

Segmenting Lesions in Multiple Sclerosis Patients

James Chen, Jason Su

Radiologists and researchers spend countless hours tediously segmenting white matter lesions to diagnose and study brain diseases. This mundane task is ideal for a machine.

Our project builds a binary classifier to identify lesions using a novel data set. We use a set of 3-D brain volumes for each subject; it contains both the typical “clinical” images used for segmentation, as well as 10 additional quantitative “maps” that measure physical tissue properties. Using these maps for lesion segmentation has never been done before.

This work has direct application for studies we are involved in. It will enable us to collect and process extensive longitudinal data for gaining a greater understanding of multiple sclerosis, a disease which affects up to 0.1% of the population.

A Primer on Brains

Voxels in a brain are usually classified into categories like white matter, gray matter, and lesions. It is relatively easy to differentiate between gray and white matter, and between lesions and white matter. However, gray matter and lesions are hard to differentiate. They have similar pixel intensities so an algorithm that tries to segment gray and white matter will often label lesions as gray. Most gray matter is located at the surface/cortex of the brain. We can easily ignore this. The difficulty lies in differentiating between lesions and deep gray matter structures.

Data Set

Each patient has four clinical images (which we’ll refer to as Clinical) and ten quantitative maps (which we’ll refer to as mcDESPOT maps). Each image or map is a 128x128x76 matrix of values, corresponding to voxels in the brain volume.

Our training set includes data from five subjects. The data can be completely expressed in a 80,000,000x14 matrix. However, non-lesion voxels vastly outnumber those of lesions, so training on all observations would result in the naive prediction of non-lesion for everything. To fix this, we decided to balance the data set by randomly sampling rows of the full matrix so that the ratio of non-lesion to lesion voxels was 2-to-1. All lesion voxels were kept since they represent the most important data. The resulting matrix’s size is 7,251x14.

The test set contains data from one other patient with 724 true lesion voxels. We do not balance this patient’s data. The test set is a 84814x14 matrix.

Preprocessing - White Matter Mask

We use an image mask that discards the brain’s gray matter from our data set. Even though this white matter mask is edited by a human, the task is much easier and faster than selecting lesions. It is standard procedure to produce such a mask when processing MRI images. Thus our problem is simplified to finding lesions in white matter.

Learning Algorithm

We built our hypotheses on support vector machines (SVM) using the radial kernel. We also experimented with linear regression and SVM with a linear kernel, but they did not perform as well with about a 10% drop using a linear kernel. We used the libsvm library.¹

Learning the Hypothesis

Our initial feature space uses the clinical images that are typically employed by radiologists to segment lesions by hand. We wanted to evaluate how well they worked alone. Each voxel is treated as an independent sample with no information about its spatial neighbors. We looked at its misclassification, recall, precision, and similarity index as a function of training set size.

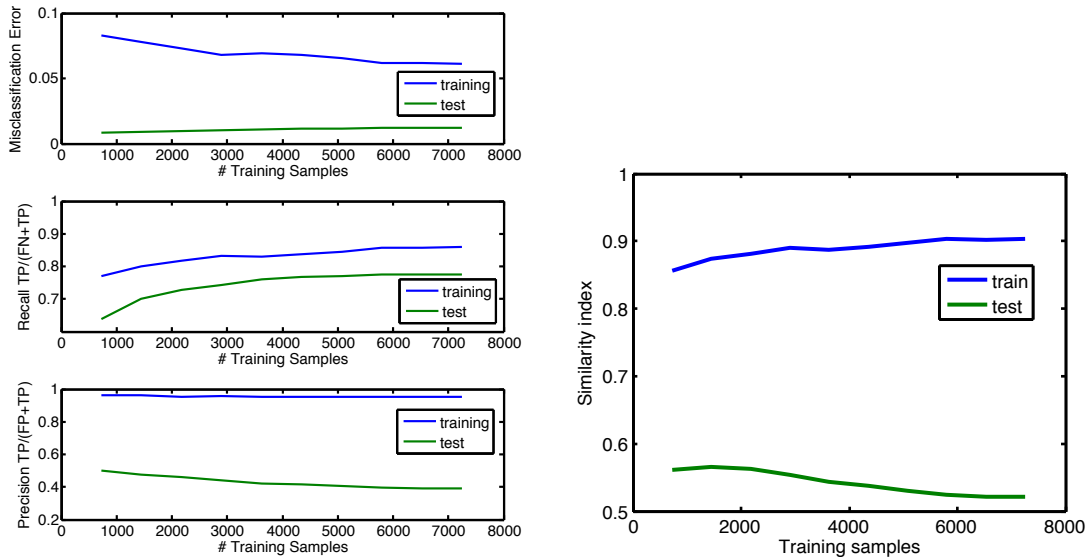


Figure 1. Misclassification error, recall, precision (left) and similarity index (right) for Clinical features

We then added the novel quantitative imaging maps to the feature space. This did not improve the training SI or test SI.

