CS736 Assignment 2

Hastyn Doshi, Kartik Gokhale & Sarthak Mittal

March 13, 2022

Contents

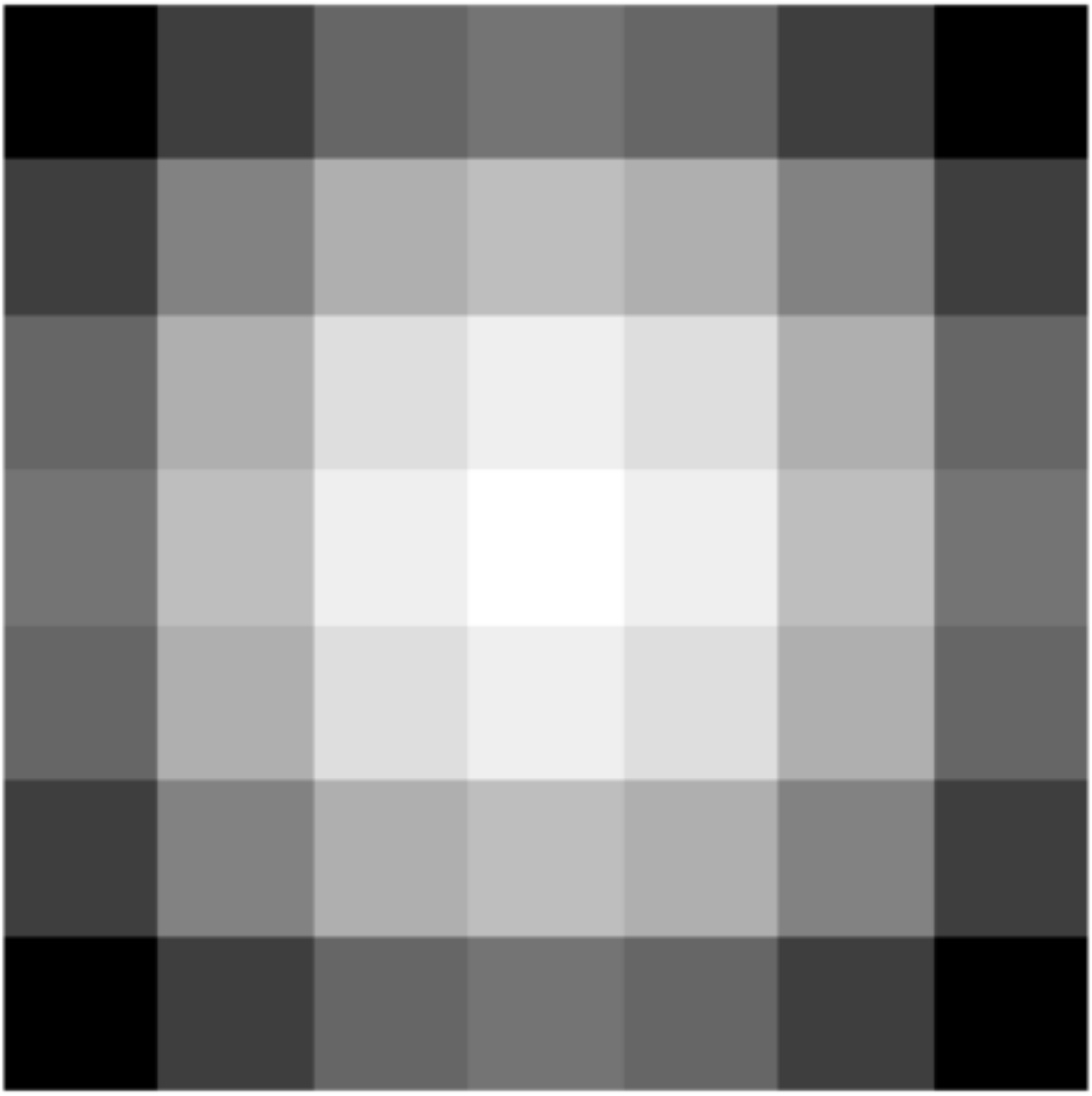
1 Segmenting a Brain MRI (FCM)

1.1 Fuzziness Parameter (q)

The chosen value of the fuzziness parameter q is 2.8.

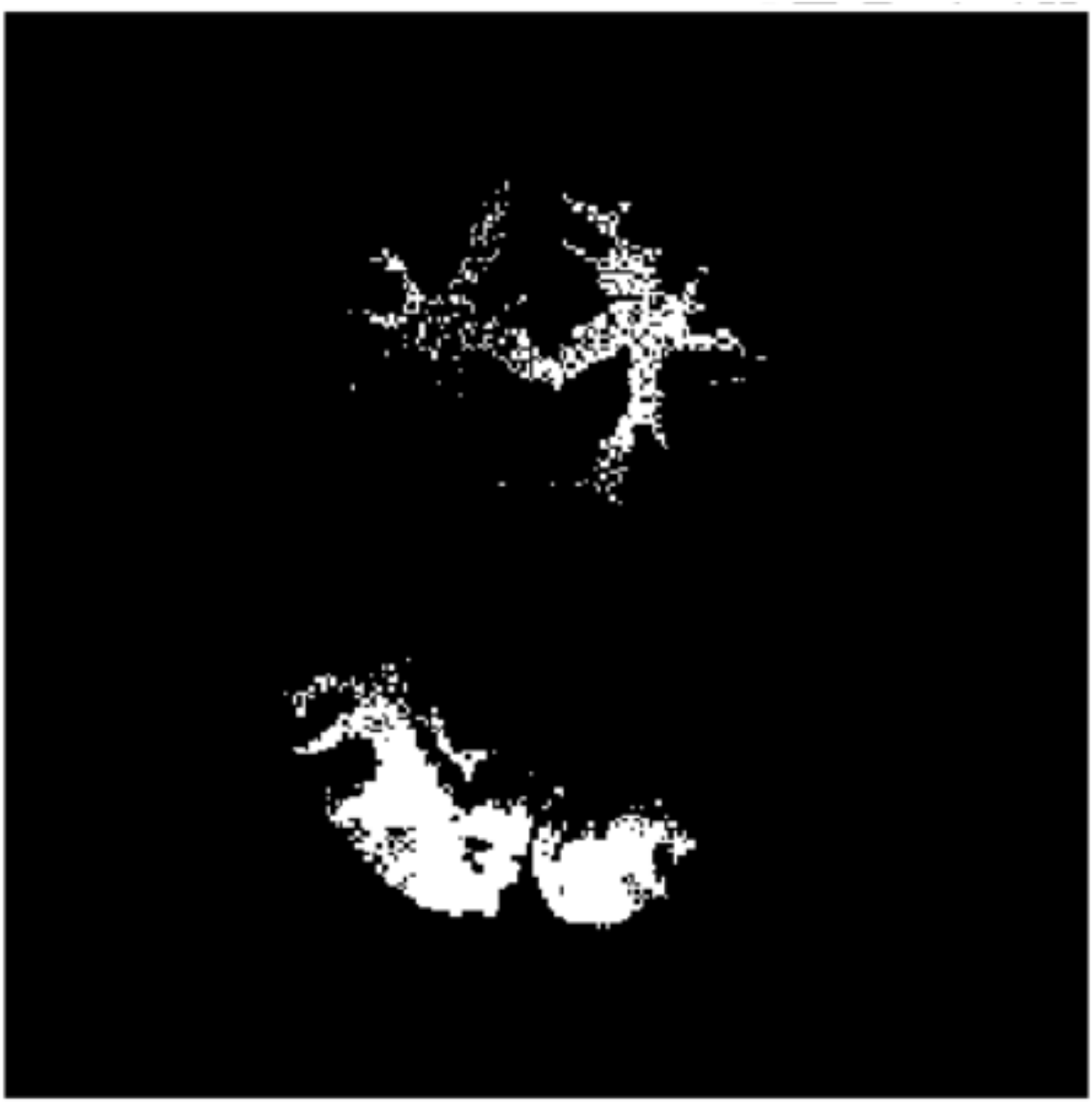
1.2 Neighborhood Mask (w)

The 7×7 neighborhood mask w_{ij} is represented as:



