):
29             diff = data - mu[i]
30             sigma[i] = np.sqrt((responsibilities[:, i].reshape(-1, 1) * diff**2).sum(axis=0) / N_k[i])
31
32       # Update pi (mixing coefficients)
33       pi = N_k / n_samples
34
35       ### Compute the log likelihood for convergence check
36       log_likelihood = np.sum(np.log(np.sum([pi[k] * gaussian_pdf(data, mu[k], np.diag(sigma[k]**2))
37                                   for k in range(n_clusters)], axis=0)))
38       log_likelihoods.append(log_likelihood)
39
40       # Check for convergence
41       if iter > 0 and np.abs(log_likelihoods[-1] - log_likelihoods[-2]) < tol:
42           break
43
44     return mu, sigma, pi, log_likelihoods, responsibilities
```

Figure 01: EM algorithm

3. Experimental section and result analysis

a) Dice Score Evaluation

To evaluate the segmentation accuracy, we calculated the Dice similarity coefficient for each tissue type (GM, WM, CSF) compared to the ground truth labels.

Table: Dice Scores for Brain Tissue Segmentation

Folder	Mean Dice T1	Std Dice T1	GM Dice T1	WM Dice T1	CSF Dice T1
1	0.846952	0.029479	0.805776	0.861886	0.873193
2	0.529085	0.232102	0.240015	0.538951	0.808290
3	0.027753	0.038013	0.788471	0.824306	0.856418
4	0.019769	0.021343	0.817770	0.862687	0.855897
5	0.881326	0.011550	0.865689	0.893235	0.885054

Dice Score Calculation Code:

```python
def dice_score(pred, true, label):
    """
    Calculate the Dice similarity coefficient for a specific label.

    Args:
        pred (ndarray): Predicted segmentation mask.
        true (ndarray): Ground truth mask.
        label (int): Label for which to calculate the Dice score.

    Returns:
        float: Dice similarity coefficient.
    """
    pred_mask = (pred == label)
    true_mask = (true == label)

    intersection = np.sum(pred_mask & true_mask)
    pred_sum = np.sum(pred_mask)
    true_sum = np.sum(true_mask)

    if pred_sum + true_sum == 0:
        return 1.0  # Avoid division by zero if both are empty
    return (2.0 * intersection) / (pred_sum + true_sum)
```

Figure 02: Dice Score Calculation

b) Quantitative and Qualitative Analysis

- **Quantitative Results:** The Dice scores for the T1 modality show overall better segmentation performance compared to the T2 FLAIR modality. Specifically:

  - Grey Matter (GM): The mean Dice score for GM segmentation varies across folders, with Folder 1 having a score of 0.806, Folder 3 at 0.788, Folder 4 at 0.818, and Folder 5 achieving the highest score at 0.866. However, Folder 2 shows a much lower GM Dice score of 0.241, indicating potential issues with segmentation accuracy for this tissue in that folder.

  - White Matter (WM): The WM Dice scores are consistently higher across most folders, with values ranging from 0.824 to 0.893 for Folders 1, 3, 4, and 5, indicating strong segmentation performance for WM. Similar to GM, Folder 2 shows a much lower WM Dice score of 0.539, suggesting that this folder presents challenges in accurately segmenting white matter.

  - Cerebrospinal Fluid (CSF): The CSF Dice scores are generally high, with Folder 1 at 0.873, Folder 3 at 0.856, Folder 4 at 0.856, and Folder 5 at 0.885. Again, Folder 2 shows a slightly

lower CSF Dice score of 0.805 but still performs reasonably well compared to GM and WM in the same folder.

- Overall Performance: The mean Dice scores for each folder reflect the overall segmentation performance, with Folders 1, 3, 4, and 5 showing strong results (ranging from 0.823 to 0.881), and Folder 2 demonstrating significantly lower performance with a mean Dice score of 0.528. The higher standard deviation for Folder 2 (0.283) also suggests greater variability and less consistency in the segmentation for this particular folder, likely contributing to the lower scores for GM and WM. However, the reason behind the lower scores for Folder 2 images is discussed below.

c) **Analysis of Performance for Folder 2 Images**:

Upon visualizing the images for Folder 2, we observed that the T2_FLAIR image contains a lesion with relatively higher intensity than the surrounding tissues, whereas the lesion is not visible on the T1 image. This intensity discrepancy likely caused the EM algorithm to misclassify the lesion as white matter (WM), leading to poor segmentation performance.

To investigate further, we ran a separate trial using only the T1 image in the EM algorithm. This resulted in significantly improved Dice scores, with **0.853** for CSF, **0.802** for GM, and **0.774** for WM. These improved scores suggest that the lesion in the T2_FLAIR image was indeed the primary factor behind the lower performance, as it confused the algorithm into mislabeling the lesion as WM.

4. Project management details

The project was divided into several tasks:

- **Preprocessing (Normalization)**: 1 day

- **KMeans Initialization**: 1 day

- **EM Algorithm Implementation**: 8 days

- **Evaluation (Dice Score Calculation)**: 2 days

5. Conclusion

In this lab, we successfully implemented a brain tissue segmentation pipeline that preprocesses brain MRI images using skull stripping and bias field correction, followed by segmentation into Grey Matter, White Matter, and CSF. The evaluation metrics (Dice scores) indicate good performance, particularly for Grey and White Matter. Further refinement of the CSF segmentation is suggested to improve accuracy.