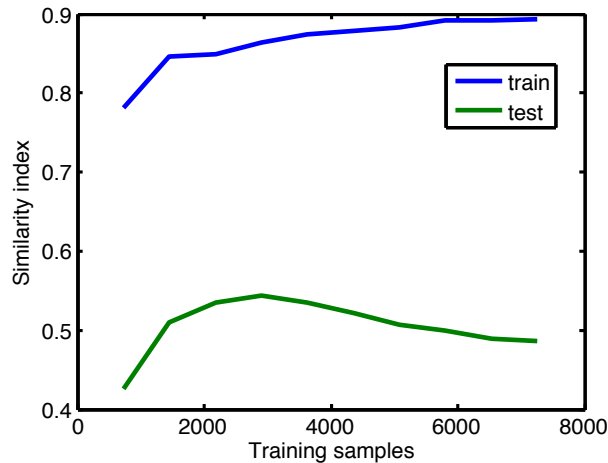


Figure 2. Learning curve for Clinical + mcDESPOT features

Similarity Index

After the initial evaluation of our data set, we chose to use similarity index (SI) as the primary metric of the performance of our features and algorithms. SI is a standard measure for

the performance of lesion segmentation in literature, and we wanted to be able to compare our results.² It is the harmonic mean of precision and recall. By optimizing SI, one is balancing between reducing false negatives and false positives, both of which are valuable in this task.

Adding features

Locally-weighted Gaussian neighborhood

We wanted to have a feature that represented the aggregate characteristics of a voxel's immediate neighborhood. Before this, each voxel was treated independently with no spatial context. For each voxel, we generate a weighted average of the 3-D neighborhood with a 3-D Gaussian kernel, akin to the kernel used in locally-weighted linear regression. With a finite kernel size of $7 \times 7 \times 7$, the Gaussian tails are cut off. We apply a Hamming window to mitigate Gibbs ringing. The kernel is normalized to have a sum of one. This adds 14 features to the data, one for each original image/map.

Edge handling

The weighted neighborhood feature is not valid at the edges of the image, where the background is zero. To account for this, we have a simple binary flag. The flag equals zero if the Gaussian neighborhood is computed with any background zeros values, and one otherwise, i.e. if the neighborhood contains all valid data. This adds one binary feature to the data.

Edge detection

Edges of lesions are particularly hard to segment because they often fade smoothly into their surroundings. We used a high pass filter to boost the contrast for each of the original 14 features, highlighting the edges, and added these as additional features. Here we used a simple $3 \times 3 \times 3$ kernel that summed to zero with equal weights on all but the center value. This adds another 14 features to the data.

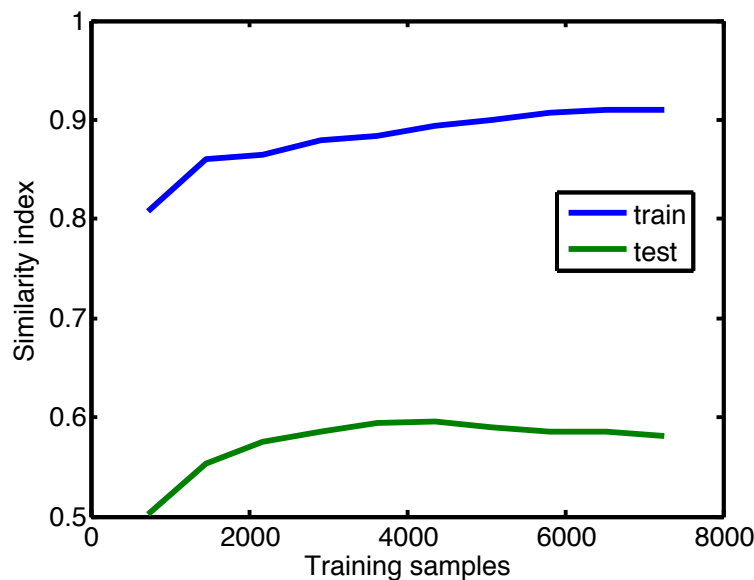


Figure 3. Learning curve for Clinical + mcDESPOT + Neighborhood + Edge-related features

Feature selection – PCA

A problem we have encountered thus far was overfitting. The training SI improves monotonically, while the test SI would decrease with more training samples. So we performed principal component analysis on all 43 features. Then we determined the optimal number of principal components to use by computing the test SI for each. The top 18 principal components were best and were kept as input features. As shown by the learning curve in below, test SI is now much more stable as training samples are increased.