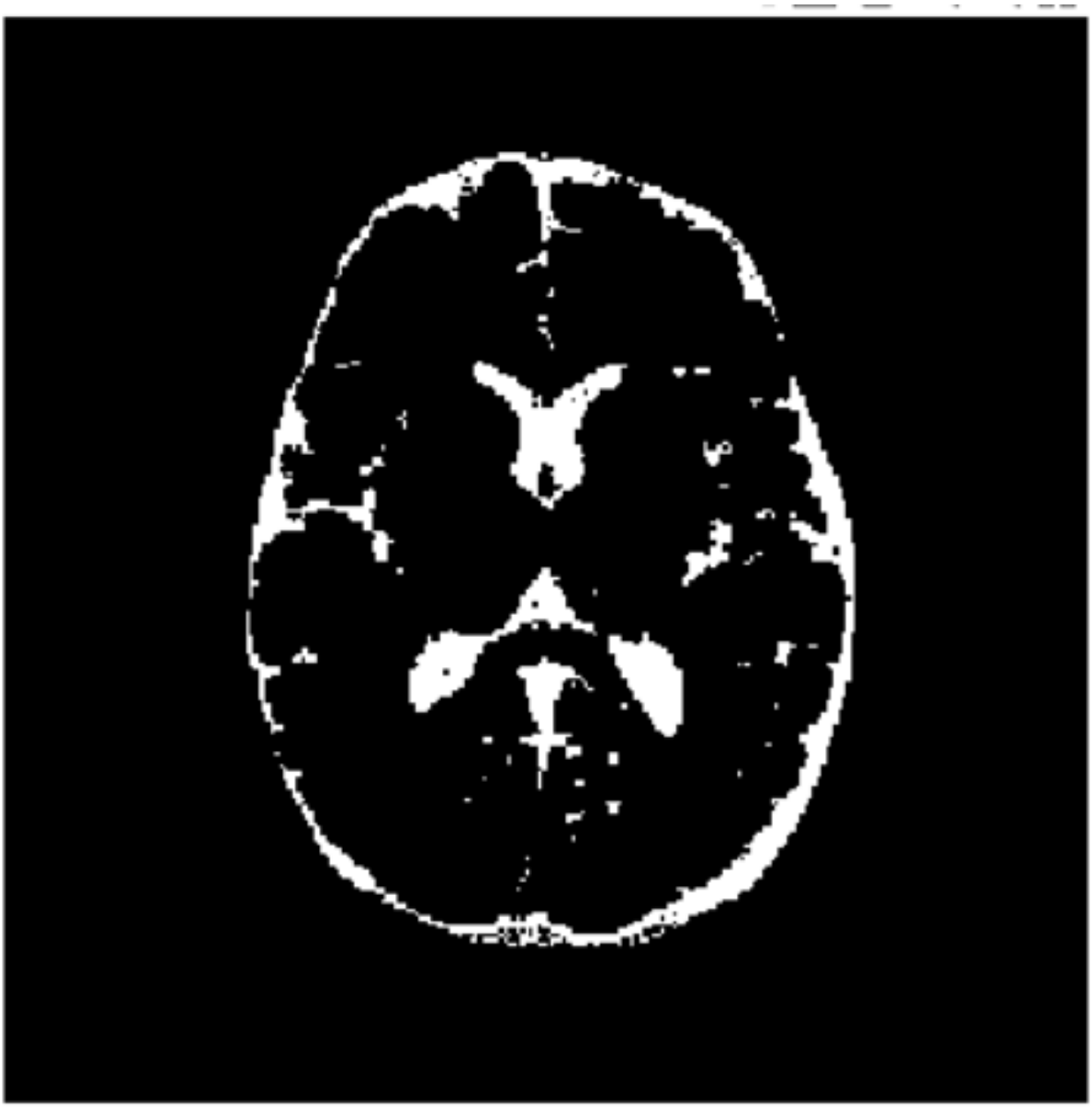
1.3 Initial Memberships

1.3.1 White Matter



1.3.2 Gray Matter

1.3.3 Cerebrospinal Fluid



1.3.4 Motivation

We decide the membership based on which partition the value lies in.

1.3.5 Algorithm

We initialise the memberships as binary values depending on the partition the pixel values lie in.

1.4 Initial Class Means

1.4.1 Values

Class	Value
White Matter	0.8
Gray Matter	0.5
Cerebrospinal Fluid	0.2

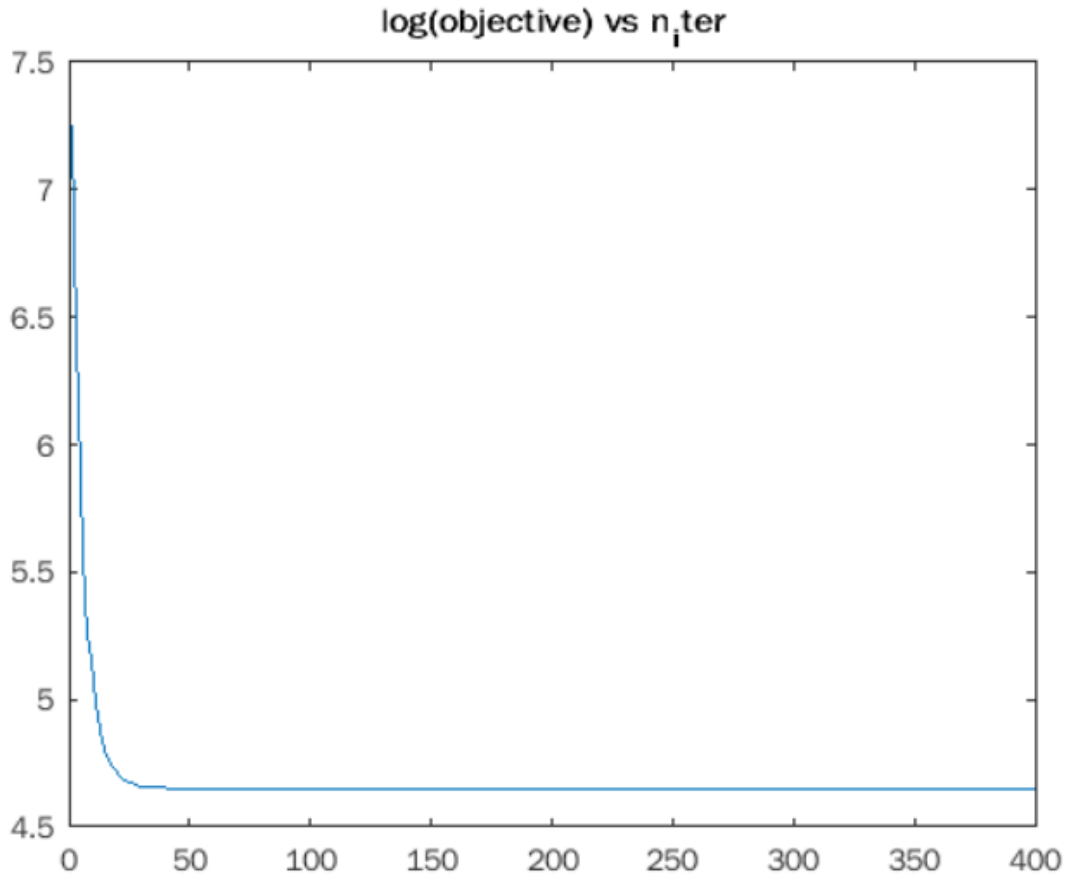
1.4.2 Motivation

We intuitively concluded that the values will roughly obey the order Cerebrospinal ¡ Gray ¡ White.

1.4.3 Algorithm

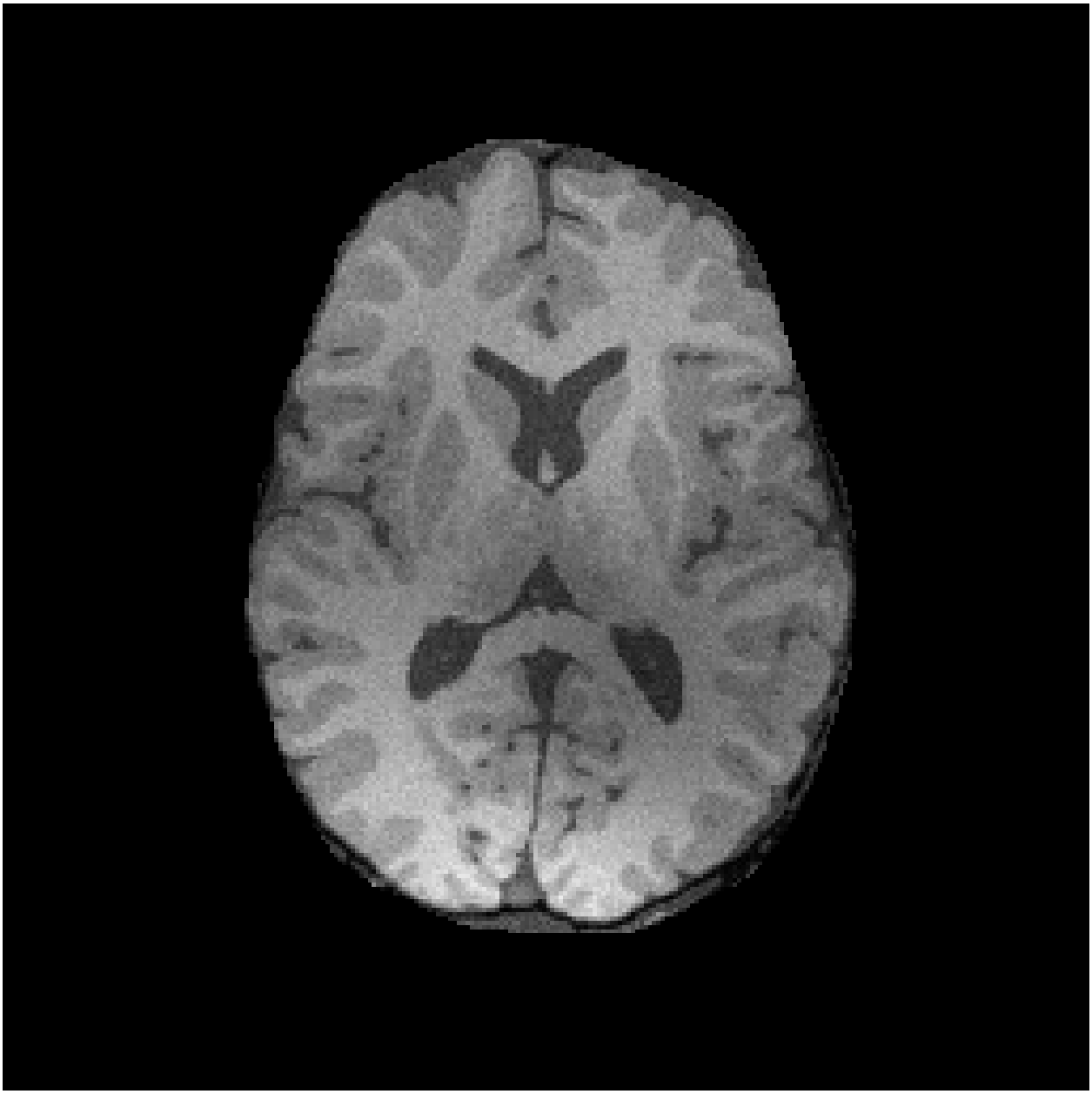
We sorted values, and then divided it into 3 segments that were roughly of equal size, and took average for the class means.

1.5 Objective Function



1.6 Images

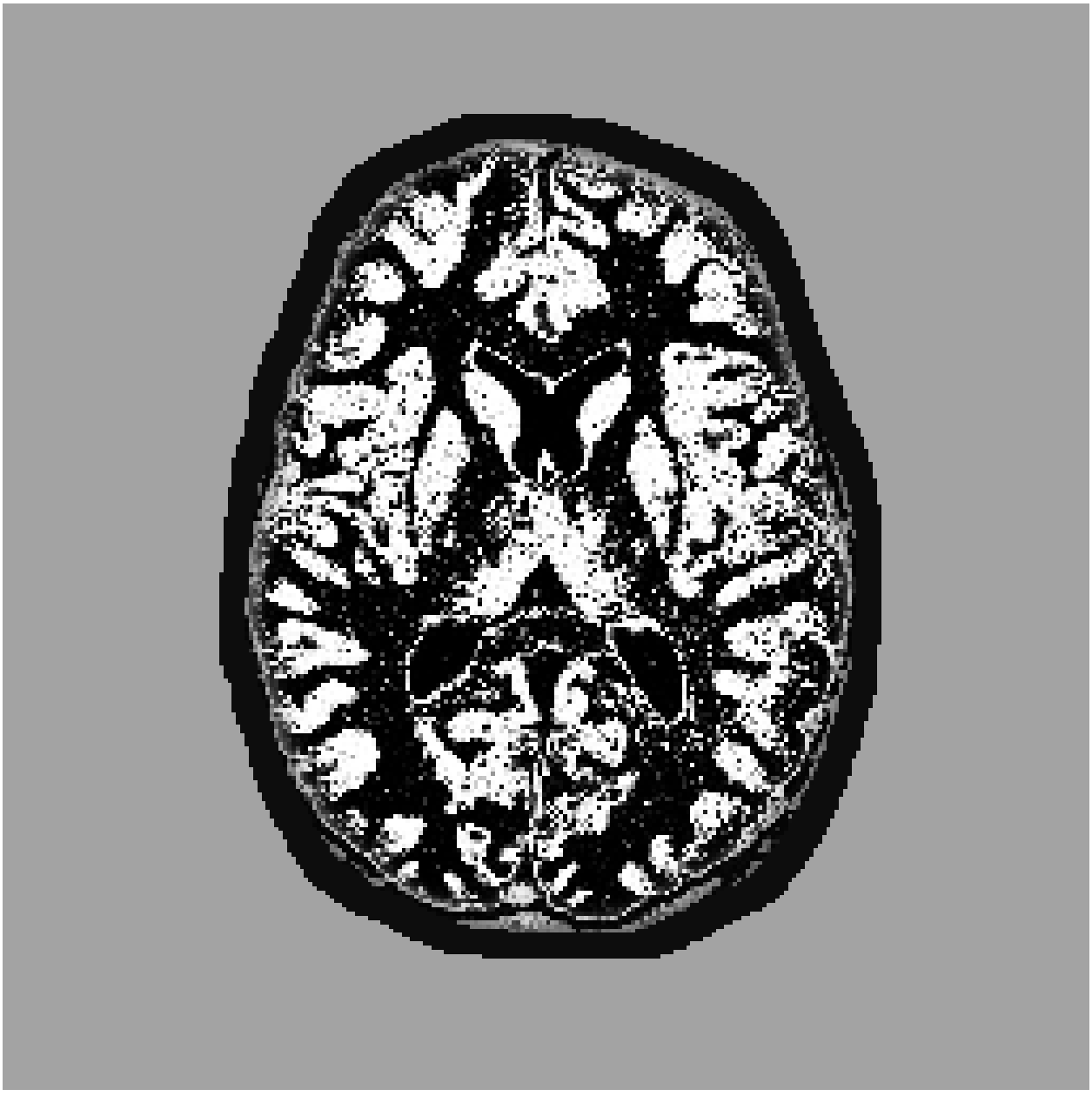
1.6.1 Corrupted Image



1.6.2 Optimal Class Memberships

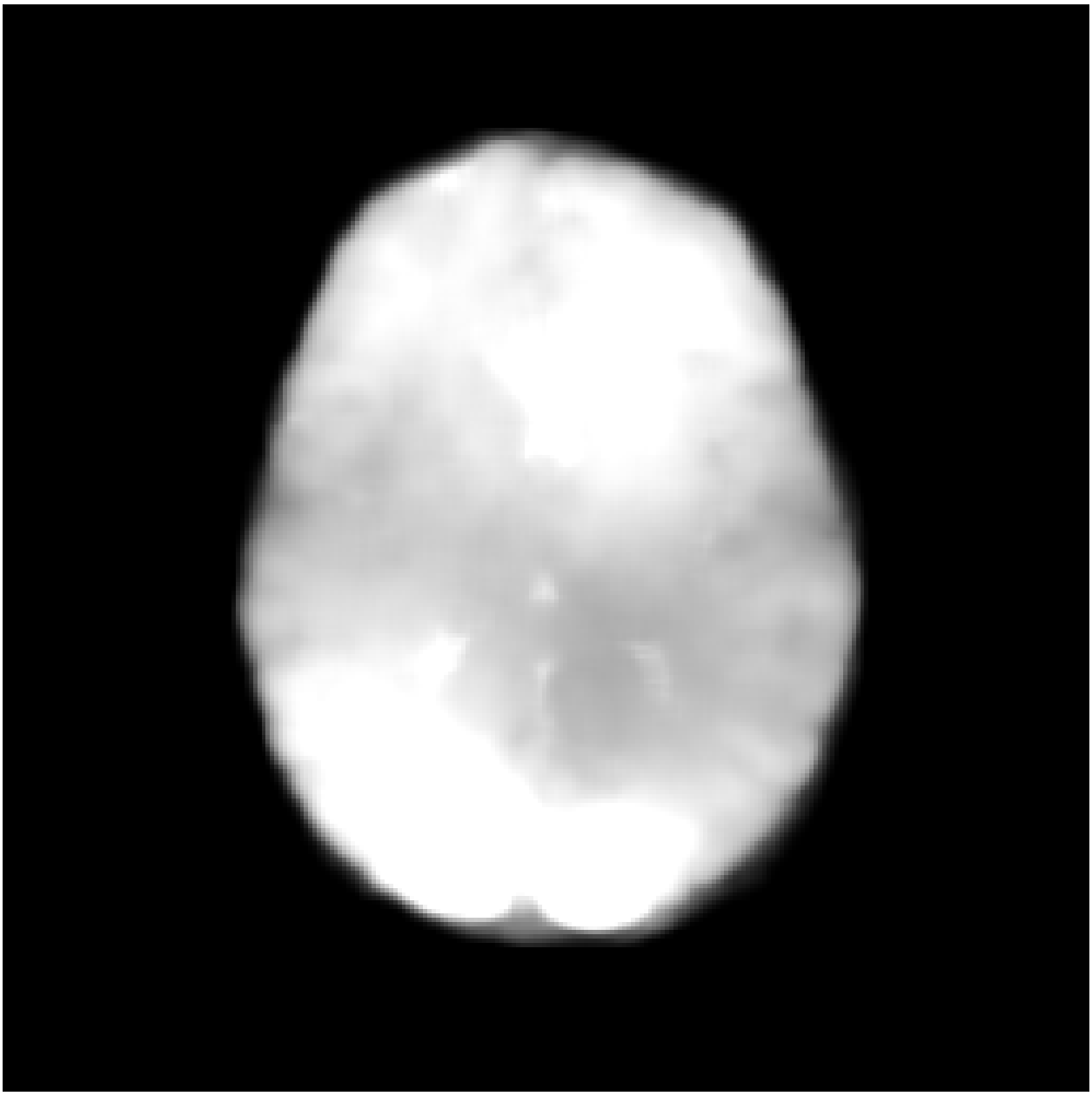
White Matter

Gray Matter

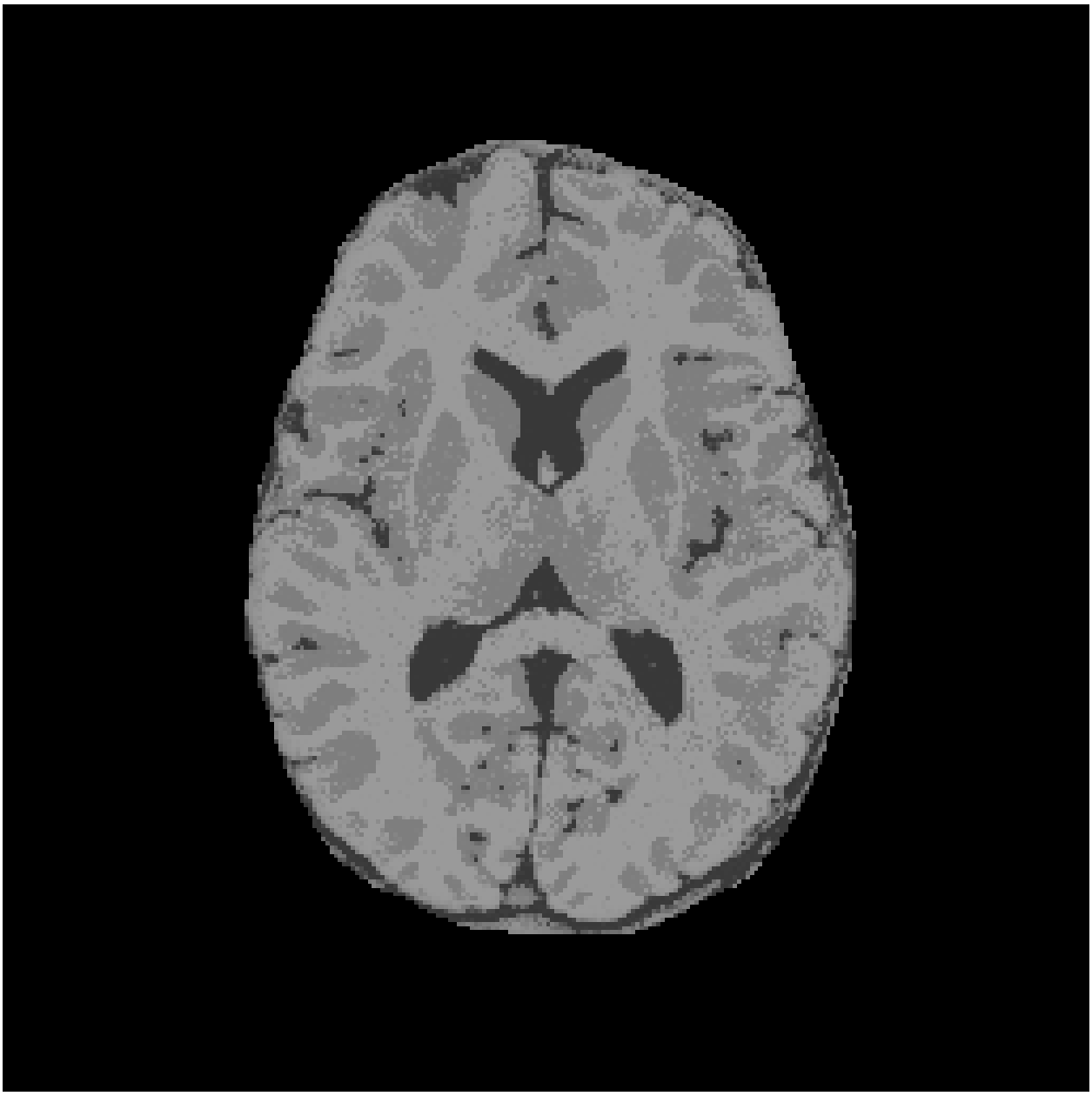


Cerebrospinal Fluid

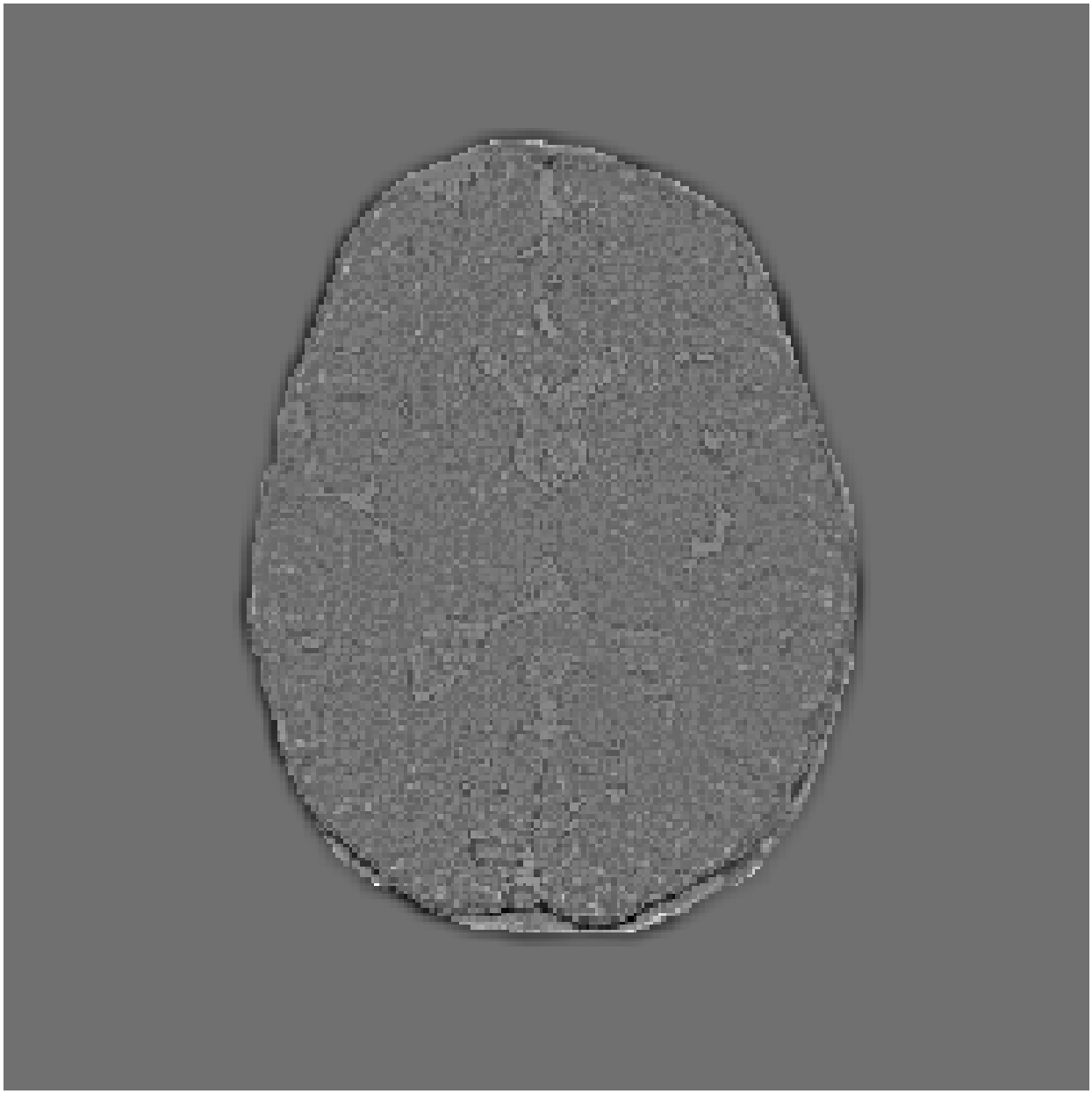
1.6.3 Optimal Bias Field



1.6.4 Bias Removed Image



1.6.5 Residual Image



1.7 Optimal Class Means

Class	Value
White Matter	0.60588
Gray Matter	0.49322
Cerebrospinal Fluid	0.21935

1.8 Uniqueness

Yes the solution will be unique (upto name of class labels). To see this intuitively we observe the objective function

$$L(\theta) = \sum_{j=1}^{N}\sum_{k=1}^{K}\mu_{jk}^{q}\sum_{i=1}^{N}w_{ij}(y_j - c_k b_i)^2 + \sum_{j=1}^{N}\lambda_j\left(1 - \sum_{k}u_{jk}\right)$$

Now we just need to show the mean and the bias will be unique (stating the question).Notice that if we simultaneously try to optimize the function with respect to means and bias it won't yield a unique solution. To see this jut observe that multiplying all b_i by a constant and dividing all c_k by the same factor will lead to the same objective function value.
Solution:While updating bias keep means fixed and while updating the means keep the bias fixed.. This algorithm will lead to a unique solution as at any time the function we are optimizing is convex(sum of squares which are convex functions). Thus each step results in a unique solution and thus the final solution is also unique.
Implementation: To implement this at each step to find the updates of bias or means just find the derivative with respect to that parameter and and equate it to zero to find the optimal value of that parameter in this iteration.
For example to find the new value of b_h set

$$\frac{\partial L}{\partial b_h} = 0$$

2 Segmenting a Brain MRI (HMRF-EM-GMM)

2.1 Smoothness Parameter (β)

The chosen value of the smoothness parameter β is 0.33.

2.2 MRF Prior on Label

2.2.1 Image

2.2.2 Motivation

We divided the values into 3 different partitions of roughly equal size based on maximum value.

2.2.3 Algorithm

We assign labels depending on which partition the pixel value lies in.