The first principal component lies primarily along the axes of several mcDESPOT quantitative features. The weight coefficients are large for these features.

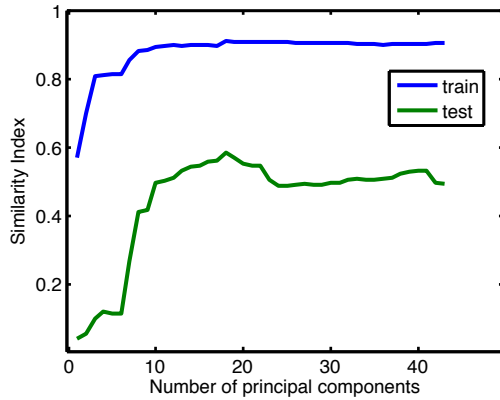


Figure 4. Determining the optimal number of principal components. (left)

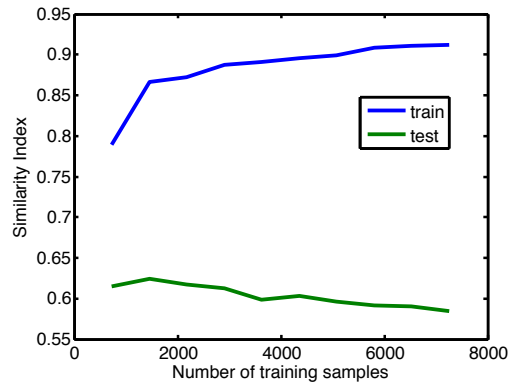


Figure 5. Learning curve with top 18 PCA features. (right)

Training features	Test SI	Training SI
Clinical	56.5	90.3
Clinical + mcDESPOT	54.4	89.4
Clinical + mcDESPOT + N&E	59.6	91.0
PCA-Top18 (Clinical + mcDESPOT + N&E)	62.4	91.2

Table 1. A comparison of learning between different feature sets

Final Results

We validated our final hypothesis on three new patients with different types of MS.

Patient number	Validation SI
P008	65.5%
P018	73.6%
P029	45.8%

Table 2. Final model on three new patients.

Discussion

State-of-the-art lesion segmentation algorithms have an SI of about 70% with similar variations in performance between subjects.² With the preliminary results of this study, we are

confident that the novel quantitative maps can complement clinical images and contribute additional information to identifying lesions.

A large gap remains between the training and actual performance. This is not due to a lack of data since even with the addition of 10 more brains to the training set, there was still a comparable disparity. We believe that the variability between the brains of individuals as well as the subjectivity of the radiologists who defined the true lesion labels are partially responsible for this difference. The boundaries of lesions are poorly defined and vary greatly even between experts doing the manual segmentation.

Additional work can be done to tune the system such as optimizing the C-parameter, radial kernel parameter, and Gaussian neighborhood size and width. We can also do better to clean the clinical images, which are typically on an inconsistent scale between patients.

Conclusion

We have achieved a lesion segmentation system that is fast and performs nearly as well as state-of-the-art methods. With so many avenues still yet to be explored, we believe that there is great potential for an even better final system.

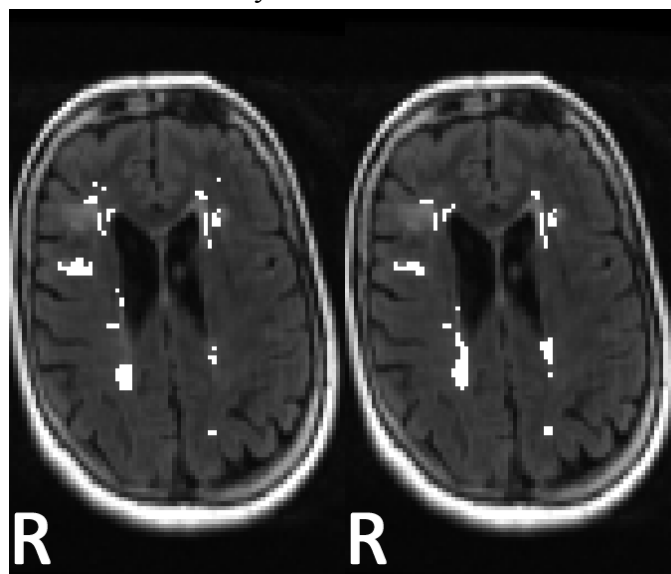


Figure 6. Patient 018 with lesions marked by hand in white. (left)
Figure 7. Patient 018 with lesions marked by this paper's final model in white. (right)

References

- [1] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1--27:27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [2] H. Soltanian-zadeh, "Segmentation of multiple sclerosis lesions in MR images : a review," *Brain*, 2011.