2.3 Initial Parameters

2.3.1 Class Means

Class	Value
White Matter	0.5949
Gray Matter	0.4058
Cerebrospinal Fluid	0.1837

2.3.2 Class Standard Deviations

Class	Value
White Matter	0.0545
Gray Matter	0.0785
Cerebrospinal Fluid	0.0522

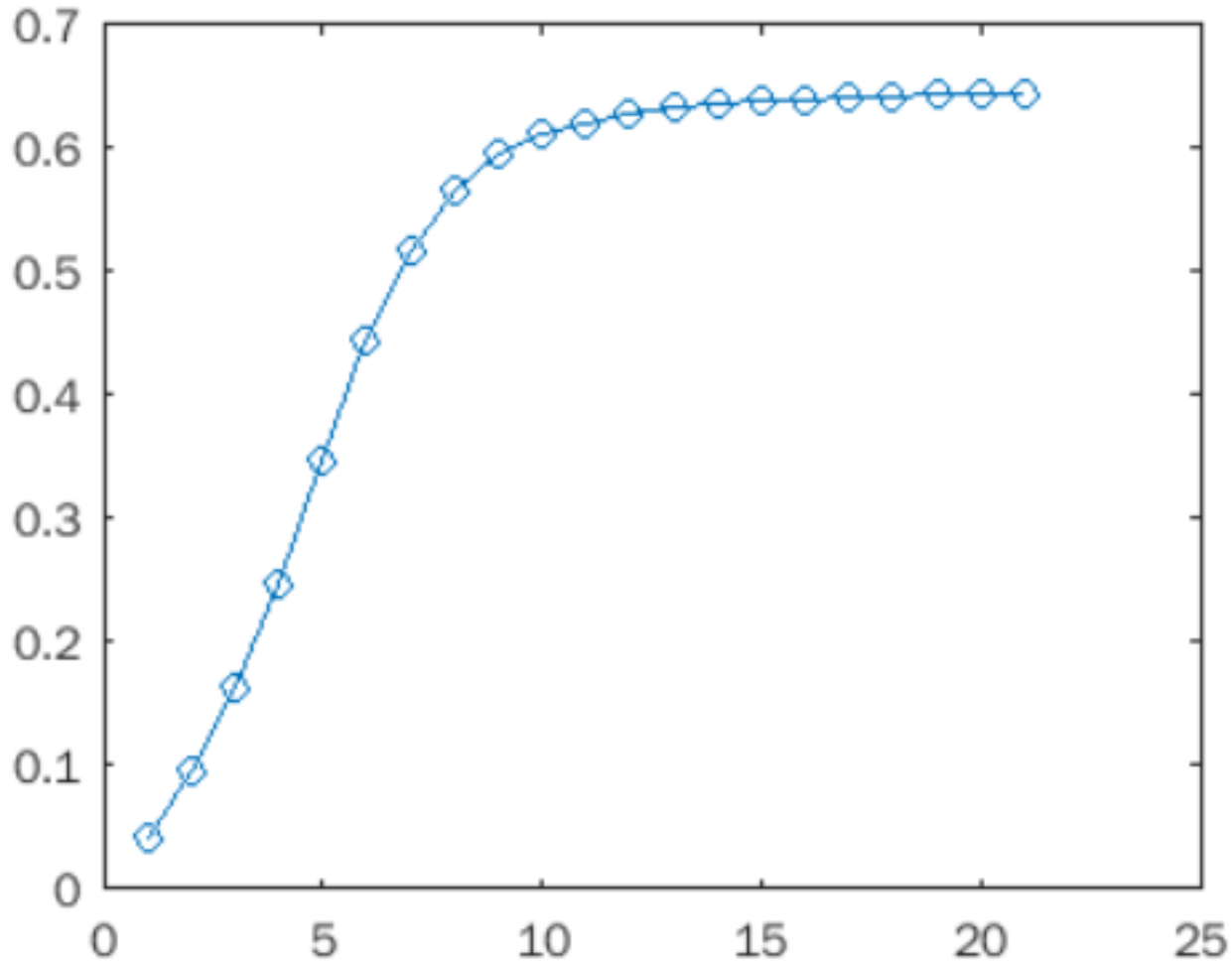
2.3.3 Motivation

Using the labels, we find the average of the pixel values to determine the class means, and the standard deviations.

2.3.4 Algorithm

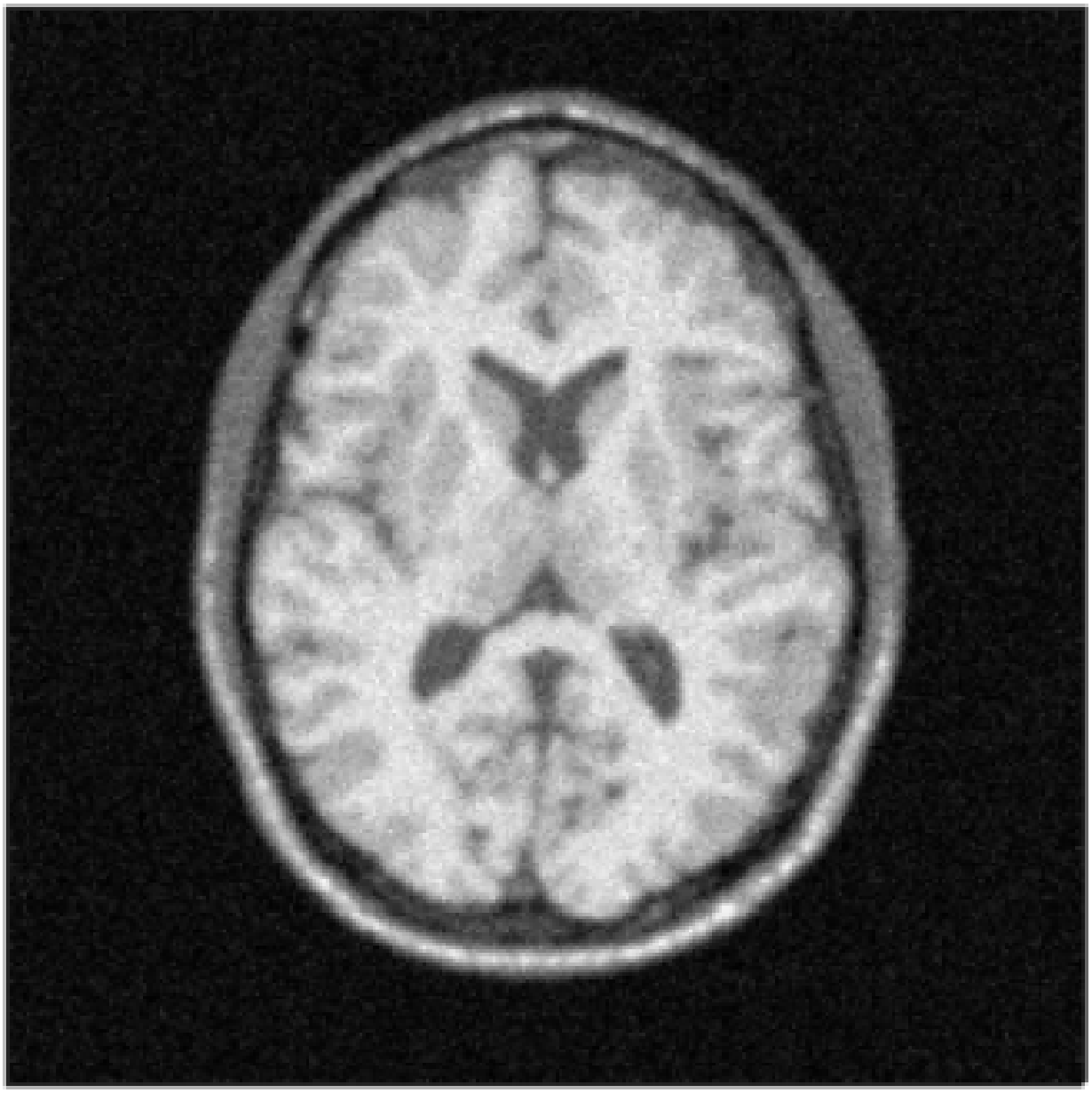
We have used standard functions to find the mean and standard deviations.

2.4 Log Posterior Probability



2.5 Images

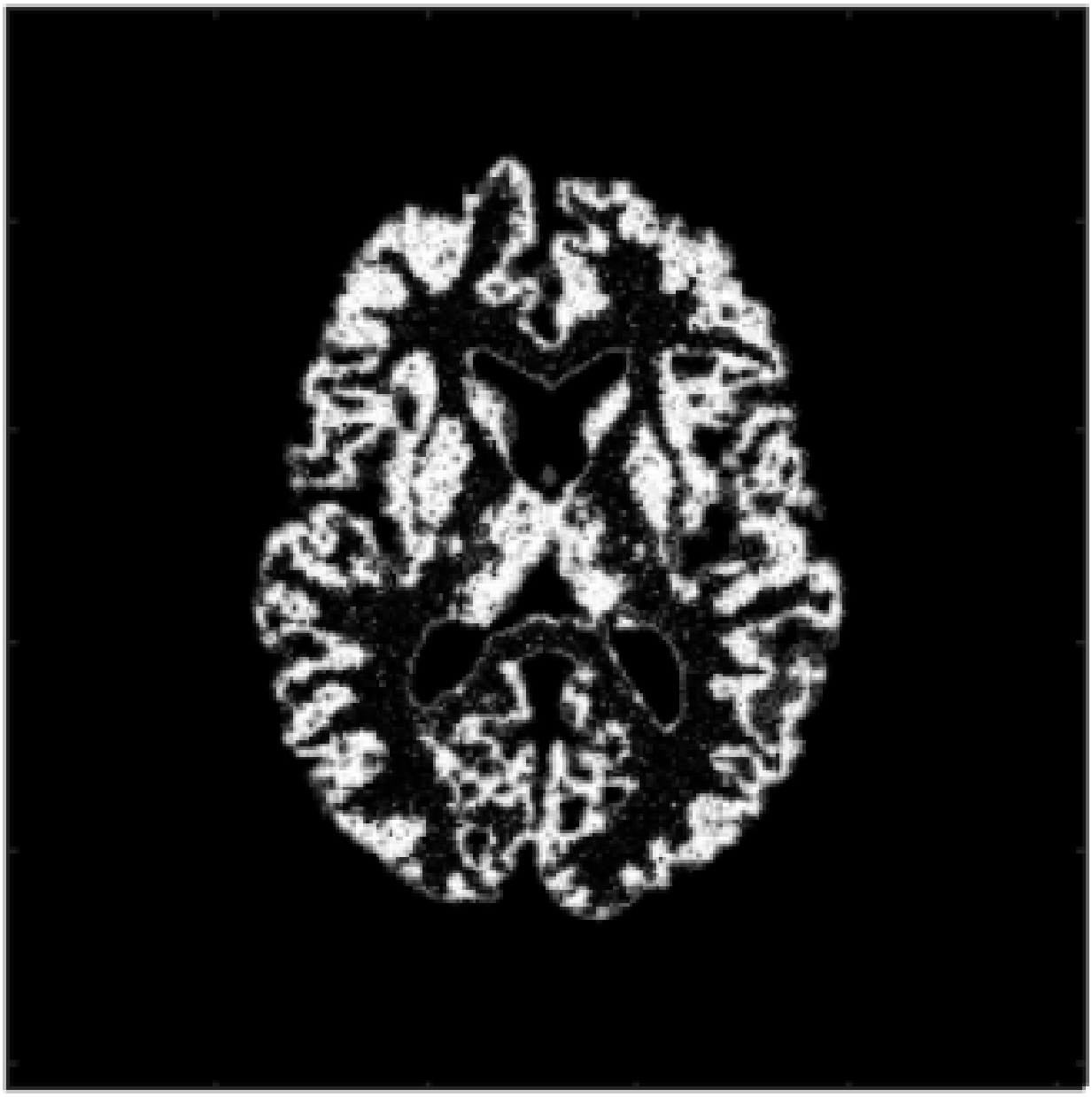
2.5.1 Corrupted Image



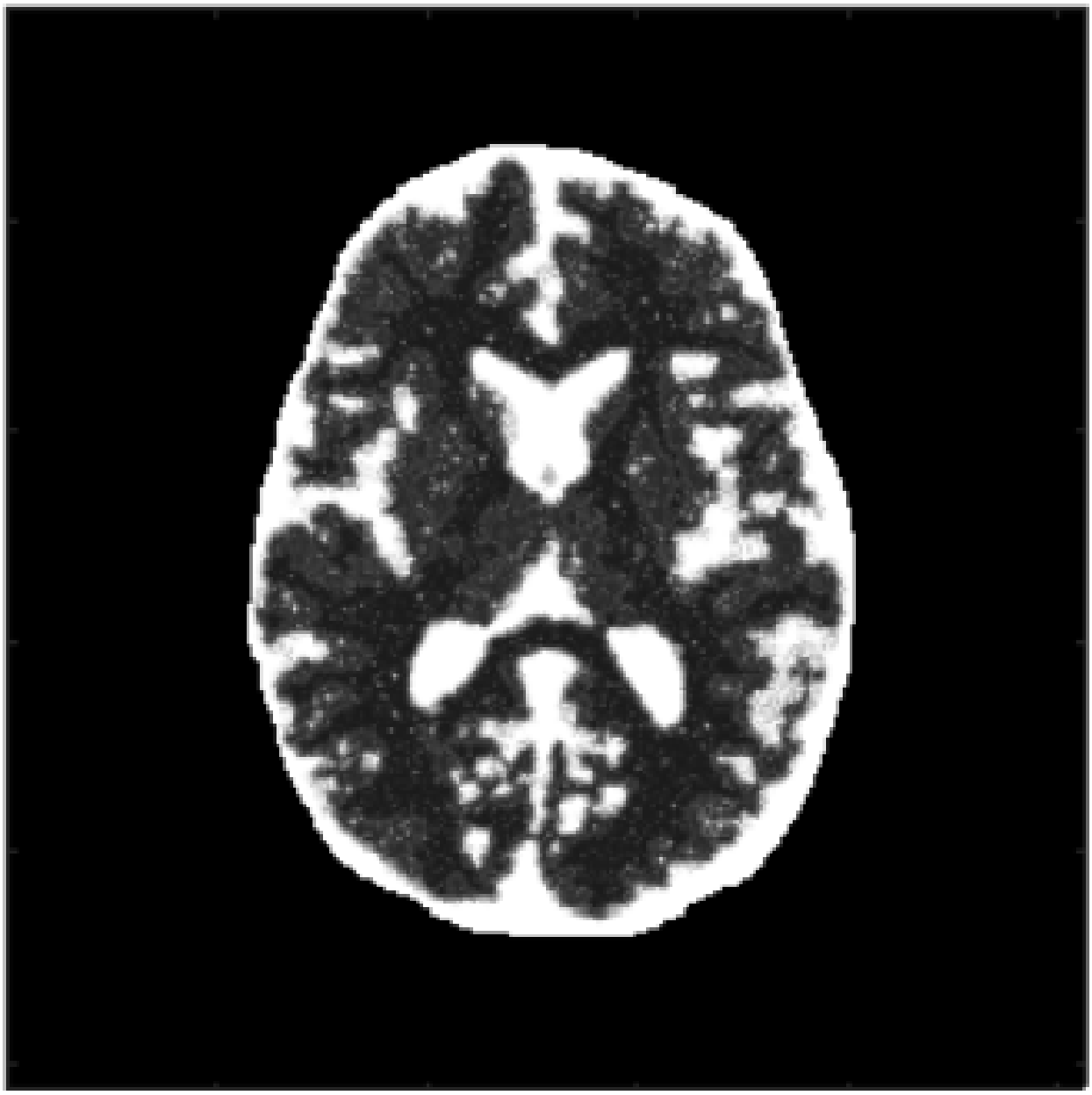
2.5.2 Optimal Class Memberships

White Matter

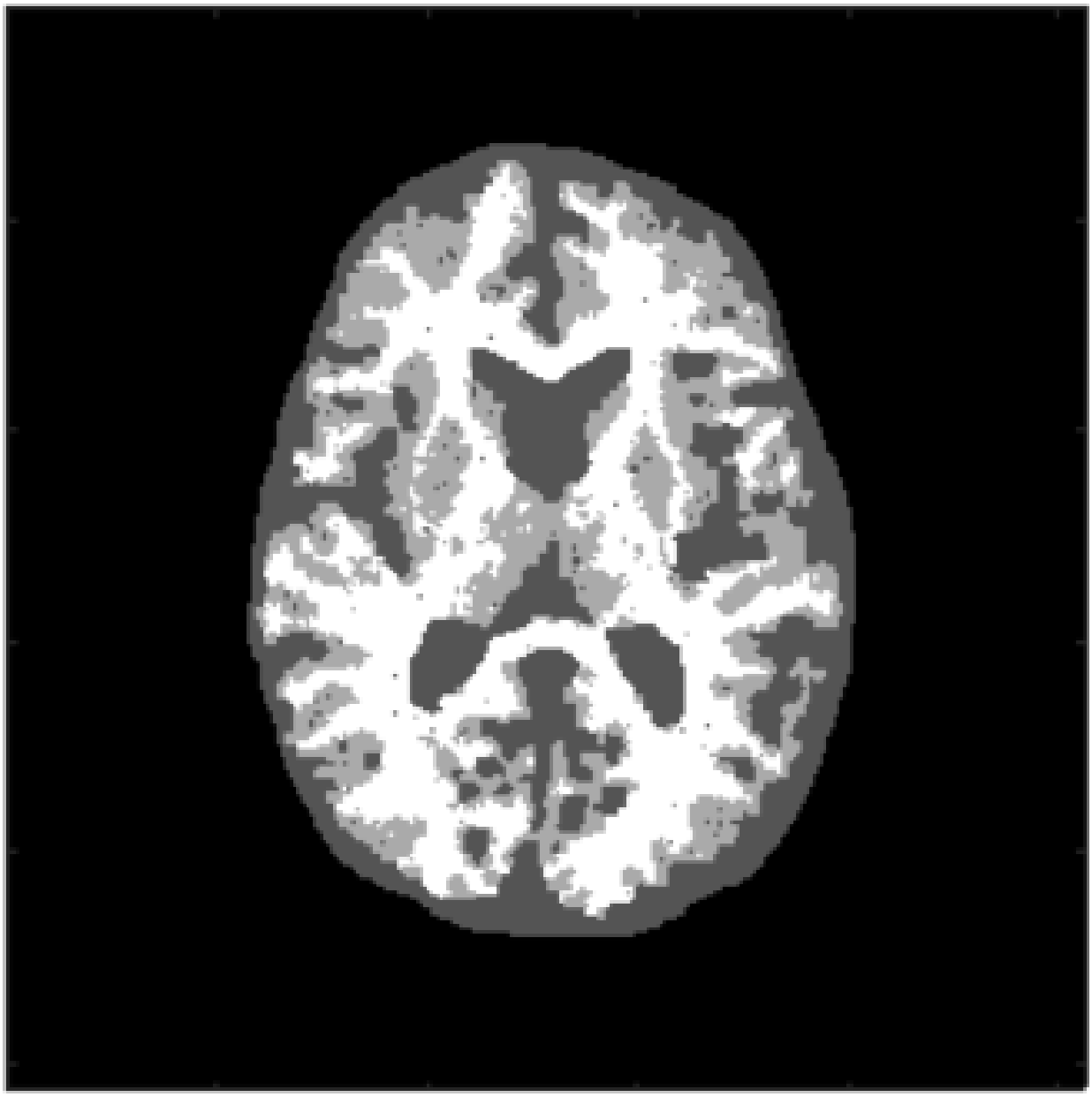
Gray Matter



Cerebrospinal Fluid

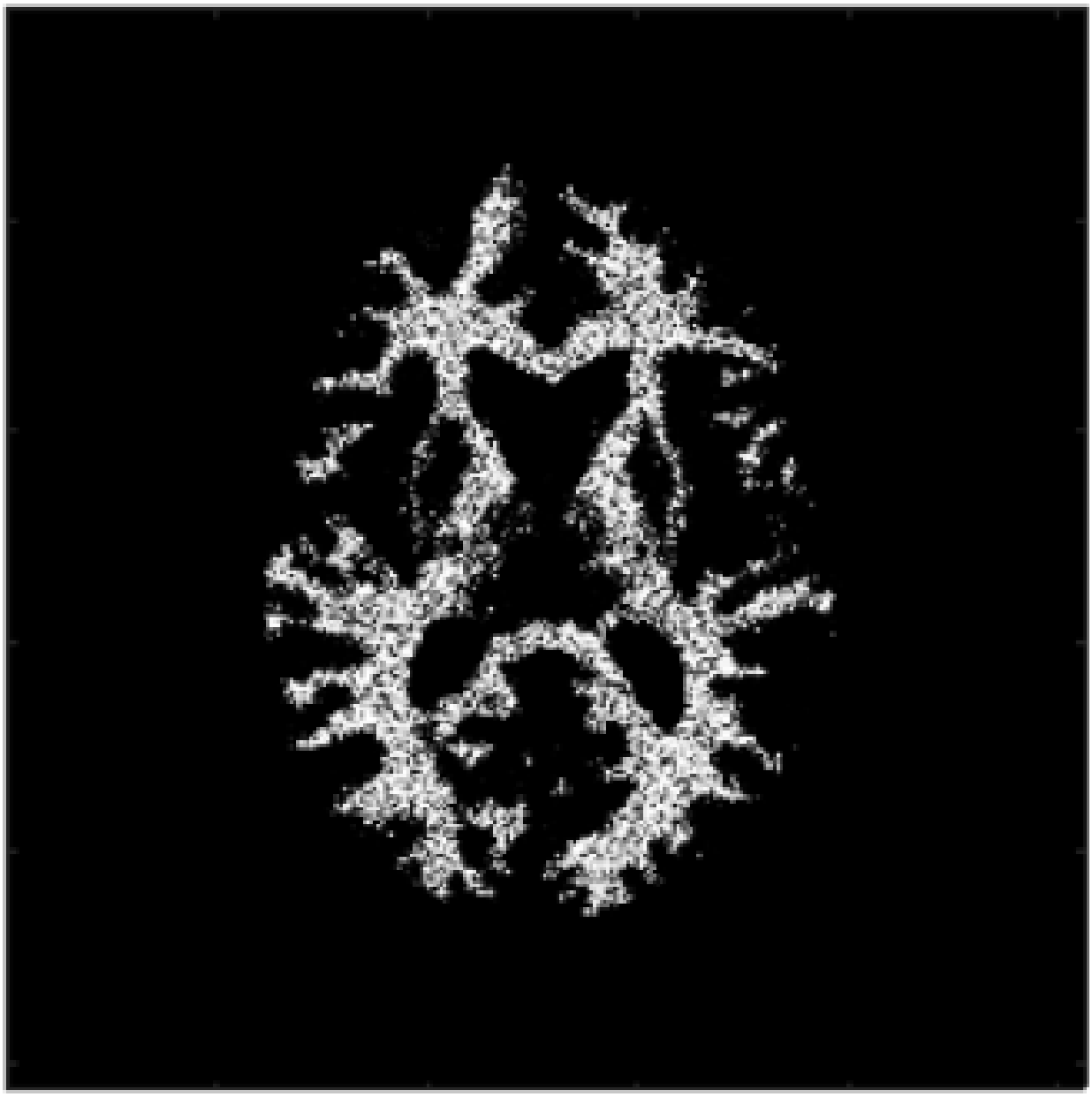


2.5.3 Optimal Label Image

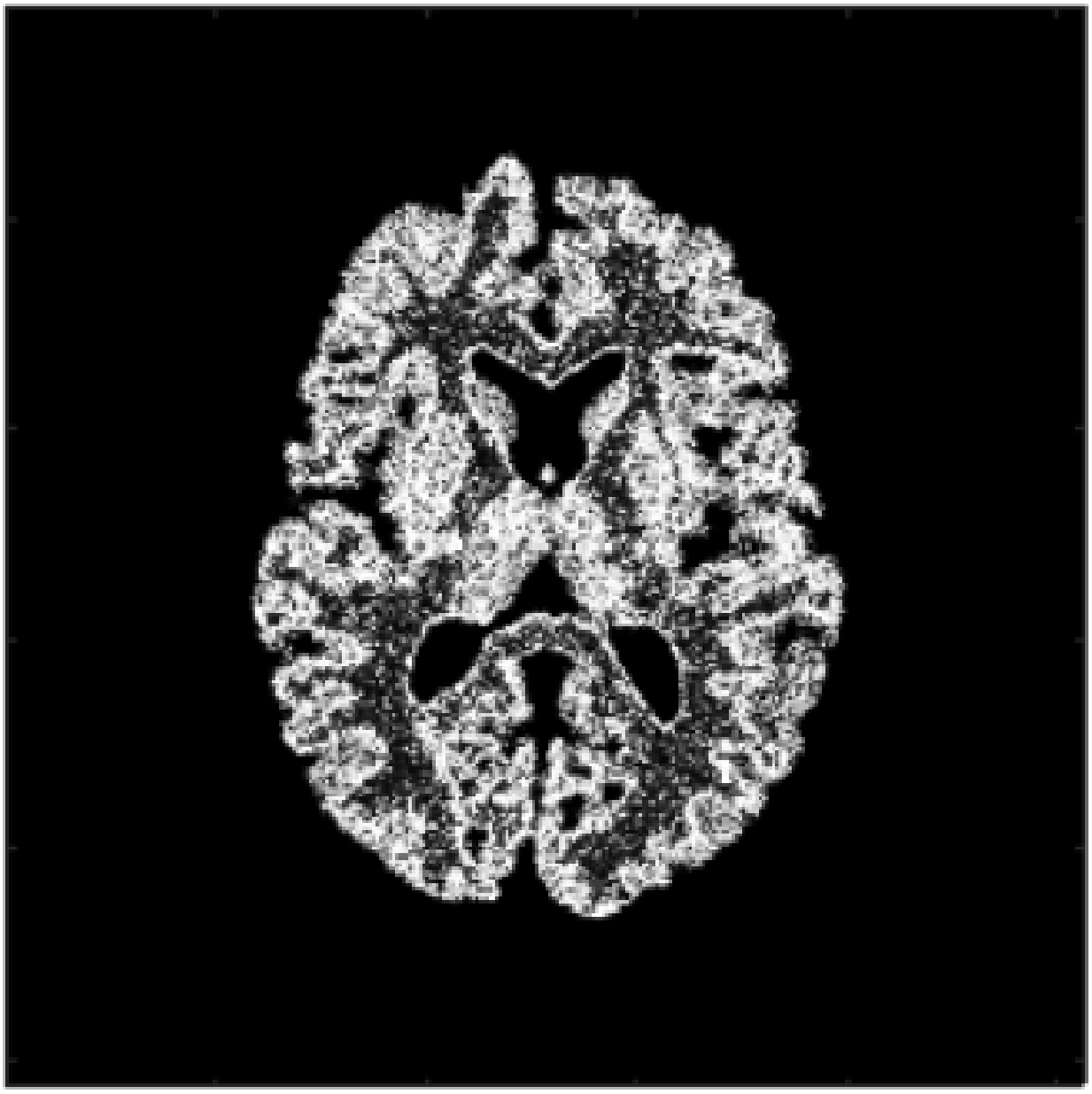


2.5.4 Optimal Class Memberships ($\beta = 0$)

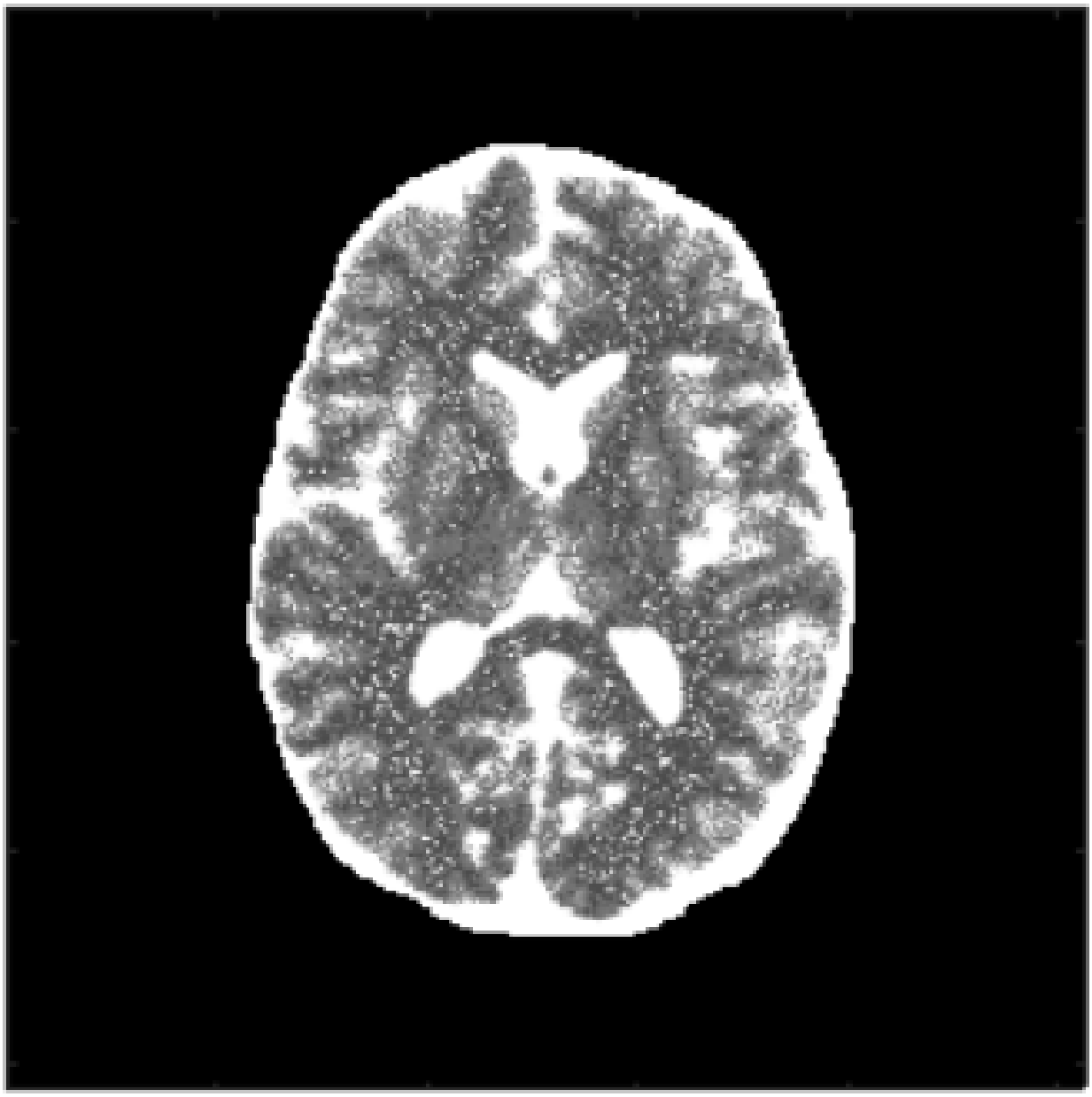
White Matter



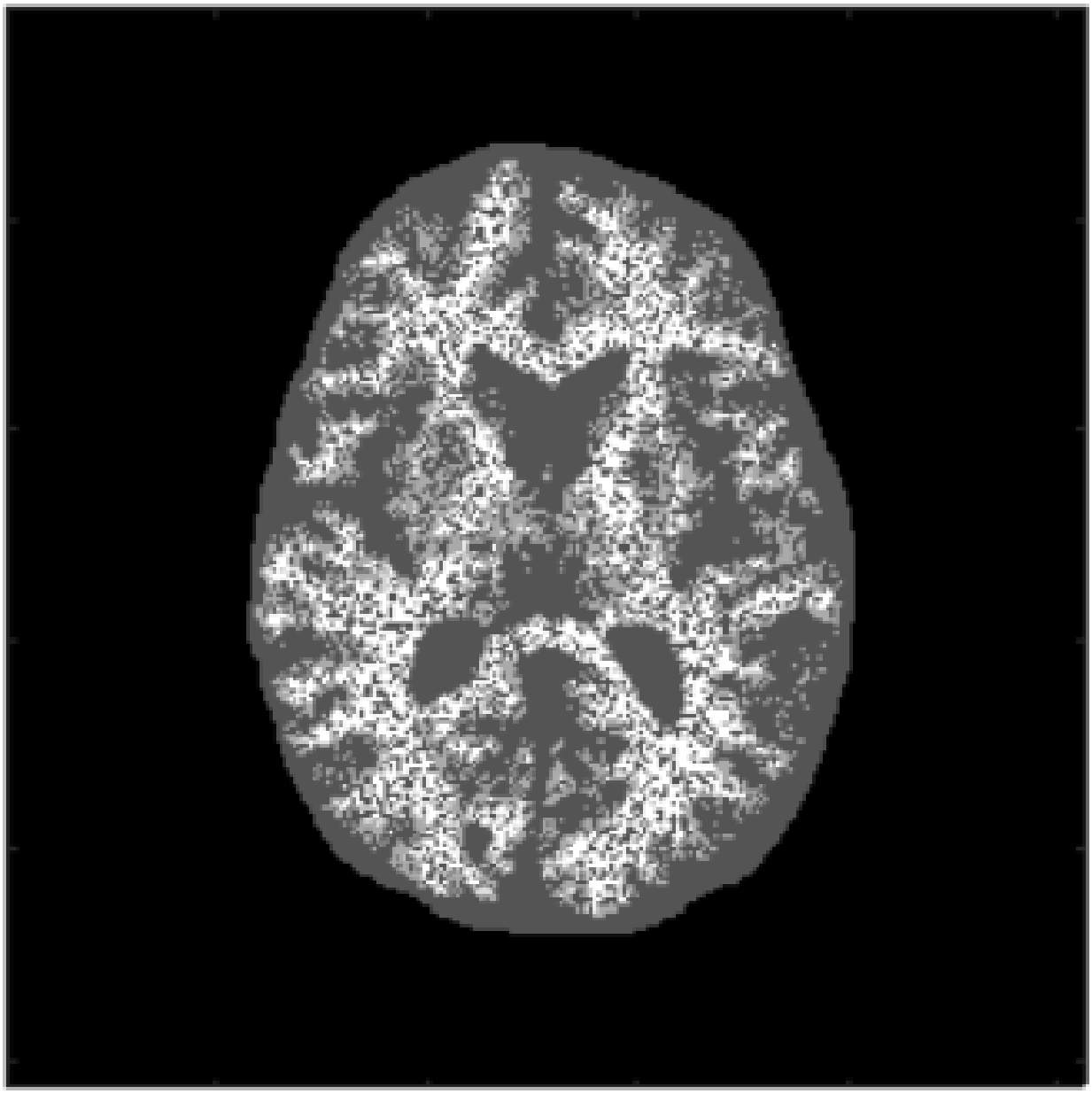
Gray Matter



Cerebrospinal Fluid



2.5.5 Optimal Label Image ($\beta = 0$)



2.6 Optimal Class Means

Class	Value
White Matter	0.6474
Gray Matter	0.5660
Cerebrospinal Fluid	0.4746

3 EM Optimization

3.1 Using Parameter Prior ($P(\theta)$)

3.1.1 Extending the Framework

If we have prior information on the parameters θ underlying the image intensity model, we can estimate θ using maximum-a-posteriori (MAP) estimation. Considering the observed data y, hidden random variables x as in the original framework, we iteratively optimize estimate for θ.

We design a function of parameters θ, id est, $F(\theta;\theta_i)$ that is a lower bound for the log-posterior function and touches it at the current estimate θ_i. We know, the posterior function is given by,

$$P(\theta|y) = \frac{P(y|\theta)P(\theta)}{P(y)}$$

where $P(y|\theta)$ is the likelihood, $P(\theta)$ is the prior and $P(y)$ is the marginal probability. So the log-posterior will be,

$$log(P(\theta|y)) = log(P(y|\theta)) + log(P(\theta)) - log(P(y))$$

Similar to the original framework, we can write log-posterior as,

$$log(P(\theta|y)) = F(q,\theta) + KL(q||P(x|y,\theta))$$

where,

$$F(q,\theta) := \sum_x q(x)\left(log\left(\frac{P(y,x|\theta)}{q(x)}\right)\right) + log(P(\theta)) - log(P(y))$$

and $KL(.||.)$ is the Kullback-Leibler divergence function. Since $KL(.||.) \geq 0$ (equality iff functions are the same), $F(q,\theta)$ is a lower bound of log-posterior for any chosen $q(x)$, and $F(q,\theta^*)$ equals $log(P(\theta^*|y))$ iff $q(x) = log(P(x|y,\theta^*))$.

3.1.2 E-Step

At every iteration i, we design $q_i(.)$ to maximize $F(q,\theta_i)$. We take,

$$q_i(x) := log(P(x|y,\theta_i))$$

which implies $F(q_i,\theta_i) = log(P(\theta_i|y))$, and that $F(.)$ touches the log-posterior function at θ_i, as desired.

3.1.3 M-Step

We choose θ_{i+1} such that $F(q_i,\theta_{i+1})$ is maximized (subject to the set constraints of a touching lower bound). We note that,

$$F(q_i,\theta_{i+1}) := E_{q_i(x)}[log\left(P(y,x|\theta_{i+1})\right)] + log(P(\theta_{i+1})) + H(q_i) + G(y)$$

where $H(q)$ and $G(y)$ are the entropy of q and negative log marginal probability of y, both of which are not functions of θ_{i+1}. So we define,

$$Q(\theta;\theta_i) = E_{q_i(x)}[log\left(P(y,x|\theta_{i+1})\right)] + log(P(\theta_{i+1}))$$

The update at each iteration should be such that $Q(\theta_{i+1};\theta_i)$ is an increasing/non-decreasing function.

3.1.4 Termination

We terminate when the relative change in θ_i between consecutive iterations falls below a user-defined threshold.

3.2 GMM Model with Priors

3.2.1 Prior Models

We define the prior PDF,

$$P(\theta) = P(w) \times \prod_{k=1}^{K} P(\mu_k|C_k)P(C_k)$$

which is chosen as an approximate conjugate prior to the likelihood function.
The weights w_k are modeled by a Dirichlet distribution,

$$P(w) \propto \prod_{k=1}^{K} w_k^{\xi_k - 1}$$

where ξ_k is the concentration hyper-parameter.
The means μ_k are modeled by Gaussian distribution,

$$P(\mu_k|C_k) = G(\mu_k|m_k, C_k/\sqrt{\kappa_k})$$

where m_k is the mean hyper-parameter, κ_k is the compactness hyper-parameter and C_k is the covariance matrix.
The covariance matrices are modeled by inverse Wishart distribution,

$$P(C_k|V_k, \lambda_k) = IW(C_k; V_k, \lambda_k)$$

$$IW(C_k; V_k, \lambda_k) \propto |C_k|^{-(\lambda_k+m+1)/2}\, e^{-\frac{1}{2}tr(V_k C_k^{-1})}$$

where V_k is the hyper-parameter for scale matrix and λ_k is the degree of freedom hyper-parameter.

3.2.2 E-Step

Similar to previous part, we design functions at every iteration to maximize the function F that is the lower bound for log-posterior, which will be the sum of log of Gaussian Mixture Model (log-likelihood) and log of aforementioned prior.

3.2.3 M-Step

In each iteration, keeping the weights fixed, we estimate the means and covariance matrices using MAP estimation. Then using the new means and covariance matrices, we update the memberships (ratio of corresponding term in the GMM and sum of all terms). For means since its a simple Gaussian finding the MAP estimates is simple. For covariance matrix estimate let us see the posterior distribution for the covariance matrix (D is data matrix, S is covariance of data, m is size of scale matrix).

$$P(C_k|D) \propto P(D|C_k)P(C_k) = |C_k|^{-n/2} e^{-(n/2)tr(SC_k^{-1})} |C_k|^{-((\lambda_k)+m+1)/2} e^{-tr((nS+V_k)C_k^{-1})}$$

This is nothing but a IW distribution itself given as,

$$IW(nS + V_k, n + m)$$

Now we can carry out Cholesky decomposition (a matrix partitioning procedure) of the inverse Wishart matrix to maximize the covariances and get independent normal chi-squared random variables along the diagonal, which can be used as estimates of the square of the standard deviations.

3.2.4 Termination

We terminate when the relative change in parameters between consecutive iterations falls below a user-defined threshold.